杭州海康机器人股份有限公司

VisionMaster 算法开发平台

用户手册

扫码可得更多产品资料

HIKROBOT

法律声明

版权所有©杭州海康机器人股份有限公司 2026。保留一切权利。

本手册的任何部分，包括文字、图片、图形等均归属于杭州海康机器人股份有限公司或其关联公司（以下简称“海康机器人”）。未经书面许可，任何单位或个人不得以任何方式摘录、复制、翻译、修改本手册的全部或部分。除非另有约定，海康机器人不对本手册提供任何明示或默示的声明或保证。

关于本产品

本手册描述的产品仅供中国大陆地区销售和使用。本产品只能在购买地所在国家或地区享受售后服务及维保方案。

关于本手册

本手册仅作为相关产品的指导说明，可能与实际产品存在差异，请以实物为准。因产品版本升级或其他需要，海康机器人可能对本手册进行更新，如您需要最新版手册，请您登录海康机器人官网查阅（ *http://www.hikrobotics.com* ）。
海康机器人建议您在专业人员的指导下使用本手册。

商标声明

- HIKROBOT 为海康机器人的注册商标。
- 本手册涉及的其他商标由其所有人各自拥有。

责任声明

- 在法律允许的最大范围内，本手册以及所描述的产品（包含其硬件、软件、固件等）均“按照现状”提供，可能存在瑕疵或错误。海康机器人不提供任何形式的明示或默示保证，包括但不限于适销性、质量满意度、适合特定目的等保证；亦不对使用本手册或使用海康机器人产品导致的任何特殊、附带、偶然或间接的损害进行赔偿，包括但不限于商业利润损失、系统故障、数据或文档丢失产生的损失。
- 您知悉互联网的开放性特点，您将产品接入互联网可能存在网络攻击、黑客攻击、病毒感染等风险，海康机器人不对因此造成的产品工作异常、信息泄露等问题承担责任，但海康机器人将及时为您提供产品相关技术支持。
- 使用本产品时，请您严格遵循适用的法律法规，避免侵犯第三方权利，包括但不限于公开权、知识产权、数据权利或其他隐私权。您亦不得将本产品用于大规模杀伤性武器、生化武器、核爆炸或任何不安全的核能利用或侵犯人权的用途。如本手册内容与适用的法律相冲突，则以法律规定为准。

目 录

第 1 章 前言 1
1.1 符号约定 1
1.2 相对路径约定 1
1.3 获得支持 2
第 2 章 发版说明 3
2.1 升级必看 3
2.1.1 与 MVS 相关的注意事项 3
2.2 V4.4.0 维护版本（2026.05） 4
2.3 V4.4.0 维护版本（2025.11） 5
2.4 V4.4.0 维护版本（2025.04） 6
2.5 V4.4.0 维护版本（2024.08） 7
2.6 V4.4.0 维护版本之前 11
第 3 章 产品概述 25
3.1 运行环境 26
3.2 授权方式 27
3.2.1 加密狗授权 28
3.2.2 远程加密狗授权 30
3.2.3 授权码授权 31
3.3 术语介绍 33
3.4 界面介绍 37
3.4.1 欢迎页 38
3.4.2 主界面 39
3.5 二次开发方式 46
第 4 章 快速入门 49
4.1 准备工作 49
4.1.1 准备图像采集设备 49

4.1.2 获取软件使用授权 49
4.2 了解方案构成 54
4.3 体验示例方案 57
4.4 搭建入门级方案 60
4.4.1 步骤一：创建方案 63
4.4.2 步骤二：搭建流程 64
4.4.3 步骤三：设计运行界面 74
4.5 运行入门级方案 79
4.6 保存方案 79
4.7 进阶参考 80
第 5 章 方案搭建-硬件设备配置 82
5.1 相机管理 82
5.1.1 添加相机 82
5.1.2 配置自定义参数相机 83
5.1.3 配置全局相机 84
5.1.4 配置 HKR 采集卡连接的相机 91
5.1.5 配置 Dalsa 采集卡连接的相机 92
5.1.6 配置 Matrox 采集卡连接的相机 93
5.2 通信管理 95
5.2.1 设备管理 95
5.2.2 接收事件 117
5.2.3 发送事件 124
5.2.4 心跳管理 128
5.2.5 响应配置 129
5.2.6 通信示例 132
第 6 章 方案搭建-流程配置 139
6.1 配置单流程运行逻辑 139
6.2 配置多流程运行逻辑 141

6.2.1 步骤一：新建流程 141
6.2.2 步骤二：编排相关实体连接 142
6.2.3 步骤三：定义各流程的输入输出 143
6.2.4 （可选）步骤四：定义各流程可选配置 144
6.2.5 步骤五：试运行并调试 148
6.3 其他操作与配置 149
第 7 章 方案搭建-模块配置 152
7.1 模块使用基础规则 152
7.2 模块通用配置 154
7.2.1 基本参数 154
7.2.2 运行参数 155
7.2.3 结果显示 157
7.2.4 绘制 ROI 区域 157
第 8 章 方案搭建-全局与局部逻辑配置 162
8.1 全局相机 162
8.2 全局/局部变量 162
8.3 全局触发 165
8.4 全局脚本 167
8.4.1 VS 调试 170
8.4.2 全局脚本开放接口 171
8.4.3 SDK 调用 182
8.4.4 程序集添加 184
8.4.5 第三方库调用方法 185
8.4.6 应用示例：流程控制 186
8.4.7 应用示例：设置模块参数 193
8.5 数据队列 197
8.5.1 使用示例 201
第 9 章 方案搭建-运行界面设计 206

9.1 设计窗口概览 206
9.2 设计运行界面 208
9.2.1 通用设计思路 208
9.2.2 控件通用操作 208
9.2.3 设计示例 208
9.3 界面元素参考 211
9.3.1 画布及固定栏配置参考 211
9.3.2 控件配置参考 212
第 10 章 方案调试与运行 252
10.1 调试方案 252
10.1.1 图像显示区域 253
10.1.2 结果显示区域 261
10.1.3 综合配置 264
10.2 运行单个方案 265
10.2.1 在运行界面运行 266
10.2.2 导出运行界面并运行 266
10.3 自动切换方案并运行 268
第 11 章 软件通用设置 270
11.1 权限设置 270
11.2 软件设置 272
11.3 方案设置 275
11.4 运行策略 276
11.5 采集策略 277
第 12 章 工具介绍 278
12.1 标定配置引导 278
12.1.1 静态标定 278
12.1.2 动态标定 279
12.1.3 映射标定 280

12.2 标定板生成 281
12.3 异常信息收集 283
12.4 参数查找工具 285
12.5 自定义模块生成 286
12.6 版本切换工具 290
12.7 加密狗写密码工具 292
12.8 读取加密狗序列号工具 294
12.9 存图大数据解压工具 294
12.10 激活工具 295
第 13 章 模块使用参考 296
13.1 采集 296
13.1.1 图像源 296
13.1.2 多图采集 307
13.1.3 输出图像 314
13.1.4 缓存图像 319
13.1.5 光源 320
13.2 定位 322
13.2.1 轮廓匹配 323
13.2.2 高精度匹配 342
13.2.3 快速匹配 351
13.2.4 灰度匹配 361
13.2.5 图形定位 375
13.2.6 位置修正 390
13.2.7 Blob 分析 395
13.2.8 Blob 分析组合 414
13.2.9 Blob 标签分析 423
13.2.10 圆查找 430
13.2.11 椭圆查找 442

13.2.12 阵列圆查找 449
13.2.13 直线查找 455
13.2.14 直线查找组合 463
13.2.15 多直线查找 467
13.2.16 边缘交点 474
13.2.17 四边形查找 480
13.2.18 平行线查找 489
13.2.19 平行线计算 496
13.2.20 矩形检测 500
13.2.21 中线查找 506
13.2.22 垂线查找 509
13.2.23 角平分线查找 513
13.2.24 卡尺工具 516
13.2.25 边缘查找 527
13.2.26 顶点检测 533
13.2.27 路径提取 537
13.2.28 目标跟踪 542
13.3 测量 549
13.3.1 线圆测量 549
13.3.2 圆圆测量 556
13.3.3 点圆测量 562
13.3.4 点线测量 566
13.3.5 线线测量 569
13.3.6 点点测量 575
13.3.7 亮度测量 578
13.3.8 间距检测 582
13.3.9 像素统计 587
13.3.10 直方图工具 590

13.4 识别 595
13.4.1 DL 读码 595
13.4.2 条码识别 604
13.4.3 二维码识别 609
13.4.4 DL 字符定位 614
13.4.5 DL 字符识别 621
13.4.6 DL 单字符检测 628
13.4.7 字符识别 634
13.4.8 ML 分类 640
13.5 缺陷检测 647
13.5.1 表面缺陷滤波 647
13.5.2 字符缺陷检测 654
13.5.3 异常检测 665
13.5.4 边缘模型缺陷检测 680
13.5.5 边缘对模型缺陷检测 691
13.5.6 直线边缘缺陷检测 702
13.5.7 直线对缺陷检测 711
13.5.8 圆弧对缺陷检测 716
13.5.9 圆弧边缘缺陷检测 725
13.5.10 边缘组合缺陷检测 732
13.5.11 边缘对组合缺陷检测 736
13.5.12 边缘位置趋势分析 741
13.5.13 边缘对位置趋势分析 746
13.6 边缘学习 752
13.6.1 学习计数 754
13.6.2 有无检测 761
13.6.3 全监督检测 768
13.6.4 异常分类 776

13.6.5 多类别分类 782
13.6.6 注册分类 789
13.6.7 注册检测 796
13.6.8 注册分割 802
13.7 深度学习 807
13.7.1 DL 快速图像分割 813
13.7.2 DL 图像分割 821
13.7.3 DL 实例分割 828
13.7.4 DL 目标检测 834
13.7.5 DL 分类 841
13.7.6 DL 图像检索 846
13.7.7 DL 无监督分类 851
13.7.8 DL 无监督分割 854
13.8 标定 858
13.8.1 N 点标定 858
13.8.2 N 图像标定 867
13.8.3 标定板标定 873
13.8.4 相机映射 880
13.8.5 映射标定 883
13.8.6 畸变标定 887
13.8.7 平移旋转标定 892
13.8.8 旋转标定 897
13.8.9 坐标系 899
13.8.10 标定加载 901
13.9 运算 905
13.9.1 单点对位 905
13.9.2 单点抓取 907
13.9.3 单点映射对位 910

13.9.4 单点纠偏 913
13.9.5 标定转换 915
13.9.6 点集对位 918
13.9.7 旋转计算 919
13.9.8 线对位 921
13.9.9 单位转换 923
13.9.10 变量计算 926
13.9.11 坐标转换 929
13.10 图像处理 931
13.10.1 图像二值化 931
13.10.2 图像滤波 939
13.10.3 掩膜工具 950
13.10.4 形态学处理 957
13.10.5 图像增强 966
13.10.6 阴影校正 972
13.10.7 图像组合 977
13.10.8 图像归一化 981
13.10.9 图像运算 987
13.10.10 图像修正 994
13.10.11 仿射变换 997
13.10.12 逆仿射变换 1002
13.10.13 图像缩放 1005
13.10.14 几何变换 1009
13.10.15 圆环展开 1012
13.10.16 拷贝填充 1017
13.10.17 畸变校正 1022
13.10.18 图像矫正 1024
13.10.19 图像拼接 1026

13.10.20 清晰度评估 1033
13.10.21 帧平均 1037
13.10.22 多图融合 1039
13.11 颜色处理 1043
13.11.1 颜色抽取 1044
13.11.2 颜色测量 1048
13.11.3 颜色转换 1051
13.11.4 颜色识别 1055
13.11.5 颜色分割 1062
13.11.6 彩图生成 1067
13.12 拆分组合 1069
13.12.1 划片拆分 1069
13.12.2 二维阵列 1072
13.12.3 多标签筛选 1076
13.12.4 Box 融合 1079
13.12.5 Box 重叠率计算 1083
13.12.6 Box 过滤 1085
13.13 图形生成 1087
13.13.1 圆拟合 1087
13.13.2 椭圆拟合 1091
13.13.3 直线拟合 1094
13.13.4 几何创建 1098
13.14 逻辑工具 1101
13.14.1 条件分支 1101
13.14.2 条件检测 1104
13.14.3 分支模块 1107
13.14.4 分支字符 1109
13.14.5 文本保存 1111

13.14.6 逻辑 1114
13.14.7 格式化 1117
13.14.8 字符比较 1120
13.14.9 脚本 1122
13.14.10 Python 脚本 1159
13.14.11 Group 1171
13.14.12 点集 1173
13.14.13 耗时统计 1175
13.14.14 数据集合 1176
13.14.15 触发模块 1178
13.14.16 数据记录 1179
13.14.17 图形收集 1181
13.14.18 数据筛选 1183
13.14.19 数据分类 1186
13.14.20 数据排序 1189
13.15 通信 1192
13.15.1 接收数据 1192
13.15.2 发送数据 1194
13.15.3 相机 IO 通信 1197
13.15.4 协议解析 1198
13.15.5 协议组装 1200
第 14 章 常见问题 1203
14.1 单台工控机是否可以安装多套 VM 软件？ 1203
14.2 是否支持加密方案？ 1203
14.3 获取软件使用授权的方式有哪些？ 1203
14.4 安装时是否可选择多种授权方式？ 1203
14.5 能否在单台工控机上使用多个加密狗？ 1203
14.6 能否在单台工控机上使用多套授权码？ 1204

14.7 是否对相机出图有分辨率限制？ 1204
14.8 如何获取 SDK 和算子开发包？ 1204
14.9 软件都需要哪些驱动？ 1204
第 15 章 问题排查 1206
15.1 安装问题排查 1206
15.1.1 安装过程中提示文件在 GAC 目录下不存在 1206
15.1.2 安装过程中提示环境变量配置异常 1207
15.1.3 双击安装包或补丁包无反应，且无进程状态显示 1207
15.1.4 安装过程中提示 MVS 正在运行中 1209
15.2 启动问题排查 1209
15.2.1 启动软件时提示找不到方法 1209
15.2.2 启动软件时提示加密狗未检测到或检测异常 1210
15.2.3 软件安装后启动异常 1212
15.3 使用问题排查 1214
15.3.1 全局相机硬触发时无法触发其他流程 1214
15.3.2 添加全局相机时提示动态库加载失败 1215
15.3.3 拖拽模块提示动态加载库失败 1215

第 1 章 前言

本文档旨在帮助您正确使用产品，避免误操作可能导致的危险或资产损失。使用本产品之前，请阅读本文档并妥善保存，以备日后参考。

PDF 版文档不支持 GIF 动图和视频，因此推荐查阅网页版文档。

1.1 符号约定

对于文档中出现的符号，说明如下所示。

符号	说明
说明	说明类文字，表示对正文的补充和解释。
注意	注意类文字，表示提醒用户一些重要的操作或者防范潜在的伤害和财产损失危险。如果不加避免，有可能造成伤害事故、设备损坏或业务中断。
危险	危险类文字，表示有高度潜在风险，如果不加避免，有可能造成人员伤亡的重大危险。

1.2 相对路径约定

本文档内，针对相对路径描述的约定如下。请根据软件适配的系统类型（Windows 或 Linux），自行判断实际生效的约定。

说明

本章节仅作相对路径约定。软件实际兼容的系统，请参见本文档内的*运行环境*章节。

- 若为适配 Windows 系统的软件，文档内的.***（例如.\Development），均表示相对路径，且相对路径对应的上级目录均为软件安装路径下的“软件名称”文件夹，例如 VisionMaster4.4.0。
- 若为适配 Linux 系统的软件，文档内的./***（例如./Development），均表示相对路径，且相对路径对应的上级目录均为软件安装路径下的“软件名称”文件夹。

1.3 获得支持

若本手册无法解决您的问题，可联系我们获得支持。

- 官网：访问 ***http://www.hikrobotics.com*** 网址查找相关文档或寻求技术服务。
- 热线：拨打 400-989-7998 热线联系技术人员获取帮助。
- 邮件：发送邮件至 tech_support@hikrobotics.com，支持人员会及时回复。
- V 社区：扫描二维码进入 V 社区（***www.v-club.com***），注册/登录后获得服务。



图 1-1 V 社区二维码

第 2 章 发版说明

本章节介绍软件各版本的发布时间和更新情况。

2.1 升级必看

本文介绍与 VM 升级相关的注意事项。

2.1.1 与 MVS 相关的注意事项

若需在同一台工控机（或普通计算机）上安装使用 VM 和 MVS，您需关注两者软件版本是否兼容。

涉及场景

MVS 是海康机器人研发，用于调试工业相机的免费软件。相较于 VM，MVS 提供更加完备的可调试参数，提高成像效果。若您在使用 VM 搭建视觉方案的同时，对成像效果有较高要求，可安装 MVS 进行调试。

不兼容可能导致的问题

VM 和 MVS 均集成工业相机 SDK，若两者版本不兼容，将导致工业相机 SDK 版本冲突，进而导致 SDK 动态库调用相关问题，影响 VM 和 MVS 的正常使用。

安装建议

建议先安装或升级 VM，再根据 VM 版本选用兼容的 MVS 版本。
在 VM 主界面菜单栏选择 *帮助 → 版本信息*，打开*版本信息*窗口，查看与 VM 版本兼容的最低 MVS 版本，如下图所示。

说明

MVS 可在海康机器人官网 *下载中心* 搜索下载。更多该软件相关信息，请参见该软件的用户手册。



图 2-1 版本信息窗口

2.2 V4.4.0 维护版本（2026.05）

该版本为 V4.4.0 的维护版本，发布于 2026 年 5 月。该版本核心聚焦于内存优化，除优化软件及流程运行的内存策略外，重点对高频模块（如 Blob 分析、Blob 标签分析、仿射变换、图像缩放、形态学处理、DL 图像分割、DL 目标检测等）进行了大幅内存优化，显著降低软件、流程和模块运行时的内存占用和耗时。

功能新增

新增 *Blob 分析组合* 模块，此模块在原“Blob 分析”模块基础升级，支持为每个 ROI 独立配置运行参数，满足更复杂的检测场景需求。

功能更新

- 增加对 PLC 各地址数据比对去重功能，有效减少冗余数据传输，提升通信效率。详情请参见 *PLC 类设备寄存器地址参数介绍* 。
- *全局变量*页面新增*详情*按钮。当变量数据类型为组合数据类型(如 BOX，POINT 等)且已绑定模块结果，或为 int、float、string 基本类型但绑定的模块结果为数组时，运行流程后可单击*详情*按钮查看数据详情。详情请参见 *全局/局部变量* 。
- “流程耗时”区域支持查看模块算法运行占用的内存。在“流程耗时”区域单击 ，进入*耗时查看*页面，此页面支持通过**算法内存(KB)**和**程序内存**分别查看当前流程中各模块算法运行占用的内存和 VM 占用内存。

功能优化

- 优化软件、流程运行内存策略，重点对 Blob 分析、仿射变换、图像缩放、形态学处理、DL 图像分割、DL 目标检测等高频模块进行了大幅内存优化，有效降低了运行时的内存占用和耗时，提升整体运行速度与稳定性。
- 升级深度学习推理库 cuDNN 至 V9.7.0 版本, 可更好地发挥 NVIDIA® GeForce® RTX™ 50 系列显卡的性能。

手册优化

新增 *二次开发方式*，帮助用户在进行二次开发时快速选择合适的开发工具。

2.3 V4.4.0 维护版本（2025.11）

该版本为 V4.4.0 的维护版本，发布于 2025 年 11 月。该版本主要新增了边缘学习模块，并对流程配置、软件设置、运行界面以及深度学习模块支持的硬件设备等进行了更新。

硬件兼容更新

更新 *深度学习* 模块的显卡配置要求和支持的硬件设备，新增支持 NVIDIA® GeForce® RTX™ 50 系列显卡和 AI 推理终端。具体显卡型号详情及 AI 推理终端相关的使用说明，请参见 *运行环境* 。

功能新增

- 新增 *学习计数* 、 *有无检测* 、 *异常分类* 和 *多类别分类* 等边缘学习模块。该类模块通过学习少量注册模板数据，可快速适配不同的场景下的检测需求，实现检测效率与准确率的显著提升。
- 图像处理模块下新增 *掩膜工具* 模块，可助您快速完成图像掩膜运算。
- 新增支持 AB EtherNet/IP CIP 协议，可实现与罗克韦尔 PLC 的通信。更多详情，请参见 *通信设备管理* 。
- *脚本* 模块界面新增查询接口，支持查看各数据类型的接口信息。

- *存图大数据解压工具* 新增批量解压功能，可通过导入文件夹批量解压多个.DAT 文件。导入文件后，还可快速预览压缩包中的图像。
- *快捷工具条* 新增*清空结果*按钮，可一键清空当前方案内的模块状态、历史模块结果及图像渲染信息。

功能更新

- 全局相机配置中新增**插值算法类型**和**分时频闪拆图**参数，可设置像素转换时使用的图像插值算法以及是否开启分屏采集图像。更多详情，请参见 *常用参数* 。
- *软件设置* 中新增**执行前清空缓存**参数，可设置流程在每次执行前自动清空一次历史结果数据。
- *运行策略* 设置中新增**全选**选项，支持在配置自定义 CPU 运行策略时一键全选或取消全选 CPU 选项。
- 流程配置中的显示设置新增*显示*选项，勾选后可显示指定图形。更多配置操作详情，请参见 *配置渲染效果与通信* 。
- *子界面* 控件编辑窗口中的基础设置参数新增 **X 方向的距离**、**Y 方向的距离**和**尺寸**参数，支持按照设置的尺寸和位置显示该控件。
- 更新 *软件设置* 中的启动关闭设置选项，当开启**自动加载指定方案**或**自动加载最近打开方案**后，方案打开时的启动状态仅支持选择**非连续执行**或**连续执行**，不再支持**静默执行**选项。您可以前往 *软件设置* → *高级设置* 中设置该参数。
- 更新 *软件设置* 中的**模块最大数量**参数范围。单个方案中允许拖入的模块数量上限扩充至 4096 个，您可以根据实际检测需求构建更加复杂的方案。
- 更新 *异常检测* 模块在初定位时使用的模板匹配算法，由 *高精度匹配* 改为 *轮廓匹配* 。具体配置操作请参见 *直接创建缺陷模型* 。

2.4 V4.4.0 维护版本（2025.04）

该版本为 V4.4.0 的维护版本，发布于 2025 年 4 月。该版本主要新增了激活工具并优化了控制器管理功能。

功能新增

- 新增 *激活工具* ，可用于自动激活当前工具所在目录对应的 VM 版本。当 PC 上同时安装 VM 的多个版本时，可使用该工具切换对应的 VM 版本。
- *控制器管理* 新增支持 VC5000 视觉控制器，支持配置 I/O 输入、输出及光源参数。

功能更新

逻辑工具下的 *文本保存* 模块新增**异步保存**选项，可异步保存文本到指定位置，以解决保存操作耗时长的问题。

问题修复

修复若干已知问题，提升 VM 运行的稳定性。

用户手册优化

- 新增 *快速入门*，帮助新手快速了解 VM 的关键概念；并通过引导新手体验示例方案、搭建入门级方案，帮助新手快速上手方案搭建的基础操作和流程。
- 整体优化 *方案搭建-流程配置*，将原有内容拆分为 *配置单流程运行逻辑* 、*配置多流程运行逻辑* 和 *其他操作与配置*，方便您根据流程配置场景查阅相应内容。
- *产品概述* 下新增 *授权方式*，提供 VM 授权方式（本机加密狗、远程加密狗和软加密）的概览性规格信息，助您合理进行授权方式选型。
- 对多个模块（如 *采集* 下各模块、 *定位* 下的模板匹配模块、 *识别* 和 *深度学习* 下的深度学习模块）的使用参考进行了优化并提供更全面的使用说明，帮忙您快速掌握模块的使用方法。

2.5 V4.4.0 维护版本（2024.08）

该版本为 V4.4.0 的维护版本，发布于 2024 年 8 月。该版本主要基于 4.4.0 版本，对编辑运行界面、部分模块、通信管理等功能进行新增、更新或优化。

- *功能新增*
- *功能更新*
- *功能优化*
- *功能废弃*
- *用户手册调整*

功能新增

编辑运行界面

新增配置引导窗口，在安装软件后首次打开编辑运行界面时，可通过配置指引快速搭建运行界面。



图 2-2 运行界面搭建引导

模块

- *轮廓匹配* 模块新增**是否输出轮廓**参数，可控制轮廓是否输出。同时模块结果中新增**横切值**，代表匹配对象 Y 轴向 X 轴的旋转角度。
- *间距检测* 模块新增**查找方向**参数，可指定检测时的查找方向。
- *DL 字符定位* 模块新增**文本对比**功能，用于筛选特定文本。
- 注册学习分类新增 *全监督检测* 模块，该模块相比注册检测模块有更高的检出率和准确率。
- *DL 无监督分割* 模块新增**获取模型 ROI** 参数。开启后，模块可自动获取模型文件中的 ROI 信息。无需在基本参数中重新配置 *ROI 区域*。
- *直线拟合* 模块的结果显示中新增**贯穿线**显示功能。
- *Group* 模块的*输出设置*和*显示设置*新增**累积输出**功能，将循环过程中输出的数据累积打包，最后一次性输出。

  说明

  可替代*数据集合*模块完成数据收集。

通信管理

控制器管理 新增控制器品牌，具体为 HKR 光源、VC2100 控制器、VT2000 控制器和 VT3000 控制器。

其他

- *帮助* 新增在线帮助文档和 V 社区的跳转。

  说明

  该功能需安装软件的工控机支持访问外网，否则无法成功跳转。
- 软件的各功能模块新增通过窗口左上角的 打开文档对应章节功能。

功能更新

编辑运行界面

个别控件新增部分参数，同时补充自带的运行界面模板。

- 设计模板中自带多个运行界面模板可供选择，您可根据实际需求选择适合的模板进行使用或调整。
- *多图像*控件新增**保留上次结果**功能。
- *按钮*控件选择**执行**时，新增对单个流程或全流程的**开启连续运行**和**关闭连续运行**功能。
- *参数组合*、*文本框组合*控件配置的数据源，增加上下调整顺序功能。
- *表格*控件新增**标题文本行**、**标题行**、**行背景**的颜色设置。

文件

*导出算子代码*功能新增自定义选择需导出的流程，并支持导出模块的 ROI 配置（需自定义选择导出哪些模块的 ROI 配置）。



图 2-3 导出算子代码

设置

- *软件设置* 新增**模块间距**参数，用于定义模块自动连接时的间距。
- *软件设置* 新增 **GPU 锁频开启**参数，用于设置 GPU 的工作频率，稳定 GPU 的耗时波动，降低 GPU 的运行耗时。

工具

参数查找工具 选中模块的运行参数时，除已显示软件中的调用方式外，新增算子开发包中调用方式的显示。同时增加运行参数在界面上显示的名称。

通信管理

- 网关 PLC 设备的地址个数上限由 60 个调整为 500 个。
- *响应配置* 的**方案加载**、**流程控制**和**相机响应配置**功能新增选择 PLC 设备时可选择**数据类型**。同时在一定条件下，可设置**结束符**。

模块

*脚本*模块新增 *BytesToPointset* 接口（将点集二进制数据转换为轮廓点数组数据）和 *PointsetToBytes* 接口（将轮廓点数组数据转换为点集二进制数据）。

功能优化

编辑运行界面

优化部分控件的交互、视觉效果、默认样式和配置逻辑。

- 优化保存模板功能，仅**名称**为必填项，其他参数选填。
- 优化*图像*和*多图像*控件的**显示辅助线**功能。仅在预览和运行时可查看效果，且未运行时可调整辅助线中心点的位置。
- 优化*表格*、*生产统计*、*文本框组合*、*参数组合*和*界面日志*控件的视觉效果。
- *线图阵列*、*饼图*、*文本框组合*和*参数组合*控件增加默认样式（之前为空白）。

设置

采集策略 增加关于该功能的使用场景和注意事项。

局部变量

针对流程和 Group 模块的局部变量，均称之为局部变量，容易引起误解。此次将两者名称进行区分，分别为*流程局部变量*和 *Group 局部变量*。

相机管理

- 调整 *相机管理* 中设备列表下拉选项的名称，分别为*全局相机*、*HKR 采集卡*、*Dalsa 采集卡*和 *Matrox 采集卡*。
- 全局相机中 *常用参数* 的**取流使能**参数名称修改为**开始采集**，并移动到**触发设置**参数列表下。

通信管理

- 优化 *控制器管理* 新增设备时，设备管理中需设置的参数项名称和布局。
- 优化 *控制器管理* 中已有控制器品牌的名称，并重新排序。
- 接收事件中 *字节匹配-协议比对* 的规则列表，所有**类型**均支持**更改为**功能。

模块

- *快速匹配*模块的运行参数**轮廓使能**名称调整为**是否获取轮廓**。
- 调整*卡尺工具*模块的**投影方向**参数下拉选项的名称，但功能无变化。
- 将 *ML 分类*模块部分与训练相关的运行参数调整至分类训练中。
- 调整 *图像拼接* 模块中**首图清空**参数的位置，由运行参数 Tab 页移至基本参数 Tab 页。
- 所有深度学习和注册学习模块：

> **说明**
>
> 涉及的模块为：*DL 字符定位*、*DL 字符识别*、*DL 单字符检测*、*DL 快速图像分割*、*DL 图像分割*、*DL 实例分割*、*DL 分类*、*DL 图像检索*、*DL 无监督分类*、*DL 无监督分割*、*注册分类*、*注册检测*、*注册分割*。

  - 运行参数 Tab 页的参数划分为**模型参数**、**运行模式**、**算法参数**、**过滤参数**等。
  - 优化模块在 NVIDIA® GeForce® RTX™ 40 系列显卡下的处理性能。
  - 优化模块在 CPU 以及 GPU 下的算法推理效率及资源占用。
- *直线拟合*模块的**角度使能**参数，名称调整为**角度拟合约束**。开启后，需设置的参数分别为**期望直线角度**和**直线角度容忍范围**。参数对应功能无变化。
- 优化 *脚本* 和 *Python 脚本* 模块的加密功能。使用加密功能时，不再针对整个模块做加密，仅针对编辑区的代码做加密。

其他

- 优化软件打开的启动速度。
- 完善软加密失败时提示窗口中的信息。

功能废弃

*工具*中去除 *自定义模块生成* 工具。如需使用，可到软件安装路径下（..\Applications\Tools\AlgorithmXMLGenerator）获取。

用户手册调整

- 重构 *数据队列* 及其 *使用示例* 章节的内容。
- *光源* 模块章节修改参数配置和模块结果的内容，之前版本存在错误。
- 重构*快速匹配*、*高精度匹配*、*轮廓匹配*、*灰度匹配*中参数配置的相关介绍，具体参见 *轮廓匹配* 章节。
- 优化 *条码识别* 和 *二维码识别* 模块章节，补充模块原理相关内容。
- *注册检测* 、 *注册分割* 、 *DL 无监督分类* 、 *Python 脚本* 、 *数据记录* 、 *数据筛选* 、 *数据分类* 、 *数据排序* 模块章节补充使用方法相关介绍。
- 修改 *脚本* 模块的作用和使用限制，之前版本存在错误。*脚本*模块只能对流程内的数据进行处理，无法控制流程的执行，但*全局脚本*可以控制流程的执行。
- 重构常见问题章节，分为 *常见问题* 和 *问题排查* 两部分，其中问题排查分为 *安装问题排查* 、 *启动问题排查* 和 *使用问题排查* 。同时结合在线文档的反馈和 V 社区的问答增加部分常见问题和问题排查。

2.6 V4.4.0 维护版本之前

软件 V4.4.0 维护版本之前的版本更新详情如下。

V4.4.0

该版本发布于 2024 年 1 月。

- 功能新增：
  - *运行环境* ：
    - 软件兼容 Windows11（64 位）操作系统。
    - 软件新增可适配的 40 系列显卡。
  - 菜单栏：
    - 菜单栏的设置新增 *采集策略*，同时 *软件设置* 和 *方案设置* 新增部分参数。
    - 菜单栏的 *文件* 新增导出算子代码功能，提供算子代码开发示例，仅型号中带 6210 或 7120 的加密狗支持。
    - 菜单栏的工具新增 *参数查找工具*，方便查看模块输入、输出和运行参数（可查看 VM 二次开发和算子二次开发的参数）。仅型号中带 6210 或 7120 的加密狗支持。
  - 新增 *存图大数据解压工具*，可将**大文件存图**生成的 dat 文件解压得到 *输出图像* 模块存储的图像。

    说明

    更多**大文件存图**相关详情，请参见 *采集策略* 。
  - 快捷工具条新增方案级别的 *综合配置* 界面。
  - *流程编辑区域* 新增搜索和注释等功能。
  - 所有模块及流程的模块结果新增“耗时（ms）”输出。
  - 新增流程固定间隔时间数据外发的功能。
  - 新增基于流程和 Group 模块的 *局部变量* 。
  - 相机管理新增 *自研采集卡*，同时 *全局相机* 新增相机参数配置控件和部分参数。
  - 通信管理：
    - *通信设备* 新增三菱 FX 设备的支持，同时部分已支持通信设备新增部分参数。
    - 新增针对整个 *通信管理* 的导入/导出功能。
  - 模块箱：
    - 标定类新增 *坐标系* 模块。
    - 模块箱新增注册学习类，该类中新增 *注册检测* 和 *注册分割* 模块。
    - 深度学习类新增 *DL 无监督分类* 模块。
    - 颜色处理类新增 *颜色分割* 和 *彩图生成* 模块。
    - 逻辑工具类新增 *Python 脚本* 、 *数据记录* 、 *数据筛选* 、 *数据分类* 、 *数据排序* 模块。
- 功能优化：
  - 订阅类参数新增浮窗订阅功能，相关介绍参见 *运行参数* 。
  - 模块屏蔽区新增订阅功能，且订阅的屏蔽区支持位置修正。
  - 优化锁定流程编辑区域的功能，相关介绍参见 *快捷工具条* 。
  - 运行界面：
    - 整体重构，优化交互和视觉 。
    - 增加布局功能，控件置于布局之中，改善运行界面放大，缩小时控件对不齐的问题。
    - 新增生产统计、折线图、饼图等新控件，可结合普通模块使用。例如结合数据记录模块使用时，可保持数据显示不重新刷新。
  - *通信管理* ：

› 将控制器管理整合到通信管理中，设备管理分为 ***通信设备*** 和 ***控制器类设备*** 。

说明

若加载的老版本方案有使用控制器功能，可正常加载，支持在此处进行设置。

› 通信管理的 ***接收事件*** 和 ***发送事件*** 支持选择控制器类设备，同时均可查看数据结果和状态。

› 将全局触发功能同步嵌入到通信管理的触发配置。

○ 流程配置：

› 优化 ***流程配置*** 功能，新增通信配置、图像显示和输出结果。

› 流程配置界面中 ROI 配置类的运行参数由弹出改为内嵌形式。

○ ***全局变量*** 新增支持的数据类型，且支持数组。

○ 模块通用功能：部分模块的基本参数新增可同时绘制多个 ROI 的功能。涉及的模块为：DL 字符识别、直线边缘缺陷检测、直线对边缘缺陷检测、圆弧边缘缺陷检测、圆弧对边缘缺陷检测、边缘对缺陷组合、DL 注册分类、DL 分类、DL 快速图像分割、DL 图像分割、DL 图像检索、DL 无监督分割。

○ 采集类模块：

› ***图像源*** 模块新增订阅图片文件夹相关功能，可根据文件夹更新图像列表。

› ***输出图像*** 模块优化部分基本参数。

○ 定位类模块：

› ***灰度匹配*** 和***轮廓匹配***模块优化全模板搜索功能。

› ***轮廓匹配***新增掩膜图建模功能。

› 优化模板匹配相关模块的模型配置。

› ***Blob 分析*** 模块优化运行参数和模块结果。

› ***直线查找*** 和 ***多直线查找*** 模块新增部分运行参数。

› ***卡尺工具*** 新增扇形 ROI，支持扇弧段卡尺。

○ 测量类模块：***线圆测量*** 、 ***圆圆测量*** 、 ***点圆测量*** 、 ***点线测量*** 、 ***线线测量*** 、 ***点点测量*** 模块优化基本参数和运行参数。

○ 识别类模块：

› ***DL 字符定位*** 和 ***DL 字符识别*** 模块优化运行参数。

› ***ML 分类*** 模块新增分类训练，优化运行参数。

○ 缺陷检测类模块：

› ***表面缺陷滤波*** 模块优化运行参数，新增滤波模式参数。

› ***字符缺陷检测*** 模块优化字符模板的训练方式。

› ***异常检测*** 模块的缺陷模型分为直接创建和继承两种方式。

○ 注册学习类模块：

› 注册分类模块从深度学习移动到注册学习下。

› 注册分类的 ***注册类别*** 处增加任务配置相关参数。

○ 深度学习类模块：

› ***DL 实例分割*** 和 ***DL 目标检测*** 模块优化运行参数。

› 深度学习的 CPU 模块支持 OpenVINO 推理加速。

- 标定类模块：*N 图像标定*、*标定板标定*、*映射标定*、*畸变标定* 模块优化运行参数。
- 图像处理类模块：
  - *图像二值化* 模块优化运行参数，新增 ROI 外处理参数。
  - *图像拼接* 模块优化拼接模型和运行参数。
- 颜色处理类模块：*颜色转换* 模块优化运行参数，新增新的转换比例。
- 拆分组合类模块：*Box 融合* 模块优化运行参数，新增新的融合依据。
- 图形生成类模块：*圆拟合*、*椭圆拟合* 和 *直线拟合* 模块优化基本参数或运行参数。
- 逻辑类模块：
  - *脚本* 模块的输入输出变量增加组合类型和数组，同时增加渲染控件和模块显示配置。
  - *耗时统计* 模块优化处理逻辑和操作方法，且支持同时统计多个模块间的耗时。
  - *Group* 模块新增全局变量，优化参数配置，同时循环效率和循环结果输出方式。
- 通信类模块：*发送数据* 模块优化基本参数。

- 功能调整：
  - 加密狗：
    - 新增部分型号加密狗，可打开 4.4 版本软件。部分功能仅该型号加密狗支持。涉及型号如下：
      - 本机加密狗：MVS-VM-6210、iMVS-VM-7210。
      - 授权码：iMVS-VMS-6210、iMVS-VMS-7210。
      - 远程加密狗：iMVS-VMNET-6210、iMVS-VMNET-7210。
    - 删除部分老版本加密狗，不再支持 4.4 版本软件。涉及型号如下：
      - 本机加密狗：iMVS-VM-1100、iMVS-VM-2100、iMVS-VM-4100、iMVS-VM-6100、iMVS-VM-7100。
      - 授权码：iMVS-VMS-1100、iMVS-VMS-2100、iMVS-VMS-4100、iMVS-VMS-6100、iMVS-VMS-7100。
      - 远程加密狗：iMVS-VMNET-6100、iMVS-VMNET-7100。
  - 调整部分模块的形状，可直观的判断模块的类型，如下图所示。



图 2-4 模块形状概览

说明

因文档中截图数量较多，仅在对应模块章节对截图进行更新，其余截图未做调整。

- 功能删除：
  - 删除 *DL 异常检测*模块，相关功能合并到 *DL 无监督分割* 模块中。4.4 版本软件仍支持加载含该模块的方案，但不可设置。该模块的模型可在 *DL 无监督分割*模块中使用。
  - 删除可切换英文界面的功能。
- 文档调整：
  - 新增 *术语介绍*，对软件及文档出现的部分专业词汇进行介绍。
  - 新增 *设计运行界面*，对如何搭建运行界面进行介绍。
  - 新增 *了解方案构成* 章节，对方案的整体构成做简单介绍。
  - 新增 *字节匹配-协议比对的控制器示例*，说明如何通过试过视觉控制器的 IO 输入触发流程执行。
  - 将所有模块的模块结果调整到 *模块使用参考* 的各个模块章节下。
  - 定位类模块： *边缘交点* 、 *四边形查找* 和 *平行线查找* 模块补充模块原理， *目标跟踪* 模块完善模块原理。
  - *标定* 补充如何选择各个标定模块的介绍。
  - 图像处理类模块： *畸变校正* 模块补充模块原理。
  - 拆分组合类模块：优化 *Box 融合* 模块的模块原理，修改 *Box 重叠率计算* 模块中模块原理的错误描述。

V4.3.0 维护版本（2023.06）

该版本为 V4.3.0 的维护版本，发布于 2023 年 6 月。

- 功能新增：
  - 新增 *远程加密狗*。
  - 定位分类新增 *阵列圆查找* 模块。
  - 识别分类新增 *ML 分类* 模块。
  - 深度学习分类新增 *DL 快速图像分割* 和 *注册分类* 模块。
- 功能优化：
  - 优化 *软加密* 功能。
  - *软件设置* 新增自动加载最近打开方案功能。
  - *相机管理* 中选择全局相机时，增加触发模式参数。
  - *控制器管理* 中新增 VB2230，同时部分控制器新增触发配置参数。
  - 运行界面的界面配置新增条件信号灯控件，运行控制控件新增单次运行/连续运行的切换，按钮控件新增保存方式的功能。
  - *数据队列* 的触发流程新增异步触发功能。
  - 优化 *Blob 标签分析* 模块的运行参数。
  - *线圆测量*、*线线测量*、*圆圆测量*、*点圆测量*、*点线测量*、*点点测量* 模块运行参数中新增输出角度范围。
  - 优化 *像素统计* 模块的运行参数。
  - *图像运算* 模块优化运算类型功能。
  - *畸变校正* 模块新增对彩色图像的校正。
  - *圆弧对缺陷检测* 模块优化运行参数。
  - *脚本* 模块新增部分接口。
  - *图形收集* 模块新增订阅的目标状态为空或未订阅时，直接输出订阅区域的功能。
- 文档调整：
  - 重构整个文档的框架。
  - 各模块补充模块原理、使用方法和模块结果，并完善参数释义。

V4.3.0

该版本发布于 2022 年 12 月 29 日。

- 功能新增：
  - *主界面* 新增日志栏。
  - 定位分类新增 *轮廓匹配*、*椭圆查找* 模块。
  - 图像生成分类新增 *椭圆拟合* 模块。
  - 图像处理分类新增 *逆仿射变换* 模块。
  - 新增通过 *主界面* 的运行模式切换至运行界面。
- 功能优化：
  - 支持打开包含当前软件中不存在的模块的方案。
  - *软件设置*中新增 *修改参数自执行* 功能。

- 图像显示区域可显示双画面。
- 流程和 *Group* 新增运行参数功能。
- *全局脚本* 支持 VM4.X 版本 C#二次开发平台接口调用。
- *脚本* / *全局脚本* 调整附加调试的进程。
- *目标跟踪* 模块新增轨迹重叠率参数。
- *标定板标定* 模块增加不同类型标定板的介绍。
- *旋转标定* 模块新增反向纠正参数。
- *单点抓取* 和 *单点纠偏* 模块新增标定矩阵和旋转拍照使能参数。
- *单点映射对位* 模块新增 N 点标定矩阵、映射标定矩阵和 N 点标定文件参数。
- *Box 融合* 模块删除 Box 标签和优先级排序参数，调整 box 融合类型参数项名称。
- *字符缺陷检测* 模块中字符精定位新增阈值类型和边缘阈值参数。

- 文档调整：
  - 新增 *版本切换工具* 和 *异常信息收集* 的介绍。
  - 将快速匹配、高精度匹配、轮廓匹配以及灰度匹配整合为 *模板匹配* 章节，并补充应用场景相关介绍。
  - *划片拆分模块* 、 *二维阵列* 、 *多标签筛选* 、 *Box 重叠* 、 *Box 过滤* 模块新增模块原理。
  - *表面缺陷滤波* 、 *边缘位置趋势分析* 、 *边缘对位置趋势分析* 模块新增模块原理及应用，并补充部分运行参数释义。

V4.2.0 维护版本（2022.06）

该版本为 V4.2.0 的维护版本，发布于 2022 年 6 月。

- 功能新增：
  - 标定分类新增 *平移旋转标定* 模块、 *旋转标定* 模块。
  - 运算分类新增 *单点抓取* 模块、 *单点映射对位* 模块、 *单点纠偏* 模块。
  - 缺陷检测分类新增 *表面缺陷滤波* 模块。
- 功能优化：
  - *相机管理* 新增对 Dasla 图像采集卡的支持。
  - *图像源* 模块新增 SN 初始值和拼接使能相关参数。
  - *灰度匹配* 模块新增部分模板配置参数和运行参数。
  - *BLOB 分析* 模块新增部分运行参数。
  - *路径提取* 模块的模板配置新增清空基准的功能。
  - *标定转换* 模块新增位姿位和运行位坐标信息输入。
  - *单位转换* 模块新增像素当量修正参数。
  - *Box 融合* 模块新增部分运行参数。
  - *边缘模型缺陷检测* 模块的模型配置新增清空基准点功能。

V4.2.0

该版本发布于 2022 年 3 月 24 日。

- 功能新增：

  - 支持写入密码进行双重加密。
  - 新增 *拆分组合* 分类，包含 *划片拆分* 、 *二维阵列* 、 *多标签筛选* 、 *Box 融合* 、 *Box 重叠* 和 *Box 过滤* 模块。
  - 深度学习分类新增 *DL 无监督* 模块。
  - 运算分类新增 *坐标转换* 模块。
  - 定位分类新增 *目标跟踪* 模块。
  - 缺陷检测分类新增 *边缘位置趋势分析* 和 *边缘对位置趋势分析* 模块。
  - 逻辑工具分类新增 *条件分支* 、 *触发模块* 和 *图形收集* 模块。
- 功能优化：
  - 新增流程超时时间设置。
  - *权限设置* 新增导入/导出功能。
  - *软件设置* 新增撤销重做设置功能。
  - *运行策略* 的策略模式新增耗时稳定选项。
  - 图像预览窗口优化十字辅助线功能。
  - 流程设置中新增输入设置，且输入、输出和显示设置均支持多选。
  - 优化 *相机管理* 功能。
  - *控制器管理* 新增自动重连功能。
  - 优化 *接收事件* 选择文本和字节匹配时的相关功能。
  - 运行界面导出程序新增工程类型参数。
  - *数据队列* 中，新增触发流程功能，且队列设置新增 IMAGE 类型数据。
  - *图像源* 模块选择本地图像时，新增显示图像名称参数。
  - *输出图像* 模块增加同步存储、启用 FTP 等功能。
  - *图像运算* 模块新增部分运算类型。
  - *仿射变换* 模块新增仿射变换类型，并增加锁定输出尺寸参数。
  - *图像拼接* 模块优化基本参数和运行参数。
  - *相机 IO 通信* 模块删除有效电平参数，调整持续时间参数。
- 文档调整：
  - 新增对加密方式的详细介绍。
  - 重构 *软加密型号和操作方式* 。

V4.1.0

该版本发布于 2021 年 10 月 11 日。

- 功能新增：
  - 工具新增 *自定义模块生成工具* 。
  - 定位分类新增 *角平分线查找* 、 *中线查找* 、 *平行线计算* 和 *垂线查找* 模块。
  - 深度学习分类新增 CPU 和 GPU 版本的 *DL 实例分割* 模块。
  - 标定分类新增 *标定加载* 模块。
  - 运算分类新增 *旋转计算* 模块。
- 功能优化：

- ***权限设置*** 增加操作员使能功能， ***方案设置*** 增加自动保存设置功能。
- 创建一键标定引导中 ***动态标定*** 的通信信号新增示教信号。
- ***控制器管理*** 的 IO 参数新增延迟时间。
- ***全局变量*** 新增位置调整和导入导出功能。
- 通信管理的 ***设备管理*** 配置 PLC 相关参数时，新增数据上传功能。
- 通信管理的 ***响应配置*** 由统一的通信设备控制多个功能，改为各个功能单独选择通信设备。
- 全局触发可通过执行模块动作清空数据队列。
- 运行界面新增或优化部分功能。
- 调整模块的分类。
- ***输出图像*** 模块分别新增渲染图和原图命名方式。
- ***缓存图像*** 模块输出图像上限调整为 15。
- ***特征匹配*** 增加是否模板存图的选项。
- ***直线查找*** 模块新增接近中线的边缘类型，增加角度归一化和查找方向反向功能。
- ***路径提取*** 支持绘制多条路径。
- ***几何创建*** 模块可同时创建多个图形，同时图形形状增加矩形。
- ***DL 读码*** 模块新增高性能模式运行参数，同时增加部分一维码码制。
- ***DL 字符定位*** 模块增加最小平均分数参数。
- ***DL 单字符识别*** 新增每个识别框可单独自定义字符过滤信息功能。
- ***N 点标定*** 、 ***映射标定*** 和 ***N 图像标定*** 模块新增部分参数。
- ***标定转换*** 模块支持 xml 格式的标定文件。
- ***变量计算*** 的输出类型新增对 POINT 的支持。
- ***图像运算*** 模块增加图像权重参数。
- 调整 ***字符缺陷检测*** 模块。
- ***文本保存*** 模块新增实时存储功能。
- ***格式化*** 增加数组间分隔符和清空功能。
- ***脚本*** 模块可直接使用定义的变量名，无需内部 get 或 set，且订阅全局变量时可显示实时数值。
- ***协议解析*** 新增字节解析功能。

- 文档调整：
  - 调整 ***运行环境*** 及 ***加密狗*** 章节内容。

V4.0.0

该版本发布于 2021 年 5 月 28 日。

- 功能新增：
  - 新增每个模块运行状态的显示，NG 为红色，OK 为绿色。
  - 新增方案的文件路径显示。
  - 新增流程耗时统计功能。
  - 新增全局触发功能。
  - 图像采集分类新增 ***多图采集*** 模块。

- 深度学习新增 *DL 图像分割 CPU* 模块、DL 异常检测模块。
- 图像处理分类新增 *多图融合* 模块。
- 缺陷检测分类新增缺陷对比模块。

- 功能优化：
  - 优化方案加载时间。
  - 支持连接 1.5 亿像素相机。
  - 支持 CTRL+C/V 实现模块的复制粘贴。
  - 撤销/重做支持流程配置区的模块生成、连线、订阅、参数配置、模型配置、导入/导出、方案加载生效。
  - 流程和 *Group* 新增全局显示，删除前项显示。
  - *结果显示区域* 新增帮助板块，解释相关参数的含义。
  - 优化 *一键标定工具* 易用性。
  - 更新 *标定板生成工具*，支持生成更大码尺寸的 I 型标定板，支持生成 CAD 文件。
  - 重构 *通信* 相关功能。
  - 运行界面新增独立启动时最大化设置，并新增多图像和信号灯控件。
  - *输出图像* 模块支持存储 MONO8 和 RGB 两种格式图像，支持同时存储原图和结果图，支持 OK 和 NG 路径单独配置。
  - *特征匹配* 模块优化模板匹配算法。
  - *DL 读码* 模块新增条码评级功能。
  - *N 图像标定* 模块更新相关参数。
  - *仿射变换* 模块可输出固定大小图片。
  - *图像矫正* 模块支持 RGB 格式。
  - *文本保存* 模块支持列名选择保存。
  - 重构 *Group* 模块。
- 文档调整：
  - 修改文档整体框架，并新增加密狗相关介绍。

V3.4.0

该版本发布于 2020 年 12 月 4 日。

- 功能新增：
  - 方案新增流程导入和导出功能。
  - 快捷工具栏新增 *一键标定引导* 功能。
  - 新增通信触发功能。
  - 定位分类新增 *图形定位* 和 *BLOB 标签分析* 模块。
  - 识别分类新增二维码等级模块。
  - 深度学习新增 *DL 字符识别 CPU* 模块、 *DL 读码* 和 *DL 图像检索* 。
  - 缺陷检测分类新增 *路径提取* 模块。
  - 逻辑工具分类新增分支结束模块。
  - 新增流程 NG 时，停止方案运行功能。
- 功能优化：

- 优化方案加载时间。
- 优化读码算法库。
- *发送数据*、*接收数据*、*协议解析*、*协议组装*、*脚本*模块支持二进制数据包。
- 三角函数计算方式调整为使用角度计算。
- 修改圆形 ROI 及直线 ROI 的形态，圆形 ROI 新增继承模式。
- *全局变量*支持绑定多个。
- IO 与光源设备合并成为 *控制器管理*。
- *全局脚本*以及 *脚本*，新增程序集动态添加和使用 VS 调试功能。
- *图像源*模块新增运行完最后一张图像时方案停止功能。
- *BLOB 分析*模块新增粘贴过滤功能。
- *卡尺工具*、*边缘查找*、*间距检测*模块新增结果排序功能。
- *位置修正*模块新增尺度属性。
- *线圆测量*模块新增绘制 ROI 功能。
- *DL 图像分割*模块新增分类功能。
- *标定转换*模块支持反方向转换。
- *颜色抽取*模块新增总面积输出以及输出总面积判断。
- 优化 *字符缺陷检测*模块。
- 优化*格式化*模块配置界面。
- 修改 *Group* 模块，增加输出选择添加功能、显示配置，并支持复制/粘贴、导入/导出。

V3.3.0

该版本发布于 2020 年 6 月 10 日。

- 功能新增：
  - 新增撤销重做按钮，最大支持保存 10 步操作，便于更好的调试编辑方案。
  - 图像采集模块新增 *光源*模块，实现对光源的控制。
  - 深度学习分类新增 *DL 单字符识别*模块。
  - 标定分类新增*映射标定*和 *N 图像标定*模块。
  - 图像处理分类新增 *几何变换*和*图像拼接*模块。
  - 逻辑工具分类新增 *数据集合*模块（用于循环内数据收集，以数组形式输出）、*协议解析*和*协议组装*模块。
  - 移除原循环模块，循环功能集成至 *Group*，同时 Group 右键可添加为自定义工具，便于方案复制实现。
- 功能优化：
  - 优化保存和加载速度，支持打开静默执行一次，支持模块错误能够打开方案并提示错误信息。
  - 所有模块支持右键重置恢复默认参数，同时支持右键复制参数粘贴至相同模块。
  - 支持设置锁定模式，锁定编辑。
  - 无需依赖 net3.5。
  - 模块并行运行时，支持配置多线程，可合理分配 CPU。
  - *全局变量*支持绑定参数和结果输出，支持全局绑定无需通过接收发送。

- ○ *PLC 通信* 新增基恩士和汇川设备的支持。
- ○ 全局触发支持触发指定流程。
- ○ *图像源*模块融合相机图像和本地图像。
- ○ *定位类* 模块支持查找四边形、查找多条直线。
- ○ *Blob 分析* 模块支持多 ROI 并优化部分参数。
- ○ 深度学习训练工具支持云服务器训练，支持统计训练新增样本，并优化部分参数。
- ○ *DL 图像分割* 模块支持对比模式。
- ○ *N 点标定* 模块支持自动生成标定文件，可在模块界面清空标定点。
- ○ *标定转换*模块支持标定文件更新后自动加载。
- ○ *颜色抽取*模块支持多范围抽取，支持 ROI 范围值编辑设置和反向抽取。
- ○ *边缘对模型缺陷检测* 模块支持自动生成轨迹。
- ○ *文本保存* 模块新增 CSV 格式保存、支持按照日期生成目录、设置最大保存天数。
- ○ *直线查找* 和*圆查找*等模块增加实际卡尺显示。
- ○ 优化*格式化*模块的格式配置。

V3.2.0

该版本发布于 2019 年 10 月 10 日。

- 功能新增：
  - ○ 新增 *软加密方式* 。
  - ○ 新增 *全局脚本* ，可进行流程运行、变量调节的脚本控制。
  - ○ *图像源* 模块新增相机 IO 控制和智能相机光源控制功能。
  - ○ 图像采集分类新增*缓存图像*模块，便于方案的追溯与调试。
  - ○ 定位分类新增 *灰度匹配* 模块。
  - ○ 图像处理分类新增 *图像矫正* 模块。
  - ○ 颜色处理分类新增*颜色识别*模块。
  - ○ 流程编辑区域新增鹰眼功能。
  - ○ 流程新增复制/粘贴功能。
  - ○ 新增输出配置功能。
  - ○ 新增模块禁用功能，右击模块可禁用模块，方案运行时直接跳过该模块，再次右击可重新启用。
- 功能优化：
  - ○ TCP 客户端支持自动重连功能。
  - ○ *控制器管理*处调整 IO 为全局通信，支持 VB2000 系列 VC4000 系列视觉控制器。
  - ○ *图像源*模块新增相机 IO 控制和智能相机光源控制功能。
  - ○ 存储图像模块支持文字与界面倍率调节。
  - ○ 优化深度学习相关模块的参数调节方式。
  - ○ *标定板标定*模块新增海康标定板，可根据实际需求选择不同类型的标定板。
  - ○ *仿射变换*与 *拷贝填充* 模块支持 RGB 和灰度图格式。
  - ○ 更改 *Group* 操作方式。
  - ○ 循环模块支持循环跳出，且删除循环时保留循环内的模块。

V3.1.0 维护版本（2019.07）

该版本为 V3.1.0 的维护版本，发布于 2019 年 7 月。

- 功能优化：
  - 新增软件对 RTX 系列显卡和 GTX1660 显卡的支持。
- 缺陷修复：
  - 修复已知缺陷。

V3.1.0

该版本发布于 2019 年 5 月 24 日。

- 功能新增：
  - 缺陷检测分类新增 *圆弧缺陷检测*、*直线缺陷检测*、*圆弧对缺陷检测*、*直线对缺陷检测*、*边缘组合缺陷检测* 以及*边缘模型缺陷检测*模块。
- 功能优化：
  - 调整深度学习训练工具的位置到工具栏中。
  - *相机图像* 模块支持第三方相机输入。
  - *存储图像* 模块支持存储之前模块的渲染结果图像。
  - *字符识别*和 *DL 字符识别* 模块新增字符过滤功能。当置信度调节为 0 时，可输出得分最高的五组字符信息。
  - 深度学习训练工具新增中断/恢复训练、训练曲线绘制、训练模型验证功能。
  - *DL 分类*支持输入和输出汉字。
  - 加载、保存方案与代理之间的方案数据操作接口由同步修改为异步。
  - *格式化*模块删除格式化合并按钮，该功能由底层 SDK 实现。
- 功能删减：
  - 软件不再支持 Windows 7/10 32 位操作系统。
  - 删除 Vericode 码读取模块。

V3.0.0 维护版本（2019.04）

该版本为 V3.0.0 的维护版本，发布于 2019 年 4 月。

- 功能新增：
  - 深度学习分类新增 CPU 版本的*字符定位*模块。
  - 软件新增英文版本，支持中英文切换。
- 功能优化：
  - *BLOB 分析*模块新增孔洞面积参数。
  - *N 点标定*模块新增旋转轴中心输出。
  - *畸变标定*模块新增透视中心输入。
  - *几何创建*和图像处理分类的*圆环展开*模块提升算法处理效率。
  - 更新 OCR 算法库，提高识别率。

V3.0.0 维护版本（2019.01）

该版本为 V3.0.0 的维护版本，发布于 2019 年 1 月。

- 功能优化：
  - 深度学习分类的 *DL 目标检测*与 *DL 分类*新增 CPU 模式的支持，不占用显卡资源。原 GPU 版本的模块名称后缀为“G”，新增 CPU 版本的模块名称后缀为“C”。
  - *文本保存*与*存储图像*模块，支持数据订阅。
  - *亮度测量*与*直方图工具*模块，输出结果中增加对比度。
- 缺陷修复：
  - 解决圆环展开 ROI 区域继承不准确的问题。
  - 解决外部通信连续触发时，重启软件后触发异常的问题。
  - 解决二维码识别有大概率文本内容显示为{编码信息}的问题。
  - 解决二维码模块输出编码信息长度超过 256 后，模块崩溃的问题。
  - DL 缺陷检测模块支持检测彩色图像和灰度图像的缺陷，可根据输入图像的像素格式自动切换，同时增加算法库应用模式参数（可选高性能、极速模式）。

V3.0.0

该版本发布于 2018 年 11 月 27 日。

- 功能新增：
  - 新增一个方案中创建多个流程的功能。
  - 新增*全局变量*、*通信管理*功能。
  - 深度学习分类的*目标检测*和*分类*模块新增 CPU 版本。
  - 标定分类新增*畸变标定*模块。
  - 新增对位分类，包括*相机映射*、*单点对位*、*点集对位*和*线对位*模块。
  - 图像处理分类新增*图像组合*、*拷贝填充*、*帧平均*、*图像归一化*模块。
  - 新增缺陷检测分类，包括*字符缺陷检测*模块。
  - 逻辑分类新增新增*分支字符比较*、*文本保存*和*耗时统计*模块。
  - 通信分类新增接收数据、发送数据和 PLC 通信模块，配合通信管理使用。
- 功能优化：
  - 方案管理支持通过配置通信字符串实现方案自动切换。
  - *二维码识别*模块增加 DM 码类型参数。

第 3 章 产品概述

Vision Master 算法开发平台（简称为：VM）是海康机器人自主研发的视觉算法开发平台，由近千个图像处理算子和 150+交互式视觉功能模块组成。该平台功能丰富、简单易用，支持多种开发模式，可助力开发者快速搭建定位、测量、缺陷检测、信息识别等各类视觉方案。

- *功能特性*
- *开发模式*

功能特性

界面操作简单易上手

- 提供友好的图形用户界面（Graphical User Interface，GUI），功能图标直观易懂。
- 支持通过拖拽式操作快速搭建方案。
- 支持实时显示视觉功能模块的运行状态，问题排查无忧。

二次开发简单易上手

- 支持软件二次开发（基于 SDK）。SDK 接口简单易上手。相较于自行开发视觉方案，基于 SDK 开发可节省 90%的代码量。
- 支持将新增控件可一键导入 Visual Studio 中，且支持基于 QT、MFC、WPF 和 WinForm 开发界面。

自定义能力丰富

- 支持自定义视觉方案的运行界面。可在运行界面上集成背景图片或您公司的 Logo，满足个性化需求。
- 支持将自定义算子封装为视觉功能模块。

兼容性强大

- 兼容 TCP/IP、串口、UDP、ModBus、Ethernet/IP 等多种常见的工业通信协议和主流 PLC 设备。
- 兼容 GigE Vision、USB3 Vision 和 Camera Link 等协议标准。支持接入多种图像采集设备并处理实时图像，也可处理本地图像。

开发模式

VM 提供三种开发模式：

- 模式一：直接在 GUI 上搭建视觉方案。
- 模式二：在 GUI 上搭建视觉方案后，基于 SDK 进行应用层的二次开发。
- 模式三：基于算子开发包进行算子层级的二次开发。可将算子封装为自定义的视觉工具，集成到自定义的检测流程中。

3.1 运行环境

为确保软件能正常安装及运行，对安装软件的工控机配置有所要求。

推荐配置

- 操作系统：Windows7/10/11（64 位操作系统）

  说明

  如需使用深度学习模块，建议使用 Windows 10 或 Windows 11 64 位操作系统。

- CPU：Intel Core i7-6700 3.4GHz 及以上

  说明

  如需使用 CPU 相关深度学习模块，建议配备 i7-8 代及以上。

- 内存：8 GB 及以上
- 显卡：显存 1 GB 及以上

  说明

  如需使用 GPU 相关深度学习模块，需确保显存 6 GB 及以上，具体参见 *推荐硬件* 。

- .NET FrameWork：4.6.1 及以上
- 网卡：Intel i210 系列及以上千兆网卡
- USB 接口：USB3.0

最低配置

- 操作系统：Windows7/10/11（64 位操作系统）
- CPU：Intel 3845
- 内存：4 GB
- 显卡：显存 1 GB

  说明

  如需使用 GPU 相关深度学习模块，需确保显存 6 GB 及以上，具体参见 *推荐硬件* 。

- .NET FrameWork：4.6.1
- 网卡：千兆网卡
- USB 接口：USB3.0

推荐硬件

软件中的深度学习功能，需搭配如下设备或显卡使用。

- AI 推理终端：海康机器人自主研发的 AI 推理终端设备，其内部搭载 VM 专用 AI 推理加速卡，支持运行部分深度学习和缺陷检测模块，包括 *DL 字符定位*、*DL 字符识别*、*DL 分类*、*DL 读码*、*DL 目标检测*、*DL 实例分割*、*DL 图像检索*、*DL 图像分割*、*DL 快速图像分割*、*DL 无监督分割*和*异常检测*。

说明

使用 AI 推理终端运行上述模块前，请确保您的工控机内存不低于 16 GB，并安装 Windows 10 或 Windows 11 64 位操作系统。此外，操作系统内部版本号需不低于 10.0.19041。

- NVIDIA 独立显卡：支持运行所有深度学习类和边缘学习类模块。可考虑使用以下显卡型号。
  - NVIDIA® GeForce® RTX™ 50 系列：RTX5060、RTX5060Ti、RTX5070、RTX5070Ti、RTX5080、RTX5090 D v2。
  - NVIDIA® GeForce® RTX™ 40 系列：RTX4060、RTX4060Ti、RTX4070、RTX4070Ti、RTX4080、RTX4090。
  - NVIDIA® GeForce® RTX™ 30 系列：RTX3050、RTX3060、RTX3060Ti、RTX3070、RTX3080、RTX3090。
  - NVIDIA® GeForce® RTX™ 20 系列：RTX2060、RTX2060S、RTX2070、RTX2070S、RTX2080、RTX2080S、RTX2080Ti。
  - NVIDIA® GeForce® GTX 10 系列：GTX1060、GTX1660、GTX1660S、GTX1070、GTX1070Ti、GTX1080、GTX1080Ti。
  - NVIDIA® Tesla®系列：Tesla L2。

说明

  - NVIDIA® GeForce® RTX™ 50 系列显卡的驱动版本应不低于 572.61，以上其他系列显卡的驱动版本应不低于 528.33。
  - 安装深度学习安装包时，支持选择 **AI 推理终端**和 **NVIDIA 显卡**。软件运行时，将根据设置的显卡或设备运行深度学习相关模块。更多配置详情，请参见深度学习安装包中的*安装指南*。

3.2 授权方式

VM 为收费软件，需获取授权后使用。您可按需选择通过本机加密狗、软加密或远程加密狗获取使用授权。不同的加密狗/软加密型号支持的功能有所差异，请结合实际检测需求合理选择。

表 3-1 授权方式概述

授权方式	描述
本机加密狗	通过加密狗即插即用。 可更换工控机使用，但仅可在插入加密狗的工控机（即“本机”）上使用 VM。更多信息，请参见 *通过本机加密狗* 。
软加密（授权码）	通过授权码获取授权。

授权方式	描述
	使用 VM 之前，需先通过授权码激活 VM。更多信息，请参见 *通过授权码* 。 说明 • 若通过软加密授权后需更换使用的工控机或重装系统，则需先完成软加密的迁出，并在另一台工控机或重装系统后重新迁入。具体如何操作请联系技术支持。 • 若工控机主板等核心硬件损坏，更换核心硬件后，本工控机的软加密将失效，需重新购买授权码。
远程加密狗	同样通过加密狗即插即用。 与本机加密狗的不同之处——加密狗插入局域网内某台工控机（以下简称“工控机 A”）后，若在局域网内其他工控机上将“工控机 A”设置为服务端，其他工控机无需再插入加密狗，也可直接使用 VM。更多信息，请参见 *通过远程加密狗* 。

3.2.1 加密狗授权

您可选择通过加密狗获取 VM 使用授权。不同型号的加密狗支持的能力集有所差异，您可根据业务需求选取合适的加密狗型号。

加密狗型号

不同型号加密狗支持的能力集有所差别，具体如下表所示。

表 3-2 加密狗能力集明细

型号	定位测量标定运算	识别	缺陷检测	边缘学习	深度学习	图像处理	颜色处理
iMVS-VM-1200	*	√	×	×	×	*	*
iMVS-VM-2200	√	×	√	×	×	√	*
iMVS-VM-4200	√	×	×	×	×	*	*

型号	定位 测量 标定 运算	识别	缺陷检测	边缘学习	深度学习	图像处理	颜色处理
iMVS-VM-6200 iMVS-VM-6210	√	√	√	×	×	√	√
iMVS-VM-6217	√	√	√	√	×	√	√
iMVS-VM-7200	√	√	√	*	√	√	√
iMVS-VM-7210	√	√	√	√	√	√	√

说明

- 上表中未提及的模块类别，所有型号加密狗均支持，主要为采集、拆分组合、图形生成、逻辑工具和通信等。
- 上表中*代表该类别部分模块可用，部分模块不可用，具体请以实际情况为准。
- iMVS-VM-6210、iMVS-VM-6217 和 iMVS-VM-7210 型号加密狗（其中，iMVS-VM-6210 和 iMVS-VM-7210 是 iMVS-VM-6200 和 iMVS-VM-7200 型号加密狗的升级版）支持软件与算子互通、*编辑运行界面设计*、*脚本*与其他型号加密狗等功能，具体参见相关功能介绍的章节。

相机路数限制

不同型号加密狗支持的相机路数有所差异。iMVS-VM-6210、iMVS-VM-6217 和 iMVS-VM-7210 型号加密狗不限制相机路数，其余型号加密狗遵循以下原则：

- 不带后缀：支持 4 路相机；
- 带 SE 后缀：支持 2 路相机；
- 带 PRO 后缀：不限制相机路数。

共享加密

本机加密狗还支持共享加密服务，可向海康机器人加密设备中写入加密信息，在同一加密设备中既可以实现对软件的使用，又可以实现对软件权限信息的二次开发，且我司软件许可与用户写入的加密信息完全隔离、互不干扰。
共享加密服务的具体购买流程如下图所示。

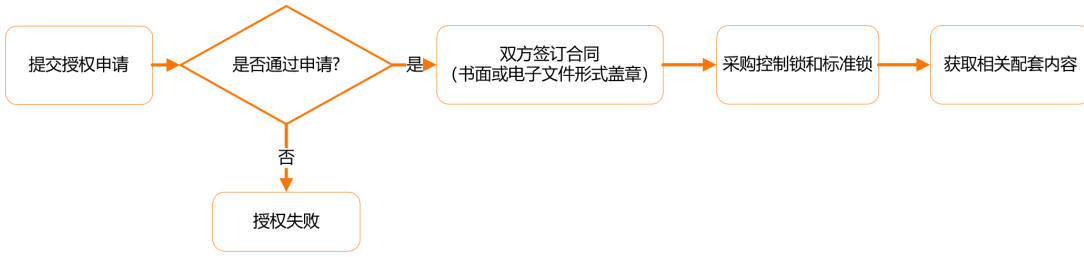


图 3-1 共享加密狗购买流程

共享加密服务的所有相关配套内容包含以下方面：

- 控制锁：又名“母锁”，主要用于创建和分发授权，可借助工具将写好的用户私有信息烧录至标准锁，以实现授权的分发，烧录次数可选择类型有 3 种（分别为 100 次、300 次和 500 次）。
- 标准锁：又名“子锁”，主要用于监控和检验授权，型号选择可参考本章节上文提及的型号情况。
- SDK：可参考 SDK 了解如何通过控制锁，将用户私有信息烧录至标准锁。
- 工具：可借助该工具对其他标准锁（子锁）进行分发授权。
- 帮助文档：可参考如何使用共享加密服务。

3.2.2 远程加密狗授权

若选择通过远程加密狗获取 VM 使用授权，您仅需一个加密狗，即可让您的团队在局域网内任意工控机上使用 VM。具体实现方式：在局域网内任意一台工控机上插入加密狗（将其作为局域网内的服务器），并在局域网内其他工控机上进行服务配置；配置完成后，在局域网内未插入加密狗的工控机上，也可正常使用 VM。为确保远程加密狗授权成功，插入加密狗的工控机与安装 VM 的工控机必须处于同一网段。

加密狗型号

远程加密狗型号如下表所示，不同型号支持的功能有所差异。

表 3-3 远程加密狗能力集明细

型号	定位 测量 标定 运算	识别	缺陷检测	边缘学习	深度学习	图像处理	颜色处理
iMVS-VMNET-6200	√	√	√	×	×	√	√

型号	定位 测量 标定 运算	识别	缺陷检测	边缘学习	深度学习	图像处理	颜色处理
iMVS-VMNET-6210							
iMVS-VMNET-6217	√	√	√	√	×	√	√
iMVS-VMNET-7200	√	√	√	*	√	√	√
iMVS-VMNET-7210	√	√	√	√	√	√	√

说明

- 上表中未提及的模块类别，所有型号加密狗均支持，主要为采集、拆分组合、图形生成、逻辑工具和通信等。
- 上表中*代表该类别部分模块可用，部分模块不可用，具体请以实际情况为准。
- 上表中，iMVS-VMNET-6210、iMVS-VMNET-6217 和 iMVS-VMNET-7210 型号（其中，iMVS-VMNET-6210 和 iMVS-VMNET-7210 型号加密狗是 iMVS-VMNET-6200 和 iMVS-VMNET-7200 型号加密狗的升级版）支持软件与算子互通、*编辑运行界面设计*、*脚本*与其他型号加密狗等功能，具体参见相关功能介绍的章节。

相机路数限制

不同型号的远程加密狗支持的相机路数有所差异，具体如下：

- iMVS-VMNET-6200 和 iMVS-VMNET-7200：支持 4 路相机。
- iMVS-VMNET-6210、iMVS-VMNET-6217 和 iMVS-VMNET-7210：不限制相机路数。

3.2.3 授权码授权

VM 除了支持硬加密（通过“加密狗”），也支持软加密（通过“授权码”）。选择软加密型号并下单后，您可通过海康机器人销售人员获取对应的授权码。

软加密型号

不同软加密型号支持的功能有所差别，具体参见下表。

表 3-4 各型号能力集

型号	定位 测量 标定 运算	识别	缺陷检测	边缘学习	深度学习	图像处理	颜色处理
iMVS-VMS-1200	*	√	×	×	×	*	*
iMVS-VMS-2200	√	×	√	×	×	√	*
iMVS-VMS-4200	√	×	×	×	×	*	*
iMVS-VMS-6200 iMVS-VMS-6210	√	√	√	×	×	√	√
iMVS-VMS-6217	√	√	√	√	×	√	√
iMVS-VMS-7200	√	√	√	*	√	√	√
iMVS-VMS-7210	√	√	√	√	√	√	√

说明

- 上表中未提及的模块类别，所有型号加密狗均支持，主要为采集、拆分组合、图形生成、逻辑工具和通信等。
- 上表中*代表该分类部分模块可用，部分模块不可用，具体请以实际情况为准。
- 上表中，iMVS-VMS-6210、iMVS-VMS-6217 和 iMVS-VMS-7210 型号加密狗（其中，iMVS-VMS-6210 和 iMVS-VMS-7210 是 iMVS-VMS-6200 和 iMVS-VMS-7200 型号加密狗的升级版）支持软件与算子互通、*编辑运行界面设计*、*脚本*与其他型号加密狗等功能，具体参见相关功能介绍的章节。

相机路数限制

软加密型号后缀不同，支持的相机路数有所差别。iMVS-VMS-6210、iMVS-VMS-6217 和 iMVS-VMS-7210 型号加密狗不限制相机路数，其余型号加密狗遵循以下原则：

- 不带后缀：支持 4 路相机。
- 带 SE 后缀：支持 2 路相机。
- 带 PRO 后缀：不限制相机路数。

注意事项

- 将某授权码在某台工控机上激活后，该授权码无法在其他工控机上再度激活。若需更换使用软件的工控机或工控机需重装系统，则应先迁出软加密信息，后续再重新迁入进行使用。具体如何操作请联系本公司技术支持。
- 使用多个授权码激活时，若不同授权码支持的能力集有差异，您可以使用多个授权码支持的全部能力集。

3.3 术语介绍

本章节对软件中出现的专业词汇进行介绍。

方案

也称为视觉方案或视觉配方，是视觉处理方法的总和。

流程

即视觉处理方法，由不同视觉功能模块按照指定的逻辑顺序组成，一般与相机相关，一个相机对应一个流程。

模块

视觉处理的最小单元，做某一个特定的图像算法或逻辑运算。

模型

通过对图像或图像上的特征进行注册或训练得到理想目标的数据块，以此告知算法目标的特征。

Group

即组合模块，指多个模块的组合。

脚本

是使用一种特定的描述性语言，依据一定格式编写的可执行文件。

位姿

目标相对于参考坐标系的位置和姿态。

基准点

设定为基准的点，可用于测量目标物的位置偏移。

图像坐标系

基于图像自身的坐标系。以图像左上角为原点、向右横轴为 X 正轴、向下竖轴为 Y 正轴，顺时针为正角度。

物理坐标系

基于真实世界中实际存在的物体（例如机械结构）为原点、X 轴、Y 轴的坐标系。物理坐标系下的距离对应真实的距离。

左手坐标系

X 轴向右，Y 轴向上，Z 轴背向“自己”。

右手坐标系

X 轴向右，Y 轴向上，Z 轴指向“自己”。

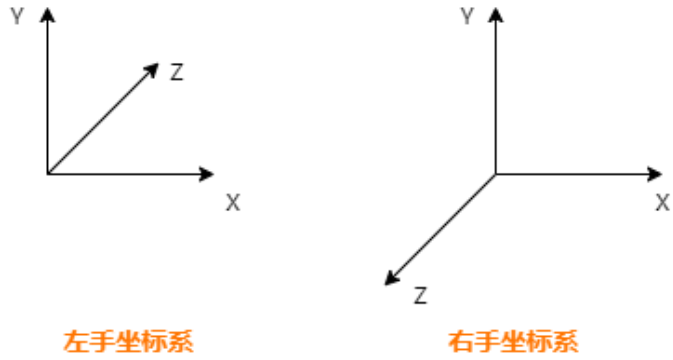


图 3-2 左/右手坐标系

手性一致性

分析两个坐标系是否同为左手坐标系或同为右手坐标系，相同则为 1。

坐标原点

图像坐标系的原点，即图像最左上角的点。

物理点

图像坐标系上的点转换到物理坐标系即物理点。

机械臂

指高精度，多输入多输出、高度非线性、强耦合的复杂系统。也可理解为用于抓取或放置物体的运动装置。

共轴

指两根或多根轴共享一个中心轴线的装置。假设 A 运动到 A1 的圆心，与 B 运动到 B1 的圆心是同一个点，则 A 与 B 共轴。

示教点

指机械臂执行任务时需要到达的固定点位，通过示教点可实现机械臂执行任务的自动化操作。

运行点

在实际生产中，对当前物料拍照定位，得到的目标位置点。

工控机

一种专门用于工业控制和自动化领域的计算机设备，也称为工业控制计算机或嵌入式工控计算机。与个人电脑相比，工控机具有更高的可靠性、稳定性、抗干扰能力和安全性，适合于在恶劣环境下长期运行和稳定控制工业生产过程。

单像素精度

也称为像素精度，表示采集的图像上一个像素在真实世界中的实际尺寸是多大。该数值越小，说明精度越高。

直线

在软件中直线以起点和终点表示，类似数学中的线段。

直线角度

直线在图像坐标系中与 X 轴正方向的角度。顺时针为正，逆时针为负。

掩膜

告知算法计算怎样的区域。一般计算灰度值为 255 的区域，不计算灰度值为 0 的区域。

BOX

矩形框，由矩形中心点（X,Y）、宽、高和角度组成。

卡尺

用于测量灰度值的一个矩形区域，详细内容参见 *卡尺工具* 模块。

检测区域

即进行算法处理的区域。

ROI 区域

即感兴趣区域，是需要设置为检测区域的区域。

质心

图像所指区域中所有点的算术平均值。假设图像区域由 n 个 x_i 组成，则质心 c 由以下公式得出。

$$c = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Blob

图像中具有相似颜色、纹理等特征所组成的一块连通区域。

概率图

反应检出效果强弱/得分的图，图像中像素值越大，表示得分越高。

RGB 颜色空间

由 R、G、B 三原色为基础构成，通过这是那种颜色的不同比例叠加，可生成各种颜色。每个颜色通道（R、G、B）的数值范围均为 0~255，0 表示没有该颜色，255 表示改颜色的最大强度。三个通道的不同组合可表示的颜色范围广泛，因此在图像显示方中被广泛使用。缺点是颜色与 R、G、B 三通道分量紧密相关，当图像中存在光照变化或阴影时，RGB 值可能存在较大的变化，不便于进行图像处理。

HSV 颜色空间

由色调（Hue）、饱和度（Saturation）和明度（Value）构成，通过非线性变换将颜色的色调、饱和度和明暗分离，更符合人类对色彩的感知方式。这种特性使得 HSV 颜色空间在色彩对比和分割任务中，相较于 RGB 颜色模型更具有优势。HSV 到 RGB 的转换公式如下：

$$V = max\{R, G, B\}$$

$$S = \begin{cases} \dfrac{V - min\{R,G,B\}}{V} & if\ \ V \neq 0 \\ 0 & other \end{cases}$$

$$H = \begin{cases} \dfrac{60(G-B)}{V - min\{R,G,B\}} & if\ \ V = R \\ 120 + \dfrac{60(B-B)}{V - min\{R,G,B\}} & if\ \ V = G \\ 240 + \dfrac{60(R-G)}{V - min\{R,G,B\}} & if\ \ V = B \end{cases}$$

$$IF\ \ H < 0, then\ \ H = \mathrm{h} + 360. \quad 0 \leq \{\mathrm{R}, \mathrm{G}, \mathrm{B}\} \leq 1$$

说明

在图像处理中，通常将 H、S、V 值的范围归一化为 0~255 表示。

HSI 颜色空间

由色调（Hue）、饱和度（Saturation）和强度（Intensity）构成，类似于 HSV 颜色空间，通过非线性变换将色调、饱和度和亮度分离，更符合人类对色彩的感知方式，有利于色彩对比和分割。HSI 到 RGB 的转换公式如下：

$$I = \frac{R+G+B}{3}$$

$$S = \begin{cases} 1 - \dfrac{3min\{R,G,B\}}{R+G+B} & if\ \ I \neq 0 \\ 0 & other \end{cases}$$

$$H = \begin{cases} \theta & G \geq B \\ \theta + \pi & G < B \end{cases}$$

$$where\ \ \theta = \frac{\pi}{2} - tan^{-1}\left(\frac{2R-G-B}{\sqrt{3}(G-B)}\right), 0 \leq \{R,G,B\} \leq 1$$

说明

在图像处理中，通常将 H、S、I 值的范围归一化为 0~255 表示。

YUV 颜色空间

Y 表示明度（Luminance），即灰度值，UV 则表示色差（Chrominance），用于描述颜色的色彩和饱和度。YUV 颜色空间广泛引用于电视系统，包含多种存储格式，常见的包括 YUV444、YUV422 和 YUV420。 ***颜色转换*** 模块中，将 RGB 图像转换为 YUV 颜色空间时，默认转换为 YUV444 格式。转换公式如下：

$$Y = 0.299R + 0.587G + 0.114B$$

$$U = -0.147R - 0.289G + 0.436B$$

$$V = 0.615R - 0.515G - 0.100B$$

3.4 界面介绍

VM 提供图形化用户界面（Graphical User Interface，GUI），方便您基于简单的 GUI 操作，快速搭建视觉方案，基本无需编写代码。VM 客户端的 GUI 主要包括欢迎页和主界面。启动 VM

客户端时默认进入欢迎页，您可通过欢迎页进入主界面。本章概览性介绍欢迎页和主界面的主要功能。

3.4.1 欢迎页

双击 启动软件，会弹出欢迎页，如下图所示。



图 3-3 欢迎页

欢迎页中包含方案类型选择、最近打开方案及其他等功能，具体介绍如下：

- **方案类型选择**：可根据所需方案类型选择。
- **最近打开方案**：显示最近打开的方案记录，选中方案并双击可快速打开。也可点击 选择工控机上的方案并快速打开。
- **学习使用 VISIONMASTER**：可打开软件用户手册。
- **查看示例方案**：可查看软件自带的示例方案。

  说明

  该功能需确保已安装示例方案工具包，否则无法使用。

- **获取更多支持和帮助**：可进入海康机器人官网。
- **不再显示**：勾选后，后续双击 打开软件时，会跳过欢迎页直接进入 *主界面*。

3.4.2 主界面

主界面为 VM 图形化用户界面的中枢。视觉方案搭建相关的主要配置与操作，均需在主界面上进行。本节概览性介绍主界面的主要功能。

在 *欢迎页* 选择任意方案类型（*通用方案*、*定位测量*、*缺陷检测*、*用于识别*）即可进入主界面。

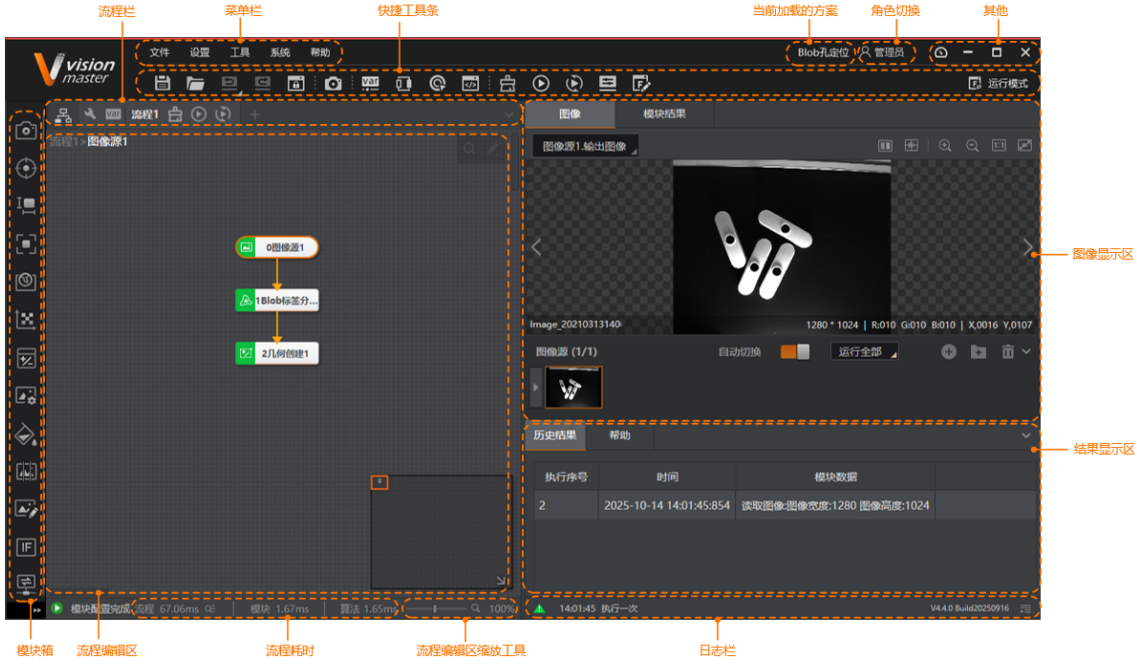


图 3-4 主界面

该界面各区域的说明参见下表：

表 3-5 主界面简介

序号	名称	描述
1	菜单栏	提供*文件*、*设置*、*工具*、*系统*等菜单选项。详情参见 *菜单栏* 。。
2	快捷工具条	提供方案保存、相机管理、控制器管理、全局变量等工具的快捷入口。详情参见 *快捷工具条*
3	当前加载的方案	显示当前加载的方案的名称。

序号	名称	描述
4	角色切换	切换角色登录软件。如果您的角色为管理员，可通过设置不同角色的权限，限制技术员和操作员访问软件的特定功能模块。 权限设置相关详情，参见 ***权限设置*** 。
5	其他	该区域的图标由左至右依次用于： ● 查看内存使用率、CPU 使用率和各硬盘的使用情况。 ● 最小化主界面窗口。 ● 最大化主界面窗口。 ● 缩小主界面窗口。 ● 关闭软件。
6	图像显示区	执行流程后，可在该区域查看经流程中各模块处理后的图像和各模块的输出结果。详情参见 ***图像显示区域*** 。
7	流程栏	提供流程相关操作入口。
8	模块箱	包含图像采集、定位、测量、识别、标定、对位、图像处理、颜色处理、缺陷检测、逻辑工具、通信等可构成流程的视觉功能模块。
9	流程编辑区	可根据业务需求，从模块箱中选择模块并拖入该区域，再对各模块进行排序和配置，定义视觉处理流程。该区域右下角为鹰眼区，可在鹰眼区拖动橙色矩形框全局查看流程编辑区。 相关操作和配置详情参见 ***方案搭建-流程配置*** 。
10	流程耗时	显示当前流程耗时。当选中模块时，显示流程耗时、模块耗时和模块算法处理耗时。 在此区域单击  ，进入***耗时查看***页面，可查看流程耗时、VM 占用内存、各模块耗时、各模块算法处理耗时及各模块算法运行占用内存。
11	流程编辑区缩放工具	可拖动滑块，对流程编辑区进行缩放控制。

序号	名称	描述
12	日志栏	可查看软件的日志信息，默认显示最近一条日志。也可单击最右侧的 ，并滚动鼠标滚轴查看全量日志。
13	结果显示区	显示流程的当前运行结果和历史运行结果，并支持查看所选的某个算法模块的简要使用说明。详情参见 *结果显示区域* 。

菜单栏

菜单栏提供软件的基本功能和基本配置的快捷入口。

文件

- *新建方案*：创建新方案。
- *打开方案*：选择已有方案并载入。
- *最近打开方案*：选择某个最近载入的方案进行加载。
- *打开示例*：可选择软件自带的示例方案并打开。需额外安装当前版本对应的示例程序安装包（名称包含 Sample 的安装包）。
- *保存方案*：将当前方案保存至默认的本地方案路径。
  - 方案首次保存时，需先选择保存路径。并可通过*加密选项*设置方案是否加密（见下图）。方案加密保存后，载入该方案时需输入密码。

    说明

    通过 *加密狗写密码工具* 设置加密狗密码后，若需载入加密方案，需满足如下两个条件：
    - 工控机上插入的加密狗的密码，与通过 *加密狗写密码工具* 设置的密码相同。
    - 输入的方案密码正确。
  - 方案非首次保存时，直接将当前方案更新保存至上述保路径。
  - 如果方案保存失败，请联系技术支持获取帮助。

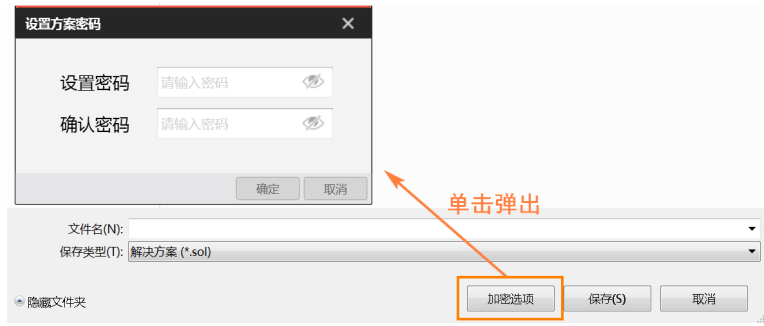


图 3-5 方案加密设置

- *方案另存为*：将当前方案另存至其他本地路径。
- *导出算子代码*：可将当前方案流程中的模块搭建逻辑导出成算子代码的书写逻辑，并以工程文件的形式存储查看编辑，使用方法如下。

说明

- 仅型号中带 6210 或 7120 的加密狗支持此功能。
- 仅使用单算子的模块支持此功能；对于使用了多个算子的模块不支持此功能，例如*图像组合*、*阵列圆查找*、*边缘组合缺陷检测*、*边缘对组合缺陷检测*模块。
- 逻辑工具模块中仅*分支模块*支持此功能，其他逻辑工具模块不支持此功能。
- 当模块间存在输入输出订阅时，模块中的输出字段与算子输出存在关联（模块输出值使用了对应算子输出值）时，支持此功能；模块订阅了自定义输出时，不支持此功能。

1. 在软件顶部菜单栏，选择 *文件* → *导出算子代码*。
2. 在*导出算子代码*对话框，配置如下参数。

编程语言

支持 **C++**和 **C#**两种编程语言，根据实际需要进行选择。

编译平台

当前暂只支持选择 **VS2013**。

工程名称

设置导出工程文件夹的名称。

导出目录

设置导出工程文件的存放路径。

导出流程

选择要导出的流程，可选择一个或多个流程导出。

导出 ROI 配置

开启此开关后，可导出模块对应的 ROI 操作代码。

说明

支持导出算子代码功能且含有 ROI 的模块支持此功能。

3. 单击*导出*。
4. 在*导出成功*提示框，单击*确定*，即可自动跳转至导出目录。

说明

- 对于需要使用模型文件的算子，因模型大小、算子接口规格不统一，在不做特殊化定制的情况下，模型文件不易导出。在导出算子代码生成的工程源码中，会将输入模型的内容预留出来，当您编译时，以 error 信息提示您填入模型文件路径或补充模型数据。
- 导出代码工程的运行效果，可能会与实际的 VM 流程存在差异。导出的代码尽量在算子操作流程与 VM 保持一致，但因一些参数或实现细节的不一致，无法做到生成完全对等的源代码。

- *导入流程*：将已导出至本地的流程文件（.prc 格式）导入至当前方案。

说明

导出流程操作，详见 *其他操作与配置* 中提及的*流程导出*。

- *退出*：退出当前的软件，退出时可选择保存方案后再退出或者直接退出。

设置

- *权限设置*：设置不同角色的软件访问权限。详情参见 *权限设置* 。
- *软件设置*：设置软件系统的配置选项，包括**开机软件自启动**、**缓存步数设置**和**模块最大数量**等。详情参见 *软件设置* 。
- *方案设置*：设置方案相关配置选项，包括**方案管理**（方案自动切换设置）、**回调设置**和**自动保存设置**。详情参见 *方案设置* 。
- *运行策略*：设置**运行模式**和**策略模式**。详情参见 *运行策略* 。
- *采集策略*：设置是否启用*大文件存图*功能。详情参见 *采集策略* 。

工具

软件提供多种工具方便您进行标定、自定义模块和采集异常信息。

- **标定配置引导**：提供步骤式引导，帮助您快速完成静态标定、动态标定或映射标定的配置。详情参见 *标定配置引导* 。
- **标定板生成**：支持自定义标定信息，生成标定图像。详情参见 *标定板生成* 。
- **异常信息收集**：支持收集 PC 配置信息、VM 版本信息、算子版本信息、dump 和日志信息（VM 软件日志、算子 SDK 日志、二次开发相关日志）。收集的信息可提供给技术支持进行问题排查。详情参见 *异常信息收集* 。
- **参数查找工具**：可查看方案中各个模块输入/输出参数和运行参数的当前值以及对应的算子参数名称。详情参见 *参数查找工具* 。

系统

- *日志*：查看软件运行过程的日志信息。
- *通信管理*：配置软件与通信设备、控制器设备等之间的数据传输和相关参数。详情参见 *通信管理*。
- *相机管理*：添加和配置相机。详情参见 *相机管理*。

帮助

- *语言*：切换软件语言。
- *帮助文档*：查看软件的帮助文档，了解并上手使用软件。
- *帮助文档（在线）*：查看软件的在线帮助文档。
- *版本信息*：查看当前软件版本、算子版本、支持的 MVS 客户端版本以及版权信息。
- *更多支持*：可进入海康机器人官网。
- *V 社区*：可进入海康机器人 V 社区。V 社区为您提供机器视觉软硬件相关的资源工具、技术支持、学习交流、应用实践、活动竞赛等一站式服务，覆盖图像处理、AI 算法、3D、深度学习、机器人导航、运动控制等热门领域及汽车、光伏、锂电、3C 等热门行业，旨在打造面向开发者、合作伙伴、终端用户、技术爱好者和高校师生的互动交流平台。
- *打开欢迎页*：打开软件的启动引导页面。

快捷工具条

快捷工具条位于主界面菜单栏下方，提供 VM 部分常用功能的快捷入口。



图 3-6 快捷工具条

表 3-6 图标简介

图标	名称	描述
	保存方案	将当前加载的方案保存到本地 PC。 • 首次保存时需选择保存路径，并可按需单击*加密选项*设置方案密码。 • 其他情况下，将当前加载的方案直接覆盖保存至原方案文件。
	打开方案	加载本地 PC 上的方案文件。 如果方案文件加密，则加载时需输入对应的密码，否则无法加载。

图标	名称	描述
	撤销	撤销最近一次操作。可单击该图标右下角的 ◢，并单击*历史记录*查看历史操作。
	重做	取消撤销操作，恢复至撤销前的状态。 **说明** • 所有支持撤销和重做功能的操作，在进行自然操作时都会有正常的内存增涨，因为需缓存操作数据。 • 新建、加载、保存方案会清空撤消重做缓存的所有数据。
	锁定	单击该图标并在弹窗中选择*是*，可开启锁定。开启锁定后，无法对流程编辑区的模块、连线进行新增、删除、移动、复制粘贴、撤销等操作。
	相机管理	单击进入相机管理界面。详情参见 *相机管理* 。
var	全局变量	单击进入全局变量界面。详情参见 *全局/局部变量* 。
	通信管理	单击进入通信管理界面。详情参见 *通信管理* 。
	全局触发	单击进入全局触发界面。详情参见 *全局触发* 。
	全局脚本	单击进入全局脚本界面。详情参见 *全局脚本* 。
	清空结果	单击后可清空当前方案中所有模块的执行状态、模块结果数据及结果渲染数据。 **说明** 任一流程或模块处于执行状态时，该按钮不可使用。
	单次执行	单击执行一次当前方案。 **说明** 单击*单次运行*相当于使用相机软触发方式获取输入图像。针对使用相机硬触发方式获取输入图像的情况，需注意仅首个绑定相机的图像源所在的流程能够得到触发，此时通过工具栏的*单次运行*按钮将无法控制其他流程的触发顺序。建议您在使用硬触发方式获取输入图

图标	名称	描述
		像时，一个流程对应一个相机，以确保对应流程正确执行。
	连续执行	单击连续执行当前方案。 **说明** 单击*连续执行*相当于使用相机软触发方式获取输入图像。针对使用相机硬触发方式获取输入图像的情况，需注意仅首个绑定相机的图像源所在的流程能够得到触发，此时通过工具栏的*连续执行*按钮将无法控制其他流程的触发顺序。建议您在使用硬触发方式获取输入图像时，一个流程对应一个相机，以确保对应流程正确执行。
	综合配置	单击进入综合配置界面。该工具将常用的运行参数、输出结果以及图像整合，可快速查看相关效果。详情参见*综合配置*。
	运行界面设计	单击进入运行界面设计窗口。详情参见*方案搭建-运行界面设计*。

3.5 二次开发方式

VM 提供软件开发工具包（SDK）和算子开发包两套二次开发工具包，分别对应基于 SDK 的方案集成开发方式和基于算子开发包的代码驱动开发方式。追求开发高效、维护简单，并希望少写代码，请选择基于 SDK 的方案集成开发方式；追求灵活可控、深度定制，可接受写大量代码，请选择基于算子开发包的代码驱动开发方式。

开发方式说明

- 基于 SDK 的方案集成开发方式：在 VM 可视化界面中，通过拖拽功能模块（如定位、测量、识别等）并用连线定义数据流向，快速搭建并调试出完整的视觉方案。方案验证成功后，导出方案文件。在最终部署时，上位机软件通过 SDK 加载这个方案文件，仅需通过简单的接口调用即可运行整个复杂的视觉方案，无需在代码中实现具体的算法逻辑。

说明

基于 SDK 的方案集成开发方式详细说明请查看 VM 算法开发平台 SDK 开发指南，文档可在“.\Development\V4.x\Documentations”路径下获取。

- 基于算子开发包的代码驱动开发方式：通过调用一系列预定义的函数（算子）来构建图像处理流程。整个逻辑完全由代码控制，从图像采集、预处理、特征提取到结果输出，每一步都需要开发者手动编写代码串联。此方式强调脚本化和灵活性，流程的跳转、循环和条件判断完全依赖编程逻辑实现。

说明

基于算子开发包的代码驱动开发方式详细说明请查看算子 SDK 开发指南，文档可在“.\MVDAlgorithmSDK\Documentations”路径下获取。

开发方式对比

简单来说，基于算子开发包的代码驱动开发方式与基于 SDK 的方案集成开发方式两者的核心区别在于“视觉逻辑在哪里实现”和“代码工作量的大小”。两种开发方式多维度对比如下表所示。

对比项	基于 SDK 的方案集成开发方式	基于算子开发包的代码驱动开发方式
视觉逻辑实现	图形界面拖拽模块+少量代码	纯代码
代码工作量	少（仅调用少量接口和控件）	多(完全使用代码调用模块接口和控件)
开发效率	快（所见即所得，即时验证）	慢（需编写和调试大量代码）
灵活性	受限（依赖于模块功能边界）	极高（可自定义任意复杂逻辑）
方案维护	便捷（通常只需替换方案文件）	复杂（需修改代码并重新编译）
版本管理	依赖方案文件管理	友好（易于纳入 Git 等版本控制）
适用人群	应用工程师或快速交付团队	资深开发工程师

选型建议

当您进行二次开发时，请根据项目特征参考以下场景选择开发方式。

- 建议选择基于 SDK 的方案集成开发方式：
  - 快速交付项目：项目周期紧，希望快速完成视觉方案设计并验证。
  - 标准化需求：通常仅涉及定位、测量、图像处理等常见任务，项目逻辑相对简单，VM 现有模块已能覆盖大部分功能需求。

  - 便于后期维护：调试时，修改视觉参数不需要重新编译，维护简便。
  - 团队分工解耦：视觉工程师负责搭方案，软件工程师负责业务逻辑调用，职责清晰。
- 建议选择基于算子开发包的代码驱动开发方式：
  - 复杂逻辑控制：视觉方案中存在大量动态分支、递归处理或多相机同步等复杂控制逻辑的任务。
  - 非标准需求：现有的模块无法满足特定的数据处理结构，需要深度定制算法步骤。
  - 极致优化：需要对每一个算法步骤的控制逻辑进行微调，以达到性能极限。
  - 纯代码部署：项目架构限制不允许依赖外部方案文件，所有逻辑必须编译在程序中。

对于二次开发方式的选择，本质上是在"灵活性/可控性"与"开发效率/易用性"之间做权衡。如果项目需求标准、追求快速落地，基于 SDK 的方案集成开发方式是首选；如果项目极具挑战性、逻辑复杂多变，基于算子开发包的代码驱动开发方式则更具优势。
在实际的大型项目中，两种开发方式也可混合使用：利用图形化界面快速构建核心检测模块，再通过算子开发包进行复杂的流程控制和数据管理。在实际使用过程中，请根据上述指南进行技术评估，选择最优的二次开发方式。

第 4 章 快速入门

本章节介绍如何快速上手使用 VM，内容涵盖使用前提、示例体验、搭建入门级方案和运行入门级方案等。

4.1 准备工作

开始前，您需已明确选用的 VM 授权方式，同时确保安装软件的工控机运行环境符合要求。

说明

- 软件已集成加密狗以及硬件所需驱动，无需下载并安装其他驱动。
- 不排除未知杀毒软件将该软件识别为病毒的情况。为方便使用，建议将本软件加入该杀毒软件的白名单中或关闭电脑上的杀毒软件。对于 360 安全卫士建议关闭。

4.1.1 准备图像采集设备

VM 兼容的图像采集设备包括面阵相机、线阵相机和 X86 开放平台。

表 4-1 兼容的图像采集设备

设备类型	描述
面阵和线阵相机	支持多种相机接口，包括千兆网口、USB3.0 口、万兆网口、Camera Link 口、CoaXPress 口等。软件可兼容遵循 GigE Vision、USB3 Vision、Camera Link 或 CoaXPress 协议的相机。 **说明** 推荐使用海康机器人相机，其他厂商的相机可能存在兼容性问题。
X86 开放平台	支持 X86 开放平台，但该类型设备不支持使用深度学习模块。

4.1.2 获取软件使用授权

VM 为收费软件，无法直接打开。使用前需通过加密狗或授权码获取使用授权。

说明

- VM 授权方式包括本机加密狗、软加密和远程加密狗，可在软件安装时选择其中一种。若安装时**授权方式**选择错误，需重装 VM 并选择正确的**授权方式**。更多安装相关信息，请参见 *VM 安装指南*。
- 不同授权方式的具体规格和能力集详情，请参见 ***授权方式*** 。

通过本机加密狗

授权方式选择***本机加密狗***时，在使用软件前需通过工控机的 USB 口插入加密狗。

说明

各加密狗型号支持的能力集详情，请参见 ***加密狗授权*** 。

通过授权码

授权方式选择***软加密***时，在使用软件前需在本机上使用授权码进行激活。

说明

不同软加密型号所支持的能力集详情，请参见 ***授权码授权*** 。

注意事项

- 将某授权码在某台工控机上激活后，该授权码无法在其他工控机上再度激活。若需更换使用软件的工控机或工控机需重装系统，则应先迁出软加密信息，后续再重新迁入进行使用。具体如何操作请联系本公司技术支持。
- 使用多个授权码激活时，若不同授权码支持的能力集有差异，您可以使用多个授权码支持的全部能力集。

激活方式

操作步骤：

1. 首次打开软件时，会提示未激活，如下图所示。选择***是***进入激活工具。



图 4-1 未激活提示

说明

激活工具可在软件安装路径下获取，相对路径为：.\Applications\Tools\AuthActiveTool\SoftEncryptActive.exe。

2. 完成软加密激活。软加密激活可选离线激活和在线激活两种方式。
   - **离线激活**：点击上方的*生成 C2D 文件*将生成的文件和授权码一起提供给海康机器人技术支持。他们会提供对应的 D2C 文件，通过下方的*导入 D2C 文件*完成导入即可激活。

     说明

     - C2D 和 D2C 文件一一对应，通过其他工控机的 C2D 文件生成的 D2C 文件，不能在当前工控机上使用。为避免出现对应错误的情况，建议一对一做好管理。
     - 对应的授权码联系海康机器人销售获取。
   - **在线激活**：联系海康机器人销售获取**授权码**，填入后单击*在线激活*完成激活即可。

     说明

     若选择在线激活，需确保激活的工控机能连接网络，否则无法成功激活。
3. （可选）完成激活后，通过激活工具的*授权信息*页面，点击*查询本地软加密狗*可查询当前工控机已激活的软加密授权信息，并可进行如下操作。
   - *查看许可*：可查看授权码的相关许可信息。
   - *离线解绑*：可对使用离线激活方式授权的工控机进行解绑，解绑后会生成 C2D 文件。将该文件提供给海康机器人技术同事，获取解绑协助。
   - *在线解绑*：若使用在线激活方式授权的工控机能连接网络，可直接进行在线解绑。在另一台可连接网络的工控机上，可通过原授权码重新在线激活。



图 4-2 授权信息

说明

解绑后，通过授权码可重新进行离线或在线激活。

注意

请务必妥善保管授权码。

通过远程加密狗

授权方式选择*远程加密狗*时，在使用软件前需在局域网的服务器上插入远程加密狗并通过 Virbox 工具完成服务器配置，局域网内安装软件的工控机方可使用。

注意

请务必确保局域网内服务器和其他工控机间网络能 ping 通。

服务器插上远程加密狗后，局域网内已安装软件的工控机通过任务栏右下角 打开 Virbox 工具，可进行加密狗的服务器配置。

- 打开工具后，切换到**服务设置**部分，服务模式可选择**客户端模式/客户端/服务器模式**，点击*添加*并输入服务器信息，点击*保存&重启*后点击*连接测试*。若服务器状态显示为“可用”，此时可打开软件。
  不同服务模式，功能存在部分差别。在**客户端/服务器模式**下，可通过**黑白名单设置**对局域网内的黑名单或白名单进行设置。
  - 选择*白名单*时，局域网内 IP 地址在白名单内的工控机可打开软件，其他工控机无法打开；
  - 选择*黑名单*时，局域网内 IP 地址在黑名单内的工控机可无法打开软件，其他工控机可打开。

  说明

  以上设置在局域网内设置一次即可。若后续使用无法打开，可查看服务器状态。若显示为“不可用”，则重新点击*保存&重启*后点击*连接测试*，确保状态变为“可用”。



图 4-3 服务设置

- 工具左侧选择**硬件锁**，可查看加密狗的基本信息。若服务器端也安装软件，进行服务设置前，加密狗属于本地加密锁；进行服务设置后，既属于本地加密锁，也属于网络加密锁。完成服务设置后，工具只显示网络加密锁信息，如下图所示。

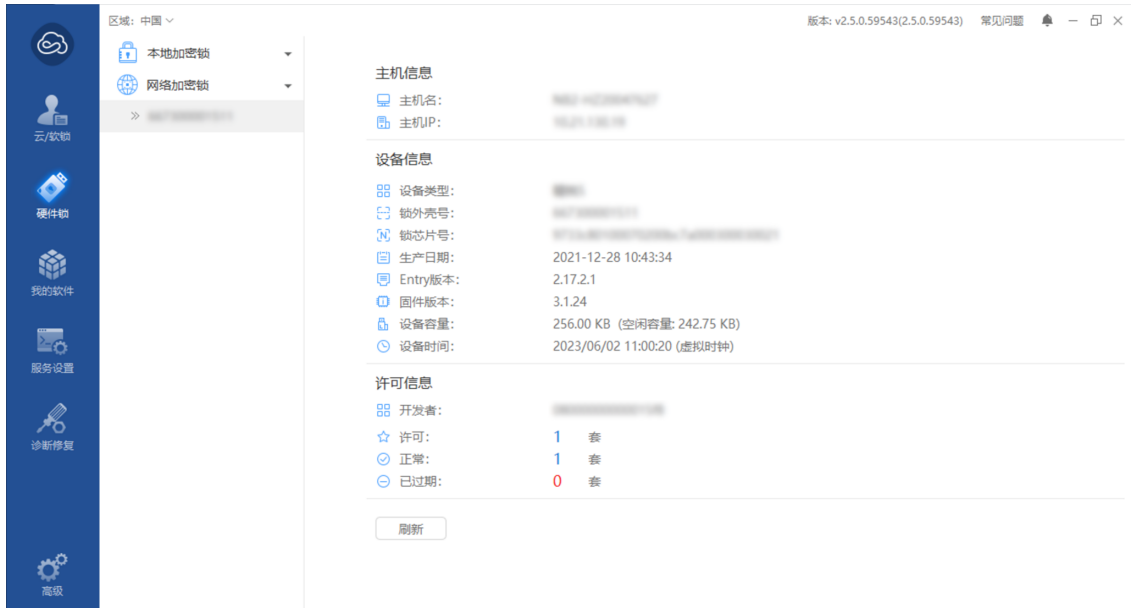


图 4-4 硬件锁

- 工具左侧选择**我的软件**，可查看加密狗信息，双击具体的加密狗后可查看该加密狗的并发数，即局域网内最多支持同时打开软件的工控机数量。



图 4-5 我的软件

4.2 了解方案构成

VM 的使用流程围绕“方案”这一概念进行。对方案及其构成的深入理解，有助于您在方案搭建阶段，将业务需求转化为清晰的方案搭建思路。建议您在入门阶段至少先了解构成方案的必需元素，即视觉流程和视觉功能模块。

本节内容包含：

- *方案是什么*
- *方案构成简介*
- *常见问题：为什么相机并非方案必需元素？*

方案是什么

方案的全称为视觉方案，是实现视觉检测所需的“数据与逻辑配置”的集合，包括图像采集、视觉处理流程、通信和脚本等。

方案构成简介

方案的具体构成如下图所示。

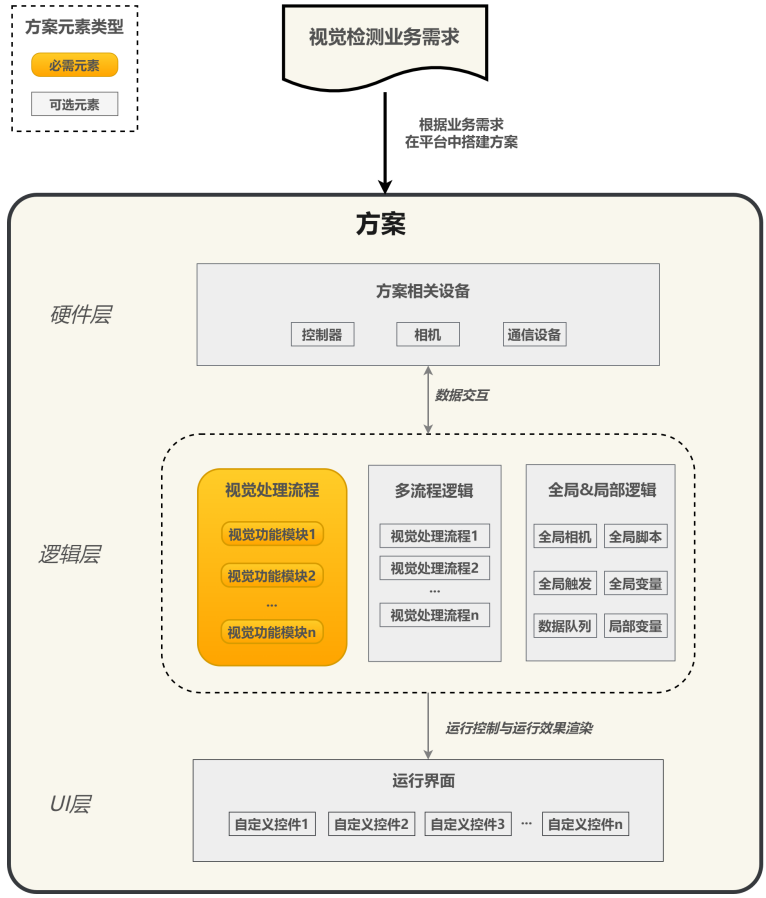


图 4-6 方案构成示意图

- 方案中的必需元素为视觉功能模块、视觉处理流程。一个可运行的方案必需包含这两个必需元素。

表 4-2 必需元素

元素	描述
视觉功能模块	简称“模块”，即处理图像数据的最小功能单元。 更多详情，请参见 *方案搭建-模块配置* 及其子章节。
视觉处理流程	简称“流程”，由视觉功能模块构成，用于定义视觉系统如何处理获取到的图像数据。流程的处理逻辑包含图像采集逻辑和视觉检测逻辑。 方案中需至少包含一条流程。 更多详情，请参见 *方案搭建-流程配置* 及其子章节。

- 方案中绑定的硬件设备、多流程逻辑、全局逻辑和运行界面均为可选元素。

表 4-3 可选元素

元素	描述
运行界面	可自行设计的的 GUI 界面。方案交付后，业务现场的终端用户可通过运行界面操控方案运行，甚至控制流水线作业。 更多详情，请参见 *方案搭建-运行界面设计* 及其子章节。
方案相关设备	与方案绑定的硬件设备，包括相机、视觉控制器和 PLC 等硬件设备。这些设备与方案绑定后，与方案进行数据交互。例如，在方案中添加相机后，相机可作为方案的图像源，为方案提供图像输入。 更多详情，请参见 *方案搭建-硬件设备配置* 。
多流程运行逻辑	方案中多个流程的联动运行逻辑。 更多详情，请参见 *配置多流程运行逻辑* 。
方案全局与局部逻辑	在整个方案维度内全局生效的逻辑定义，包括全局相机、全局脚本、全局触发、全局变量和数据队列。 ○ **全局相机**：可被方案中多个流程同时关联为图像源的相机。 *相机管理* 中配置的相机实际上均为全局相机。 ○ **全局脚本**：控制方案内其他元素（例如流程、通信设备和模块）的脚本，可用于实现多流程运行控制、通信数据收发、模块参数自动设置等特定需求。 ○ **全局触发**：可在方案内全局生效的触发逻辑。可定义平台在指定事件或字符串输入时触发特定动作，如方案切换、流程运行、模块运行等。 ○ **全局变量**：可被方案内任意模块和方案运行界面的任意控件订阅的外部变量。全局变量在方案内全局生效。 ○ **数据队列**：在流程之间调用的“数据缓存逻辑”，用于解决多流程联动运行时，不同流程耗时差异导致的流程间数据阻塞。 ○ **局部变量**：可被单个流程或单个 *Group* 模块订阅的外部变量。局部变量仅在方案内的单个流程或单个 *Group* 模块维度生效。 更多详情，请参见 *方案搭建-全局与局部逻辑配置* 及其子章节。

常见问题：为什么相机并非方案必需元素？

图像源 模块支持将流程的图像源设置为工控机上或局域网共享路径内的图像文件（即本地图）。将本地图作为图像源时，无需添加相机即可调试方案运行效果。

4.3 体验示例方案

VM 在其安装路径下提供了丰富的示例方案供您体验和学习。

前提条件

已安装示例方案安装包（VisionMaster_Sample_CN_STD）。

示例方案文件存储于 VM 安装目录下的.\Applications\Samples 文件夹中，您可以在 VM 中打开并运行这些示例，体验软件功能。同时您可以参考示例方案搭建出符合实际业务需求的方案。

说明

示例方案所在的文件夹中包含方案示例的解读，帮助您了解对应方案的目的、逻辑搭建思路和搭建步骤。

操作步骤

1. 运行 VM。

   系统弹出如下欢迎页。



图 4-7 欢迎页

2. 在欢迎页，单击*查看示例方案*。

   说明

   您也可以在进入主界面后，选择 *文件 → 打开示例* 进入示例方案所在的文件夹（.\Applications\Samples）中。

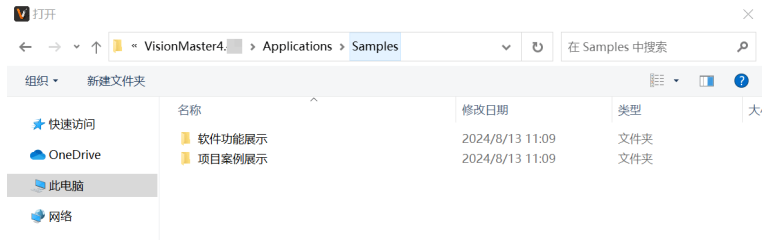


图 4-8 示例方案文件夹

3. 选择您希望体验的示例方案并打开。

   在 VM 主界面的流程编辑区域查看对应方案流程。本文以打开“工件缺陷及朝向检测”示例方案为例，对应方案流程如下图所示。

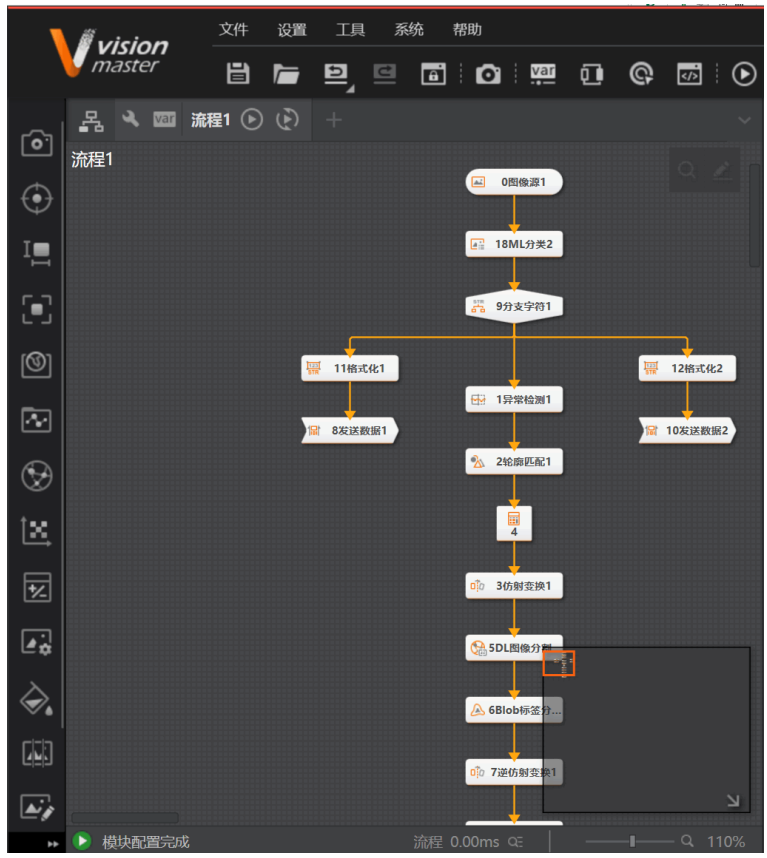


图 4-9 方案流程

4. 在流程栏，单击 或 ，单次或连续运行流程。

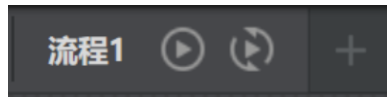


图 4-10 流程栏

5. 在 *图像显示区域* 和 *结果显示区域* 查看运行结果。

   选中流程中的某个模块，即可在右侧的图像显示区域和结果显示区域查看该模块输出的图像和结果。

说明

如果您需要获取示例方案中的本地图像，在图像显示区域的本地图像配置区域，右击任一本地图像后，单击*打开文件所在目录*，打开当前方案的本地图像存储文件夹获取图像。

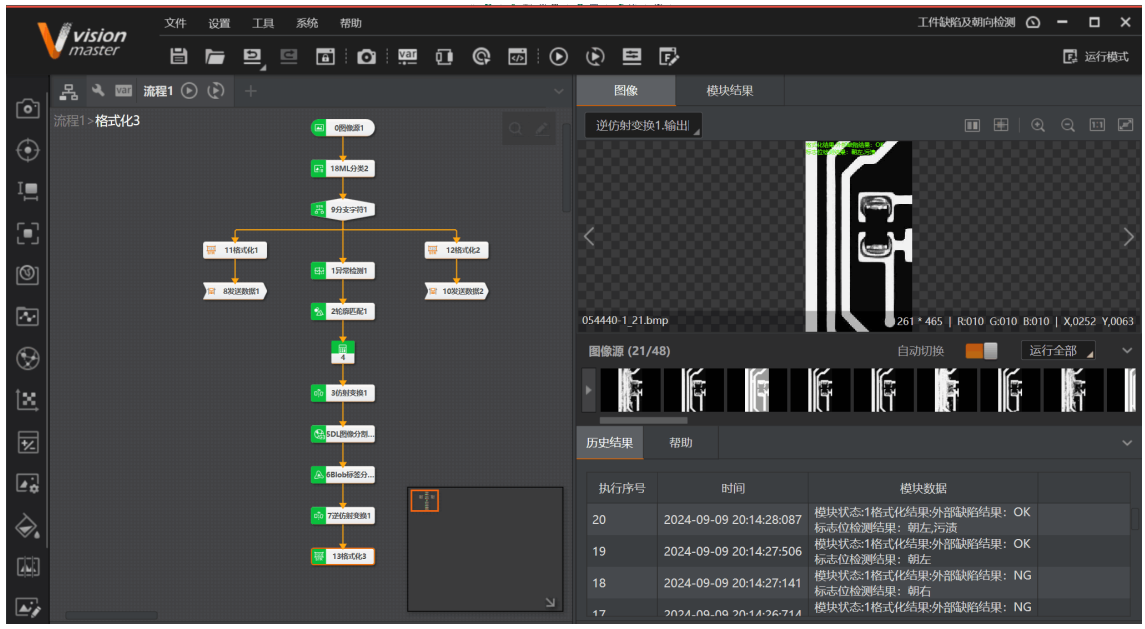


图 4-11 运行结果

6. 在主界面右上角，单击*运行模式*。

   即可进入运行界面，运行界面用于直观操控和监控方案运行，业务现场的终端用户通过运行界面操控方案运行（甚至流水线作业）。

7. 在运行界面，单击 或 ，单次或连续运行方案。

   运行界面支持自定义配置，可按照工作需求调整运行界面展示控件的类型、位置、大小等，详情请参见 *方案搭建-运行界面设计* 。示例运行界面如下图所示。

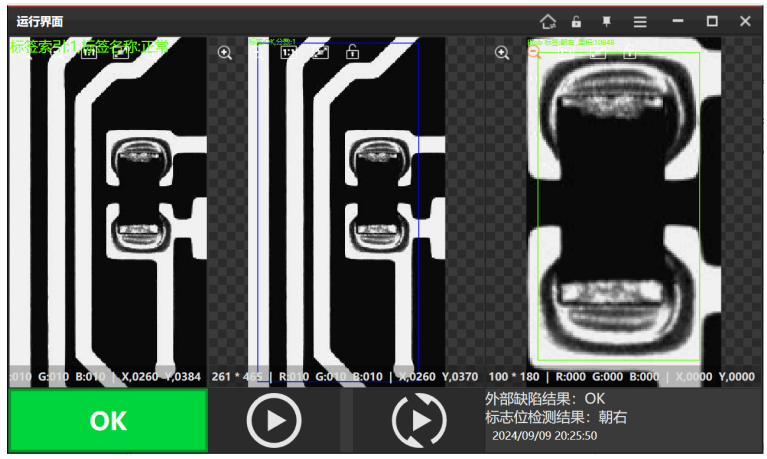


图 4-12 运行界面

4.4 搭建入门级方案

“入门级方案”指仅包含单个流程和简单运行界面的方案，不涉及全局逻辑配置、通信设备配置和控制器配置等复杂逻辑。本节基于示例方案中的“电池片检测”方案介绍如何自行从头开始搭建此示例方案，帮助您快速上手使用 VM。

方案需求

该示例的检测需求如下：

表 4-4 检测需求

需求项	描述
需求 1	计算电池片的**宽度**和**高度**，并格式化输出。
需求 2	计算电池片的中心点，并格式化输出。 由于电池片的实际形状相较于标准件可能存在偏差，为降低误差，因此需计算出电池片**中线交点**坐标和**角平分线交点**坐标的平均值，作为电池片的中心点坐标。
需求 3	为在机器人抓取电池片时方便判断定位精度，需生成电池片的**定位框**，并输出带定位框的图像。 若定位不准，则定位框相较于电池片存在肉眼可察觉的偏移。

上表中提及的**宽度**、**高度**、**中线交点**、**角平分线交点**和**定位框**的示意图如下。

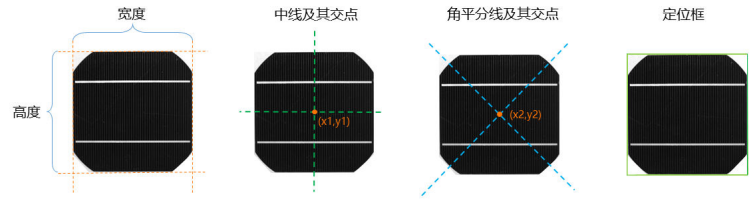


图 4-13 检测需求中的概念

方案思路

需求 1（计算电池片宽高）、需求 2（计算电池片中心点坐标）和需求 3（生成电池片定位框）的实现思路如下图所示。其中需求 3 的定位框生成，需基于需求 1 和需求 2 实现后输出的如下数据：

- 定位框宽度和高度——即电池片的宽度和高度，需求 1 实现后即可获取。
- 定位框中心点——即电池片的中心点，需求 2 实现后即可获取。
- 定位框旋转角度——即电池片的旋转角度。需求 1 实现思路中，粗定位电池后即可获取其旋转角度。

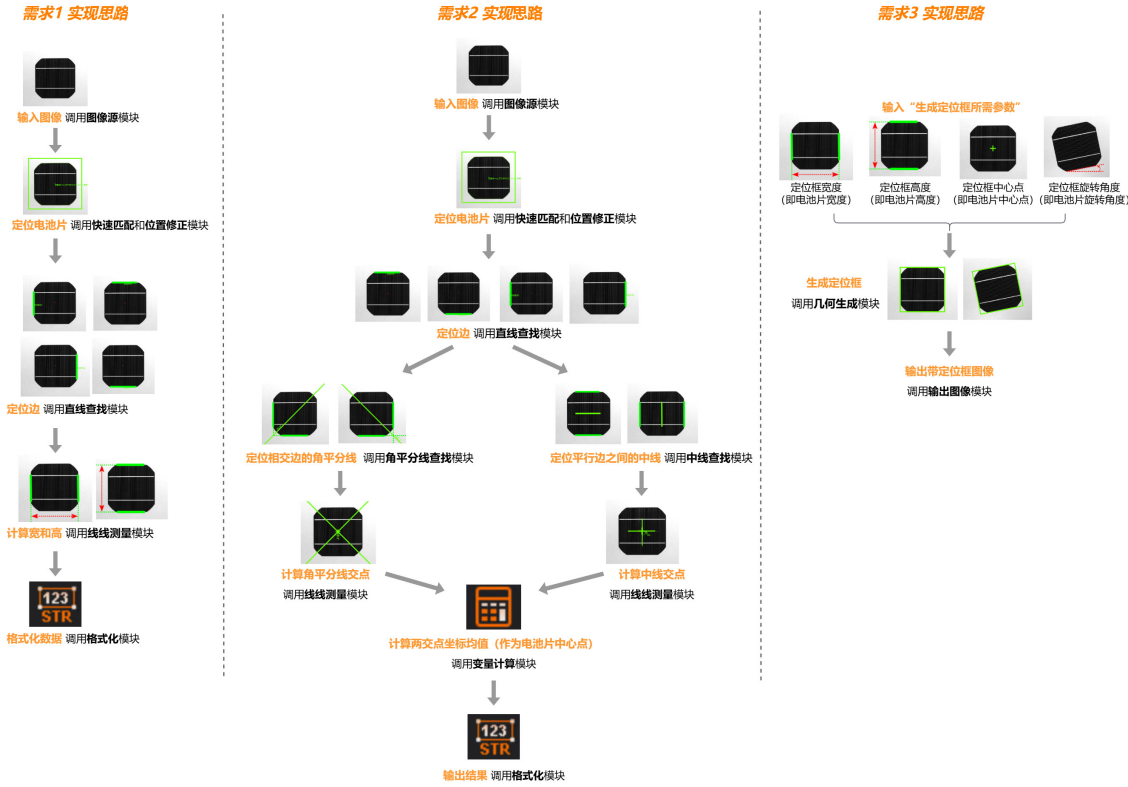


图 4-14 电池片检测思路

上图中提及的各模块的主要作用如下。

表 4-5 模块作用

模块	作用
图像源	输入电池片图像。
快速匹配	对电池片进行粗定位。
位置修正	修正电池片来料时的位置偏移，实现精确定位。
直线查找	定位电池片的四条边。
中线查找	基于***直线查找***定位到两组平行边，定位平行边之间的中线。
角平分线查找	基于***直线查找***定位到两组相交边，定位相交边之间的角平分线。
线线测量	• 计算电池片两组平行边的间距，即电池片宽度和高度。 • 计算两条中线的交点坐标。 • 计算两条角平分线的交点坐标。
变量计算	计算电池片中线交点坐标和角平分线交点坐标 X、Y 值的平均值，分别作为电池片中心点坐标 X、Y 值。
几何创建	基于***快速匹配***模块输出的电池片旋转角度以及***线线测量***模块计算出的电池片宽高和中心点，生成电池片的定位框。
格式化	将电池片的宽度、高度和中点坐标数据整合并格式化成字符串输出。
输出图像	将经过***几何创建***模块处理的图像输出。

由于以下两个原因，该 3 个需求的实现思路整合为单个流程实现。

- 需求 1、需求 2 的实现思路中，“输入图像”、“定位电池片”和“定位电池片的边”检测逻辑相同。
- 需求 3 的实现，需要获取需求 1 和需求 2 实现后输出的数据。

最终流程的整体设计逻辑如下图所示（仅供参考）。

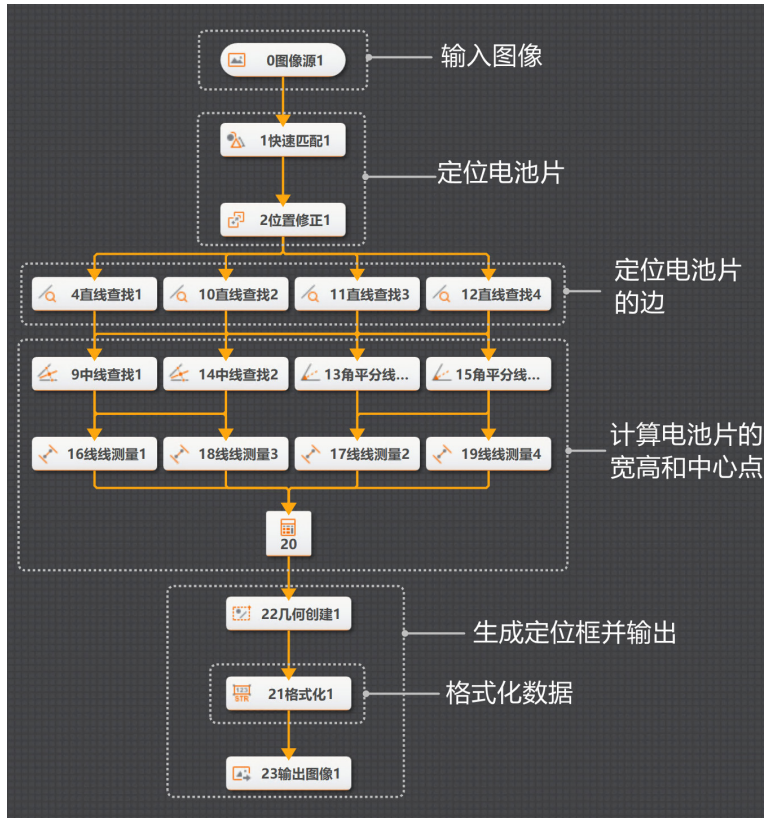


图 4-15 流程搭建思路示意

4.4.1 步骤一：创建方案

首次运行 VM 系统将弹出欢迎页。本节介绍如何从欢迎页开始搭建方案。

操作步骤

1. 运行 VM。

   系统弹出如下欢迎页。



图 4-16 欢迎页

2. 在欢迎页，单击*通用方案*，即可创建方案，并进入主界面。

说明

- 若未勾选欢迎页右上角的*不再显示*，后续重新运行 VM 时，欢迎页将继续弹出，且“最近打开方案”区域显示最近曾创建（打开）的方案。可单击最近打开的方案快速打开。
- 进入主界面后，还可通过主界面菜单栏的 *文件 → 新建方案* 创建方案。

4.4.2 步骤二：搭建流程

“搭建流程”即定义视觉处理流程的过程。在该过程中，您可定义：1.VM 从何处获取图像输入；2. VM 如何处理图像数据并输出处理结果。

编排流程运行逻辑

按照上文*方案思路*中的流程搭建思路，编排流程的整体运行逻辑。流程由模块任意排布连接而成。将所需模块从模块箱中拖入流程编辑区域，并按需连接即可。

- 若拖入的模块与另一个模块对齐，则两者自动连接，如以下动图所示。



图 4-17 自动连接

- 若拖入模块时未与另一个模块对齐，参考如下动图手动建立连接。

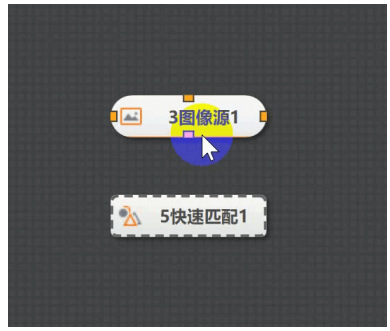


图 4-18 手动连接

说明

各模块的前后序模块以及多模块如何搭配使用相关详情，请参见 *模块使用参考* 各子章节中对于“前后序模块”的说明。

输入图像

在 VM 方案中输入图像基于*图像源*模块实现。本节基于示例方案中的“电池片检测”方案介绍如何通过*图像源*模块配置图像输入。

前提条件

已准备本地电池片图片。

说明

- 请关注图像质量是否已满足业务需求。一般情况下，图像源获取的输入图像越清晰检测结果越准确。
- 您如果没有本地电池片图片，安装示例方案安装包后，可在 VM 安装目录下的.\Applications\Samples\软件功能展示\定位测量\电池片检测文件夹中打开“电池片检测”方案获取示例电池片图片，操作详情请参见 *体验示例方案* 。

操作步骤

1. 双击*图像源*模块，打开其配置窗口。
2. 设置该配置窗口上基本参数中的**图像源**。

   本示例中*图像源*选择**本地图像**，并导入本地存储的电池片图片：

   说明

   本地图像适用于非现场调试，*图像源*也可以选择**相机**或 **SDK**，详细说明请参见 *图像源* 。

   - 在图像显示区域，单击 从本地路径添加单张图像。
   - 在图像显示区域，单击 添加本地文件夹内的全量图像。

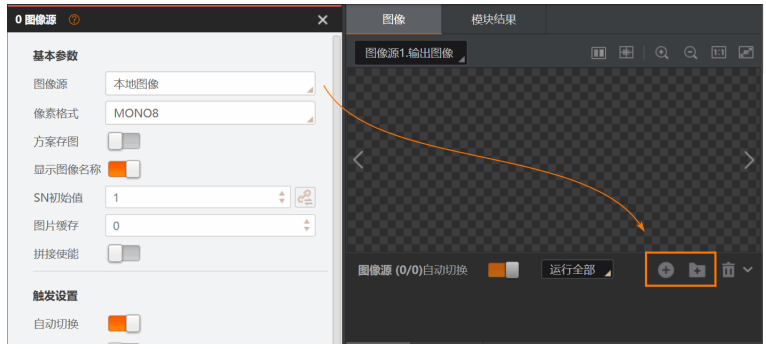


图 4-19 添加本地图像

3. **可选操作**: 按需配置该模块的其他参数。

   其他参数在简单方案中一般采用默认值即可。相关配置详情，请参见 *图像源* 。

定位电池片

操作步骤

1. 在流程工具栏，单击 执行一次流程使*图像源*将图像数据输出至后序模块。
2. 配置*快速匹配*模块，实现粗定位。
   1) 双击*快速匹配*模块，配置模块关键参数。

   说明

   关键参数以外的参数采用默认值即可。

输入源

在*基本参数*页签配置**输入源**选择前置图像源图像，例如“图像源 1.图像”。

模板

a. 在*特征模板*页签，单击*创建*。

b. 在*模板配置窗口*，创建特征模板。操作步骤如下图所示，特征模板相关详细说明请参见 *快速匹配* 的模板配置参数。



图 4-20 创建特征模板

2) 在*快速匹配*窗口右下角，单击*确定*。

3. 配置*位置修正*模块，实现精定位。

1) 双击*位置修正*模块，配置模块关键参数，实现精定位。

选择方式

在*基本参数*页签，配置**选择方式**参数为“按点”。

原点

配置*原点*参数订阅前置*快速匹配*模块的**匹配点**数据，例如“快速匹配 1.匹配点”。

角度

配置*原点*参数订阅前置*快速匹配*模块的**角度**数据，例如“快速匹配 1.角度”。

2) 在*位置修正*窗口，单击*确定*。

定位电池片的边

操作步骤

1. 双击*直线查找*模块，配置模块关键参数。

输入源

在*基本参数*页签，配置**输入源**参数为*图像源*模块图像，例如“图像源 1.图像”。

ROI

在 *ROI 区域*单击 图标，在图像显示区域的电池片图像上绘制直线卡尺，操作如下图所示。直线查找更详细说明请参见 *直线查找* 。

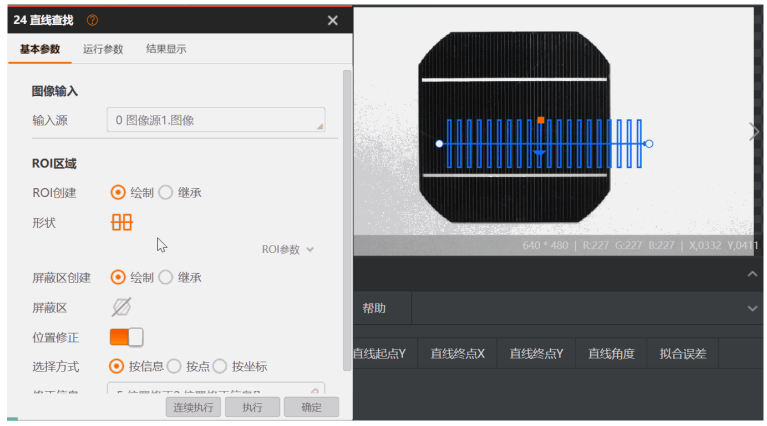


图 4-21 绘制直线卡尺

位置修正

打开*位置修正*开关；配置*修正信息*参数订阅前置*位置修正*模块数据，例如“位置修正 1.位置修正信息”。

2. 在*直线查找*窗口右下角单击*执行*和*确定*。
3. 重复以步骤，配置另外 3 个*直线查找*模块。

方案中合计 4 个*直线查找*模块，分别用于定位电池片的四条边，*直线查找*模块对应的绘制图形如下图所示。

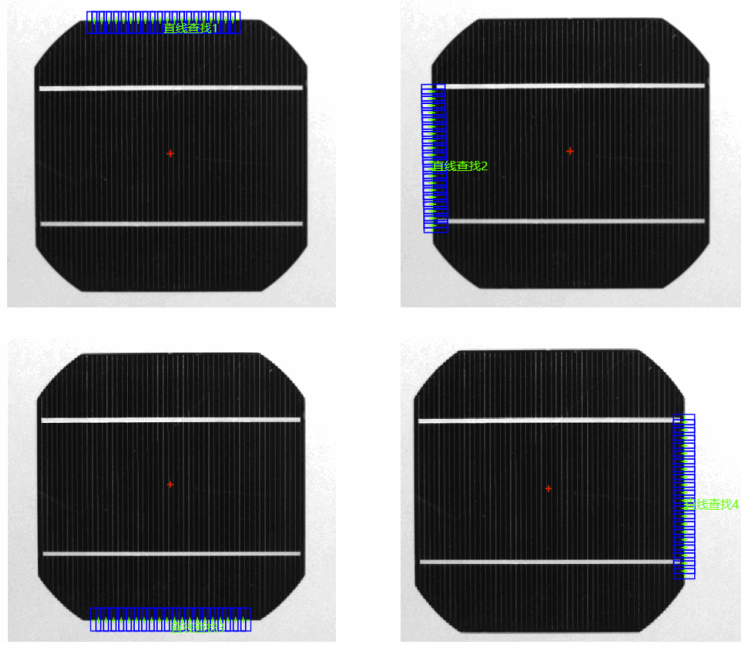


图 4-22 直线查找模块对应绘制图形

计算电池片宽高和中心点

操作步骤

1. 配置 2 个*线线测量*模块（此处以示例中的*线线测量 3* 和*线线测量 4* 为例）并分别执行，计算电池片的宽和高。
   1) 双击*线线测量 3* 模块，配置模块关键参数。

      输入源

      在*基本参数*页签，配置**输入源**参数为图像源图像，例如“图像源 1.图像”。

      数据来源

      选择*订阅*。

      线

      *线*参数分别订阅 2 个*直线查找*模块输出的直线，分别对应电池片一组对边。例如分别订阅“直线查找 1.输出直线”、“直线查找 3.输出直线”。

   2) 在*线线测量*窗口，单击*执行*和*确定*。

      单击*执行*后，可在图像显示区查看线线测量结果，如下图所示。

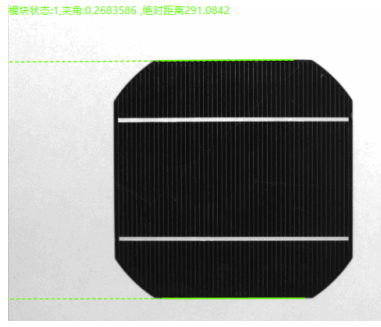


图 4-23 电池片高

   3) 重复以上 2 个子步骤，配置*线线测量 4* 模块，用于测量电池片的宽。参数分别订阅另外两个*直线查找*模块输出的直线，例如分别订阅“直线查找 2.输出直线”、“直线查找 4.输出直线”。

2. 依次配置*中线查找*、*角平分线查找*、*线线测量*和*变量计算*模块，计算电池片的中心点坐标值。
   1) 配置*中线查找 1* 和*中线查找 2* 的关键参数，并执行该两个模块，定位电池片的 2 条中线。

      输入源

      在*基本参数*页签，配置**输入源**参数为图像源图像，例如“图像源 1.图像”。

      线

      *线*参数分别订阅 2 个*直线查找*模块输出的 1 组对边。例如分别订阅“直线查找 1.输出直线”、“直线查找 3.输出直线”。

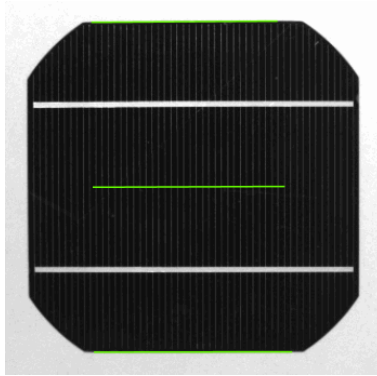


图 4-24 中线查找结果

2) 配置*角平分线查找 1*和*角平分线查找 2*模块的关键参数，并执行该两个模块，定位电池片的 2 条角平分线。

输入源

在*基本参数*页签，配置**输入源**参数为图像源图像，例如“图像源 1.图像”。

线

*线*参数分别订阅 2 个*直线查找*模块输出的 1 组邻边。例如分别订阅“直线查找 3.输出直线”、“直线查找 4.输出直线”。

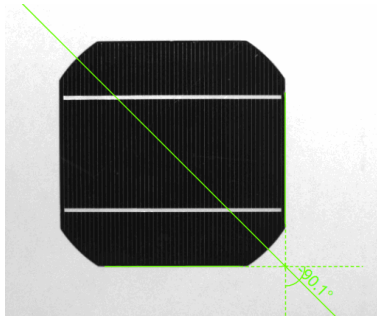


图 4-25 角平分线查找结果

3) 配置*线线测量 1*和*线线测量 2*模块并执行，分别计算 2 条中线的交点坐标和 2 条角平分线的交点坐标。

- *线线测量 1*模块的关键参数配置如下。

  输入源

  在*基本参数*页签，配置**输入源**参数为图像源图像，例如“图像源 1.图像”。

  数据来源

  选择*订阅*。

  线

  *线*参数分别订阅 2 个*中线查找*模块输出的中线。例如分别订阅“中线查找 1.中线”、“中线查找 2.中线”。

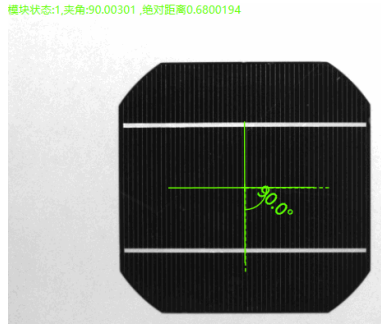


图 4-26 中线交点

- *线线测量 2* 模块的关键参数配置如下。

  输入源

  在*基本参数*页签，配置**输入源**参数为图像源图像，例如“图像源 1.图像”。

  数据来源

  选择*订阅*。

  线

  *线*参数分别订阅 2 个*角平分线查找*模块输出的角平分线。例如分别订阅“角平分线查找 1.角平分线”、“角平分线查找 2.角平分线”。

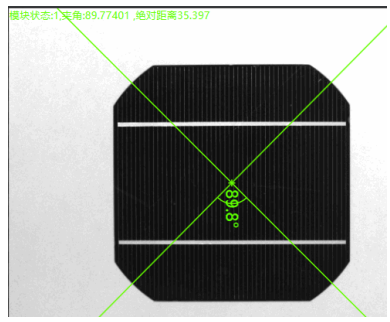


图 4-27 角平分线交点

3. 配置*变量计算*模块，计算中线交点坐标和角平分线交点坐标的平均值。

   在*基本参数*页签，添加两个名称为“var0”和“var1”的变量，分别代表平均中点坐标 X 值和 Y 值，并配置如下表达式分别计算中线交点坐标 X 值和角平分线交点坐标 X 值的平均值、中线交点坐标 Y 值和角平分线交点坐标 Y 值的平均值：

   - var0：(<16 线线测量 1.交点 X>[0]+<17 线线测量 2.交点 X>[0])/2。
   - var1：(<16 线线测量 1.交点 Y>[0]+<17 线线测量 2.交点 Y>[0])/2。

   变量计算模块配置如下图所示。



图 4-28 变量计算配置

生成电池片定位框

操作步骤

1. 双击*几何创建*，配置模块关键参数。



图 4-29 几何创建配置

2. 在*几何创建*窗口，单击*执行*和*确定*。

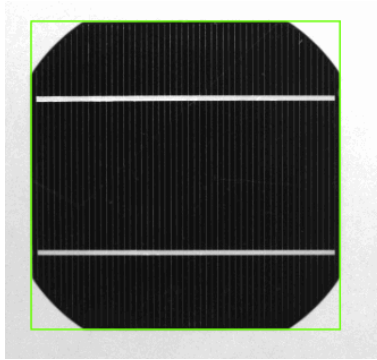


图 4-30 几何创建结果

格式化数据并输出图像

操作步骤

1. 配置*格式化*模块，格式化电池片的宽度、高度和中心点坐标数据。
   1) 双击*格式化*模块，配置模块关键参数。

      在*基本参数*页签，添加数据行用于整合检测信息，如下图所示。



图 4-31 格式化配置

   2) 在*格式化*窗口，单击*保存*。
2. 配置*输出图像*模块，将经*几何创建*模块处理的图像输出。
   1) 双击*输出图像*，配置模块关键参数。

      输入源

      在*基本参数*页签，配置*输入源*参数为图像源图像，例如“图像源 1.图像”。

      前项存储设置

在*渲染设置*页签，配置*前项存储设置*参数为“几何创建”。

2) 在*输出图像*窗口，单击*确定*。

后续处理

至此，您已完成流程搭建。您可试运行流程，并在 *图像显示区域* 和 *结果显示区域* 查看运行结果。若结果不符合预期，调试模块配置（例如运行参数），直至运行结果符合预期。
调试中建议重点关注：

- 检测精度是否已满足业务需求。在流水线运行过程中可能出现的所有场景中，目标需尽可能百分百被成功检测。
- 检测精度通常与 ROI 选取、运行参数取值和模型（特征模板）精度有关。以运行参数为例，若采用默认值运行后效果不佳，可根据 *模块使用参考* 具体模块介绍章节的参数配置说明，尝试调大或调小参数取值。

4.4.3 步骤三：设计运行界面

运行界面用于直观操控和监控方案运行，您可根据实际业务场景及需求设计运行界面。本节以“电池片检测”方案为例，指导您设计并编辑运行界面。

前提条件

已停止运行方案或流程。

说明

如果未停止运行，运行界面将无法配置。

基于“电池片检测”方案，假设需要在运行界面实现以下需求。以下操作任务以实现以下需求为目的。

- 控制方案运行与停止。
- 查看包含电池片角平分线和电池片四条边线的检测结果图。
- 查看*输出图像*模块的结果图像。
- 显示电池片中心、高度、宽度数据。

说明

您也可以根据实际需求自定义运行界面，详情请参见 *方案搭建-运行界面设计* 。

操作步骤

1. 在主界面快捷工具条，单击 。

   即可打开*运行界面设计*窗口。

2. 配置画布布局。

   1) 单击 *分页布局区域 → 2 画面* 。

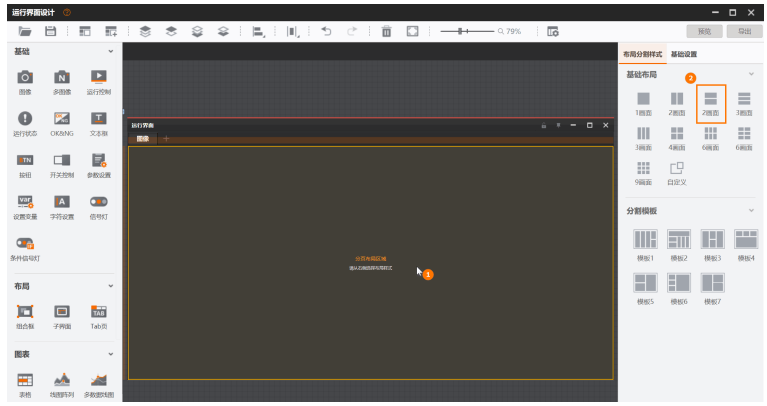


图 4-32 配置画布整体布局

2)选中画布第二行区域，在*布局分割样式*页签选择 *2 画面*。

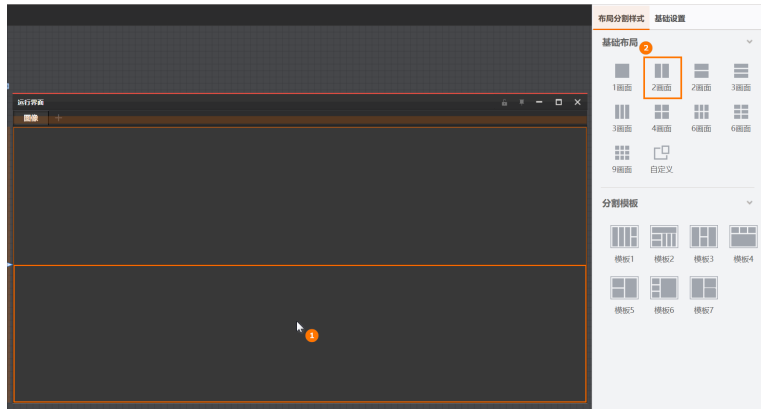


图 4-33 配置画布第二行区域布局

3)通过网格线微调画布每个区域的大小。

画布布局整体效果如下图所示。

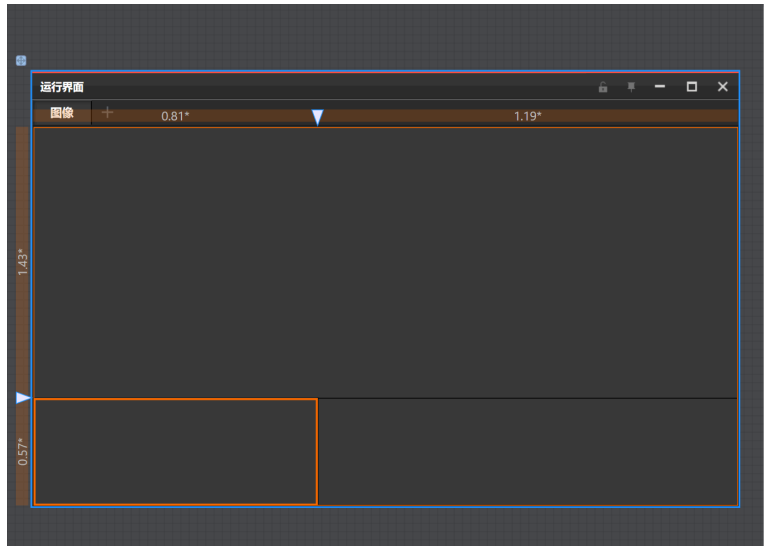


图 4-34 画布布局

3. 配置控件。

1) 在左侧控件区，选中*多图像*控件并拖入画布的右侧区域，并配置控件参数。

控件关键参数配置如下，其他参数采用默认值即可。

画面数量

在下拉框中选择“1X2”。

数据源

单击*配置*，进入*配置*窗口，配置画面的**图像源**和**图形源**参数。

- 画面 1：

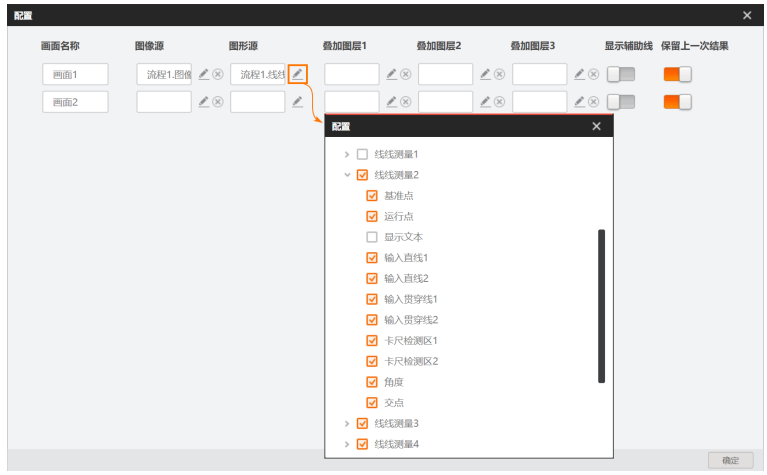


图 4-35 多图像控件图形源配置

a. 单击*图像源*列的 ，选择 *流程 1 → 图像源 1 → 输出图像* 。
b. 单击*图形源*列的 ，勾选*线线测量 2*、*线线测量 3* 和*线线测量 4* 前的复选框，并去勾选线线测量 2、3、4 中的*显示文本*，如上图所示。

- 画面 2：单击*图像源*列的 ，选择 *流程 1 → 输出图像 1 → 输出图像* 。

自动填充

开启*自动填充*开关，使*多图像*控件填满画布右侧区域。

2) 在左侧控件区，选中*运行控制*控件拖入画布的左下角区域，并配置控件参数。

在属性设置区域开启*自动填充*开关，使*运行控制*控件填满左下角区域。

说明

您也可以关闭*自动填充*开关后，通过*尺寸*和*边距*参数或鼠标悬浮在控件边框上拖动双箭头调整控件显示大小。

3) 在左侧控件区，选中*文本框*控件并拖入画布的右下角区域，并配置控件。

此文本框用于显示“检测结果：”，控件关键参数如下，其他参数采用默认值即可。

字符信息

文本框中输入“检测结果：”。

尺寸

调整尺寸数值为 190*30, 您也可以鼠标悬浮在控件边框上拖动双箭头调整控件显示大小。

字体

在下拉框中**字体**选择 22。

4) 在左侧控件区，选中*文本框*控件并拖入画布的右下角区域，并配置控件参数。

此文本框用于显示检测结果数据，置于“检测结果：”文本框下，控件关键参数如下，其他参数采用默认值即可。

数据源

单击 ...，在配置窗口选择 *流程 1 → 格式化 1 → 格式化结果* 。

数据绑定

文本框中删除“格式化结果:”，即只保留“<%s>”。

尺寸

调整尺寸数值为 400*100，您也可以鼠标悬浮在控件边框上拖动双箭头调整控件显示大小。

字体

在下拉框中**字体**选择 22。

配置控件后效果如下图所示。

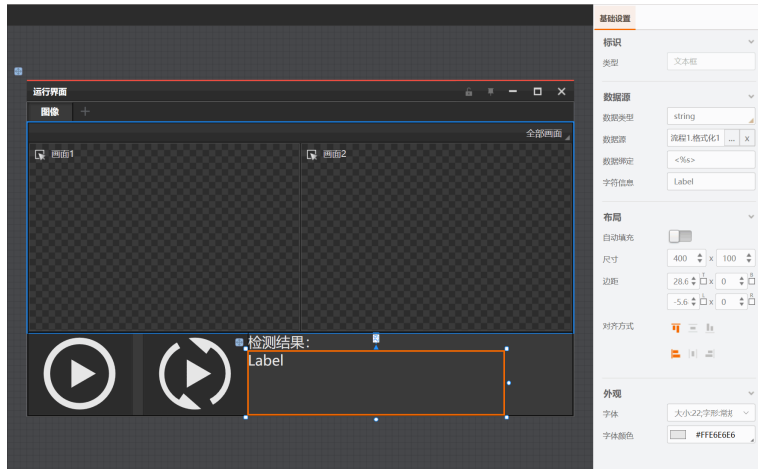


图 4-36 控件配置结果

4. 在运行界面设计窗口右上角，单击*预览*。

在预览窗口，单击执行按钮可通过综合运行界面查看方案运行效果，如下图所示。

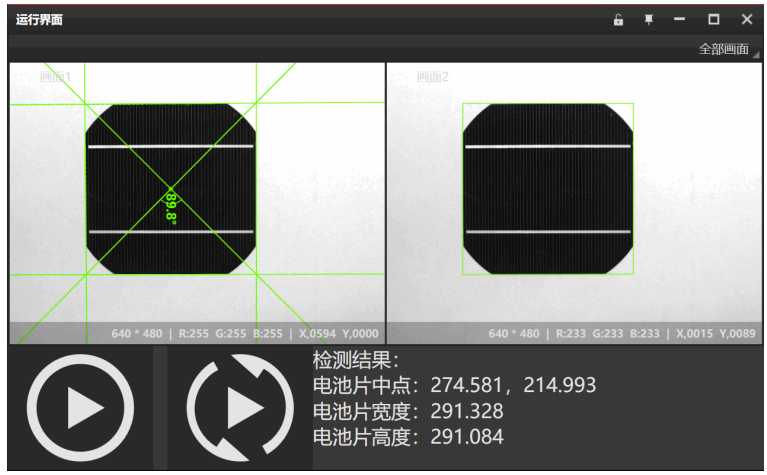


图 4-37 预览运行界面

说明

若结果不符合预期，调试控件配置参数（例如位置、大小、数据源等），直至符合预期。

5. 单击*运行界面设计*窗口右上角的 ✕，并在弹出的提示窗口上选择*是*，保存运行界面设计。

后续处理

至此，您已完成该入门级方案的搭建，可前往运行界面运行该方案，详情请参见 *运行入门级方案*。

4.5 运行入门级方案

完成“电池片检测”入门级方案搭建之后，即可运行该方案，并查看该方案运行效果。

操作步骤

1. 在主界面的快捷工具条上单击 / 单次试运行或连续试运行方案。

   若运行效果欠理想，继续调整流程和模块配置，直至方案运行效果达到预期。

   注意

   快捷工具条上的 / 与流程页签上 / 分别用于运行方案和流程，请勿混淆。

2. 在菜单栏选择 *文件 → 保存方案* 或 *文件 → 方案另存为* 保存方案。
3. 单击主界面右上角的*运行模式*，切换至已在 *步骤三：设计运行界面* 中设计完成的运行界面。
4. 在运行界面单击 / 运行方案。

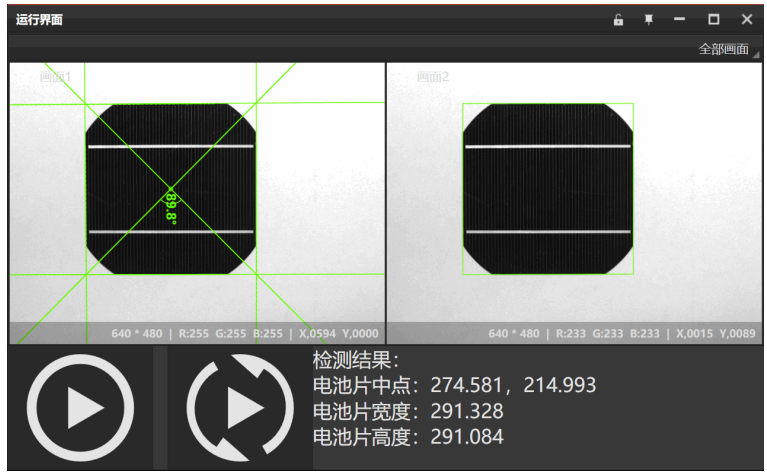


图 4-38 运行界面

至此，您已完成入门级方案——“电池片检测”的运行。若需了解方案运行更为高阶的功能，请参见 *方案调试与运行* 及其子章节。

4.6 保存方案

方案调试过程中或方案调试完成后，均建议及时保存方案，以防方案配置信息丢失。

VM 支持的方案保存操作如下。

- 单击主界面快捷工具条上的 。
- 在主界面菜单栏上选择 *文件 → 保存方案*，将方案保存至默认的保存路径。

- 方案首次保存时，需先选择保存路径。并可通过*加密选项*设置方案是否加密（见下图）。方案加密保存后，载入该方案时需输入密码。

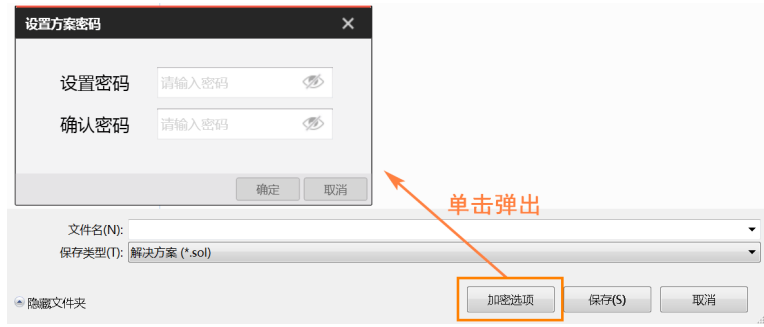


图 4-39 方案加密设置

> **说明**
>
> 当您完成方案密码设置后，还可使用 *加密狗写密码工具* 设置加密狗密码，完成双重加密设置。VM 在保存方案时默认使用已设置的加密狗密码和方案密码进行加密。若需载入双重加密的方案，您需满足如下两个条件：
>
> › 工控机上插入的加密狗的密码，与通过 *加密狗写密码工具* 设置的密码相同。
>
> › 输入的方案密码正确。

- 方案非首次保存时，直接将当前方案更新保存至上述保存路径。

> **说明**
>
> 如果方案保存失败，请联系技术支持获取帮助。

- 在主界面菜单栏上选择 *文件 → 方案另存为*，将方案保存至默认路径以外的路径。

4.7 进阶参考

了解如何搭建并运行入门级方案之后，您可参考如下信息，学习搭建更为复杂或更贴合业务需求的方案。

搭建入门级方案 的方案示例中，未涉及全局/局部逻辑和通信等复杂逻辑。但在实际业务中，复杂逻辑在实现业务需求上至关重要。

您可参照如下流程图搭建更加复杂的方案。



图 4-40 流程概览

上图中：

- 配置方案相关设备：硬件设备添加及配置相关详情，请参见 *方案搭建-硬件设备配置* 。
- 搭建流程：流程配置和流程逻辑配置相关详情，请参见 *方案搭建-流程配置* 。
- 配置全局逻辑：全局脚本、全局变量、全局触发等全局逻辑相关详情，请参见 *方案搭建-全局与局部逻辑配置*

说明

  - 针对单个流程和单个 *Group* 模块支持配置局部变量，局部变量仅在方案内的单个流程或单个 *Group* 模块维度生效。局部变量详情请参见 *全局/局部变量* 。
  - “上下相机映射对位”、“钢筋计数&横截面积计算”、“胶路及表面缺陷检测”、“TRY 盘物料定位抓取”和“机械 8421 码识别”等示例方案中配置了全局变量和全局触发逻辑，供您学习和参考。打开示例方案请参见 *体验示例方案* 。

- 设计运行界面：设计窗口概览、通用设计思路和控件相关详情，请参见 *方案搭建-运行界面设计* 。

说明

示例方案中每个方案已配置对应的运行界面，您可以参考示例方案设计运行界面，打开示例方案请参见 *体验示例方案* 。

第 5 章 方案搭建-硬件设备配置

在视觉系统中按需部署完相机、PLC 和视觉控制器等硬件设备后，您需将其关联至 VM，并进行相应配置，建立硬件设备与 VM 服务端的通信。以相机为例，将其关联至 VM 后，相机实时采集的图像数据可传输至 VM 进行视觉处理。

5.1 相机管理

VM 可添加全局相机（即可在单个方案内全局应用的相机）、HKR 采集卡连接的各接口相机、Dalsa 采集卡连接的 Camera Link 相机和 Matrox 采集卡连接的 CameraLink 相机。

说明

各种采集卡连接的相机本质上也同属全局相机，可全局应用于方案中的所有流程，作为流程的图像输入源。当您添加全局相机后，可采用相机硬触发或软触发的方式采集输入图像。当相机硬触发时，对应流程将自动执行，需注意此时仅首个绑定全局相机的图像源所在的流程能够被触发，通过工具栏的*单次运行*或*连续运行*按钮将无法控制其他流程的触发顺序。建议您在使用相机硬触发方式获取输入图像时，一个流程对应一个相机，确保对应流程正确执行。

5.1.1 添加相机

请参照如下步骤添加相机。上述相机的添加步骤基本相同。

前提条件

请确保相机已上电并处于可用状态，同时确保相机 IP 配置正确。

说明

除了本节所介绍的相机添加方式，还可：

- 通过*图像源*模块的**关联相机**参数添加相机。
- 在 *步骤二：编排相关实体连接* 时添加全局相机。

操作步骤

1. 单击主界面快捷工具条的 打开*相机管理*窗口。
2. 在该窗口单击 打开相机类型选择对话框。
3. 选择相机类型，并单击*确定*。

HKR 采集卡

通过 HKR 采集卡连接的各接口相机。

全局相机

在单个方案内可进行全局应用的相机。

Dalsa 采集卡

通过 Dalsa 采集卡连接的相机。

Matrox 采集卡

通过 Matrox 采集卡连接的相机。

*相机管理*窗口的设备列表中显示当前步骤中所选类型的相机。

4. **可选操作**: 在设备列表中右键单击相机，并单击*重命名*或*删除*将相机重命名或删除。
5. 按需配置相机参数，并单击*确定*。

不同类型相机的配置有所差异，具体见如下说明。

- *配置 HKR 采集卡连接的相机*
- *配置全局相机*
- *配置 Dalsa 采集卡连接的相机*
- *配置 Matrox 采集卡连接的相机*

5.1.2 配置自定义参数相机

当您添加自定义参数相机后，可根据实际需求添加自定义相机参数并创建参数分组，以便快速管理高频使用的相机参数。

常用参数

相机连接

选择相机

选择相机进行连接。软件将自动枚举在同一网段内的所有相机，您可以在下拉列表中查看当前枚举到的所有相机。若未在下拉列表中找到目标相机，可单击 刷新相机列表。

相机参数

单击*相机参数*进入相机参数配置界面，查看或编辑当前连接相机的属性。右键单击任一属性，并选择**添加至收藏夹**，可将高频使用的相机参数添加到*自定义参数*。

说明

关于如何设置相机参数的说明，可查看当前相机对应的用户手册或查看该相机客户端用户手册中的对应内容。

实时取流

单击*实时取流*，在弹出的*相机预览*窗口查看相机实时采集的图像，并判断当前图像是否符合要求。

说明

仅支持组播的相机才能进行实时取流操作。

自定义参数

在*自定义参数*，推荐您按照以下步骤管理已添加的自定义相机属性。

说明

自定义参数属于全局相机，修改连接的相机，自定义参数不变。

1. 单击*相机参数*进入相机参数配置界面，可继续收藏相机属性或调整相机属性值。

   说明

   关闭相机参数配置界面时，所选参数将自动同步至*自定义参数*的**默认分组**下。
2. 单击*实时取流*，在*相机预览*窗口预览图像，实时查看相机参数的调整效果。在该界面左下角，可查看相机的**丢包数**和**图像数**。
3. 单击*自定义编辑*进入*配置*界面，按需添加相机属性分组。该界面支持完成以下操作：
   - 新增分组：单击*添加分组*可添加一个分组。
   - 删除分组或参数：单击指定分组或参数右侧的 ⊗ 按钮，即可删除所选内容。

     说明

     删除所选分组的同时，将删除该分组下的所有参数。
   - 调整参数分组：勾选指定分组左侧的单选框，单击*勾选参数移动至*并在下拉列表中选择目标分组，可一键将已收藏的所有相机参数从当前分组迁移至另一分组。

5.1.3 配置全局相机

添加相机时，如果相机类型选择为**全局相机**，则需按照实际应用场景配置本节介绍的所有参数。全局相机可配置的参数，按大类可分为常用参数和设备控制类参数。常用参数可进一步分为相机参数、图像参数、触发参数和软件参数等。

常用参数

相机连接

GenTL 相机

开启 **GenTL 相机**时，需**选择 Cti 文件**，并**选择相机**进行连接。

选择 Cti 文件

从本地路径加载.cti 文件。

选择相机

选择相机进行连接。开启 **GenTL 相机**时的，该下拉列表中匹配的相机均为符合 GenTL 标准的相机。如果未在下拉列表中找到目标相机，可单击 刷新相机列表。

类型选择

若您未开启 **GenTL 相机**，可自行选择连接相机的方式。

枚举

类型选择默认为**枚举**，此时直接在*选择相机*下拉框中选择同网段内未被占用的相机即可。

跨网段 IP

通过相机 IP 地址手动连接跨网段相机。需设置如下参数。

本机 IP

安装 VM 的计算机的 IP 地址。

相机 IP

待连接的其他网段内的相机的 IP 地址。

连接相机

如果开启，则 VM 与目标相机建立连接。如果关闭的，则断开连接。

自动重连

如果开启，则 VM 在相机掉线时自动重连该相机。

加载用户集 1

设置相机自动加载*用户集 1* 中保存的参数。开启后，当您手动连接相机、加载包含相机设置的方案或遇到相机掉线重连时，相机将自动加载*用户集 1* 中保存的参数。

实时取流

单击*实时取流*，可在弹出的预览窗口查看相机采集的实时图像，判断图像是否符合要求。

说明

仅支持组播的相机才能进行实时取流操作。

当相机图像结果中出现“图像获取超时”或“图像显示区画面丢失”等问题时，可通过*实时取流*查看丢包状况。

相机参数

单击*相机参数*进入相机客户端的参数设置窗口，调整相机参数。

说明

- 部分参数必须在停止取流后才能配置，具体请以界面实际显示为准。
- 更多有关参数设置窗口的使用说明，请参见所使用相机的客户端用户手册。更多参数介绍详情，请参见所使用相机的用户手册。

保存用户集 1

单击*保存用户集 1*，可将已设置相机参数保存至用户集。

触发设置

开始采集

该功能一般在相机通过触发模式出图时使用。启用该功能后，相机收到触发信号可出图；否则，相机收到触发信号也无法出图。

说明

- VM 执行方案时，以软触发方式取图。软触发时，该参数若为关闭状态，则会自动开启。
- 该参数默认开启，但选择相机后需 1~2 秒的时间方可生效。部分相机参数设置时需先先关闭该功能。

触发模式

设置是否开启触发模式。

触发源

根据需求选择相机出图的触发源。

- 选择 SOFTWARE 时，VM 每次执行流程时给相机发送软触发信号。
- 选择其他信号源时，触发信号来源于 IO 接线连接的外部设备。

触发延迟

设置相机接收到触发信号后，延迟多长时间（单位：μs）响应出图。

条件触发数

可设置软件接收到多少张相机图像后才开始触发流程执行。

假设设置为 5，则软件接收到 5 张相机图像后，触发流程连续运行，流程运行 5 次后停止运行。

说明

该参数仅在**触发源**选择硬触发信号时有效。

条件触发超时

可设置软件等待**条件触发数**图像的时间，单位为 ms。若等待超时时间后，相机图像未达到设置的**条件触发数**，也进行流程触发，流程触发次数为实际采集到的图像张数。

说明

该参数仅在**触发源**选择硬触发信号时有效。

图像参数

图像宽度
图像高度

查看并设置当前被连接相机的图像宽度和高度。

像素格式

查看并设置当前选择相机的像素格式。

插值算法类型

设置像素转换时使用的图像插值算法。可从以下选项中选择合适的图像插值算法。

Quick

快速算法，计算复杂度低，运算速度快，插值效果较差，适用于对图像处理效率有较高要求的场景。

Balance

均衡算法，默认选项，计算复杂度中等，兼顾插值效果与效率，可在保证较高处理效率的同时，较好地保留图像纹理细节，适用于大部分场景。

Optimal

最优算法，计算复杂度较高，运算速度较慢，图像高频纹理区域细节还原较好，适用于对图像高频纹理细节还原有较高要求的场景。

OptimalPlus

最优+算法，计算复杂度高，运算速度慢，图像高频纹理区域细节还原最好，适用于对图像高频纹理细节还原有高要求且不考虑图像处理效率的场景。

帧率

设置当前连接相机的帧率。

实际帧率

查看当前选择相机的实际帧率。

曝光模式

查看并设置当前选择相机的曝光模式，包括 **OFF**、**ONCE** 和 **CONTINUOUS** 三种。

OFF

即手动曝光模式。相机根据下文提及的**曝光时间**取值进行曝光。

ONCE

即单次自动曝光模式。相机根据场景亮度变化自动调整曝光时间。相机执行一次曝光后，自动切换为手动曝光模式 。

CONTINUOUS

即连续自动曝光模式。相机根据场景亮度变化自动调整曝光时间，连续执行曝光。

曝光时间

查看并设置当前选择相机的曝光时间（单位：μs）。

增益

增益

查看并设置当前选择相机的增益。当**增益模式**选择 OFF 时，可手动修改增益。

增益模式

查看并设置当前选择相机的增益模式，包括 OFF、ONCE 和 CONTINUOUS 三种。

OFF

即手动增益模式。相机根据手动设置的**增益**取值调整模拟增益。

ONCE

即单次自动增益模式。相机根据场景亮度变化自动调整模拟增益，自动调整一次后自动切换为手动增益模式。

CONTINUOUS

即连续自动增益模式。相机根据场景亮度变化，连续自动调整模拟增益。

最小增益
最大增益

增益模式设置为 CONTINOUS 时，可设置增益范围的最小值和最大值。

Gamma

Gamma 使能

开启后，可调整相机的 Gamma 数值。

Gamma

查看并设置当前选择相机的 Gamma 数值。通常相机图像传感器的输出与照射在图像传感器上的光子是线性的，Gamma 可提供一种输出非线性的映射机制。Gamma 数值在 0 ~ 1 之间，图像暗处亮度提升；数值在 1 ~ 4 之间时，图像暗处亮度下降。

选择器

可设置为用户或 sRGB。

用户

可自行设置 Gamma 数值。

sRGB

Gamma 数值默认为标准协议模式定义的数值，不可自行设置。

线阵相机参数

当连接的相机为线阵相机时，可配置或查看**行频**和**实际行频**。

行频

设置线阵相机的行频大小。

实际行频

显示当前相机运行的实际行频大小。

软件参数

相机类型

选择当前连接的相机的类型，可选**标准面阵相机**、**智能相机**和**线阵相机**。其中，**智能相机**表示 X86 智能相机和基于 X86 开放平台的智能相机。

有效电平

设置相机输出结果时所需要的逻辑电平类型。该参数定义了在满足*相机 IO 通信*模块设置的输出条件时，相机输出的电平信号类型。为确保电平信号被准确识别，选择**高电平有效**时，相机模块会将相机当前电平状态置为低电平；选择**低电平有效**时，相机模块会将相机当前电平状态置为高电平。

高电平有效

满足上述条件时，相机输出高电平。

低电平有效

满足上述条件时，相机输出低电平。

断线重连时间

设置相机断线后，自动重连的超时时长（单位：秒）。如果达到该时长相机仍未重连成功，则不再继续进行重连。

软触发超时

软件主动触发相机执行一次图像采集时，相机可能因为内部故障或网络故障，需要很长的时间响应。此处设置的超时时间（单位：秒）可用于避免流程因相机未响应而卡顿过久。

如果软触发后相机未在超时时间内采集图像，流程将以无图像源的形式执行并报错（NG）。

图像缓存数量

设置相机输出的图像在软件中缓存的数量，一般在硬触发时使用。

等待连接时间

设置软件在完成方案加载后等待连接相机的时间，单位为毫秒。

分时频闪拆图

请结合实际场景选择是否开启分屏采集图像。开启后，单台设备可同时采集多张图像，以适配不同检测需求。

设备控制

IO 控制

可配置相机输入/输出控制相关参数，具体如下。

IO 选择项

连接面阵相机或线阵相机时，有该参数。可通过该参数选择硬触发的信号源。

IO 模式

连接面阵相机或线阵相机时，有该参数。可查看 **IO 选择项**选择的信号源模式（**input** 或 **strobe**）。部分相机的信号源可自行切换。

IO 输出选择器

连接智能相机时，有该参数。可选择信号源作为输出。

IO 输出源

连接智能相机时，有该参数。可设置为 **OFF** 或 **SOFTWARETRIGGER**。

OFF

关闭 **IO 输出选择器**中选择的输出信号源。

SOFTWARETRIGGER

开启 **IO 输出选择器**中选择的输出信号源，且可配置如下参数。

反相输出

设置是否输出反相的电平信号。

电平触发

单击***电平触发***可使相机输出电平信号。

光源控制

连接的相机为智能相机时，可配置如下参数控制相机光源。

光源使能

连接智能相机时，有该参数。开启后，相机光源被点亮。开启前，请确保相机光源已完成部署和配置。

光源模式

连接智能相机时，有该参数。可将光源模式设置为 **strobe** 和 **constant**。

strobe

设置为光源频闪模式。设置为频闪后，可设置**光源延迟时间**。

光源延迟时间

设置相机光源频闪的延迟点亮时间。

constant

设置为光源常亮模式。

光源持续时间

设置相机光源点亮的持续时间。

光源提前时间

设置光源相对于相机曝光提前点亮的时间。

自定义参数

全局相机的自定义参数配置步骤和自定义参数相机一致。操作详情请参见 *配置自定义参数相机* 中的 *自定义参数* 。

5.1.4 配置 HKR 采集卡连接的相机

先*选择采集卡*再*选择相机*即可连接相机。

连接相机后，可设置相关参数。

配置文件

可将相机的参数以 hcf 文件形式导入，从而修改相机相关参数。

说明

hcf 文件通过采集卡客户端导出属性的方式生成。

图像宽度
图像高度

可分别查看并设置相机采集图像的宽度/高度。

脉冲数

可读取相机设置的脉冲数情况。

触发源

按需选择相机出图的触发源。

触发选择器

可选 FrameBurstStart（帧触发）或 LineStart（行触发）。

说明

面阵相机只有帧触发选项。

触发模式

设置是否开启触发模式。

5.1.5 配置 Dalsa 采集卡连接的相机

需先通过**配置文件**加载采集卡的.ccf 文件，再选择需连接的相机。连接相机后，可通过**缓存张数**调整采集卡的图像缓存空间。

说明

Dalsa 采集卡连接的相机仅支持硬触发出图，不支持软触发。

配置文件

从本地路径上传 Dalsa 采集卡的.ccf 文件。

说明

连接相机前请务必先配置**配置文件**。未配置**配置文件**的情况下连接相机将报错。

选择相机

上传采集卡的.ccf 文件后，此处将枚举出该采集卡连接的相机。如未成功枚举出相机，可单击 刷新。

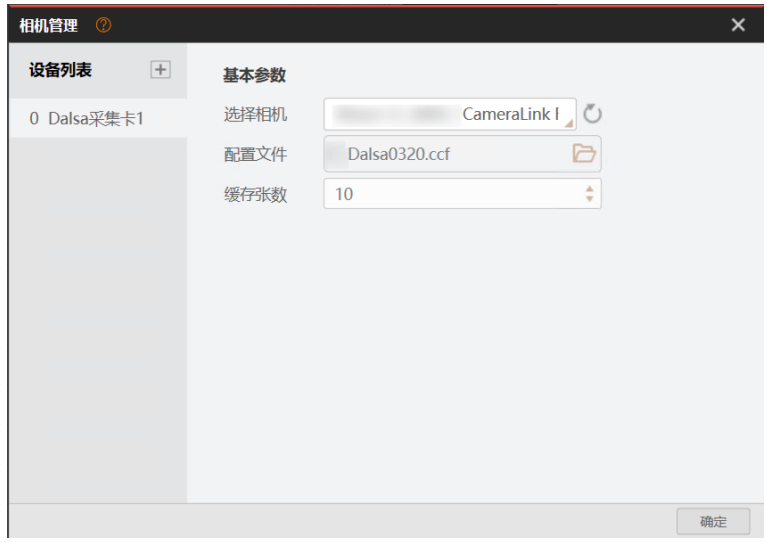


图 5-1 Dalsa 采集卡配置

5.1.6 配置 Matrox 采集卡连接的相机

如需将 Matrox 采集卡连接的相机添加至 VM，需先通过**配置文件**加载采集卡的.dcf 文件，再选择需连接的相机。连接相机成功后，可配置图像参数和增益。

说明

Matrox 采集卡连接的相机目前仅支持硬触发出图，不支持软触发。

本节内容包含：

- *基本参数*
- *图像参数与增益*

基本参数

配置文件

从本地路径上传 Matrox 采集卡的.dcf 文件。

选择相机

上传采集卡的.dcf 文件后，此处将枚举出该采集卡连接的相机，可在此处的选择相机。如未成功枚举出该相机，可单击 刷新。

说明

选择相机后，可能弹出如下窗口。此时请将波特率（**Baud rate**）设置为采集卡端所设的值，即 **115200**（如下图所示）。

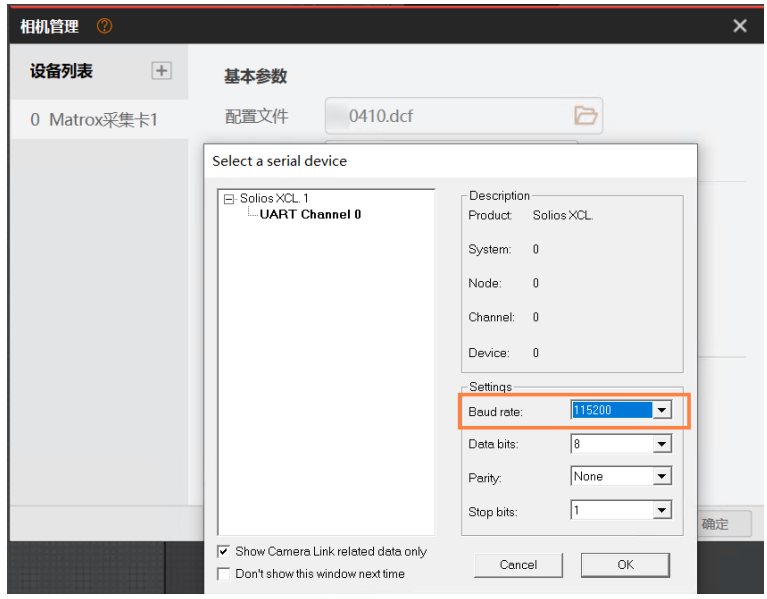


图 5-2 设置波特率

图像参数与增益

- 图像参数详情见 *图像参数* 。
- 增益详情见 *增益* 。



图 5-3 Matrox 采集卡配置

5.2 通信管理

通信管理是连接 VM 和外部设备（包含通信设备和控制器设备）的桥梁。VM 既支持读取外部设备的数据，用于实现触发相机拍照或软件运行等功能；也支持发送数据给外部设备，将软件处理结果发送给外部设备，用于控制光源和其他机构件运行等。

通信管理分为 *设备管理* 、 *接收事件* 、 *发送事件* 、触发配置（即 *全局触发* ）、 *响应配置* 和 *心跳管理* 。

通过窗口右上角的 可将通信管理中的所有配置以 gtd 格式文件导入/导出。

5.2.1 设备管理

*设备管理*可对通信设备和控制器类设备进行添加、删除、参数配置等操作。其中*通信设备列表*用于管理通信设备，*控制器列表*用于管理控制器类设备。

通信设备管理

通信设备管理可创建设备并设置相关参数，从而与对应的外部通信设备连接。

通信协议分为基本通信协议和 PLC 通信协议两大类。使用时，可根据实际情况选择不同的协议类型。

表 5-1 基本通信协议和 PLC 通信协议的区别

对比项	基本通信协议	PLC 通信协议
协议类型	底层传输协议或物理层协议	应用层或协议层协议
寄存器支持情况	不支持	支持
支持的设备类型	广泛适用于各种设备（不限于 PLC）	专为 PLC 设计，支持 PLC 特有的数据类型
支持的通信方式	单一通信方式	支持串口、TCP、UDP 等单一或多种通信方式
协议举例	TCP 客户端、TCP 服务端、UDP、串口协议	汇川 H2U、松下 MewTocol、三菱 MC、欧姆龙 Fins TCP、西门子 S7、基恩士 KV 等

基本通信协议

基本通信协议主要包括 TCP 客户端、TCP 服务端、UDP、串口通信协议，其操作方法参见 *基本通信协议* 。

表 5-2 基本通信协议介绍

协议类型	说明
TCP 客户端	TCP（Transmission Control Protocol）是一种面向连接的、可靠的、基于字节流的传输层通信协议。 使用 TCP 服务时，通信的双方，一方为 TCP 客户端，一方为 TCP 服务端。
TCP 服务端	
UDP	Internet 协议集支持一个无连接的传输协议，该协议称为用户数据报协议（UDP，User Datagram Protocol）。UDP 为应用程序提供一种无需建立连接就可以发送封装 IP 数据包的方法。
串口	串行接口简称串口，也称串行通信接口或串行通讯接口（通常指 COM 接口），是采用串行通信方式的扩展接口。串行接口 （Serial Interface）是指数据一位一位地按顺序传送。特点是通信线路简单，只要一对传输线就可以实现双向通信，从而大大降低成本，适用于远距离通信，但传送速度较慢。

PLC 通信协议

PLC 通信协议包括汇川 H2U、松下 MewTocol、三菱 MC、三菱 FX、ModBus 通信、欧姆龙 Fins TCP、西门子 S7、基恩士 KV、EtherNet/IP CIP、AB EtherNet/IP CIP 等，具体操作方法参见 *PLC 通信协议* 。

表 5-3 PLC 通信协议介绍

协议类型	说明
汇川 H2U	汇川 H2U 通信是以串行链路（485 串口通信）为基础，按照一定的规则（计算机链接协议）组合成规定的命令。
基恩士 KV	基恩士 KV 通信以串行链路（串口通信）和 TCP/IP 网络通信为基础，按照一定的规则（基恩士 KV 模式（上位链路）协议）组合成规定的命令。

协议类型	说明
松下 MewTocol	松下 MewTocol 通信是松下公司开发的一种通信协议，以串行链路（串口通信）和 TCP/IP 网络通信为基础，按照一定的规则（松下 MewTocol 协议）组合成规定的命令。具有高效性、稳定性、兼容性、安全性的特点。
三菱 MC	MC 协议开放 PLC 内部寄存器给外部设备，实现外部设备和 PLC 的数据交互。以串行链路（串口通信）、UDP 和 TCP/IP 网络通信为基础，按照一定的规则组合成规定的通讯格式。通讯内容分为二进制和 ASCII 文本。
三菱 FX	三菱 FX 系列 PLC 是一种常用的工业自动化控制设备，用于控制和监控工业自动化进程。常用的通信协议有串口通信协议和以太网通信协议。 • 串口通信协议通过串行口与上位机进行通信（包括 RS232、RS485 等协议）。在串口通信中，PLC 作为从机，上位机作为主机，PLC 需预先设置好通信参数，如波特率、数据位、停止位等，与上位机进行通信。 • 以太网通信协议可以通过以太网口与上位机进行通信，其中包括 TCP/IP 协议和 UDP 协议。
ModBus 通信	ModBus 是一种串行通信协议，是 Modicon 公司（现在的施耐德电气 Schneider Electric）于 1979 年为使用可编程逻辑控制器（PLC）通信而发表的协议。ModBus 通信以串行链路（串口通信）和 TCP/IP 网络通信为基础，在此基础上按照一定规则（Modbus 协议）组合成特定报文。
AB EtherNet/IP CIP	AB EtherNet/IP CIP 是由罗克韦尔自动化公司开发的工业协议。该协议基于标准以太网、TCP/IP 协议和 CIP 协议，兼容性高，可实现 PLC、I/O 模块等工业设备间的控制指令传输、设备组态配置及数据采集。
EtherNet/IP CIP	EtherNet/IP（Ethernet/Indstrial Protocol，以太网/工业协议）是一种基于以太网和 TCP/IP 技术的工业协议。其物理层和数据链路层使用以太网协议，网络层和传输层使用 TCP/IP 协议，应用层使用 CIP 协议。

协议类型	说明
	说明 CIP（Common Industry Protocol）是一种端对端的面向对象协议，它规范了工业设备（传感器、执行器）和高级设备（控制器）之间的连接。CIP 作为开放性应用层协议，独立于物理媒体和数据链路层。EtherNet/IP 利用标准以太网和 TCP/IP 技术传输 CIP 通信数据包。
欧姆龙 Fins TCP 欧姆龙 Fins UDP	Fins（Factory Interface Network Service）协议是欧姆龙公司所开发的一个工业自动化网络通信协议，用于实现 PLC 和各种设备之间的通信的协议。具有高效、可靠、安全的特点。Fins 协议封装在 TCP/UDP 之上。 说明 基于 TCP 的 Fins 数据包和基于 UDP 的包在头部上差异较大。
欧姆龙 HostLink	Hostlink 协议是欧姆龙公司开发的上位计算机与 PLC 设备通讯的协议，通过串行通讯的方式收、发数据。通讯命令分为：C-code 命令的 Hostlink 和 FINS 命令，也可分为网口（CMND/SEND/RECV）的 Hostlink 命令和串口的 Hostlink 命令。
横河 PClink	横河 PClink 通信以 TCP/IP 网络通信为基础，按照一定的规则组合成规定的命令。
西门子 S7 西门子 S7-200Smart	S7 通信协议是指西门子公司用于其 S7 系列 PLC 之间通信的标准通信协议。它是一种基于串行通信方式进行通信的协议，支持点对点和多点通信方式，能够实现 PLC 之间的数据传输和互相控制。 说明 S7 和 S7-200smart 的主要区别是连接的握手协议不同。

基本通信协议

本章节仅对 TCP 客户端、TCP 服务端、UDP 及串口这四种通信协议在软件中如何操作进行介绍。

前提条件

打开*通信管理*并选择*设备管理*。

操作步骤

1. 点击**通信设备列表**右侧的 ⊞ 添加设备。
2. 在弹出的设备管理页中，**协议类型**根据需求自行选择。

   说明

   本章节仅适用于**协议类型**选择 TCP 客户端、TCP 服务端、UDP 及串口的这四种类型。
3. 可选操作: **设备名称**根据需求自定义命名，默认与选择的协议类型同名。

   说明

   *设备列表*选中设备，右键单击选择*重命名*也可修改**设备名称**。
4. 根据实际情况完成*通信参数*的设置。

   不同设备的通信参数部分有所差异，部分完全一致。

   - 有所差异的参数：
     - 协议类型选择 **TCP 客户端**：需设置的参数为**目标 IP** 和**目标端口**。此处填写作为 TCP 服务端设备的 IP 地址和端口号即可。

       说明

       目标 IP 默认为 127.0.0.1，此为环回地址，主要用于网络软件测试以及本地机进程间的通信。实际使用时，需根据情况修改。协议类型选择 UDP 时，也有此类参数。
     - 协议类型选择 **TCP 服务端**：需设置的参数为**本机 IP** 和**本机端口**。此处填写本机的 IP 地址和端口号即可，作为 TCP 客户端设备的 IP 地址与端口号需与此处保持一致。
     - 协议类型选择 **UDP**：需设置的参数为**本机 IP**、**本机端口**、**目标 IP** 和**目标端口**。**本机 IP** 处填写本机的 IP 地址，**目标 IP** 处填写需接收 UDP 信息的设备 IP，**本机端口**和**目标端口**根据需求填写，未被占用即可。
     - 协议类型选择**串口**：需设置的参数为**串口号**、**波特率**、**数据位**、**校验位**、**停止位**和**字符超时（ms）**。除**字符超时（ms）**外的其他参数，根据外部设备的串口对应参数设置即可。**字符超时（ms）**参数可设置字符间的最大的延时时间。读取串口数据时，若两个字符间的传输时间差超过设置的时长，将直接返回现有的数据。
   - 完全一致的参数：
     - **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

       说明

       无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。
     - **自动重连**：当软件与外部设备断开连接时，启用该参数可自动重连。
     - **接收结束符**：启用该功能后，接收到的数据会根据设置的结束符进行数据分割，有效处理数据的黏包和分包问题。结束符可选择\r（回车）、\n（换行）、\r\n（回车换行），也可自定义设置，以右侧窗口显示的内容作为结束符使用。未启用时，每次读取到的内容不做处理，直接上传。

说明

*通信参数*也可在创建设备后，通过*设备管理*页的右侧进行修改。

5. 点击*创建*即可完成设备添加。
6. **可选操作**: 测试*接收数据*是否正常。

   若软件与 TCP 服务端设备通信正常，TCP 服务端发送的内容将在*接收数据*处呈现。勾选*16 进制显示*，可将接收的内容转换为 16 进制数据呈现。点击*清空*可清空当前*接收数据*处的内容。
7. **可选操作**: 测试*发送数据*是否正常。

   在*发送数据*处输入需发送的信息后并点击*发送*，若软件与 TCP 服务端设备通信正常，设备能接收到发送的内容。勾选 *16 进制显示*，可将发送的内容转换为 16 进制发送。点击*清空*可清空当前*发送数据*处已发送的内容。

说明

*发送数据*处仅支持输入 String 类型的字符。

8. 启用*设备列表*本次添加设备右侧的 ，则软件与外部设备进行数据传输。
9. **可选操作**: *设备列表*选中设备，右键单击选择*删除*可删除选中的设备。

PLC 通信协议

本章节主要对专为 PLC 设计的通信协议在软件中如何操作进行介绍。

前提条件

打开*通信管理*并选择*设备管理*。

操作步骤

1. 单击**通信设备列表**右侧的 添加设备。
2. 在弹出的设备管理页中，**协议类型**根据需求自行选择。
3. **可选操作**: **设备名称**根据需求自定义命名，默认与选择的协议类型同名。

说明

*设备列表*选中设备，右键单击选择*重命名*也可修改**设备名称**。

4. 根据需求选择**通信方式**。

   不同协议类型，支持选择的通信方式有所差别，具体参见下表。

表 5-4 不同协议类型支持的通信方式

协议类型	支持的通信方式
汇川 H2U	串口
松下 MewTocol	串口、TcpClient
三菱 MC	TcpClient、UDP、串口
三菱 FX	TcpClient、UDP、串口
ModBus 通信	串口、TcpClient
欧姆龙 Fins TCP	TcpClient
欧姆龙 HostLink	串口、TcpClient
欧姆龙 Fins UDP	UDP
横河 PClink	TcpClient
基恩士 KV	串口、TcpClient
AB EtherNet/IP CIP	TcpClient
EtherNet/IP CIP	TcpClient
西门子 S7	TcpClient
西门子 S7-200Smart	TcpClient

5. 根据选择的通信方式，完成相关参数的设置。

说明

- *通信参数*也可在创建设备后，通过*设备管理*页的右侧进行修改。
- 不同通信协议中需要设置的通信参数有所差异，请以实际界面为准。

需要配置的参数整体可分为各协议均需配置的“通用参数”和协议间存在差异的“更多参数”。

- **通用参数**：
  - 通信方式选择**串口**：需设置的参数为**串口号**、**波特率**、**数据位**、**校验位**、**停止位**。
  - 通信方式选择 **TcpClient**：需设置的参数为**目标 IP** 和**目标端口**。此处填写作为 TCP 服务端设备的 IP 地址和端口号即可。

说明

目标 IP 默认为 127.0.0.1，此为环回地址，主要用于网络软件测试以及本地机进程间的通信。实际使用时，需根据情况修改。通信方式选择 UDP 时也有此类参数。

- 通信方式选择 **UDP**：需设置以下参数。

  本机 IP

  本机的 IP 地址。

  目标 IP

  需接收 UDP 信息的设备 IP。

  本机端口/目标端口

  根据需求填写，未被占用即可。

- **更多参数**：

  字符超时（ms）

  设置字符间的最大的延时时间。读取数据时，若两个字符间的传输时间差超过设置的时长，将直接返回现有的数据。

  轮询间隔（ms）

  可设置开启**轮询读取**功能的寄存器地址读取数据的间隔时长。

  自动重连

  当软件与外部设备断开连接时，启用该参数可自动重连。

  机架号

  用来分配 CPU 模块、电源模块、接口模块、I/O 信号模块、通信模块、功能模块等在机架上的具体位置。此为通信设备自身参数。

  槽号

  用来标识扩展单元所在的位置。此为通信设备自身参数。

  设备地址

  设备的一个标识字段，推荐使用默认值。

  字节顺序

  可设置占内存多于一个字节类型的数据在内存中的存放顺序，可选 ABCD、BADC、DCBA、CDAB。例如数据为 0x41 0x42 0x43 0x44，则 CDAB 的数据为 0x43 0x44 0x41 0x42。

  字符串反转

  开启后可交换字符串相邻字节，如果字符串长度为奇数则末尾补零后再交换。若需先反转字符再发送给目标 PLC 设备，需开启该功能；一般场景无需开启。

说明

仅支持转换发送的字符串数据，无法转换接收到的数据。

字符串结束符

开启该功能后，发送字符串结尾默认加一个'\0'结束符。

数据转换

设置接收到的数据的显示格式。若不开启，接收的数据默认以十六进制显示；若开启，数据显示格式取决于配置的**数据类型**。例如，**数据类型**配置为 **String**，收到的数据按 ASCII 码显示。

自动关联

推荐使用该功能。该功能自动将可一起读取的地址关联起来再与 PLC 交互，以提高轮询读取效率，减少与 PLC 交互的次数。该功能仅针对读取 PLC 数据，并且相邻的地址协议和数据类型等参数需保持一致。

站号

用于标识 PLC 通讯站点的标识符，每个 PLC 都有一个唯一的 HostLink 站号。

PLC 单元号

每个 PLC 都有一个唯一的单元号，用于标识该 PLC 在通讯网络中的位置。单元号的范围是 0 到 31，其中 0 号单元为主单元，其他单元是从单元。

帧格式

表示协议的报文格式。

报文类型

设置通信数据的代码类型，可选 ASCII 数据或二进制数据。

目标网络 ID

标识通信目标设备的唯一地址。每个设备在 TwinCAT 网络中必须拥有唯一的目标网络 ID。

发送网络 ID

标识通信发起方的设备地址。每个设备在 TwinCAT 网络中必须拥有唯一的目标网络 ID。

连接 ID

用于唯一标识通信会话或链路的标识符。

SID

服务标识符。一般使用默认值。

6. 单击*创建*完成设备添加。

7. 添加*寄存器地址*并完成相关设置。
   1) 在*地址空间*选择需添加的地址，单击*+新建地址*添加地址。
   2) **设备名称**处根据需求自定义。
   3) 完成寄存器地址的**通信参数**设置。

      说明

      不同协议类型设备，添加寄存器地址时，需设置的通信参数有所差别，具体参见 *PLC 类设备寄存器地址参数介绍* 。
   4) 单击*确定*完成寄存器地址的添加。
8. **可选操作**: 完成寄存器地址配置后，还可进行数据同步和发送测试。
   - *数据同步*：读取所有寄存器地址当前的数据。
   - *发送测试*：完成相关参数配置后，单击*发送测试*测试数据发送效果。具体操作如下。
     a. 单击*发送测试*。
     b. 在**地址选择**处选择需测试的地址。
     c. 根据选择的**发送数据类型**输入*发送数据内容*。
     d. 单击*发送*。
   - *最大化*：可将地址空间部分全屏显示。可通过 Esc 按键退出全屏。
   - *清空地址*：单击*地址空间*右侧的*清空*，清空当前地址空间已配置的所有内容。
   - **测试数据收发是否正常**：若软件与外部 PLC 设备通信正常，则单击任意地址的**数据**列，即可查看当前地址中的数据。勾选 *16 进制显示*，可将当前地址中的数据转换为 16 进制数据呈现。单击*清空*可清空当前地址中的数据。
9. 启用*设备列表*本次添加设备右侧的 ，则软件与外部设备进行数据传输。
10. **可选操作**: 在*设备列表*选中设备，右键单击选择*删除*可删除选中的设备。

PLC 类设备寄存器地址参数介绍

本章节对 PLC 类设备添加寄存器地址时需设置的通信参数进行介绍。

汇川 H2U

- **站号**：用于标识 PLC 通讯站点的标识符，每个 PLC 都有一个唯一的 HostLink 站号。
- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 D 代表数据寄存器。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。

- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

基恩士 KV

- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 DM 代表数据寄存器，MR 代表辅助寄存器，W 代表链路寄存器。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。
- **16 进制接收**：开启时，可将接收的数据转换为 16 进制数据。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。
- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

松下 MewTocol

- **站号**：用于标识 PLC 通讯站点的标识符，每个 PLC 都有一个唯一的 HostLink 站号。一般情况下，HostLink 站号的范围是 1 到 32，其中 1 号站是主站，其他站是从站。
- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 D 代表数据寄存器，R 代表辅助寄存器。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit、32bit 或 BOOL。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。

- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

三菱 MC

- **帧格式**：表示协议的报文格式，可选 3E 帧、3C 帧格式 3 和 4C 帧格式 5。

说明

仅**通信方式**设置为 **TcpClient** 或 **UDP** 时，**帧格式**可选 **3E 帧**；当通信方式设置为**串口**时，**帧格式**可选 **3C 帧格式 3** 和 **4C 帧格式 5**。

- **报文类型**：设置通信数据的代码类型，可选 ASCII 数据或二进制数据。
- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 D 代表数据寄存器，M 代表辅助寄存器。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

ModBus 通信

ModBus 通信设备只支持保持寄存器。

- **主从模式**：主站具有访问从站的权限，可以主动向从站发送数据，任何一次数据交换，都由主站发起。从站不能主动向主站发送数据，无论何时，从站都处于接收状态。一般分为主机模式和从站模式，当前软件只支持作为主机的模式使用。
- **协议类型**：设置通信数据的代码类型，可选 RTU 数据或 ASCII 数据。
- **Int 类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **发送顺序**：可设置占内存多于一个字节类型的数据在内存中的存放顺序，可选 ABCD、BADC、DCBA、CDAB。例如数据为 0x41 0x42 0x43 0x44，则 CDAB 的数据为 0x43 0x44 0x41 0x42。
- **设备地址**：设备的一个标识字段，推荐使用默认值。
- **寄存器地址**：需使用的寄存器起始地址。
- **寄存器个数**：需使用的寄存器个数。从配置的寄存器地址开始计数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

EtherNet/IP CIP

- **槽号**：标识扩展单元所在的位置，设备自身参数。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 INT16、UINT16、INT32、UINT32、BOOL。
- **标签名称**：设置的变量名称，可自定义修改。
- **字节长度**：可设置读取数据的字节个数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。
- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

AB EtherNet/IP CIP

该通信协议支持的通信参数与 EtherNet/IP CIP 保持一致。更多详情，参见上文的 *EtherNet/IP CIP* 。

欧姆龙 Fins TCP、欧姆龙 Fins UDP

- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit、32bit 或 BOOL。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。
- **发送顺序**：可设置占内存多于一个字节类型的数据在内存中的存放顺序，可选 ABCD、BADC、DCBA、CDAB。例如数据为 0x41 0x42 0x43 0x44，则 CDAB 的数据为 0x43 0x44 0x41 0x42。
- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 DM 代表数据寄存器，WR 代表辅助寄存器。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

欧姆龙 HostLink

欧姆龙 HostLink 通信设备只支持数据寄存器。

- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。
- **站号**：用于标识 PLC 通讯站点的标识符，每个 PLC 都有一个唯一的 HostLink 站号。一般情况下，HostLink 站号的范围是 1 到 32，其中 1 号站是主站，其他站是从站。
- **PLC 单元号**：每个 PLC 都有一个唯一的单元号，用于标识该 PLC 在通讯网络中的位置。单元号的范围是 0 到 31，其中 0 号单元为主单元，其他单元是从单元。
- **上位机单元号**：上位机标识，一般用默认值即可。
- **设备标识符**：设备的一个标识字段，推荐使用默认值。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

横河 PClink

- **CPU 号**：根据设备自身参数设置 CPU 模块，CPU1 为 CPU 模块 ,CPU2 为附加 CPU 模块 1，CPU3 为附加 CPU 模块 2，CPU4 为附加 CPU 模块 3。推荐使用默认值。
- **协议类型**：设置通信数据的代码类型，可选 RTU 数据或 ASCII 数据。
- **Int 类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 D 代表数据寄存器。
- **寄存器地址**：需使用的寄存器起始地址。
- **寄存器个数**：需使用的寄存器个数。从配置的寄存器地址开始计数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

说明

关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

西门子 S7、西门子 S7-200Smart

- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。DB 代表数据寄存器，M 代表辅助寄存器。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit、32bit、BOOL 或 8bit。
- **DB/M**：对应**软元件类型**选择寄存器的数据存储区。
- **偏移量**：表示数据在 DB 块中的存储位置。偏移量为整数，表示数据距离 DB 块起始地址的字节数。

说明

仅**软元件类型**选择 DM 时可设置。

- **字节长度**：可设置从偏移量开始需要操作的字节个数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。

- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。

  说明

  仅**软元件类型**选择 DM 时可设置。

- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

  说明

  关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

  说明

  无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

三菱 FX

- **站号**：用于标识 PLC 通讯站点的标识符，每个 PLC 都有一个唯一的 HostLink 站号。
- **软元件类型**：表示寄存器类型，是可通过 CPU 模块直接访问智能功能模块的缓冲存储器的软元件。 D 代表数据寄存器。
- **整型数据类型**：用于设置整型数据占用的内存大小，可选 16bit 或 32bit。
- **软元件地址**：需使用的软元件起始地址。
- **软元件点数**：需使用的软元件个数。从配置的软元件地址开始计数。
- **超时时间（ms）**：表示该地址进行读写操作时，等待 PLC 应答的最大时间。超过该时间，返回失败；在时间段内收到数据，则立即返回。
- **关联地址**：推荐使用该功能。该功能可将此地址会随同关联地址一起与 PLC 交互从而获取数据。可提高轮询读取效率，减少与 PLC 交互的次数。关联地址时，建议选择比较接近的地址作为轮询地址，建议相差在 20 个地址以内。
- **轮询读取**：开启后，软件会主动实时从 PLC 设备获取该地址的数据。

  说明

  关联地址和**轮询读取**功能互斥，使用时只能二选一开启或都不开启。

- **轮询数据去重**：开启后，寄存器比对获取到的当前与历史数据，仅在数据变动时上传最新数据至软件。
- **数据上传**：启用该功能时，外部通信设备的数据可上传给软件，进行显示或实现相关功能。例如设备管理处显示接收到的数据、还可传递给全局脚本、全局触发的字符串触发、方案自动切换等。未启用时，数据不会上传给软件。

说明

无论该参数是否启用，不影响接收事件处接收外部通信设备的数据。

控制器管理

控制器列表可连接光源控制器或视觉控制器，设置控制器的光源或 I/O 功能。

在*设备管理*中**控制器列表**右侧单击 ⊞ 即可添加控制器，根据实际需求选择对应的**控制器协议**并完成相关参数设置，最后通过控制器列表处对应设备的 启用即可。

说明

设备列表选中控制器后，右键单击可对设备进行*删除*或*重命名*操作。

不同品牌的控制器，需设置的参数有所差别，主要分为 *控制器参数* 、 *IO 输出* 、 *IO 输入* 和 *光源参数*，具体参见下表。

表 5-5 不同品牌控制器参数支持情况

品牌类型	说明	控制器参数	光源参数	IO 输出	IO 输入
VC2000/2100	对应 VC2000/2100 系列视觉控制器	支持	支持	支持	支持
VC3000H/X	对应 VC3000 系列中型号带 H 或 X 的视觉控制器	支持	支持	支持	支持
VC3000	对应 VC3000 系列视觉控制器的自带 IO	支持	不支持	支持	支持
VC3000（IO 扩展板）	对应 VC3000 系列视觉控制器的 IO 扩展模块	支持	不支持	支持	支持
VC3000(Light 扩展板)	对应 VC3000 系列视觉控制器的光源扩展模块 说明 该控制器品牌需在设备固件升级后方可使用。如需了解更多详情，请联系技术支持。	支持	支持	支持	不支持
VC4000	对应 VC4000 系列视觉控制器	支持	支持	支持	支持

品牌类型	说明	控制器参数	光源参数	IO 输出	IO 输入
	说明 该控制器品牌需在设备固件升级后方可使用。如需了解更多详情，请联系技术支持。				
VC5000	对应 VC5000 系列视觉控制器	不支持	支持	支持	支持
VB2200	对应 VB2200 系列中的 VB2210 和 VB2220 视觉控制器	支持	支持	支持	支持
VB2230	对应 VB2200 系列中的 VB2230 视觉控制器	支持	支持	支持	支持
VT2000	对应系列的光源控制器	支持	不支持	支持	支持
VT3000	对应系列的光源控制器	支持	不支持	支持	支持
HKR 光源控制器	对应系列的光源控制器	支持	支持	不支持	不支持
DPS2	对应系列的光源控制器	支持	支持	不支持	不支持
MV-AP1024-2T	对应系列的光源控制器	支持	支持	不支持	不支持
MV-LEVD-125-4-SY	对应型号的光源控制器	支持	支持	不支持	不支持
MV-LEVD-200-6-SY	对应型号的光源控制器	支持	支持	不支持	不支持

说明

选择品牌为视觉控制器时，需确保软件安装在对应的视觉控制器上使用。选择光源控制器时，需确保安装软件的工控机已通过串口连接对应的光源控制器。

控制器参数

控制器参数主要用于将控制器与软件通过串口建立通讯，从而可设置光源参数、IO 输出和 IO 输入。相关参数介绍如下。

说明

品牌选择 **VC3000** 时，只需选择是否开启*自动重连*。

类型

说明

品牌选择 **HKR 光源控制器**时，需配置此参数。

选择该品牌下具体设备型号，可选择 MV-LE100-60W24-4D、MV-LE100-200W24-2BD、MV-LE100-500W24-1ED、MV-LE200-48W24-2D、MV-LE200-90W24-6D、MV-LE200-120W24-4TD、MV-LE200-200W24-4TD、MV-LE201-10W5-2D、MV-LE201-30W5-2FD、MV-LE201-200W48-2TD、MV-LE201-500W48-2TD、MV-LE201-750W48-2TD、MV-LE201-1200W48-4T、MV-LE201-2000W48-4T、MV-LE202-300W24-1T 或 MV-LAS-808-50W-LV0。

串口号

根据实际连接使用的端口选择**串口号**。

不同系列视觉控制器出厂默认控制 IO 及光源的串口号信息如下：

- VB2200、VB2230、VC2000/2100：对应 COM2。
- VC3000(Light)及 VC3000(IO)：与 IO 模块上的拨码相关，具体请查看 VC3000 系列视觉控制器的用户手册。
- VC3000H/X、VC4000：对应 COM3。

波特率

用于衡量符号传输速率的参数，表示每秒钟传送符号的个数。

说明

使用串口通信时，需确保两个进行通信的端口**波特率**、**数据位**、**校验位**和**停止位**参数的设置完全匹配。

数据位

用于衡量通信中实际数据位的参数。

校验位

表示串口通信的检错方式，一般分为无、奇、偶、高、低这几种情况。

停止位

表示单个数据包的最后一位，一般为 1、1.5、2。

自动重连

打开此开关，若软件检测到控制器失去连接，会自动进行重连，直至重新连接上控制器。

IO 输出

IO 输出可设置视觉控制器的 IO 输出信号。选择不同品牌的控制器，具体的 IO 输出参数有所差别，请以实际显示的 IO 输出参数为准。

说明

仅在控制器品牌选择 VB2200、VB2230、VT2000、VT3000、VC2000/2100、VC3000、VC3000（IO 扩展板）、VC4000 或 VC5000 时，支持该功能。

相关参数介绍如下：

输出类型

可设置满足绑定的条件时，各个 IO 输出信号跟按照设置输出电平信号。可选 **OK/NG 时输出**。

输出极性

可设置 IO 信号的输出时的极性，可选 **NPN** 或 **PNP**。

说明

该参数仅选择 VC3000（IO 扩展板）、VC3000 和 VC5000 品牌时，可进行设置。

发送间隔（ms）

可设置两个相邻信号的时间间隔。

说明

控制器选择 VC3000 或 VT3000 时，无该参数。

延迟时间（ms）

可设置信号延迟多长时间输出。

各个 IO 输出信号

可设置各个 IO 端口的**持续时间（ms）**和**有效电平**。使用前需确保选择的 IO 端口号与控制器上实际的 IO 输出接口对应，相关内容请查看对应系列视觉控制器的用户手册。

说明

不同品牌控制器的 IO 输出信号路数有所差异，请以实际显示为准。

持续时间（ms）

可设置 IO 信号输出时的持续时间。

说明

品牌选择 VC3000 或 VT3000 时，直接设置该参数即可，与其他品牌的控制器存在差异。

有效电平

可设置输出信号的有效电平，可选**低电平有效**（符合要求时输出低电平）和**高电平有效**（与前面相反）。

IO 输入

IO 输入可设置软件是否轮询读取各个 IO 输入信号的状态（输入高电平时，对应的 IO 信号显示绿色，否则为灰色），并通过*接收事件*进行解析，在通过*触发设置*进行相关控制。

说明

仅在控制器品牌选择 VB2200、VB2230、VT2000、VT3000、VC2000/2100、VC3000、VC3000（IO 扩展板）、VC3000H/X、VC4000 或 VC5000 时，支持该功能。

轮询开关

可设置是否定期读取 IO 输入端口的状态。根据读取的 IO 状态, 可进行方案或流程的切换。详情参见 *字节匹配-协议比对的控制器示例* 。

轮询间隔

可设置间隔多长时间读取 IO 输入端口的状态，单位为微秒。

说明

仅在开启**轮询开关**的情况下设置有效。

光源参数

光源参数可对连接的光源控制器或视觉控制器的光源进行控制。选择不同品牌的控制器，具体的光源参数有所差别。

说明

除 VC3000、VC3000（IO 扩展板）、VT2000 和 VT3000 外，其他控制器品牌均支持该功能。

- 品牌选择 DPS2、MV-AP1024-2T、MV-LEVD-125-4-SY、MV-LEVD-200-6-SY、HKR 光源控制器时，可分别设置各个通道的**光源亮度**。MV-LEVD-200-6-SY 为 6 路光源，HKR 光源控制器为 16 路光源，其余为 4 路光源。
- 品牌选择 VB2200、VB2230、VC2000/2100 或 VC5000 时，可分别设置各通道的**光源亮度**和**光源状态**。VB2200 仅 1 路光源，VB2230 为 2 路光源，VC2000/2100 为 4 路光源，VC5000 为 8 路光源。
- 品牌选择 VC3000(Light)、VC3000H/X 和 VC4000 时，可分别设置各通道的**光源亮度**、**沿定义**、**光源状态**、**持续时间(ms)**，共有 4 路光源。

以上控制器涉及的光源参数介绍如下：

- **光源亮度**：可控制各通道光源的亮度。
- **光源状态**：可选**触发后灭（常亮）**和**触发后亮（常灭）**。

- **沿定义**：通过光源触发接口控制光源时，通过该参数可设置触发光源的信号类型。可选**上升沿**和**下降沿**。
- **持续时间（ms）**：可设置光源触发后亮/灭的持续时长。

5.2.2 接收事件

接收事件可对外部设备发送给 VM 数据进行解析配置，使其从一段数据解析成所需要的值。

说明

外部设备可以是通信设备，也可以是带 IO 功能的视觉控制器，但不支持光源控制器和仅带光源功能的视觉控制器。

完成数据解析后，可将接收到的数据或自定义数据重新组装再回复给外部设备，也可作为 *全局触发* 中的触发事件执行相关操作。

接收事件包括 *文本-协议解析* 、 *文本-协议比对* 、 *字节匹配-协议比对* 和 *脚本* 四种方式。

操作方法：

1. 打开*通信管理*并选择*接收事件*。
2. 点击*接收事件列表*右侧的 ⊞ 。
3. 根据需求选择**处理方式**和**事件类型**。
4. 点击*创建*即可完成接收事件的添加。
5. （可选）*接收事件列表*处选中事件并右键单击可重命名事件或删除事件。
6. （可选）点击当前界面右下角的*打开消息*可查看接收事件的解析结果。
   - **16 进制显示**：可将解析结果的内容转换为 16 进制显示。
   - **清空**：可将消息窗口解析的内容清空。

文本-协议解析

文本-协议解析可将外部设备发送给 VM 的数据根据设置的分隔符和规则进行解析。只有内容完全符合设置的规则，才可解析成功。
相关参数说明如下：

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

分隔符

可下拉选择或自定义输入分隔符，将接收的字符串数据根据设置的分隔符隔开。

字符长度比较

启用该功能后，只有接收的数据长度与设置的**字符长度**完全一样，才会进行数据解析，否则不对接收的数据进行解析。

说明

- **字符长度**默认为 0，此时解析会失败，需根据需求自行修改。
- 分隔符也代表一个字符。

输出列表

点击 ⊞ 可添加一条输出规则，每条规则需根据需求进行设置。支持添加多条规则。

说明

解析字符串数据时，按照**输出列表**添加的规则顺序从上往下依次执行解析。当其中一条解析失败时，则说明发送的字符串数据不符合规则，无法解析。

名称

解析后数据的名称，可自定义修改。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

数据结果

显示解析后每条数据的内容。解析失败时，此处不显示数据。

状态

可查看最后一次接收的数据与解析结果是否匹配。解析成功，则显示绿色 √ ；否则显示红色×。

文本-协议比对

文本-协议比对可理解为在文本-协议解析的基础上，增加单个数据的比较以及组装功能，同时还可以将组装结果回复给外部设备。参数配置过程包括*输入配置*和*组装配置*两个步骤。

输入配置

将外部设备发送给 VM 的字符串数据根据设置的分隔符和规则进行解析。

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

分隔符

可下拉选择或自定义输入分隔符，将接收的字符串数据根据设置的分隔符隔开。

字符长度比较

启用该功能后，只有接收的数据长度与设置的**字符长度**完全一样，才会进行数据解析，否则不对接收的数据进行解析。

说明

- **字符长度**默认为 0，此时解析会失败，需根据需求自行修改。
- 分隔符也代表一个字符。

解析列表

点击 ⊞ 可添加一条解析规则，每条规则需根据需求进行设置。支持添加多条规则。

说明

解析数据时，按照**解析列表**添加的规则顺序从上往下依次执行解析。其中一条解析失败，说明发送的字符串数据不符合规则，无法解析。

名称

解析后数据的名称，可自定义修改。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

比较规则配置

可选**不比较**、**！=**（不等于）和**=**。

不比较

按照设置的**类型**解析，无需设置具体比较的内容。

！=

=

需设置具体比较的内容，int 或 float 类型数据需设置数值范围，string 类型数据设置具体的字符内容即可。

数据结果

显示解析后每条数据的内容。解析失败时，此处不显示数据。

状态

可查看最后一次接收的数据与解析结果是否匹配。解析成功，则显示绿色 √；否则显示红色**×**。

组装配置

将输入配置解析后的数据组装成新的数据，并设置是否回复给外部设备。

回复给设备

将按照组装列表组装后的数据发送后外部设备。

分隔符

可下拉选择或自定义输入分隔符，将发送的数据根据设置的分隔符隔开。

组装列表

点击 ⊞ 可添加一条组装规则，每条规则需根据需求进行设置。支持添加多条规则。

说明

组装数据时，按照**组装列表**添加的规则顺序从上往下依次进行组装。

名称

解析后数据的名称，可自定义修改。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

内容

可从输入配置的解析列表订阅相同类型的数据，也可自定义。

数据结果

显示解析后每条数据的内容。解析失败时，此处不显示数据。

字节匹配-协议比对

字节匹配-协议比对与前面两种接收事件有所差别，是对外部设备发送的 16 进制字节数据进行解析和组装。该接收事件根据字节位置以及其他规则进行解析。参数分为*解析配置*和*组装配置*。

解析配置

将外部设备发送给 VM 的字节数据根据设置的规则进行解析。

解析数据的判断逻辑为：*判断字节长度是否符合要求 → 判断规则列表中的每条数据是否符合设置的要求 → 根据规则列表每条数据的比较结果，结合规则匹配逻辑判断是否符合要求*。

说明

任何一步解析失败，则整体解析失败。

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

字节长度比较

启用该功能后，只有 VM 接收的数据长度与设置的**字节长度**完全一样，才会进行数据解析，否则不对接收的数据进行解析。

说明

字节长度默认为 0，需根据实际情况自行修改。设置的字节长度最大不可超过 512 字节，否则将导致解析失败。

ASCII 数据

未开启该功能时，只对外部设备发送的 HEX 数据进行解析；开启该功能后，外部设备发送的 ASCII 数据将在转换为 HEX 数据后进行解析。

规则匹配逻辑

可选**与**和**或**。设置规则列表解析后的数据必须同时满足设置的比较规则（对应**与**）或只满足其中一条即可（对应**或**）。

规则列表

点击 ⊞ 可添加一条解析规则，每条规则需根据需求进行设置。支持添加多条规则。

名称

解析后数据的名称，可自定义修改。

字节起止位置

选择需解析字节的起始和终止位。第一位为起始位，第二位为终止位。VM 对选择的起始和终止位间的数据（包含起始和终止位）进行解析。

类型

可设置此条解析数据的类型，可选 int、float、string 和 byte。

顺序

可设置数据解析后的排列顺序，可选 ABCD、BADC、DCBA、CDAB。

假设：收到的数据为 01 00 64 00 00 00，类型设置为 int，字节起止位置为 2、5，则需要 64 00 00 00 进行解析。当顺序选择 ABCD 时，输出为 1677721600；选择 DCBA 时，输出为 100。

说明

类型选择 byte 时，该参数无效，无需设置。

比较规则配置

可选项包括**不比较**、**！=**（不等于）、**=**、**上升沿**、**下降沿**、**更改为**。具体选项说明如下：

说明

不同数据类型对应可选的比较规则选项可能不同，以上仅列出该参数支持选择的所有比较规则选项。当您设置该参数时，请以实际显示的选项为准。

不比较

表示按照设置的**类型**解析，无需设置具体比较的内容。

！=

=

需设置具体比较的内容，int 或 float 类型数据需设置数值范围，string 或 byte 类型数据设置具体的字符内容即可。

上升沿

表示和上次解析的数据比较，本次解析数据由其他值变为 1 时，则符合要求。

说明

仅当您在**类型**字段选择 int 时，才可使用该选项。

下降沿

表示和上次解析的数据比较，本次解析数据由 1 变为其他值时，则符合要求。

说明

仅当您在**类型**字段选择 int 时，才可使用该选项。

更改为

表示和上次解析的数据比较，本次解析的数据由其他值变为设置的数值时，则符合要求。

数据结果

显示解析后每条数据的内容。解析失败时，此处不显示数据。

状态

可查看最后一次接收的数据与解析结果是否匹配。解析成功，则显示绿色 √ ；否则显示红色×。

组装配置

将经过解析后的数据组装成新的数据，并设置是否回复给外部设备。

说明

此处的组装配置操作与**文本-协议比对**处的组装配置操作大体相同，唯一区别是该处组装配置操作支持组装 byte 类型数据，其他操作及功能基本一致，具体参见文本-协议比对的 *组装配置*，此处不再赘述。

脚本

该方式支持通过 python 脚本对外部设备发送的数据进行解析和组装。
VM 安装后自带接收事件的脚本示例 RecvEventTest.py，您可以到软件安装路径下查看该示例文件，相对路径为：.\Applications\ModuleProxy\x64。

说明

- 示例脚本仅供参考，可根据需求自行修改或全新开发。
- 此处使用的脚本为通信相关脚本，使用 python 语言开发。与 VM 中的全局脚本、脚本模块所使用的脚本存在差别，不可混用。

相关参数释义如下：

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

回复给设备

将按照组装列表组装后的数据发送后外部设备。

分隔符

可下拉选择或自定义输入分隔符，将发送的数据根据设置的分隔符隔开。

载入路径

点击 选择接收事件的脚本文件。

组装列表

显示加载脚本后解析的内容。下图为 VM 自带脚本示例解析的内容。



图 5-4 示例脚本效果

名称
类型

由 getOutputParam()函数定义，仅支持 string、int、float、byte 四种数据类型。

数据结果

由 handleMessage(info)函数处理。

满足要求时，在**数据结果**处显示并上传接收事件；不满足要求时，**数据结果**处不显示且不上传事件。

说明

- handleMessage(info)函数定义默认分隔符为**#**，故外部设备发送的数据需以**#**隔开。
- 更多示例脚本相关功能，请查看脚本文件的注释。

5.2.3 发送事件

发送事件可将 VM 中固定格式的数据发送给外部通信设备。其中具体数据格式通过发送事件进行配置。

说明

外部设备仅能选择通信设备使用。

该事件需搭配 *发送数据* 模块使用，包括 *文本-直接输出* 、 *文本-组装输出* 和 *脚本* 三种方式。

操作方法：

1. 打开*通信管理*并选择*发送事件*。
2. 点击*发送事件列表*右侧的 ⊞ 。
3. 根据需求选择**处理方式**和**事件类型**。
4. 点击*创建*即可完成发送事件的添加。
5. （可选）*发送事件列表*处选中事件并右键单击可重命名事件或删除事件。
6. （可选）点击*打开消息*可查看发送事件的解析结果。点击**清空**可将消息窗口解析的内容清空。

说明

发送事件处完成相关参数设置后，需通过发送数据模块进行下图所示的设置，方可将 VM 的数据在流程运行时输出给外部设备。



图 5-5 发送数据模块对应设置

文本-直接输出

文本-直接输出不处理发送数据模块提供的数据，仅将**参数列表**添加的参数按照从上到下的顺序组装后输出给外部设备，各数据间使用分隔符隔开。

相关参数释义如下：

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

分隔符

可下拉选择或自定义输入分隔符，将接收的字符串数据根据设置的分隔符隔开。

参数列表

点击 ⊞ 可添加一条数据，每条数据需根据需求进行设置。支持添加多条数据。

名称

解析后数据的名称，可自定义修改。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

数据结果

显示解析后每条数据的内容。解析失败时，此处不显示数据。

状态

可查看最后一次接收的数据与解析结果是否匹配。解析成功，则显示绿色 √；否则显示红色×。

文本-组装输出

文本-组装输出会将发送数据模块提供的数据根据组装列表添加的参数按照从上到下的顺序组装后输出给外部设备，各数据间使用分隔符隔开。参数分为*输入配置*和*组装配置*。

输入配置

将需输出的数据绑定外部设备并创建需输出的参数。

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

参数列表

点击 ⊞ 可添加一条数据，每条数据需根据需求进行设置。支持添加多条数据。

名称

解析后数据的名称，可自定义修改。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

数据结果

显示解析后每条数据的内容。解析失败时，此处不显示数据。

状态

可查看最后一次接收的数据与解析结果是否匹配。解析成功，则显示绿色 √；否则显示红色×。

组装配置

设置序输出数据的组装规则以及分隔符。

分隔符

可下拉选择或自定义输入分隔符，将接收的字符串数据根据设置的分隔符隔开。

组装列表

点击 ⊞ 可添加一条数据，每条数据需根据需求进行设置。支持添加多条数据。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

内容

可订阅输入配置中**参数列表**的参数，也可自定义。

脚本

脚本通过加载的 python 脚本将 VM 的数据发送给外部设备。
VM 安装后自带发送事件的脚本示例 SendEventTest.py，可到 VM 安装路径下获取，相对路径：.\Applications\ModuleProxy\x64。

说明

- 示例脚本仅供参考，可根据需求自行修改或全新开发。
- 此处使用的脚本为通信相关脚本，使用 python 开发。与 VM 中的全局脚本、脚本模块存在差别，不可混用。

相关参数释义如下：

绑定设备

选择该事件接收哪个外部设备发送的数据。若选择的是 PLC 类型的通信设备，还需在**绑定地址**处选择具体的寄存器地址。

载入路径

点击 🗁 选择发送事件的脚本文件。

组装列表

显示加载脚本后解析的内容。下图为 VM 自带脚本示例解析的内容。



图 5-6 示例脚本效果

名称
类型

由 getInputParam()函数创建，仅支持 string、int、float、byte 四种类型。

发送数据模块的输出数据

由 handleMessage(list)函数对多个发送数据组装后输出。

说明

- handleMessage(list)函数定义默认分隔符为#，故发送给外部设备的数据使用#隔开。
- 更多示例脚本相关功能，请查看脚本文件的注释。

5.2.4 心跳管理

VM 与外部设备建立通信后，启用心跳管理可间隔固定时间向外部设备发送配置的内容。支持同时向多个外部设备发送心跳信号。可通过外部设备能否正常接收信号，判断 VM 与设备间的通信是否存在异常。

前提条件

通信管理的 *通信设备管理* 处添加至少一个外部通信设备。

说明

若添加的设备为 PLC，还需确保完成地址的设置。

操作步骤

1. 点击 ⊞ 。
2. **设备**处下拉选择需接收内容的通信设备。

   说明

   - 只能选择未被配置心跳管理的通信设备。
   - PLC 设备需选择到具体的地址。若一个 PLC 设备有多个地址，每个地址均可被选择。

3. **心跳类型**处根据需求选择发送的数据类型，并完成**发送内容**的设置。
   - **心跳类型**选择单数据时，**发送内容**处仅需设置 1 组数据。
   - **心跳类型**选择多数据时，**发送内容**处需设置 2 组数据。VM 会往对应外部设备循环发送这两组数据。
4. **时间间隔**处根据需求设置两次信号发送的间隔时间，默认为 1000ms。
5. 开启**启用/关闭**参数后，软件即可发送数据给外部通信设备。
6. 重复以上步骤，可添加多个需进行心跳管理的外部通信设备。

结果说明

- **心跳类型**选择单数据时，软件每隔一段时间给外部通信设备发送**发送内容**的内容。
- **心跳类型**选择多数据时，软件每间隔一段时间给外部通信设备轮询发送**发送内容**的 2 个内容。

5.2.5 响应配置

响应配置可配置软件在发生指定动作时，发送指定信息给对应的通信设备。

指定的动作主要为 *方案加载* 、 *流程控制* 和 *相机响应配置* 。启用对应的参数，并完成相关参数配置即可。

方案加载

启用**方案加载**并完成以下参数的设置后，当方案加载成功时，软件会发送指定信息给外部通信设备。

通信设备

选择一个外部通信设备。

- 若您选择的通信设备为 PLC，则还需指定具体使用的寄存器地址，并设置待发送信息的数据类型：

  数据类型

设置待发送信息的数据类型。支持发送 **int**、**float** 和 **string** 类型的数据。如果您选择 **string** 类型，则还可以设置字符串的**结束符**。

- 若您选择的通信设备非 PLC 设备，则无需设置待发送信息的数据类型。

通信超时时间(s)

可设置方案加载后，延迟多久发送指定内容给外部通信设备。

结束符

可设置待发送内容的结束符。打开***结束符***右侧的按钮，在输入框填写自定义的结束符，或者单击输入框右下角的三角按钮下拉选择如下结束符：**\r**、**\n** 和**\r\n**。

触发字符

可自定义方案加载后，输出给外部通信设备的具体内容。

流程控制

启用**流程控制**并完成以下参数的设置后，当各流程状态发生变化时，软件会发送指定信息给外部通信设备。流程状态分为空闲和忙碌两种。

通信设备

选择接收特定信息的外部通信设备。

- 若您选择的通信设备为 PLC，则还需指定具体使用的寄存器地址，并设置待发送信息的数据类型：

  数据类型

  设置待发送信息的数据类型。支持发送 **int**、**float** 和 **string** 类型的数据。如果您选择 **string** 类型，则还可以自定义字符串内容的结束符。打开***结束符***右侧按钮，可显示结束符输入框。您可以输入自定义结束符，也可以单击输入框右下角的三角按钮，选择如下结束符：**\r**、**\n** 和**\r\n**。

- 若您选择的通信设备非 PLC 设备，则无需设置待发送信息的数据类型。

结束符

可设置待发送内容的结束符。打开***结束符***右侧的按钮，在输入框填写自定义的结束符，或者单击输入框右下角的三角按钮下拉选择如下结束符：**\r**、**\n** 和**\r\n**。

分隔符

可设置待发送信息中自定义内容与流程 ID 之间的分隔符，可下拉选择，也可自定义设置。

空闲时触发
忙碌时触发

可自定义流程状态变为空闲/忙碌时发送的内容。

发送给外部通信设备的内容规则为：{配置的内容}{分隔符}{流程 ID}。

说明

此处的流程 ID 可通过流程编辑区域查看，下图中框选处的数值减去 9999 即可。

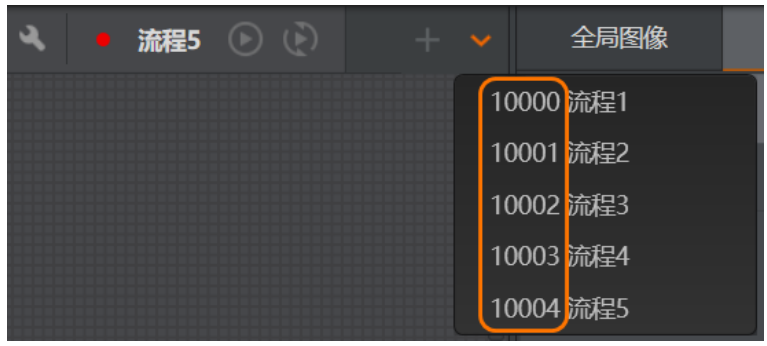


图 5-7 流程 ID

相机响应配置

启用**相机响应配置**并完成以下参数的设置后，当相机状态发生变化时，软件会发送特定信息给外部通信设备。相机状态分为连接和掉线两种。

通信设备

选择接收特定信息的外部通信设备。

- 若您选择的通信设备为 PLC，则还需指定具体使用的寄存器地址，并设置待发送信息的数据类型：

  数据类型

  设置待发送信息的数据类型。支持发送 **int**、**float** 和 **string** 类型的数据。如果您选择 **string** 类型，则还可以自定义字符串内容的结束符。打开***结束符***右侧按钮，可显示结束符输入框。您可以输入自定义结束符，也可以单击输入框右下角的三角按钮，选择如下结束符：**\r**、**\n** 和**\r\n**。
- 若您选择的通信设备非 PLC 设备，则无需设置待发送信息的数据类型。

分隔符

可设置发送的信息中自定义内容与相机 ID 之间的分隔符，可下拉选择，也可自定义设置。

连接时触发/掉线时触发

可自定义相机状态变为连接/掉线时发送的内容。

发送给外部通信设备的内容规则为：{配置的内容}{分隔符}{相机 ID}。

说明

此处的相机 ID 可通过相机管理查看，下图中框选处的数值即相机 ID。



图 5-8 相机 ID

5.2.6 通信示例

本章节提供以下关于通信设备、视觉控制器和全局触发相关的示例。

- 通信示例 1： *文本-协议解析的通信示例*
- 通信示例 2： *字节匹配-协议比对的通信示例*
- 控制器示例： *字节匹配-协议比对的控制器示例*

说明

如需了解如何通过指定的字符串内容触发流程执行，请参见 *字符串触发* 相关介绍。

文本-协议解析的通信示例

示例需求：VM 接收外部 TCP 服务端发送的字符串数据，能解析{字符串,整型数据}格式的数据，其他格式数据无法解析。解析成功时，将解析后的整型数据赋值给流程 1 直线查找模块中的运行参数**边缘阈值**，并触发方案中的流程 1 运行。

根据以上需求，可提炼的方案要求如下：

- 需要新建 1 个流程，且流程名称为流程 1，流程中至少需包含图像源和直线查找模块。
- 通信管理的设备列表中需创建一个 TCP 客户端，用于接收外部 TCP 服务端的数据。
- 通信管理的接收事件需新建一个文本-协议解析的事件，对 TCP 客户端接收的数据通过接收事件进行数据解析，且只能解析{字符串,整型数据}格式的数据。
- 全局触发需添加一个事件触发，用于触发流程执行，并将整型数据赋能给直线查找模块对应的参数。

操作步骤

1. 新建流程 1，涉及的模块和顺序如下图所示。



图 5-9 新建流程

2. 通过通信管理*设备管理*的**通信设备列表**添加设备。
   1) **协议类型**选择 TCP 客户端。
   2) **目标 IP** 和**目标端口**填写 TCP 服务端的 IP 地址和端口号。
   3) 在设备列表选中 TCP 客户端并启用。

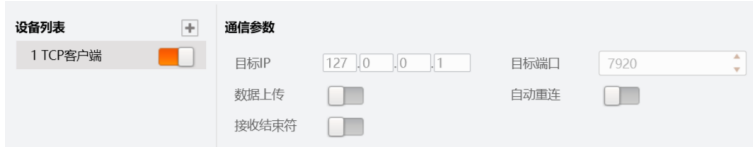


图 5-10 通信设备设置

3. 切换到通信管理的*接收事件*，新增文本-协议解析的接收事件。
4. 完成接收事件的相关参数设置。
   1) **绑定设备**选择步骤 2 中新增的 TCP 客户端。
   2) 根据需解析的数据格式要求，将**分隔符**设置为“,”。
   3) *输出列表*添加 2 个参数，第一个参数**类型**选择 string，第二个参数**类型**选择 int。



图 5-11 接收事件设置

5. 切换到通信管理的*触发配置*，添加触发事件并完成相关参数设置。
   1) 在事件触发页面添加一个触发事件。
   2) **触发事件**选择“0 文本-协议解析”。
   3) 点击 在内容查看页签中将 out2 的*目标输出*订阅到流程 1 中直线查找模块的**边缘阈值**参数。参见如下视频。
   4) **触发命令类型**选择执行流程。
   5) **触发配置**选择流程 1。
   6) **触发字符**订阅“0 文本-协议解析”中的 out1。



图 5-12 全局触发设置

示例

完成以上操作后，当 TCP 服务端发送的数据为“ABC,2”时，接收事件中的 out1 参数赋值为 ABC，out2 参数赋值为 2，且直线查找模块的边缘阈值参数被赋值为 2，流程 1 自动运行 1 次。

字节匹配-协议比对的通信示例

示例需求：VM 接收外部 ModBus 设备（使用 TCP 客户端形式通信）发送的 4 位字节数据，前两位解析为整数型数据，后两位解析为字符类数据。解析成功时，触发方案中所有流程运行。

根据以上需求，可提炼的方案要求如下：

- 需要新建至少 1 个流程。
- 通信管理的设备列表中需创建一个 ModBus 通信并新建一个含 2 个寄存器个数的寄存器地址，用于接收外部 ModBus 设备的数据。
- 通信管理的接收事件需新建一个字节匹配-协议比对的事件，对 ModBus 设备发送的数据通过接收事件进行数据解析，且将数据的前两位字节解析为整数型数据，后两位解析为字符类数据。
- 全局触发需添加一个事件触发，用于触发所有流程执行。

操作步骤

1. 完成方案中各流程的搭建。

   说明

   该示例对方案流程搭建无要求，可根据需求自行搭建。

2. 通过通信管理*设备管理*的**通信设备列表**添加设备。
   1) **协议类型**选择 ModBus 通信。
   2) **通信方式**选择 TcpClient。
   3) **目标 IP** 和**目标端口**填写 ModBus 设备的 IP 地址和端口号。
   4) 添加 1 个寄存器地址，其中**协议类型**选择 RTU，**Int 类型**选择 16bit，**寄存器个数**设置为 2。
   5) 在设备列表选中 ModBus 通信设备并启用。



图 5-13 通信设备设置

3. 切换到通信管理的*接收事件*，新增字节匹配-协议比对的接收事件。
4. 完成接收事件中解析配置部分的参数设置。
   1) **绑定设备**选择步骤 2 中新增的 ModBus 设备，**绑定地址**选择步骤 2 中添加的寄存器地址。
   2) **规则匹配逻辑**设置为与。

3) *规则列表*添加 2 个参数并完成规则设置。

- 第一个参数 in1：**字节起止位置**设置为 0~1，**类型**选择 int，**顺序**选择 ABCD，**比较规则配置**选择不比较。
- 第二个参数 in2：**字节起止位置**设置为 2~3，**类型**选择 string，**顺序**选择 ABCD，**比较规则配置**选择不比较。



图 5-14 接收事件的解析配置

5. 组装配置的组装列表处添加 2 个参数并完成设置。

- 第一个参数 out1：**类型**选择 int，**内容**订阅*规则列表*的 in1。
- 第二个参数 out2：**类型**选择 string，**内容**订阅*规则列表*的 in2。



图 5-15 接收事件的组装配置

6. 切换到通信管理的*触发配置*，添加触发事件并完成相关参数设置。

1) 在事件触发页面添加一个触发事件。
2) **触发事件**选择“2 字节匹配-协议比对”。
3) **触发命令类型**选择执行流程。
4) **触发配置**选择全流程。
5) **触发字符**订阅“2 字节匹配-协议比对”中的 out2。



图 5-16 全局触发设置

7. 回到*设备管理*的 ModBus 通信设备，双击打开已创建的寄存器地址，启用**轮询使能**。

示例

完成以上操作后，当 ModBus 通信发送的 16 进制数据为“00 5C 41 44”时，接收事件中的 out1 参数赋值为 92，out2 参数赋值为 AD，所有流程自动执行一次。

字节匹配-协议比对的控制器示例

示例需求：安装在 VC2000 系列视觉控制器上的 VM，通过视觉控制器 IO 输入的 IO 4 触发“流程 1”运行。其中 IO 4 需接收到上升沿信号，方可触发“流程 1”运行。

根据以上需求，可提炼的方案要求如下：

- 需要新建 1 个流程，且流程名称为流程 1。
- 通信管理的控制器列表中需新增一个 VC2000 设备并开启 IO 输入的**轮询开关**。
- 通信管理的接收事件需新建一个字节匹配-协议比对的事件，对 VC2000 设备发送的 IO 输入信号通过接收事件进行解析，判断 IO 4 是否为上升沿信号。
- 全局触发需添加一个事件触发，用于触发“流程 1”执行。

操作步骤

1. 新建流程 1。

   说明

   该示例对方案流程搭建无要求，可根据需求自行搭建。

2. 通过通信管理*设备管理*的**控制器列表**添加设备。
   1) **协议类型**选择 VC2000。
   2) **串口号**选择 COM2。
3. 切换到通信管理的*接收事件*，新增字节匹配-协议比对的接收事件。
4. 完成接收事件中解析配置部分的参数设置。
   1) **绑定设备**选择步骤 2 中新增的 VC2000 设备。
   2) *规则列表*添加 1 个参数并完成规则设置。

      字节起止位置设置为 3~3，**类型**选择 int，**顺序**选择 ABCD，**比较规则配置**选择上升沿。



图 5-17 接收事件的解析配置

说明

组装配置无需设置。

5. 切换到通信管理的*触发配置*，添加触发事件并完成相关参数设置。
   1) 在事件触发页面添加一个触发事件。
   2) **触发事件**选择“0 字节匹配-协议比对”。
   3) **触发命令类型**选择“执行流程”。
   4) **触发配置**选择“流程 1”。



图 5-18 全局触发设置

6. 回到*设备管理*的 VC2000 设备，打开 IO 输入处的**轮询开关**。

示例

完成以上操作后，当 VC2000 的 IO 4 输入信号接收到上升沿信号时，“流程 1”自动执行一次。

第 6 章 方案搭建-流程配置

流程即“视觉处理流程”，是方案中视觉处理逻辑的集合。流程由不同的视觉功能模块（简称“模块”）并行或串行连接而成。

若单条流程无法解决复杂的视觉处理问题，您可在方案中创建多条流程，并配置方案运行时多条流程的联动运行逻辑加以解决。

6.1 配置单流程运行逻辑

测距、有无检测、计数等简单检测场景下，您可优先考虑通过单条流程对检测目标进行视觉处理。

新建或导入方案后，VM 主界面的流程编辑区域已默认创建一条空流程。

操作步骤

1. 为流程添加模块。
   - 选项 1：右键单击流程编辑区，并从弹出的右键菜单中选择模块（见下图）。



图 6-1 右键菜单选择模块

   - 选项 2：按需从模块箱中选择模块，并拖入流程编辑区（见如下视频）。
2. 按执行流程对模块进行排序，并建立模块间的连接（具体操作见如下视频）。

说明

- 各模块的前后序模块以及多模块如何搭配使用相关详情，请参见 ***模块使用参考*** 中的对应模块章节。
- 若在上一步拖入某个模块时与流程编辑区中已有模块垂直对齐，两者之间将自动连接。

3. **可选操作**: 进行如下可选操作。

删除连接线	将光标悬浮至模块间的连接线，直至连接线略微增粗显示。此时右键单击连接线，并在弹出菜单单击*删除*。
删除模块	右键单击某个模块或框选多个模块，并在右键菜单选择*删除*。

禁用/启用模块 右键单击某个模块或框选多个模块，并在右键菜单选择*禁用*。模块被禁用后置灰。禁用后在右键菜单选择*启用*重新启用该模块。

复制与粘贴模块 右键单击某个模块或框选多个模块，并在右键菜单选择*复制*。复制后再在需粘贴该模块的区域，右键菜单选择*粘贴*。

复制与粘贴参数 如果存在两个相同模块（例如两个*圆查找*），可将其中一个的参数取值复制粘贴至另一个。

a. 右键单击某个模块，并在右键菜单选择*复制参数*。
b. 右键单击“待复制参数取值”的相同模块，并在右键菜单选择*粘贴参数*。

搜索模块 单击流程编辑区右上角的 后，选择**全流程**或**当前流程**，并输入搜索条件（即模块显示名称）即可搜索到符合条件的模块。选中搜索到的模块，可直接定位到该模块所在位置。

添加注释 可自定义绘制区域并设置注释。

a. 单击流程编辑区右上角的 。
b. 单击 并框选想注释的内容。
c. 双击“请填写注释内容”，并自定义输入注释内容。
d. 选中单个注释框，单击 可将注释框置顶、置底。

说明

也可通过选中单个注释框后，右键单击实现。

e. 单击 可清空所有注释信息。

说明

选中单个注释框后，右键单击可删除该注释框。

4. 配置流程中各模块。

说明

模块配置详情，请参见 *模块使用参考* 中对应模块章节。

5. **可选操作**: 在流程栏上右键单击当前流程，并在弹出的右键菜单上按需选择杂项可选操作和配置。

相关详情请参见 *其他操作与配置* 。

6. 试运行该流程。
   - 查看运行效果：单击流程页签上的 单次运行流程。
   - 模拟在产线运行的效果：单击流程页签上的 连续运行流程。

说明

建议先单次运行，若效果符合预期，再连续运行。

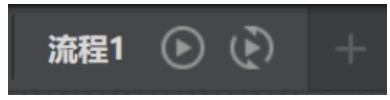


图 6-2 流程页签

7. 若该流程运行结果未达到预期，调试流程中模块的运行参数。

   更多调试相关信息，请参见 *方案调试与运行* 。

   说明

   若经多番调试后，该流程的运行结果仍未达到预期，可尝试 *配置多流程运行逻辑*，通过配置多流程联动运行逻辑加以解决。

6.2 配置多流程运行逻辑

在复杂检测场景下，若单条流程经调试后仍无法达到预期的运行效果，您可配置多流程运行逻辑加以解决。

多流程运行逻辑中可引入并联动如下三类逻辑实体，灵活实现跨流程数据传递与处理，解决棘手的视觉处理问题。

表 6-1 可引入的逻辑实体

逻辑实体	描述
流程	视觉处理逻辑的集合。通过引入多条流程，可联动多个集合的数据处理逻辑。
全局相机	多条或单条流程的图像数据来源。
数据队列	可在流程间引入的逻辑实体，用于解决多流程联动运行时，不同流程耗时差异导致的流程间数据阻塞。

6.2.1 步骤一：新建流程

前提条件

已配置默认流程的运行逻辑，详情请参见 *配置单流程运行逻辑* 。

操作步骤

1. 在流程栏上单击 添加流程。

   说明

   最多可创建 64 条流程。

2. 配置并试运行该流程。
   1) 在流程中添加模块，编排模块间的连接逻辑，并按需配置模块参数。
   2) 试运行该流程。

说明

更多相关操作详情，请参见 *配置单流程运行逻辑* 。

6.2.2 步骤二：编排相关实体连接

新建多个流程并为其配置运行逻辑后，您需定义多流程之间的连接逻辑。您还可引入全局相机和数据队列，并将两者（或两者中的任意一种）编排至多流程连接逻辑之中。

操作步骤

1. 在流程栏单击 进入多流程编辑区域。

   若已创建流程、 *全局相机* 以及 *数据队列*，则该区域显示这些逻辑实体。

2. **可选操作**: 继续添加流程、全局相机或数据队列。

   添加流程

   选项 1：将光标悬浮至左侧的 ，直至其变为橙色，再将其拖入右侧的编辑区。

   选项 2：在编辑区任意空白区域右键单击，并在弹出的右键菜单上选择 *工具 → 流程* 。如果本地路径中已有流程文件，也可在右键菜单上选择*导入流程*，将流程文件导入。

   添加数据队列

   选项 1：将光标悬浮至左侧的 ，直至其变为橙色，再将其拖入右侧的编辑区。

   选项 2：在编辑区任意空白区域右键单击，并在弹出的右键菜单上选择 *工具 → 数据队列* 。

   可双击数据队列对其进行配置，详情参见 *数据队列* 。

   添加全局相机

   选项 1：将光标悬浮至左侧的 ，直至其变为橙色，再将其拖入右侧的编辑区。

   选项 2：在编辑区任意空白区域右键单击，并在弹出的右键菜单上选择 *工具 → 全局相机* 。

   添加全局相机后，可双击对其进行配置，详情参见 *配置全局相机* 。

3. 拖动上述任意实体，按需排布。
4. 建立实体之间连接。

   连接方式与单个流程中的模块间连接相同。

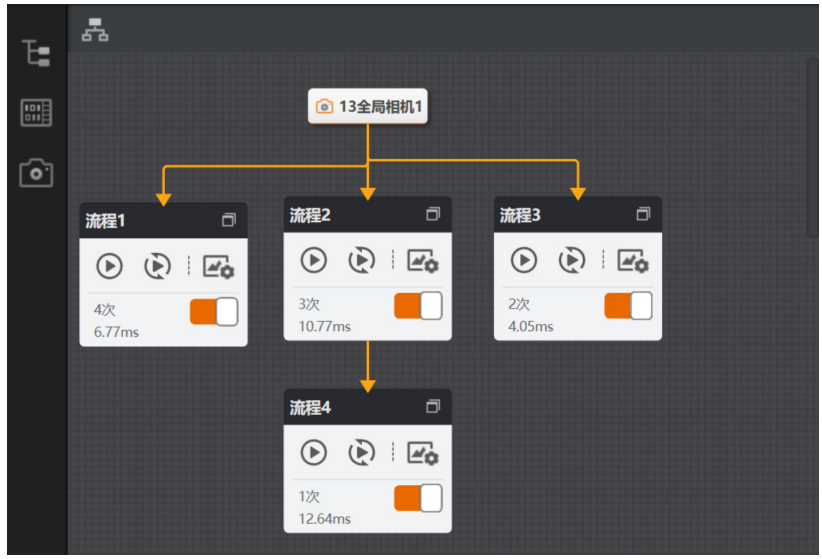


图 6-3 多流程连接

5. **可选操作**: 执行如下可选操作。

删除连接线	将光标悬浮至实体间的连接线，直至连接线略微增粗显示。此时右键单击连接线，并在弹出菜单上单击*删除*。
执行单个流程的右键菜单操作	右键单击某个流程弹出右键菜单，根据右键菜单选项执行流程导出、流程删除、流程拷贝粘贴等操作。详情参见 *其他操作与配置* 。
执行单个流程	单击某流程上的 或 ，分别执行一次或连续执行该流程。
启用/禁用单个流程	将某流程上的开关关闭，禁用该流程。禁用后，多流程异步执行时，该流程不执行，且软触发和硬触发对该流程均不生效；打开开关，重新启用该流程。
返回单流程界面	单击某流程上的 ，返回对应的单流程界面。

6.2.3 步骤三：定义各流程的输入输出

为实现多流程联动运行，您需至少定义初始流程的输出、终止流程的输入以及中间流程的输入与输出。

可通过以下方式打开流程配置窗口，设置各流程的输入和输出。

- 选项一：在流程栏单击某个流程的 。
- 选项二：在流程栏单击 进入多流程配置界面，并单击某条流程的 。

输入设置

可选择前序流程的输出参数作为当前流程的输入参数，供当前流程中的模块进行订阅。
选择参数的**类型**后，单击 ⊞ 即可添加一个输入参数。自定义**参数名称**并通过**订阅关系**订阅前序流程输出的参数即可。
类型可设置为**多选**、**int**（整型）、**float**（浮点型）、**string**（字符串）、**byte**（字节）、**pointset**（点集）、**IMAGE**（图像）、**PIXELIMAGE**（像素图）、**BOX**（目标区域）、**POINT**（点）、**LINE**（线）、**FIXTURE**（修正信息）、**ANNULUS**（圆环）、**CONTOURPOINT**（轮廓点）、**RECT**（矩形）、**CLASSINFO**（类别）、**POLYGON**（多边形）、**POSTURE**（位姿）、**ELLIPSE**（椭圆）。

说明

- **类型**设置为**多选**时，可批量添加多个多种类型的输入参数。
- **类型**设置为 **IMAGE** 或 **PIXELIMAGE** 时，可设置对应参数是否勾选（默认勾选）。勾选的参数在**显示设置**中自动添加，对应渲染图像可在图像显示窗口勾选（若勾选则在该窗口显示）。

输出设置

可选择当前流程中各模块的输出参数作为当前流程的输出参数，输出至后序流程。
输出参数的设置方法与输入参数基本一致，此处不再赘述。
在*输出设置*页签选择当前流程下属模块的输出参数（可多选）。

说明

除 **IMAGE** 或 **PIXELIMAGE** 类型的参数，部分参数也可勾选是否显示，默认不勾选。涉及的类型为 **string**、**pointset**、**BOX**、**POINT**、**LINE**、**ANNULUS**、**CONTOURPOINT**、**RECT**、**POLYGON**、**ELLIPSE**。

6.2.4 （可选）步骤四：定义各流程可选配置

您还可在流程配置窗口按需设置各流程的其他配置项，例如单个流程的运行参数和流程结果数据的渲染效果，方便试运行时进行调试。

单击某个流程上的 进入流程配置窗口。

自定义各流程运行参数

可设置为“流程运行参数”的参数包括各模块基本参数中的 ROI 相关参数、所有运行参数、匹配模型参数以及结果显示中的结果判断相关参数等。

操作步骤

1. 在流程配置窗口选择*运行参数*页签。
2. 单击*编辑*打开*配置*窗口。
3. **可选操作**: 在如下图所示的下拉列表处，选择参数数据类型（默认为**全部**）。



图 6-4 选择数据类型

4. 展开选择列表中的可选项，并通过勾选对应参数，将参数添加至右侧的*已选择参数*列表。

说明

- 通过参数数据类型右侧的搜索框，可快速查找参数。
- 首次添加参数时，该参数自动添加至首个分组。首个分组默认名称为“分组 0”，可自定义；若需将参数添加至其他分组，可先单击右上角的*添加分组*，再添加参数。最多可添加 32 个分组。
- 存在多个分组时，需先选中分组，再添加参数。选中的分组显示在橙色矩形框中，如下图所示。



图 6-5 配置窗口

5. **可选操作**: 执行如下可选步骤。

删除分组/参数	单击 ⊗ 删除对应分组或参数。
修改分组/参数排序	单击 ↑ 将对应分组或参数在列表中上移；单击 ↓ 将对应分组或参数在列表中下移。

清空列表　　单击右上角的*清空*，将*已选择参数*列表清空。

修改订阅数据　　单击 修改已添加参数订阅的数据（即修改该参数绑定的数据）。

说明

若修改后订阅的数据源数量为两个或两个以上（例如下图所示的情况），则需在修改后单击*数据同步*列的 进行数据同步，使修改生效。



图 6-6 从订阅单数据源修改为订阅双数据源

6. 单击*配置*窗口右下角的*确定*，返回*运行参数*页签。
   - 可在该页签内配置所选的运行参数。
   - 选择某个运行参数，右侧图像显示区域会显示该参数所在模块的图像。

配置渲染效果与通信

您可在流程配置窗口的*显示设置*和*通信配置*页签分别设置流程运行结果的渲染效果和流程与 PLC 等设备的通信方式。

单击某个流程上的 进入流程配置窗口。

显示设置

设置当前流程运行时，需在图像显示窗口呈现的图像数据及叠加显示的信息。

选择**参数名称**后，单击 即可添加一个显示参数。**参数名称**可自定义修改，通过**关联关系**订阅当前流程各模块的模块结果即可。

说明

- 已在*输入设置*或*输出设置*中添加且勾选的 IMAGE 或 PIXELIMAGE 类型参数，将被默认关联至*显示设置*，作为“需在图像显示窗口呈现的数据”。若无需呈现，请前往*输入设置*或*输出设置*取消勾选。
- 除**输出图像**外的其他图像显示参数均支持勾选*显示*单选框。勾选后，所选图形将在图像显示窗口显示。

通信配置

设置流程间隔固定时间或流程结束时发送数据至外部设备。**发送方式**可选**定时发送**或**流程结束时发送**。单击 ⊞ 可添加多个通信设置。

定时发送

设置流程间隔固定时间发送指定信息至外部设备。

时间间隔（ms）

设置发送信息给外部设备的间隔时间。

发送对象

选择*通信管理*窗口中添加的通信设备和控制器设备作为接收信息的设备。可单击 前往*通信管理*窗口的 *设备管理* 页签进行设置。

条件判断

设置判断 OK 或 NG 的条件。一般订阅各模块的模块状态。若模块状态结果为 1，则该条件判断为 OK；若为 0，则该条件判断为 NG。

当**条件判断**关联了其他 Int 类型的输出参数时，当输出结果为 0 时，则该条件判断为 OK；当参数结果为非 0 时，则该条件判断为 NG。

OK 发送
NG 发送
超时发送

分别设置 OK、NG 及超时时发送的信息。

流程结束时发送

设置流程结束时发送指定信息给外部设备。

发送对象

选择*通信管理*窗口中添加的通信设备和控制器设备作为接收信息的设备。可单击 前往*通信管理*窗口的 *设备管理* 页签进行设置。

条件判断

设置判断 OK 或 NG 的条件。一般订阅各模块的模块状态。若模块状态结果为 1，则该条件判断为 OK；若为 0，则该条件判断为 NG。

当**条件判断**关联了其他 Int 类型的输出参数时，当输出结果为 0 时，则该条件判断为 OK；当参数结果为非 0 时，则该条件判断为 NG。

OK 发送
NG 发送

分别设置 OK、NG 时发送的信息。

6.2.5 步骤五：试运行并调试

配置完多流程联动运行逻辑后，需试运行并调试方案，使运行结果更贴合您的检测需求。

试运行

单次运行或连续运行多流程。

- 查看运行效果：单击快捷工具条上的 单次运行方案。
- 模拟在产线运行的效果：单击快捷工具条上的 连续运行方案。

调试多流程运行逻辑

若多流程运行逻辑不满足检测需求，可调整如下配置：

- 各流程内部的模块连接逻辑
- 各流程内模块的参数配置
- 流程间的连接逻辑

若您已为各流程配置常用运行参数和显示设置，您也可进入流程配置窗口，快速调试单条流程。

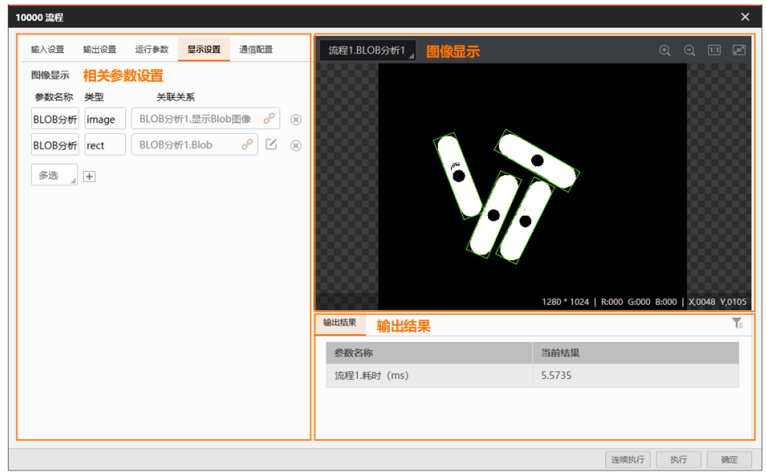


图 6-7 流程配置窗口

该窗口各区域的主要功能如下。

表 6-2 各区域功能

区域	功能
“相关参数设置”区域	支持通过调整各参数配置，对单个流程进行快速调试。
图像显示区域	• 支持勾选**显示设置**中配置的图像数据。 • 支持通过左上角的下拉列表叠加图像数据，增强图像渲染效果，以方便调试。 • 支持选中某个运行参数后，显示该参数所在模块的图像。
输出结果区域	支持单击 筛选需实时查看的流程结果相关数据，包括流程耗时和流程输出。

6.3 其他操作与配置

配置流程后，支持对单条流程进行重命名、导出、删除、拷贝、设置流程超时时间等操作与配置，并支持调整流程的页签顺序。

流程导出

在流程栏上右键单击某个流程，单击右键菜单上的*流程导出*，将该流程（包括该流程的参数设置）以.Prc 格式文件保存至指定的本地路径。

流程拷贝

在流程栏上右键单击某个流程，单击右键菜单上的*流程拷贝*，将选中的流程复制为新流程。

删除

在流程栏上右键单击某个流程，单击右键菜单上的*删除*，删除选中的流程。

导出算子代码

在流程栏上右键单击某个流程，单击右键菜单上的*导出算子代码*，即*文件*下的**导出算子代码**，具体参见 *导出算子代码* 。

设置运行间隔

在流程栏上右键单击某个流程，单击右键菜单上的*设置运行间隔*，在弹出窗口上设置流程连续执行时，前后两次执行的间隔时间（单位：ms）。 该设置主要用于定义前后两次执行之间的等待时长（如下图所示）。

说明

- 若流程耗时大于此处设置的间隔，则间隔不生效。
- 若同时设置了此处的*设置运行间隔*和 *软件设置* 中的**流程延时**，*流程延时*的生效逻辑可分为如下图所示的三种情况。

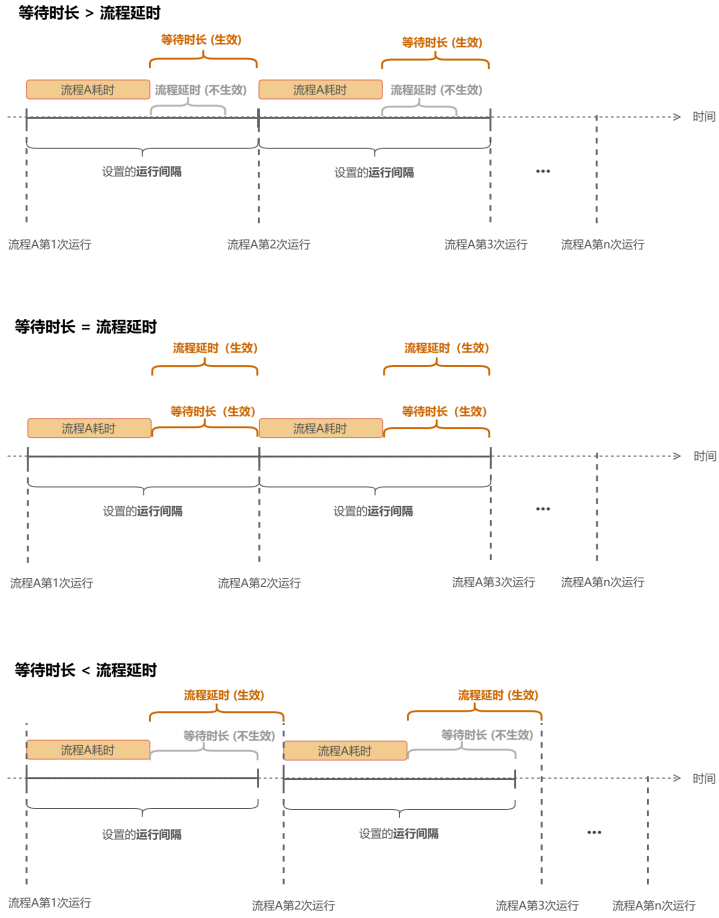


图 6-8 该两项设置的生效情况

流程执行 NG 时停止

在流程栏上右键单击某个流程，可选择开启右键菜单中的*流程执行 NG 时停止*。开启后，若所选流程的模块运行状态为 NG，则该流程停止运行执行流程。

重命名

在流程栏上右键单击某个流程，单击右键菜单上的*重命名*，对选中的流程进行重命名。

设置流程超时时间

在流程栏上右键单击某个流程，单击右键菜单上的*设置流程超时时间*，在弹出窗口设置选中流程单次执行的最大时间。如果该流程实际执行时间达到超时时长，该流程中当前正在执行

的模块执行完毕后，该流程将停止继续执行并报错。超时时间默认为 0，表示不限制流程执行时间。

调整流程页签顺序

当方案中配置了两个及以上流程，支持在流程栏中通过鼠标左键长按流程名称并拖动的方式调整流程页签顺序，如下图所示。

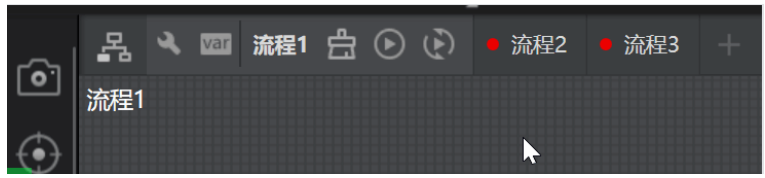


图 6-9 调整流程页签顺序

第 7 章 方案搭建-模块配置

视觉功能模块（简称“模块”）是视觉处理的最小功能单元。单个模块中通常包含特定的算法处理或运算逻辑。

按照视觉处理类型，模块分为采集、定位、测量、识别等类别，此处不再赘述，具体请以实际界面上的划分为准。

模块调用及配置说明请参见 *模块使用基础规则* 。

7.1 模块使用基础规则

搭建视觉流程时一般需要连接多个模块，其中模块调用和配置需要遵循一定的规则。

模块调用

单个模块通常无法解决视觉处理问题，将多个模块串行或并行连接成流程之后，才可解决。但特定模块前后并非可连接任意模块。

- A 模块前序可连接什么模块，由“前序模块输出的结果数据（即模块结果）与可输入 A 模块的数据是否匹配”决定。
- A 模块后序可连接什么模块，由“A 模块输出的结果数据与后序连接模块可输入的数据是否匹配”决定。

说明

- 某个模块前序和后序可连接的其他模块，可参考 *模块使用参考* 各模块内容中的“前后序模块”相关说明。
- 某个模块输出的结果数据相关详情，请参见 *模块使用参考* 各子章节的*模块结果*小节。
- 某个模块可输入数据相关详情，请参见各模块配置窗口的***基本参数***页签。

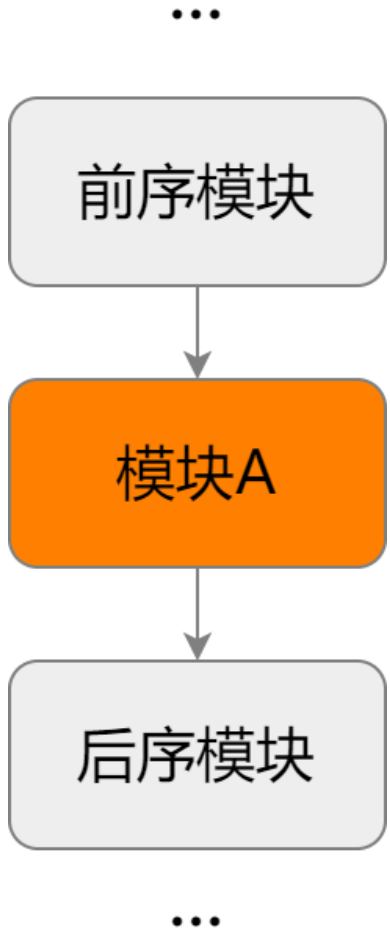


图 7-1 前后序模块示意

模块配置

双击已拖入流程编辑区的模块即可打开模块配置窗口。大部分模块的配置窗口都分为*基本参数*、*运行参数*和*结果显示* 3 个页签。少量涉及特征模型的模块还有模型（模板）配置页签。

- *基本参数*页签用于定义模块的输入数据。具体配置请参见 *模块通用配置* 。
- *运行参数*页签用于定义模块运行时如何处理输入的数据，具体配置请参见 *模块使用参考* 中各模块内容的*参数配置*。
- *特征模板*页签（或其他类似名称的页签）用于创建匹配模型。具体配置请参见 *模块使用参考* 相应模块内容中与模板配置相关的内容。
- *结果显示*页签用于定义模块如何将模块输出的结果渲染在图像上。具体配置请参见 *模块通用配置* 。

7.2 模块通用配置

模块作为方案中的重要组成部分，可实现各种视觉检测。

大部分模块配置界面主要由基本参数、运行参数和结果显示页签组成。个别模块包含的页签有所差别，请以实际模块内容为准。在模块配置界面的左上角，单击 ，即可查看对应模块的帮助文档。

本章节主要对模块中的 *基本参数* 、 *运行参数* 和 *结果显示* 页签中的通用操作进行介绍。

说明

各个模块的用途、使用方法、具体参数详情，请参见 *模块使用参考* 。

7.2.1 基本参数

基本参数可订阅图像输入，并对需检测区域进行 ROI 相关设置。

说明

部分模块无基本参数或需设置的内容有所差别，具体请以模块实际内容为准。本章节仅对基本参数的图像输入和 ROI 区域进行介绍。

图像输入

图像输入可订阅该模块的图像来源，可通过**输入源**订阅前序模块输出的图像。

ROI 区域

ROI 区域可设置输入图像的的检测区域。

说明

不同模块 ROI 区域的设置有所差别，本章节基于轮廓匹配模块的 ROI 区域参数进行介绍，具体请以模块实际显示参数为准。

输出掩膜

可设置模块是否输出 ROI 区域的掩膜图像。

ROI 区域类型

可设置 ROI 区域的类型，可选**图形类型**和**图像类型**。

图形类型

自定义设置 ROI 区域，同时可设置屏蔽区和位置修正。

ROI 创建

选择创建 ROI 的方式，可选**绘制**和**继承**。

绘制

选择**形状**处的图标后在预览图像中进行绘制即可。如何绘制参见 *模块检测的 ROI 区域* 。

继承

可直接从前序模块继承区域或参数，从而构建 ROI 区域。

继承方式

可选**按区域**继承或**按参数**继承。

按区域

通过**区域**订阅前序模块的图像区域即可。

按参数

通过各个参数订阅前序模块的各个点的 X 和 Y、宽、高、角度等数据即可。

屏蔽区创建

可将创建的 ROI 区域中部分区域屏蔽，模块运行时不检测 ROI 区域中的屏蔽区域。选择创建屏蔽区的方式，可选**绘制**和**继承**。

绘制

如何绘制参见 *模块检测的 ROI 屏蔽区域* 。

继承

可通过**屏蔽区**订阅的方式继承前序模块屏蔽区域。

位置修正

使用该功能时，需确保前序模块中有位置修正模块。相关功能和原理参见 *位置修正* 章节。使用该功能时，需完成以下参数的设置。

选择方式

可选择位置修正信息的获取方式，可选**按信息**、**按点**和**按坐标**。无论选择哪种方式，均需订阅相关参数。

图像类型

通过**掩膜图像**直接订阅前序模块的图像作为 ROI 区域。

7.2.2 运行参数

运行参数可对模块进行视觉检测时的算法参数进行设置，从而达到理想检测效果。

说明

部分模块无运行参数，具体请以模块实际内容为准。

各模块需实现的视觉检测需求不同，故算法参数各有不同。针对各算法参数的功能和使用效果，本章节不详细展开，具体请见 ***模块使用参考*** 中对各个模块的详细介绍。本章节仅对运行参数的通用设置方法进行介绍。

运行参数按照操作方式一般分为以下几种类型：

- 下拉选择类的参数：根据实际需求下拉选择合适的选项即可。部分参数选择不同的选项时，会出现一些需要设置的其他参数。
- 启用类的参数：根据实际需求设置是否开启即可。部分参数开启后，会出现一些需要设置的其他参数。
- 设置数值类的参数：此类参数分为两种。
  - 只支持自定义数据：可通过三种方式自行设置数值，参见如下视频。
  - 即支持自定义数据，又支持订阅数据：默认为自定义数据的方式，点击参数最右侧的 后可切换为订阅的方式，点击参数文本框右侧的 可订阅前序模块的模块结果。

说明

订阅参数时，点击 可弹出参数订阅窗口，如下图所示。勾选**默认方式**后，则订阅类参数默认以该方式呈现。

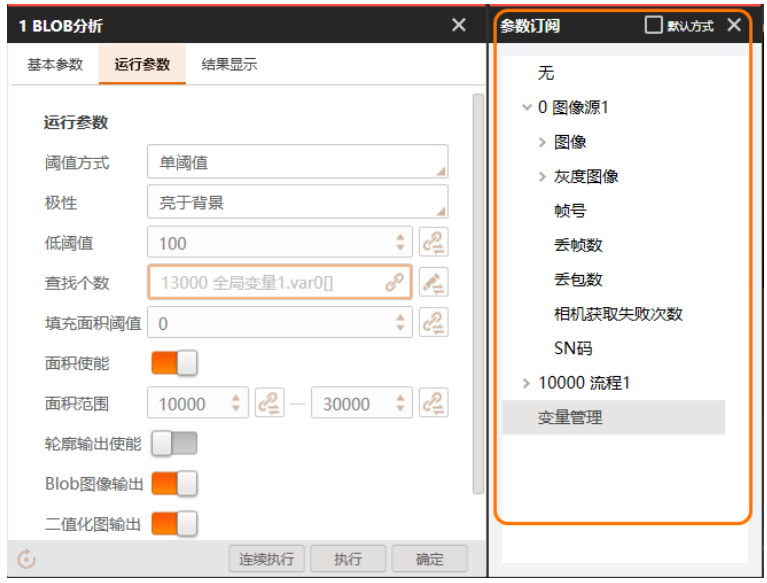


图 7-2 参数订阅

点击模块运行参数的左下角的 可将当前设置的参数重置为默认配置。

7.2.3 结果显示

结果显示部分可对模块视觉检测后输出图像上显示哪些结果进行设置。

结果显示分为 ***结果判断*** 、***图像显示*** 和 ***文本显示*** 三类。不同模块的结果显示支持设置的参数有所差别，具体请以各模块实际显示的参数为准。本章节仅从通用操作方面进行介绍。

结果判断

结果判断可设置判断模块检测结果为 OK 或 NG 的标准。若设置多个判断条件，当所有判断条件均满足时，则模块输出 OK；否则，输出 NG。
根据实际需求开启需使用的判断条件，同时针对该判断条件设置判断为 OK 的范围即可，具体操作如下视频所示。

图像显示

图像显示可设置模块输出图像中显示哪些图像相关的信息以及显示的颜色、透明度、粗细等。点击需显示信息右侧的 / 即可切换是否在输出图像上显示该图像信息。通过 还可对显示信息 OK 或 NG 时的颜色、透明度和粗细进行设置，输出图像上的显示效果会实时更新，具体操作如下动画所示。

文本显示

文本显示可对模块输出图像中叠加的文本信息进行设置。
需设置的参数如下：

- **内容**：对应输出图像上显示的文字内容，可自定义设置。
- **OK 颜色/NG 颜色**：可设置模块输出结果为 OK/NG 时，显示文字的颜色。
- **字号**：可设置显示文字的大小。
- **透明度**：可设置显示文字的透明度。
- **位置 X/位置 Y**：对应文字左上角在图像中的坐标点信息，用于设置文字叠加在输出图像中的位置。点击 后从前序模块或当前模块的模块结果中订阅即可。

7.2.4 绘制 ROI 区域

编辑方案时，大部分模块的基本参数和模型配置中需要绘制 ROI 区域进行视觉检测。本章节主要介绍如何绘制 ROI 区域。

说明

- 在绘制 ROI 区域之前，需确保该方案流程已至少执行一次，且图像输入区域的***输入源***已完成配置。
- 各模块 ROI（此处代表 ROI 区域、ROI 屏蔽区域、模型的掩膜区域和掩膜屏蔽区域）支持的形状、可同时添加的个数有所差异，请以软件实际情况为准。

编辑方案时，以下三个地方需绘制 ROI 区域：

- *模块检测的 ROI 区域*
- *模块检测的 ROI 屏蔽区域*
- *模块中模型配置的掩膜区域*

模块检测的 ROI 区域

模块的 **ROI 创建**选择**绘制**时，需在图像预览窗口自行绘制 ROI 区域。支持绘制的 ROI 形状以及相关操作参见下表。

表 7-1 模块检测的 ROI 区域介绍

ROI 形状	图标	操作方法
全屏		单击 后，该模块的检测区域调整为**输入源**处订阅的整幅图像。 **说明** 各模块的 ROI 区域默认为全屏。
矩形		单击 后，在预览窗口的图像上选中某一点，按住鼠标左键并拖动，即可完成绘制。 选中矩形区域，可调整位置、角度和大小，具体参见如下视频。也可通过单击 *ROI 参数*，修改**中心点 X/Y**、**宽/高**、**角度**参数进行调整。
多边形		单击 后，在预览窗口的图像上选中某一点作为多边形的起点，并依次在多边形的其他点处单击，最后一个点双击即可。 选中多边形区域，可调整位置；选中多边形的交点可调整多边形的形状，具体参见如下视频。也可通过单击 *ROI 参数*，修改**点*(x,y)**参数进行调整。
圆		单击 后，在预览窗口的图像上，选中某一点作为圆心，再选一点作为圆周上的点，即可完成绘制。 选中圆区域，可调整位置、大小、内径、外径等，具体参见如下视频。也可通过单击 *ROI 参数*，修改**圆心点 X/Y**、**内径**、**外径**参数进行调整。
扇形		单击 后，在预览窗口的图像上，选中某一点作为圆心，再选一点作为圆周上的点，即可完成绘制。 选中扇形区域，可调整位置、大小、内径、外径、起始角度和角度范围等，具体参见如下视频。也可通过单击 *ROI 参数*，修改**圆心点 X/Y**、**内径**、**外径**、**起始角度**、**角度范围**参数进行调整。

ROI 形状	图标	操作方法
阵列圆		单击 后，在预览窗口的图像上选中某一点，按住鼠标左键并拖动，即可完成绘制，选中的点为四边形左上角的点。 选中阵列圆区域，可调整位置、矩形大小和角度、圆的大小和形状等，具体参见如下视频。也可通过单击 *ROI 参数*，修改**横向圆个数 M**（只能通过此处调整）、**纵向圆个数 N**（只能通过此处调整）、**中心点 X/Y**、**宽/高**、**角度**、**内径**、**外径**参数进行调整。
直线卡尺		预览窗口的图像默认已完成直线卡尺的绘制。 其中各个矩形框为单个卡尺的输入区域。 选中直线卡尺区域，可调整位置，还可调整直线的位置及长度、以及卡尺的大小，具体参见如下视频。也可通过单击 *ROI 参数*，修改**起点 X/Y**、**终点 X/Y**、**卡尺数量**（只能通过此处调整）、**卡尺宽/高**参数进行调整。
圆卡尺		单击 后，在预览窗口的图像上，选中某一点作为圆卡尺的圆心，再选一点作为圆卡尺的圆周上的点，即可完成绘制。 其中各个矩形框为单个卡尺的输入区域。 选中圆卡尺区域，可调整位置，还可调整圆以及卡尺的大小，具体参见如下视频。也可通过单击 *ROI 参数*，修改**圆心点 X/Y**、**半径**、**卡尺数量**（只能通过此处调整）、**卡尺宽/高**参数进行调整。
圆弧卡尺		单击 后，在预览窗口的图像上，分别依次选择两个点作为圆弧的起点和终点，即可完成绘制。 其中各个矩形框为单个卡尺的输入区域。 选中圆弧卡尺区域，可调整位置，还可调整圆弧的半径、起点、终点、方向以及卡尺的大小，具体参见如下视频。也可通过单击 *ROI 参数*，修改**起点 X/Y**、**终点 X/Y**、**半径**、**卡尺数量**（只能通过此处调整）、**卡尺宽/高**、**方向**、**圆弧大小**参数进行调整。
双矩形		单击 后，在预览窗口的图像上自动完成绘制。 选中双矩形中的每个矩形区域，可分别调整位置，还可调整宽、高以及角度。
四矩形		预览窗口的图像默认已完成直线卡尺的绘制。

ROI 形状	图标	操作方法
		选中四矩形中的每个矩形区域，可分别调整位置，还可调整宽、高以及角度。也可通过单击 *ROI 参数*，修改**中心点 X/Y**、**宽/高**、**角度**参数进行调整。
点	×	单击 × 后，在预览窗口的图像上选中某个点即可完成绘制。 选中绘制的点，可调整位置。也可通过单击 *ROI 参数*，修改**点 X/Y 坐标**参数进行调整。
线	⟍	单击 ⟍ 后，在预览窗口依次选择两个点作为线的起点和终点，即可完成绘制。 选中绘制的点，可调整线及起点/终点的位置。也可通过单击 *ROI 参数*，修改**起点 X/Y 坐标**、**终点 X/Y 坐标**参数进行调整。
坐标系	(x,y)	单击 (x,y) 后，在预览窗口选择一个点即可完成以该点为坐标系原点、水平向右为 X 轴、垂直向下为 Y 轴的坐标系。 选中坐标系，可调整坐标系位置和角度。也可通过单击 *ROI 参数*，修改**中心点 X/Y**、**角度**参数进行调整。

在预览窗口选中绘制的 ROI 区域，右键单击可进行*删除*或*复制*操作。其中复制仅在模块支持绘制多个 ROI 区域时，可进行使用，否则不支持。
其次，部分 ROI 区域的线条上带箭头，该箭头用于表示 ROI 区域的方向。一般卡尺类模块使用时会根据选中区域的方向做检测。

模块检测的 ROI 屏蔽区域

模块的 **ROI 区域类型**选择**图形类型**时，可绘制 ROI 区域的屏蔽区域，使检测时屏蔽该区域。单击 ⌀ 后，在图像预览窗口进行绘制即可。使用方法与多边形 ROI 区域基本一致，此处不再赘述。具体操作参见如下视频。

模块中模型配置的掩膜区域

模块的模型配置选择图像后，需创建掩膜区域。单击模型配置窗口左上角各类创建掩膜的图标后，需在图像预览窗口自行绘制掩膜区域。
掩膜形状包括扇形（ ）、扇圆形（ ）、矩形（ ）、多边形（ ） 和轨迹（ ）。

- 掩膜形状使用扇圆形时，需选中某一点作为圆心，再选一点作为圆周上的点，即可完成绘制。选中圆区域，可调整位置、大小、内径、外径等，操作方式和圆大同小异。
- 掩膜形状使用轨迹时，需依次单击选择轨迹线上的每个点，双击选择的终点即可。选中轨迹区域，可移动位置并调整每个点的位置，操作方式和线大同小异。
- 除扇圆形和轨迹外，其余掩膜形状绘制方法与模块检测的 ROI 区域一致，此处不再赘述。

通过 可设置屏蔽区域。操作方式和轨迹的大同小异。

说明

通过 可锁定/解锁模型配置时的掩膜区域和屏蔽区域。

第 8 章 方案搭建-全局与局部逻辑配置

全局逻辑是在方案范围内全局生效的逻辑定义，包括全局相机、全局/局部变量、全局脚本、全局触发和数据队列。配置后，您可实现向方案中的流程提供数据订阅（通过全局相机）、赋值变量以实现软件与外部数据传递（通过全局/局部变量）、设置方案/流程的执行切换（通过全局脚本）等功能。

8.1 全局相机

全局相机对应相机管理中全局相机的功能，如何添加全局相机请参考 ***添加相机*** 章节，完成相机添加后对全局相机参数进行配置请参考 ***配置全局相机*** 章节。

8.2 全局/局部变量

全局/局部变量是在方案中所有函数外部定义的变量，可以被运行界面、各模块的输入/输出参数和算法参数、*发送数据*模块等订阅。

- 全局变量针对整个方案，局部变量分别两种，一种针对单个流程，一种针对单个 *Group* 模块。
- 全局/局部变量可设置分组、自定义变量名称、类型和当前值。全局/局部变量是覆盖更新机制，当新数据传输进来时，旧数据就会被覆盖。

操作步骤

1. 进入全局/局部变量配置界面。
   - 全局变量：通过快捷工具条的 var 进入全局变量配置界面。
   - 流程/*Group* 的局部变量：选中需添加局部变量的流程/*Group* 模块，通过流程栏的 var 进入流程/*Group* 局部变量配置界面。

     说明

     Group 模块的局部变量仅供单个 *Group* 模块使用，不支持被该模块内嵌套的其他 *Group* 模块使用；流程的局部变量仅供当前流程使用（流程中的 *Group* 模块无法使用）；方案的全局变量供整个方案使用。
2. 点击分组名称右侧的 ＋ 添加变量的分组。

   说明

   当您首次进入全局变量或局部变量的配置界面时，软件自动为您添加一个默认分组。默认分组名称为**分组 0**。
3. 点击首个分组名称上方的***+添加变量***新增变量。
4. 按照以下说明设置每个变量的相关参数。

名称

可自定义全局/局部变量的名称。

注释

可对全局/局部变量增加相关注释，可不填。

类型

可选 int（整型）、float（浮点型）、string（字符串）、byte（字节）、pointset（点集）、IMAGE（图像）、BOX（目标区域）、POINT（点）、LINE（线）、FIXTURE（修正信息）、ANNULUS（圆环）、RECT（矩形）、POLYGON（多边形）、ELLIPSE（椭圆）。

当前值

可读取当前全局/局部变量的值。也可自定义输入，设置后全局/局部变量当前值变为自定义数值。

说明

当变量数据类型为组合数据类型(如 BOX，POINT 等)且已绑定模块结果，或为 int、float、string 基本类型但绑定的模块结果为数组时，运行流程后可单击*详情*按钮查看数据详情。IMAGE 和 pointset 类型暂不支持保存和显示。

输入来源

显示模块结果订阅全局/局部变量的情况，可将某个模块结果作为全局/局部变量的输入源。您可以在具体模块的*模块结果*区域，找到特定的参数并单击*变量管理*列的订阅按钮，屏幕将显示*选择变量*界面。在该界面，选择具体的全局或局部变量后单击*确定*，可将全局/局部变量的输入源设置为当前选中的模块结果参数。

说明

Group 模块下的*模块结果*区域，您可以可订阅该模块的 Group 局部变量以及全局变量，不可订阅流程局部变量。

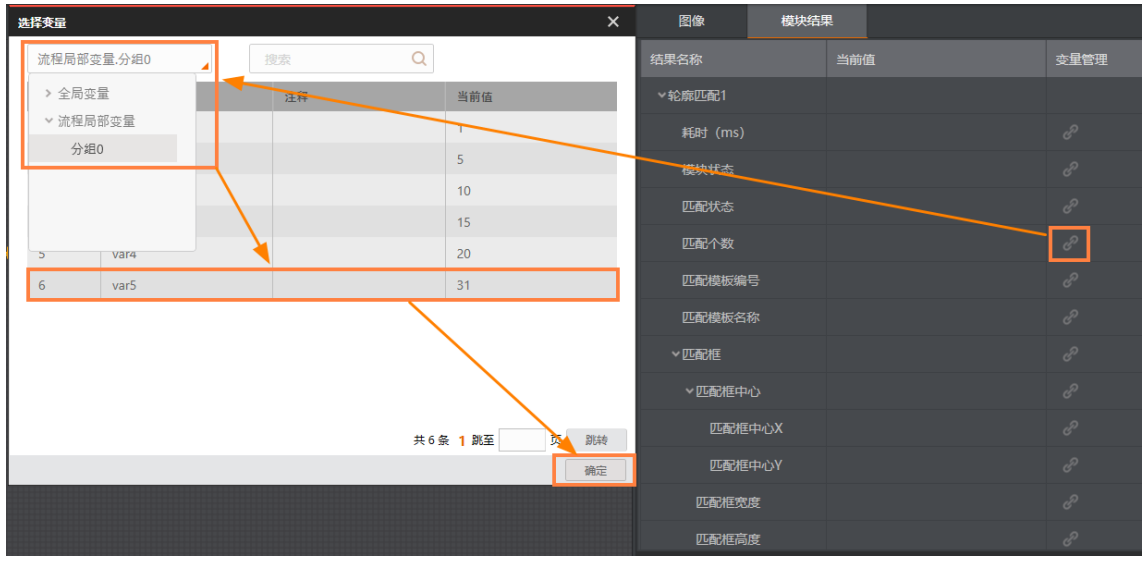


图 8-1 模块结果订阅全局/局部变量

目标输出

可将全局/局部变量赋值给模块的目标参数，可勾选多个。

说明

在目标输出列，*Group* 局部变量仅可订阅该组合模块中的所有参数，流程局部变量仅可订阅该流程中的模块参数，全局变量可订阅方案中各流程以及各组合模块中的模块参数。

初始化

启用后，全局变量可通过发送固定格式的字符串（前缀：变量名称=数值），实现对变量初始值的设置。

- 若初始化单个变量（如变量 var0），可发送 SetGlobalValue:var0=0 将该变量值设为 0。
- 若初始化多个变量（如变量 var0 和 var1），可发送 SetGlobalValue:var0=123,var1=12.25 将该变量值分别设为 123 和 12.25，即多个变量之间需要用逗号隔开。

说明

局部变量无该参数。

5. 重复以上步骤可添加多个全局/局部变量。
6. **可选操作**: 全局/局部变量窗口还可进行如下相关操作：

- *导入/导出*：可通过 gvar 格式文件导入或导出全局/局部变量信息。

说明

导出变量时，**输入来源**和**目标输出**订阅的信息无法导出，只能导出其他信息。

- *搜索*：当全局/局部变量较多时，可快速搜索。

- *置顶/ 上移/ 下移*：可对变量的顺序位置进行上下调整。
- *保存变量*：在打开的方案中，若修改了全局变量，通过*保存变量*可将修改后的变量保存到方案中，无需重新保存方案。该功能仅全局变量支持。

> 说明
> 全局变量和局部变量添加后，均实时生效。

后续处理

添加全局/局部变量后，可在方案中订阅全局/局部变量进行数据传递。

8.3 全局触发

通过快捷工具条中的 进入全局触发界面，可选择事件触发或字符串触发，通过触发事件或字符串来执行相应操作，如切换方案、执行流程、执行模块、执行模块动作。

> 说明
> 如需对全局触发配置导入或导出，可通过 *通信管理* 实现。

- *事件触发*
- *字符串触发*

事件触发

事件触发根据配置的触发事件满足所设定的条件后，能够响应切换方案、执行流程、执行模块、执行模块动作等命令，实现对操作的准确控制。事件触发的具体配置方式如下图所示。



图 8-2 事件触发

1. 单击 新建触发。
2. 设置触发事件，触发事件的具体配置可参考 *文本-协议解析的通信示例* 章节。
3. 单击 对触发事件的参数进行绑定。
   - **目标输出**需绑定到流程模块的运行参数，当事件触发时，会将该参数值赋值到所绑定的运行参数；
   - **设置输入**需绑定到流程模块的输入参数，当事件触发流程运行时，该参数值会作为所绑定的输入参数值。



图 8-3 配置触发事件参数

4. 选择**触发命令类型**，包括如下 4 种。
   - 切换方案：切换到指定路径下的方案。
   - 执行流程：触发流程运行。
   - 执行模块：单独执行订阅模块。
   - 执行模块动作：支持 N 点标定模块的清空标定点动作等。
5. 根据选择的命令类型完成对应的**触发配置**。
   - 触发命令类型选择***切换方案***时，需选择指定方案路径并设置密码。
   - 触发命令类型选择***执行流程***时，需要选择流程 ID 并订阅触发字符。
   - 触发命令类型选择***执行模块***时，需要订阅执行的模块。
   - 触发命令类型选择***执行模块动作***时，需要订阅执行的模块并配置触发动作，目前仅支持 N 点标定的清空标定点操作。

字符串触发

通过**字符串触发**功能，外部通信设备（主要为 TCP、UDP 和串口类型的设备）发送**触发字符**处设置的字符串时，软件执行对应的动作命令。动作命令包括切换方案、执行流程、执行模块、执行模块动作等。

说明

使用该功能时，需确保外部通信设备开启**数据上传**功能，否则软件无法接收字符串信息。

1. 单击 ⊞ 新建字符串触发。
2. 设置**匹配模式**明确字符串触发的方式，可选完全匹配、部分匹配和不匹配。

   完全匹配

   当软件收到的通信字符串和设置的**触发字符**完全一致时，触发设置的动作执行。

   部分匹配

   当软件收到的通信字符串中包含设置的**触发字符**内容时，触发设置的动作执行。

不匹配

软件收到通信字符串后，直接触发设置的动作执行。

3. **匹配模式**选择完全匹配或部分匹配时，通过**触发字符**设置具体的字符串内容。
4. 选择**触发命令类型**，包括以下 4 种。

切换方案

软件运行的方案切换为**触发配置**处指定的方案。

执行流程

软件执行**触发配置**处选择的流程。

执行模块

软件执行**触发配置**处选择的模块。

执行模块动作

软件执行**触发配置**处选择的模块动作。

5. 根据选择的命令类型完成对应的**触发配置**。
   - 触发命令类型选择*切换方案*时，需选择指定方案路径并设置密码。
   - 触发命令类型选择*执行流程*时，需要选择流程 ID 并订阅触发字符。
   - 触发命令类型选择*执行模块*时，需要订阅执行的模块。
   - 触发命令类型选择*执行模块动作*时，需要订阅执行的模块并配置触发动作。例如可重置*变量计算*模块的变量、清空 *N 点标定*模块的标定点。

8.4 全局脚本

全局脚本为方案的全局资源，可控制方案下的其他资源，包括流程、通信数据、模块参数等 。您可调用全局脚本自身的开放接口和 SDK 接口，自行开发代码逻辑，快速实现多流程执行控制、通信数据收发、模块参数设置等业务需求。

本节内容包含：

- *界面概览*
- *使用限制*
- *使用方法*
- *应用示例*
- *与脚本的区别*

界面概览

在 VM 主界面的快捷工具条单击  即可打开*全局脚本*窗口。

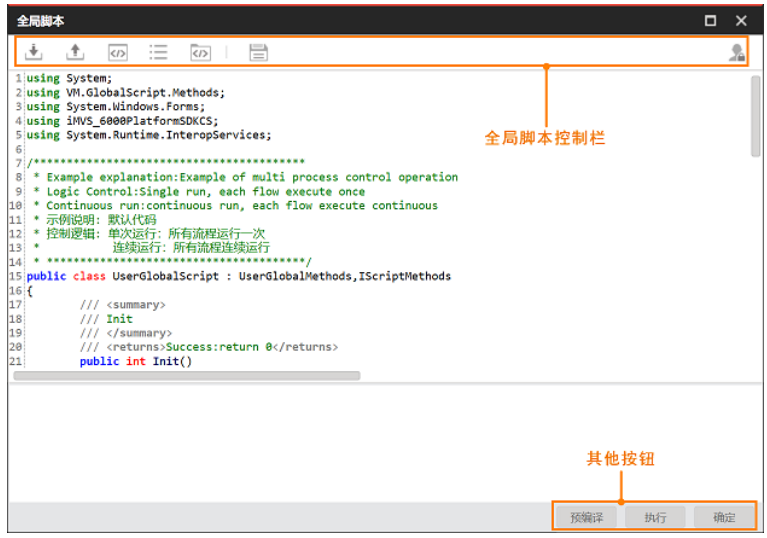


图 8-4 全局脚本窗口

该窗口顶部控制栏的图标介绍如下：

表 8-1 全局脚本控制栏

图标	描述
	导入之前保存至本地的脚本文件（格式：.CS）。
	将当前的脚本文件导出至本地计算机。
	打开全局脚本的示例程序。
	打开全局脚本的工程目录，可 *使用 Visual Studio 进行调试* 。
	打开*引用程序集*窗口。可在该窗口添加或删除引用程序集，详情见 *程序集添加* 。
	保存当前的全局脚本。
	设置密码并确认后，即启用全局脚本加密。 如果启用加密，打开*全局脚本*窗口前，必需先进行密码验证。

该窗口其他按钮的说明如下：

表 8-2 其他按钮

按钮	描述
预编译	预编译全局脚本程序。单击该按钮即调用 方法。
执行	执行流程。单击该按钮即调用 *方法* 。
确定	保存修改后的脚本代码并退出*全局脚本*窗口。

使用限制

- 全局脚本仅支持使用标准 C#语言（Windows 版本）开发。
- 全局脚本无法控制硬触发、通信触发和全局触发。
- 单流程执行按钮（无论是主界面的还是运行界面的）无法触发全局脚本执行。



图 8-5 主界面的单流程执行

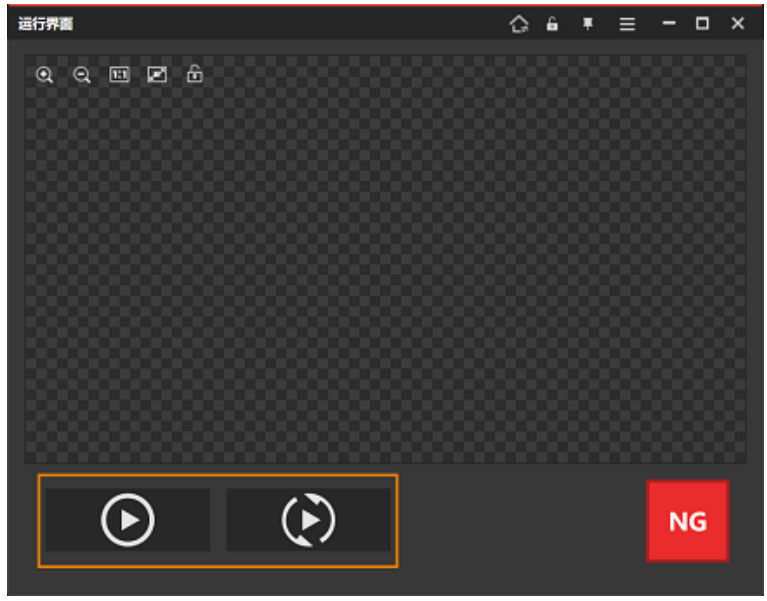


图 8-6 运行界面的单流程执行

使用方法

可调用 *全局脚本自身的开放接口* 自定义代码逻辑。其中的核心接口为 *Init* 和 *Process* 。

- 可在 Init 中实现变量初始化和句柄创建等初始化逻辑，相关工作会在加载方案时完成。
- 可在 Process 中实现变量计算和逻辑处理等具体的功能。具体的功能在多流程执行时执行。

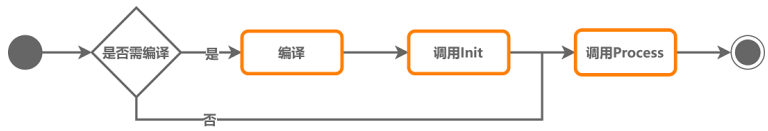


图 8-7 Init 与 Process 方法的执行顺序

说明

- 全局脚本中除了支持调用其自身的开放接口，也支持 *调用 SDK 的接口* 自定义脚本逻辑。两者可混搭调用。
- 全局脚本还支持调用 C#程序集和非托管库，详情参见 *第三方库调用方法* 。

应用示例

除了全局脚本示例程序，您还可参考如下内容，了解如何为特定方案开发全局脚本。

- *应用示例：流程控制*
- *应用示例：设置模块参数*

与脚本的区别

全局脚本与 *脚本* 的区别在于：

区别	全局脚本	脚本
流程的控制范围	可定义方案中多个流程的执行逻辑	*脚本*只能作为流程模块在方案中的某个流程中调用，定义该流程的执行逻辑
SDK 能力调用	支持	不支持

8.4.1 VS 调试

可使用 Visual Studio（以下简称为 VS）调试全局脚本代码，确保其能正常运行。

前提条件

- 已安装 2013 版或更新版本的 Visual Studio。
- 已关闭 vServerApp.exe 进程。

操作步骤

1. 单击 打开*全局脚本*窗口，并单击 打开工程目录。



图 8-8 全局脚本工程目录

2. 使用 VS 将工程目录中的的全局脚本程序（.sln 文件）打开。
3. 将全局脚本程序附加到主进程的 exe 文件。
   1) 在 VS 菜单栏选择 *调试* → *附加到进程* 打开*附加到进程*窗口。
   2) 在该窗口的可用进程选项中，选中主进程的 exe 文件，并单击该窗口右下方的*附加*。
4. 在打开的 VS 工程中，设置断点。单击 VM 快捷工具条的 ▷ 单次执行流程，同时查看 VS 中是否运行进入断点。

说明

如需继续调试，需在 VS 中重新选择附加到进程。请勿直接点击启动。

8.4.2 全局脚本开放接口

全局脚本自身提供开放接口，方便您通过几行代码快速实现多流程控制和通信控制。

全局脚本接口概览

表 8-3 全局脚本接口

实现类型	方法	描述
初始化	*Init*	初始化全局脚本。
流程逻辑处理	*Process*	定义方案下所有流程的批量执行逻辑。
	SetScriptContinuousExecuteInterval	设置 Process 方法连续运行的时间间隔。
	GetScriptContinuousExecuteInterval	获取 Process 方法连续运行的时间间隔。
释放资源	*Dispose*	释放不再需要的资源。
方案加载完成时初始化	*InitAfterLoadSol*	方案加载完成后进行初始化，实现方案加载完成所需的相关功能。

实现类型	方法	描述
获取/设置全局变量	*GetGlobalVariableIntValue*	获取 int 型全局变量。
	SetGlobalVariableIntValue	设置 int 型全局变量。
	GetGlobalVariableFloatValue	获取 float 型全局变量。
	SetGlobalVariableFloatValue	设置 float 型全局变量。
	GetGlobalVariableStringValue	获取 string 型全局变量。
	SetGlobalVariableStringValue	设置 string 型全局变量。
通信接收事件	*StartGlobalCommunicate*	初始化全局脚本与第三方通信设备之间的通信端口。
	RegesiterReceiveCommunicateDataEvent	注册通信数据接收事件。注册后，VM 收到来自第三方通信设备的数据时，将触发“通信数据接收事件”。
	UnRegesiterReceiveCommunicateDataEvent	注销通信数据接收事件。注销后，VM 收到来自第三方通信设备的数据时，不再触发“通信数据接收事件”。
	OnReceiveCommunicateDataEvent	“通信数据接收事件”的回调函数。
发送通信数据	*SendCommDeviceData*	指定某个通信设备，使其发送特定类型的数据。
调试	*ConsoleWrite*	将调试信息打印至 ***DebugView*** 中。

Init

接口原型

public int Init(){}

描述

初始化全局脚本。可在此方法中实现初始化相关操作 。该方法在加载方案或预编译全局脚本时执行。

Process

接口原型

public int Process(){}

描述

定义方案下所有流程的批量执行逻辑。具体示例参见 VM 安装路径下的*全局脚本_单流程执行*和*全局脚本_多流程控制执行*，该两个示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

使用限制

以下触发场景或需求，无法通过调用 Process 方法实现流程控制。

- 硬触发（硬件触发方案）
- 通信触发方案
- 全局触发
- 在主界面或运行界面控制单流程的执行

运行逻辑

该方法在 VM 主界面快捷工具条上的全流程执行按钮被点击后调用。



图 8-9 全流程执行按钮

被调用后，该方法的具体运行逻辑如下：

- 场景 1：
  单击 ▷ 则调用一次该方法。在全局脚本的默认代码中，定义在 Process 中的 DefaultExecuteProcess()会被调用，控制方案中的所有流程执行一次。
- 场景 2：
  单击 ⟳ 则按特定时间间隔重复调用该方法。
  重复调用该方法的时间间隔，需通过调用 *SetScriptContinuousExecuteInterval* 方法进行配置。

> **说明**
> 在场景 2 中，仅在 Process 方法第一次调用时，方案的全部流程会被连续执行。具体的逻辑见 *图 8-10*。

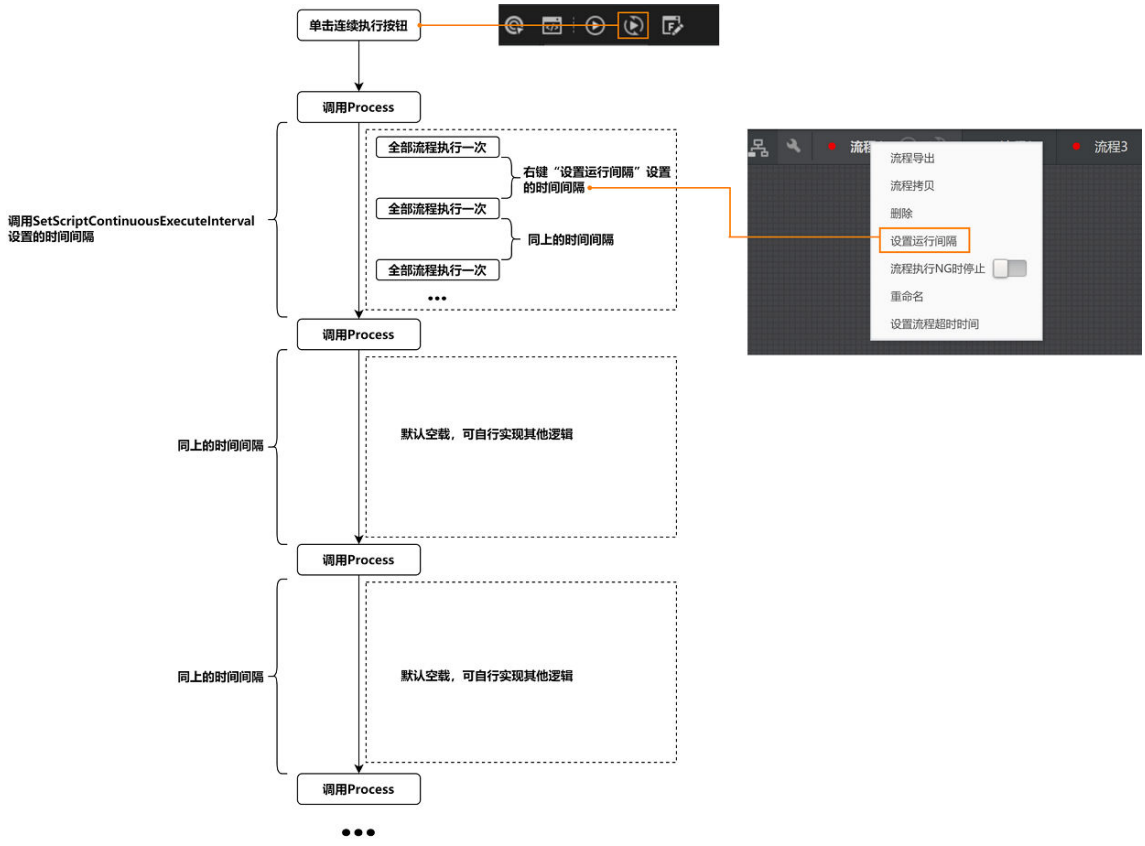


图 8-10 场景 2 中 Process 执行逻辑

SetScriptContinuousExecuteInterval

接口原型

void SetScriptContinuousExecuteInterval (uint nMilliSecond);

描述

设置 Process 方法连续运行的时间间隔 。具体示例参见 VM 安装路径下的*全局脚本_多流程控制执行*，该示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

nMilliSecond：uint 类型，时间间隔（单位：ms）

返回值

无

调用示例

SetScriptContinuousExecuteInterval(1000);

GetScriptContinuousExecuteInterval

接口原型

uint GetScriptContinuousExecuteInterval();

描述

获取 Process 方法连续运行的时间间隔。

返回值

如调用成功，则返回时间间隔值。

如调用失败，则返回-1。

调用示例

uint x = GetScriptContinuousExecuteInterval();

Dispose

接口原型

public override void Dispose(){}

调用时机

关闭程序或重新编译时。

描述

释放不再需要的资源。例如，释放 Init 中创建的句柄。

InitAfterLoadSol

接口原型

public override int InitAfterLoadSol(){}

描述

方案加载完成后进行初始化，实现方案加载完成所需的相关功能。例如，可在此方法中发送方案加载完成的信号。

GetGlobalVariableIntValue

接口原型

Int GetGlobalVariableIntValue(string paramName, ref int paramValue)

描述

获取 int 型全局变量的值 。具体示例参见 VM 安装路径下的*全局脚本_全局变量设置*, 该示例相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

paramName：string 类型，变量名称。

输出参数

paramValue：int 类型，变量值。

返回值

0：调用成功

非 0 返回值：调用失败。建议根据错误码信息排查失败原因。

SetGlobalVariableIntValue

接口原型

Int SetGlobalVariableIntValue (string paramName, int paramValue)

描述

设置 int 型全局变量的值。具体示例参见 VM 安装路径下的*全局脚本_全局变量设置*，该示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：int 类型，变量值

返回值

0：调用成功

非 0 返回值：调用失败。建议根据错误码信息排查失败原因

GetGlobalVariableFloatValue

接口原型

Int GetGlobalVariableFloatValue(string paramName, ref float paramValue)

描述

获取 float 型全局变量的值。具体示例参见 VM 安装路径下的*全局脚本_全局变量设置*，该示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：float 类型，变量值

返回值

0：调用成功

非 0 返回值：调用失败。建议根据错误码信息排查失败原因

SetGlobalVariableFloatValue

接口原型

Int SetGlobalVariableFloatValue (string paramName, float paramValue)

描述

设置 float 型全局变量值。具体示例参见 VM 安装路径下的*全局脚本_全局变量设置*，该示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：float 类型，变量值

返回值

0：调用成功

非 0 返回值：调用失败。建议根据错误码信息排查失败原因

GetGlobalVariableStringValue

接口原型

Int GetGlobalVariableStringValue (string paramName, ref string paramValue)

描述

获取 string 型全局变量。具体示例参见 VM 安装路径下的*全局脚本_全局变量设置*, 该示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：string 类型，变量值

返回值

0：调用成功

非 0 返回值：调用失败。建议根据错误码信息排查失败原因

SetGlobalVariableStringValue

接口原型

Int SetGlobalVariableStringValue (string paramName, string paramValue)

描述

设置 string 型全局变量。具体示例参见 VM 安装路径下的*全局脚本_全局变量设置*, 该示例的相对路径：.\Applications\GlobalScript\Samples\CH\3.x。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：string 类型，变量值

返回值

0：调用成功

非 0 返回值：调用失败。建议根据错误码信息排查失败原因

StartGlobalCommunicate

接口原型

bool StartGlobalCommunicate ()

描述

初始化全局脚本与第三方通信设备的通信端口。

调用前提

已在添加第三方通信设备，更多相关详情请参见 ***通信设备管理*** 。

返回值

true：调用成功

false：调用失败

相关接口

RegesiterReceiveCommunicateDataEvent

RegesiterReceiveCommunicateDataEvent

接口原型

Void RegesiterReceiveCommunicateDataEvent()

调用时机

调用 StartGlobalCommunicate 方法后。

描述

注册通信数据接收事件。注册后，VM 收到来自第三方通信设备的通信数据时，将触发 OnReceiveCommunicateDataEvent 回调函数。具体示例参见 VM 安装路径下的*全局脚本_通信接收*，该示例的相对路径：.\Applications\GlobalScript\Samples\CH\4.x。

返回值

无

相关接口

UnRegesiterReceiveCommunicateDataEvent

UnRegesiterReceiveCommunicateDataEvent

接口原型

Void UnRegesiterReceiveCommunicateDataEvent ()

描述

注销通信数据接收事件。注销后，VM 收到来自第三方通信设备的通信数据时，不再触发 OnReceiveCommunicateDataEvent 回调函数。

返回值

无

OnReceiveCommunicateDataEvent

回调函数原型

void UserGlobalMethods_OnReceiveCommunicateDataEvent(ReceiveDataInfo dataInfo)

描述

通信数据接收事件。该事件包含的参数说明见下表。

参数

可通过该回调函数获取的数据结构体为 dataInfo（类型：ReceiveDataInfo）。该结构体包含如下数据：

communicateType

：CommunicateType 型，通信设备类型，包括 TCP 客户端、TCP 服务端、UDP、串口、Modbus 和 PLC。

DeviceID

全局通信模块中通信设备索引下标，类型为 int

DeviceAddressID

PLC 或者 Modbus 设备子地址索引下标，类型为 int

DeviceData

接收到的数据，类型为 byte[]数组

SendCommDeviceData

可在全局脚本中调用如下方法，指定某个通信设备发送特定类型的数据。

表 8-4 具体方法

接口原型	说明
SendCommDeviceData(string data,int deviceID)	指定某个第三方通信设备发送 string 类型的数据。 • 输入参数： ◦ data：string 类型，待发送的数据。 ◦ deviceID：int 类型， *通信管理* 中所添加的第三方通信设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。 • 返回值：无
SendCommDeviceData(byte[] bytedata,int deviceID)	指定某个第三方通信设备发送十六进制数据。 • 输入参数： ◦ bytedata：byte[]，待发送的十六进制数组。 ◦ deviceID：int 类型， *通信管理* 中所添加的第三方通信设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。 • 返回值：无
SendCommDeviceData(string data,int deviceID,int addressID,DataType dataType)	指定某个第三方通信设备发送 int、float 或 string 类型的数据。 • 输入参数： ◦ data：string 类型，待发送的的数据，如需发送多个数据，请用“;”隔开。 ◦ deviceID：int 类型， *通信管理* 中所添加的第三方通信设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。

接口原型	说明
	○ addressID：int 类型，*通信管理*中所添加的第三方通信设备的地址 ID，即设备子地址。地址 ID 的示例见下文的 *图 8-11*。 ○ dataType：DataType 类型，数据类型（包括 int、float 和 string）。 ● 返回值：无
SendCommDeviceData(byte[] bytedata,int deviceID,int addressID,DataType.ByteType)	指定某个第三方通信设备发送十六进制数据。 ● 输入参数： ○ bytedata：byte[]，待发送的十六进制数据。 ○ deviceID：int 类型，*通信管理*中所添加的第三方通信设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。 ○ addressID：int 类型，*通信管理*中所添加的第三方通信设备的地址 ID，即设备子地址。地址 ID 的示例见下文的 *图 8-11*。 ● 返回值：无



图 8-11 设备 ID 与地址 ID 示例

上图的示例中，TCP 客户端和三菱 MC 的设备 ID 分别为 1 和 2，三菱 MC 的地址 ID 为 1。

ConsoleWrite

接口原型

void ConsoleWrite(string content)

描述

将信息打印至 DebugView 中。

输入参数

Content：string 类型，待打印的内容

返回值

无

8.4.3 SDK 调用

可通过全局脚本调用 VMSDK 的接口，实现获取模块运行参数、回调算法底层运行数据、控制方案流程等功能。

本主题内容包含：

- *SDK 回调*
- *SDK 接口调用*

SDK 回调

全局脚本默认在部分事件发生时触发 SDK 回调，获取对应的回调数据。
以下示例展示了模块输出结果数据和流程工作状态变化时触发的回调。

```
        /// <summary>
        /// SDK callback function
        /// </summary>
        public override void ResultDataCallBack(IntPtr outputPlatformInfo, IntPtr puser)
        {
            base.ResultDataCallBack(outputPlatformInfo, puser);
            ImvsSdkPFDefine.IMVS_PF_OUTPUT_PLATFORM_INFO struInfo =
(ImvsSdkPFDefine.IMVS_PF_OUTPUT_PLATFORM_INFO)Marshal.PtrToStructure(outputPlatformInfo,
typeof(ImvsSdkPFDefine.IMVS_PF_OUTPUT_PLATFORM_INFO));
            switch (struInfo.nInfoType)
            {
                //Get module result
                //获取模块结果数据
                case
(uint)ImvsSdkPFDefine.IMVS_CTRLC_OUTPUT_PlATFORM_INFO_TYPE.IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_
MODULE_RESULT:
                    {
                        ImvsSdkPFDefine.IMVS_PF_MODULE_RESULT_INFO_LIST_P resultInfo =
(ImvsSdkPFDefine.IMVS_PF_MODULE_RESULT_INFO_LIST_P)Marshal.PtrToStructure(struInfo.pData,
typeof(ImvsSdkPFDefine.IMVS_PF_MODULE_RESULT_INFO_LIST_P));
                        break;
                    }
                ///Get process execute state
                //获取流程运行状态
                case
(uint)ImvsSdkPFDefine.IMVS_CTRLC_OUTPUT_PlATFORM_INFO_TYPE.IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_
WORK_STATE:
                    {
                        ImvsSdkPFDefine.IMVS_PF_MODULE_WORK_STAUS stWorkStatus =
(ImvsSdkPFDefine.IMVS_PF_MODULE_WORK_STAUS)Marshal.PtrToStructure(struInfo.pData,
typeof(ImvsSdkPFDefine.IMVS_PF_MODULE_WORK_STAUS));
                        break;
```

```
                }
            default:
                break;
        }
    }
```

可触发回调的事件和对应的回调数据类型，具体见下表。

表 8-5 事件与回调数据类型

事件	回调数据类型
模块输出结果	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_MODULE_RESULT = 0
单个流程连续执行开始	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_START_CONTINUOUSLY = 1
单个流程停止执行（的状态信息）	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_STOP = 2
流程工作状态变化	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_WORK_STATE = 3
模块心跳异常	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_HB_MODU = 4
服务心跳异常	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_HB_SERVER = 5
平台界面心跳异常	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_HB_CLIENT = 6
加密狗异常	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_DONGLE = 7
方案加载 UI 层文件	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_SOLUTION_LOAD_INTERFACE_FILE = 8
方案保存完成	IMVS_ENUM_CTRLC_OUTPUT_PlATFORM_INFO_SOLUTION_SAVE_END = 9
方案加载完成	IMVS_ENUM_CTRLC_OUTPUT_PlATFORM_INFO_SOLUTION_LOAD_END = 10
模块进程（单进程）心跳异常	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_HB_SP_PROXY = 11
全局脚本异常	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_GLOBALSCRIPT_CRASH = 12

事件	回调数据类型
方案加载时模块报错	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_LOAD_MODULE_WARN = 13
流程导出完成	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_EXPORT_PROCESS_END = 14
流程导入完成	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_IMPORT_PROCESS_END = 15
流程注册取消	IMVS_ENUM_CTRLC_OUTPUT_PLATFORM_INFO_PROCEDURE_UNREGISTER = 16

SDK 接口调用

除可自动触发回调，全局脚本也支持调用 VMSDK 的接口。SDK 接口详情请参见*《VMSDK 开发指南》*。

说明

*《VMSDK 开发指南》*可到软件安装路径下获取。

8.4.4 程序集添加

全局脚本默认已添加 VM 二次开发需要的基础类库。如需调用其他类库的程序集，可自行添加。如果程序集无需再调用，可自行删除。

操作步骤

说明

仅支持添加 C#程序集。如需调用非托管库（如 C++库），请参见 *第三方库调用方法* 。

1. 在*全局脚本*窗口单击 打开*引用程序集*窗口。

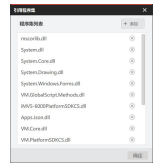


图 8-12 引用程序集窗口

说明

VM 二次开发基础类库（如上图所示）的调用方法，请参见*算法开发平台 SDK 开发指南*。

2. 单击*添加*，选择本地路径添加所需的程序集。

   示例

   如需添加 4.x 版本新增模块的类库的程序集，可从 VM 安装路径下的 Assembly 文件夹添加，该文件夹的相对路径：.\Development\V4.x\ComControls\Assembly。

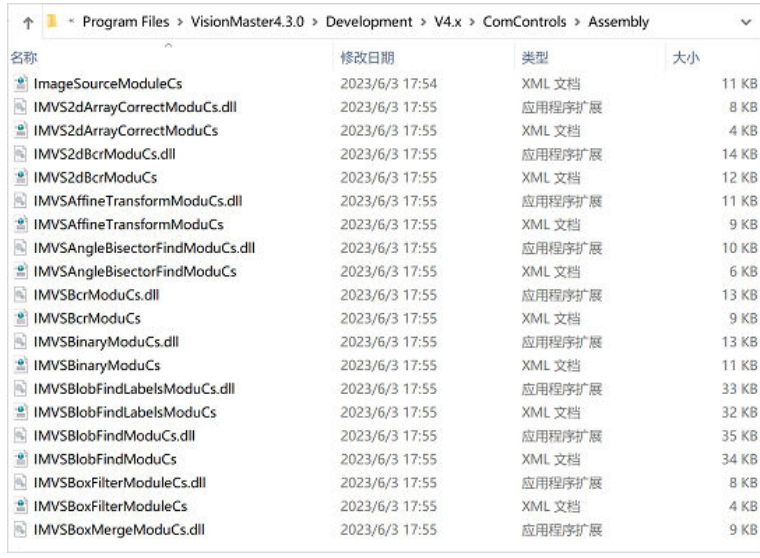


图 8-13 4.x 版本模块类库的程序集

3. **可选操作:** 单击 ⊗ 将对应的程序集从全局脚本中删除。
4. 单击*确定*。

8.4.5 第三方库调用方法

全局脚本支持调用第三方库。可调用非托管库，也可直接引用 C#库。

调用非托管库

由于全局脚本为标准的 C#程序，所以全局脚本无法直接调用 C/C++动态库，而是需要通过导出 C 接口的方式让 C#调用。以下为调用 C++动态库时的声明示例，声明中需引用 System.Runtime.InteropServices 程序集）。

```
using System.Runtime.InteropServices;
class UserScript:ScriptMethods,IProcessMethods
{
    [DllImport("ShellRTestAPI.dll", EntryPoint = "fnShellRTestAPI_GetValue")]
    public static extern int fnShellRTestAPI_GetValue();
    [DllImport("ShellRTestAPI.dll", EntryPoint = "fnShellRTestAPI_Add")]
    public static extern int fnShellRTestAPI_Add(int nValueFirst, int nValueSecond);
    public void Init()
    {  processCount = 0;}
    public bool Process()
```

```
    {  int nTemp = fnShellRTestAPI_Add(111, 888);
       SetIntValue("var1_Output", nTemp);
       return true;}
}
```

直接引用 C#库

第三方 C#库可在全局脚本中直接引用，具体操作步骤参见 *程序集添加* 。

8.4.6 应用示例：流程控制

本章节以简易业务场景为例，介绍如何通过全局脚本模块定义多个流程的执行逻辑。

本节内容包含：

- *方案需求*
- *方案思路*
- *方案搭建*
- *方案效果测试*

方案需求

假设针对某视觉业务场景的方案中原本已搭建了 3 个流程，分别为流程 1、流程 2 和流程 3。随着业务的发展，现需要实现如下多流程控制需求：流程 1 和流程 2 都执行 1 次（不分先后）之后，触发流程 3 自动执行 1 次。

由于 VM 支持执行单个流程和批量执行方案下所有流程，该需求可进一步拆分为 2 个子需求：

表 8-6 子需求

子需求	描述
子需求 1	流程 1 和流程 2 分别执行 1 次（不分先后）之后，触发流程 3 自动执行 1 次。
子需求 2	流程 1、流程 2 和流程 3 批量执行 1 次后，触发流程 3 自动执行 1 次。

方案思路

方案中多个流程的执行逻辑，可通过全局脚本实现。

具体的实现途径见下图。

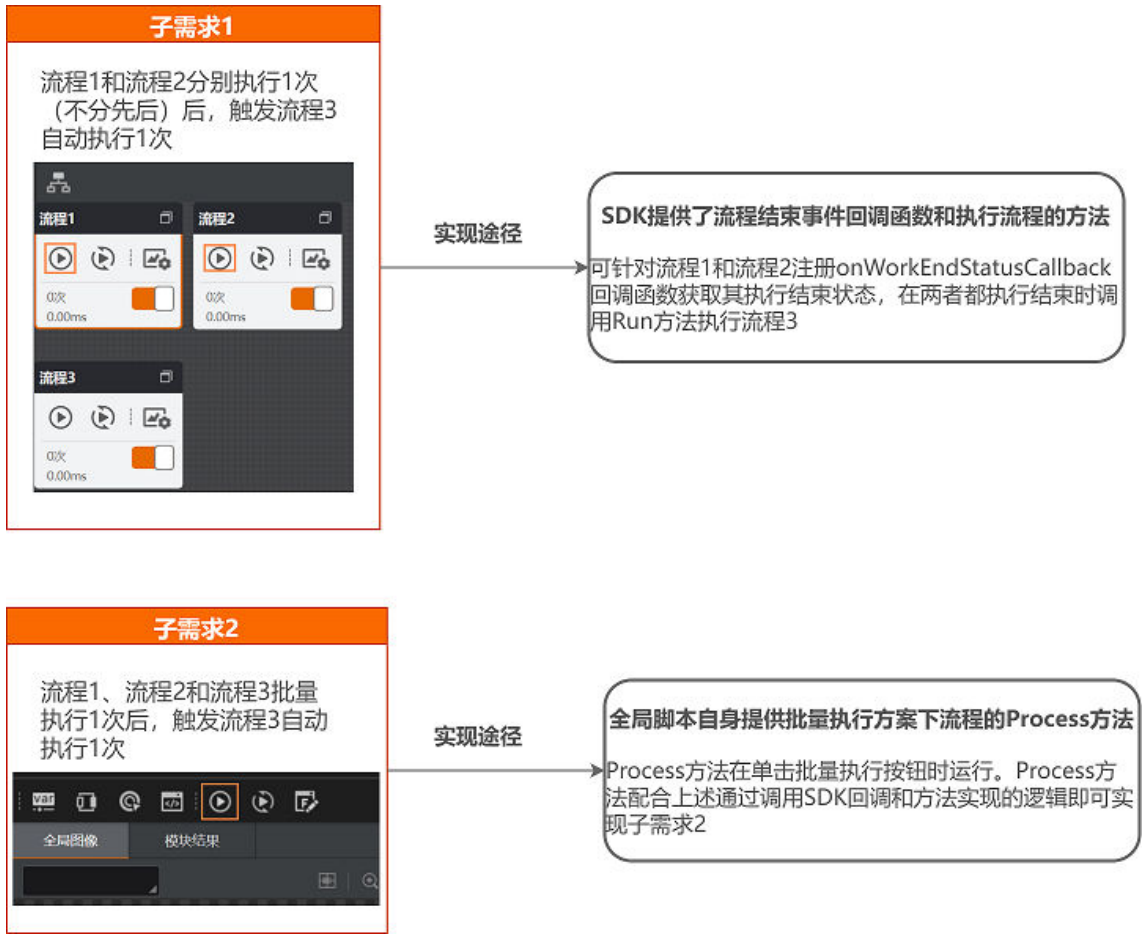


图 8-14 实现途径

方案搭建

基于上述思路，需自行开发全局脚本的代码逻辑实现多流程控制的需求。

前提条件

已在方案中的搭建 3 个流程。

说明

本节仅就全局脚本中涉及的核心接口调用做简要介绍。其中 SDK 的类、方法和属性的详情，请参见 VM 的 SDK 开发指南（.NET）。

操作步骤

1. 单击快捷工具条上的 打开*全局脚本*窗口，准备开始 C#编程。

2. 调用全局脚本的 *Init* 方法，在该方法中声明流程 1、流程 2 和流程 3 的句柄，实现全局脚本的初始化。
3. 调用全局脚本的 *InitAfterLoadSol* 方法，实现在方案加载结束时对流程 1、流程 2 和流程 3 进行初始化。
4. 在 *InitAfterLoadSol* 方法中，为流程 1 和流程 2 注册.NET SDK 的 OnWorkEndStatusCallBack 回调函数。该回调函数用于在流程执行结束时触发“流程执行结束”事件。
5. 调用.NET SDK 的 OnWorkEndStatusCallBack(object sender, EventArgs e)方法获取流程 1 和流程 2 执行结束事件的数据，并在该方法中实现多流程执行逻辑。
   1) 自行定义流程 1 和流程 2 执行结束状态的变量，并自行实现流程执行状态的判断逻辑（即流程 1 和流程 2 都执行 1 次后再执行 1 次流程 3）。
   2) 调用.NET SDK 的 Run 方法，异步执行 1 次流程 3。

   示例代码如下：

```
    private void Pro1_OnWorkEndStatusCallBack(object sender, EventArgs e)
    {
        if (e != null)
        {
            ValueEventArgs eventArgs = (ValueEventArgs)e;
            ImvsSdkDefine.IMVS_MODULE_WORK_STAUS status =
(ImvsSdkDefine.IMVS_MODULE_WORK_STAUS)eventArgs.Value;
            //流程 1 执行完成
            if (status.nProcessID == 10000)
            {
                pro1RunEnd = true;
            }
            else if (status.nProcessID == 10001)
            {
                pro2RunEnd = true;
            }
            else if (status.nProcessID == 10002)
            {
                pro1RunEnd = false;
                pro2RunEnd = false;
            }
            if (pro1RunEnd && pro2RunEnd)
            {
                //异步执行一次
                pro3.Run("", false);
            }
        }
    }
```

说明

以上示例中的 nProcessID 为 SDK 中定义“流程 ID”的属性，属于 ModuleResult.ModulInfo 结构体。“流程 ID”可从软件界面的如下路径获取：



图 8-15 流程 ID 获取

6. 调用全局脚本的 *Dispose* 方法，释放流程 1 和流程 2 的执行结束事件回调所占用的资源。至此，已实现“子需求 1”。
7. 调用全局脚本的 *Process* 方法，实现方案下所有流程在单击批量执行按钮时同时执行 1 次。该方法的逻辑与上述步骤实现逻辑共同生效，即实现“子需求 2”。

示例

上述步骤的完整示例代码如下：

```
using System;
using VM.GlobalScript.Methods;
using System.Windows.Forms;
using iMVS_6000PlatformSDKCS;
using System.Runtime.InteropServices;
using VM.Core;
using VM.PlatformSDKCS;
using ImageSourceModuleCs;
using IMVSCircleFindModuCs;
using VMControls.Interface;
using System.Drawing;
using System.Drawing.Imaging;
using System.Threading.Tasks;
/*****************************************
 * Example explanation:Example of multi process control operation
 * Logic Control:Single run, each flow execute once
 * Continuous run:continuous run, each flow execute continuous
 * 示例说明: 默认代码
 * 控制逻辑: 单次运行：所有流程运行一次
 *           连续运行：所有流程连续运行
 * *****************************************/
public class UserGlobalScript : UserGlobalMethods, IScriptMethods
{
    /// <summary>
    /// Init
    /// </summary>
    /// <returns>Success:return 0</returns>
    public int Init()
    {
        //SDK init
        int ret = InitSDK();
```

```
        //InitAfterLoadSol();
        if(pro1==null)
        {pro1 = (VmProcedure)VmSolution.Instance["流程 1"];}
        if(pro2==null)
        {pro2 = (VmProcedure)VmSolution.Instance["流程 2"];}
        if (pro3 == null)
        { pro3 = (VmProcedure)VmSolution.Instance["流程 3"]; }
        return ret;
    }

    private VmProcedure pro1 = null;
    private VmProcedure pro2 = null;
    private VmProcedure pro3 = null;
    private bool pro1RunEnd = false;
    private bool pro2RunEnd = false;

    //加载方案完成触发动作
    public override int InitAfterLoadSol()
    {
        if(pro1==null)
        {pro1 = (VmProcedure)VmSolution.Instance["流程 1"];}
        if (pro1 != null)
        {
            pro1.OnWorkEndStatusCallBack += Pro1_OnWorkEndStatusCallBack;
        }
        if (pro2 == null)
        { pro2 = (VmProcedure)VmSolution.Instance["流程 2"]; }
        if (pro2 != null)
        {
            pro2.OnWorkEndStatusCallBack += Pro1_OnWorkEndStatusCallBack;
        }
        if (pro3 == null)
        { pro3 = (VmProcedure)VmSolution.Instance["流程 3"]; }
        return 0;
    }

    private void Pro1_OnWorkEndStatusCallBack(object sender, EventArgs e)
    {
        if (e != null)
        {
            ValueEventArgs eventArgs = (ValueEventArgs)e;
            ImvsSdkDefine.IMVS_MODULE_WORK_STAUS status =
(ImvsSdkDefine.IMVS_MODULE_WORK_STAUS)eventArgs.Value;
            //流程 1 执行完成
            if (status.nProcessID == 10000)
            {
                pro1RunEnd = true;
            }
            else if (status.nProcessID == 10001)
            {
                pro2RunEnd = true;
            }
```

```
            else if (status.nProcessID == 10002)
            {
                pro1RunEnd = false;
                pro2RunEnd = false;
            }
            if (pro1RunEnd && pro2RunEnd)
            {
                //异步执行一次
                pro3.Run("", false);
            }
        }
    }

    public override void Dispose()
    {
        base.Dispose();
        if (pro1 != null)
        {
            pro1.OnWorkEndStatusCallBack -= Pro1_OnWorkEndStatusCallBack;
        }
        if (pro2 != null)
        {
            pro2.OnWorkEndStatusCallBack -= Pro1_OnWorkEndStatusCallBack;
        }
    }

    /// <summary>
    /// execute function
    /// Single run:the function execute once
    /// Continuous run:Repeat the function at regular intervals
    /// 运行函数
    /// 单次执行:该函数执行一次
    /// 连续执行:以一定时间间隔重复执行该函数
    /// </summary>
    /// <returns>Success:return 0</returns>
    public int Process()
    {
        //m_operateHandle SDK handle
        if (m_operateHandle == IntPtr.Zero)
        { return ImvsSdkPFDefine.IMVS_EC_NULL_PTR; }

        //All processes are executed by default
        //If execute in your own define logic,please remove the
function :DefaultExecuteProcess, Create your own logic function.
        //默认执行全部流程，
        //如果自定义流程执行逻辑，请移除 DefaultExecuteProcess 方法，编写自定义流程执行逻辑代码
        int nRet = DefaultExecuteProcess();
        return nRet;
    }
}
```

方案效果测试

完成该方案的搭建后，可在 VM 软件上测试方案效果。

- *测试场景 1*：先执行 1 次流程 1，再执行 1 次流程 2。
  测试结果：触发流程 3 执行 1 次（见如下动图）。



图 8-16 测试场景 1

- *测试场景 2*：先执行 1 次流程 2，再执行 1 次流程 1。
  测试结果：触发流程 3 执行 1 次（见如下动图）。



图 8-17 测试场景 2

- *测试场景 3*：批量同时执行各流程 1 次
  测试结果：流程 3 在和流程 1 和流程 2 同时执行 1 次的基础上，由于流程 1 和流程 2 都执行结束，又被触发执行 1 次。因此最终流程 3 在该场景总共执行了 2 次（见如下动图）。

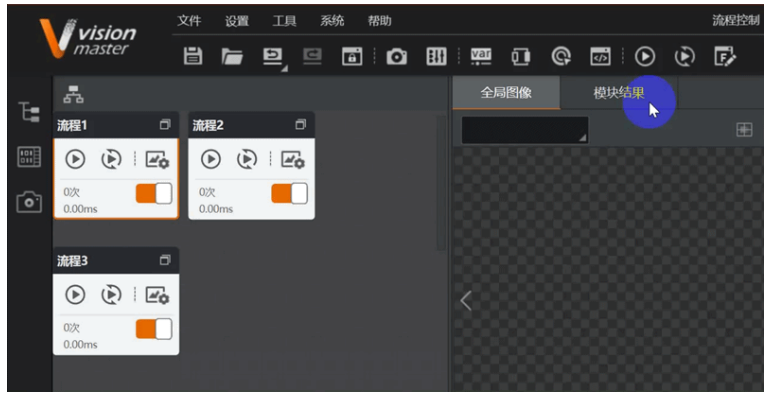


图 8-18 测试场景 3

8.4.7 应用示例：设置模块参数

本章节以简易场景为例，介绍如何通过全局脚本模块实现模块参数的自动设置。

本节内容包含：

- *方案需求*
- *方案思路*
- *方案搭建*
  - *搭建流程*
  - *自定义全局脚本*
- *方案效果测试*

方案需求

假设某视觉业务场景需实现如下需求：

表 8-7 方案需求

需求	描述
需求 1	方案中包含 2 个流程：流程 1 和流程 2，其中流程 1 中包含*快速匹配*模块。
需求 2	2 个流程批量同时执行时，自动将流程 1 中*快速匹配*模块的**最大匹配个数**设置为 *10*。
需求 3	2 个流程批量同时执行时，获取流程 1 中*快速匹配*模块结果中的**匹配个数**和**匹配点**。

方案思路

基于上述业务需求，方案搭建的思路框架如下：

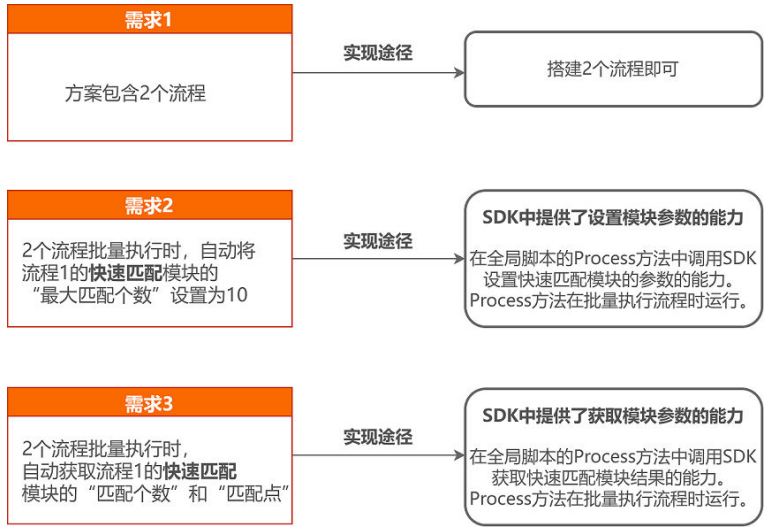


图 8-19 方案搭建思路

方案搭建

基于上述思路，需搭建流程并自行开发全局脚本。

搭建流程

本节仅对本应用示例的流程的核心操作做简要呈现，具体流程中调用的模块不作赘述。

操作步骤

1. 搭建流程 1，在流程 1 中调用*快速匹配*模块。
2. 单击流程编辑区域顶部的 新建流程（此处即“流程 2”），并搭建流程 2。

自定义全局脚本

本节仅对全局脚本中的核心接口调用作简要介绍。其中 SDK 相关的类、方法和属性的详情，请参见 VM 的 SDK 开发指南（.NET）。

操作步骤

1. 单击快捷工具条的 ，打开*全局脚本*窗口，开始准备进行 C#编程。

2. 调用全局脚本的 *Init* 方法，对全局脚本进行初始化。
3. 调用全局脚本的 *Process* 方法，并在该方法中实现如下代码逻辑。
   1) 通过.NET SDK 的 IMVSFastFeatureMatchModuTool 类，在代码中定义流程 1 中的*快速匹配*模块（即“快速匹配 1”），如下图所示。

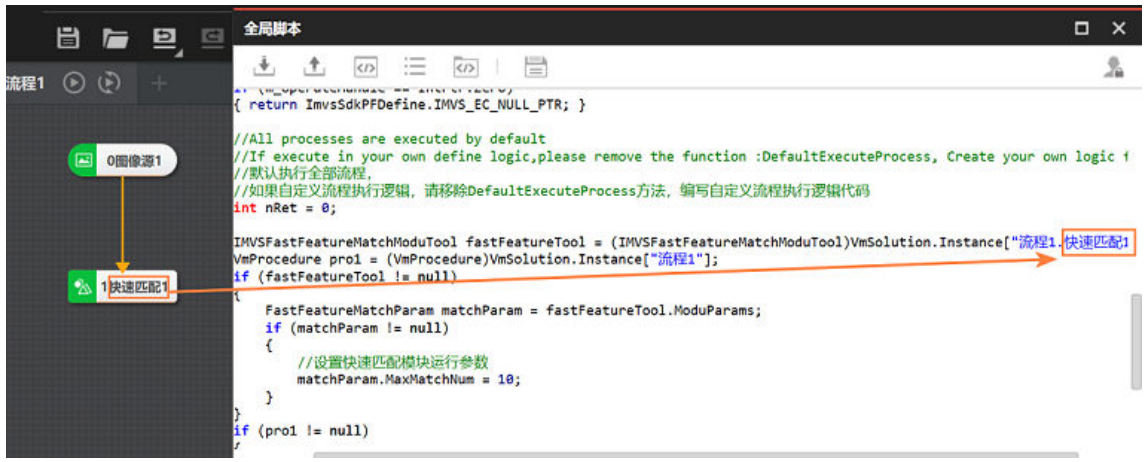


图 8-20 在代码中定义快速匹配模块

   2) 通过.NET SDK 的 FastFeatureMatchParam 类下的 MaxMatchNum 属性，将*快速匹配*模块的**最大匹配个数**的值设置为 10。
   3) 通过.NET SDK 的 FastFeatureMatchResult 类，在代码中定义*快速匹配*模块的模块结果。
   4) 通过.NET SDKFastFeatureMatchResult 类下的 MatchNum 和 MatchPoint 属性，分别获取*快速匹配*模块的**匹配个数**和**匹配点**。

示例

上述步骤的完整示例代码如下：

```
using System;
using System.Windows.Forms;
using System.Runtime.InteropServices;
using System.Collections;
using VM.GlobalScript.Methods;
using iMVS_6000PlatformSDKCS;
using VM.Core;
using VM.PlatformSDKCS;
using ImageSourceModuleCs;
using IMVSFastFeatureMatchModuCs;
using System.Collections.Generic;


/*****************************************
 * 示例说明：获取流程对象，模块对象，设置参数，运行，获取结果
 * ***************************************/
public class UserGlobalScript : UserGlobalMethods, IScriptMethods
{
    /// <summary>
    /// Init
```

```
    /// </summary>
    /// <returns>Success:return 0</returns>
    public int Init()
    {
        //SDK init
        return InitSDK();
    }

    /// <summary>
    /// execute function
    /// Single run:the function execute once
    /// Continuous run:Repeat the function at regular intervals
    /// 运行函数
    /// 单次执行:该函数执行一次
    /// 连续执行:以一定时间间隔重复执行该函数
    /// </summary>
    /// <returns>Success:return 0</returns>
    public int Process()
    {
        //m_operateHandle SDK handle
        if (m_operateHandle == IntPtr.Zero)
        { return ImvsSdkPFDefine.IMVS_EC_NULL_PTR; }

        //All processes are executed by default
        //If execute in your own define logic,please remove the
function :DefaultExecuteProcess, Create your own logic function.
        //默认执行全部流程,
        //如果自定义流程执行逻辑，请移除 DefaultExecuteProcess 方法，编写自定义流程执行逻辑代码
        int nRet = 0;

        IMVSFastFeatureMatchModuTool fastFeatureTool =
(IMVSFastFeatureMatchModuTool)VmSolution.Instance["流程 1.快速匹配 1"];
        VmProcedure pro1 = (VmProcedure)VmSolution.Instance["流程 1"];
        if (fastFeatureTool != null)
        {
            FastFeatureMatchParam matchParam = fastFeatureTool.ModuParams;
            if (matchParam != null)
            {
                //设置特征匹配模块运行参数
                matchParam.MaxMatchNum = 10;
            }
        }
        if (pro1 != null)
        {
            pro1.Run();
            //获取特征匹配模块的运行结果
            FastFeatureMatchResult matchResult = fastFeatureTool.ModuResult;
            if (matchResult != null)
            {
                //获取匹配个数
                int matchnum = matchResult.MatchNum;
                //获取匹配点
```

```
                List<PointF> matchpoint = matchResult.MatchPoint;
                int a = 0;
            }
        }
        return nRet;
    }
}
```

方案效果测试

完成该方案的搭建后，可在 VM 软件上测试方案效果。

测试需求 2 实现效果

如以下视频所示，流程 1 的*快速匹配*模块的**最大匹配个数**在流程 1 执行前的初始值为 5。流程 1 和流程 2 批量执行后，该参数的值自动设置为 10。

测试需求 3 实现效果

可在全局脚本中调用 MessageBox.Show 方法打印全局脚本获取的模块结果，验证全局脚本是否已成功获取。如果弹窗中显示了模块结果的值，则说明全局脚本获取模块结果成功。
如下视频仅以打印**匹配个数**为例进行说明。

8.5 数据队列

数据队列模块可用于在流程之间传递数据。在该模块的配置界面，您可以创建多个队列用于缓存不同数据类型的数据，并且设置软件触发指定流程从数据队列中获取数据。

流程编辑区域点击 切换为全流程时，左侧选择 并拖动到编辑区域即可新增一个数据队列模块。双击数据队列后，可对 *队列设置* 及 *触发流程* 进行设置。

模块原理

数据队列模块需要配合发送数据模块和接收数据模块使用。这三个模块之间的配合逻辑说明如下：

1. 流程 A 中包含图像源模块，并且已经在运行中。
2. 流程 A 中的 *发送数据* 模块会将采集到的图像数据缓存到与之关联的数据队列模块。
3. 当数据队列模块中所有设置的队列已缓存对应数据时，软件将根据触发流程页签下的流程添加顺序寻找并执行第一个空闲的流程，对图像数据进行进一步处理。
4. 空闲流程 B 中的 *接收数据* 模块将从数据队列模块中读取对应数据，对图像数据进行后续处理。

说明

如需了解更多以上模块之间的协作逻辑，请参考 *使用示例* 中的介绍，快速上手数据队列模块的使用。

队列设置

在*队列设置*页签下，您可以创建多个队列用于缓存不同数据类型的数据，并且指定普通队列或图像队列中可包含的数据行数。



图 8-21 队列设置

具体设置步骤说明如下：

1. 在*队列设置*页签，点击**队列列数**下方的 ⊞，添加一个数据队列。

   说明

   数据队列模块最多支持添加 16 个普通数据队列和 4 个图像队列。

2. 在**数据类型**列，单击输入框右下角的三角按钮，并下拉选择一个数据类型。
   数据队列模块中支持添加的数据类型包括 int、float、string 和 IMAGE。

   说明

   切换已存在数据的数据类型时，若该队列中已存在缓存数据，缓存的数据将会被清空。

3. 在**列名称**列，填写自定义的数据队列名称。

说明

自定义**列名称**时，需确保每个数据队列的名称互不重复。

4. 在**队列行数**区域，设置数据队列中最多可缓存多少个数据元素。
   各类型数据队列可缓存的数据个数有所差别：
   - int、float、string 类型数据队列：有效的取值范围是[1,256]。您可以根据实际需要在**队列缓存行数**字段填入自定义的数值。
   - IMAGE 类型数据队列：有效的取值范围是[1,10]。您可以根据实际需要在**图像队列行数**字段填入自定义的数值。

说明

当队列缓存的数据个数达到上限时，遵循先进先出的原则，删除最先入队的数据，用于存储新入队的数据。

触发流程

在***触发流程***页签，您可以设置指定流程得到触发，并从数据队列中获取对应数据。设置完成后，当所有数据队列已缓存指定数据时，软件会按照***触发流程***页签下的流程添加顺序寻找并执行第一个空闲流程。得到触发的流程将从数据队列中获取指定数据并做进一步处理。

说明

一旦某一流程执行成功，则终止本次触发，即一次触发最多只有一个流程执行成功。



图 8-22 触发流程

具体设置步骤说明如下：

1. 在*数据队列*模块配置页面，点击*触发流程*页签。
2. 在**触发流程列表**下方，点击 ⊞，订阅需触发的流程。

说明

触发流程列表最多支持添加 16 个流程。

3. 在*触发流程*页签，根据实际情况开启或关闭**异步触发**按钮。

说明

当您新增数据队列模块时，软件安装目录下将自动创建一个 DataQueueConfig.xml 配置文件。如果您未设置异步触发方式，当所有流程均触发失败时，软件默认会删除当前队列缓存的数据。如果您需要保留数据，可将 DataQueueConfig.xml 配置文件下的 value 字段由数值 0 改为 1。该配置文件的相对路径：.\Applications\Module(sp)\x64\Global\DataQueueModule

流程的触发方式分为异步触发和同步触发。这两种触发方式的优缺点说明如下：

表 8-8 同步/异步触发优缺点

触发类型	说明	优点	缺点
异步触发	新建数据队列模块时，会创建该模块的子线程。子线程一直处于运行中状态，每隔一定时间判断是否满足触发条件。若满足，则触发流程执行；若不满足，则不触发流程执行。	触发失败时，不需要等待数据再次入队。子线程会每隔一段时间判断该流程是否符合执行条件。	一旦触发失败，需间隔一段时间才能判断当前流程是否符合触发条件，并触发执行满足条件的流程。这可能导致流程执行过程中存在一定延时。
同步触发	当有数据进入数据队列时，若满足触发条件，则触发流程执行，无论成功与否，本次触发结束；若不满足，则不触发流程执行。	可及时触发流程执行	一旦触发失败只能等待下次数据入队。

8.5.1 使用示例

本节提供具体使用示例，介绍如何使用**数据队列**模块实现不同流程之间的数据同步。

示例需求：在某一直线查找流程中，***图像源***模块输出图像的速率是该流程处理速度的 2 倍，导致部分图像来不及被处理，无法实现对所有相机图像进行直线查找的需求。

解决方案：为了优化该流程的耗时问题，您可以新建一个流程专门用于获取相机实时采集的图像，并通过数据队列模块缓存已采集的图像数据。当指定数据存储到数据队列后，软件将根据触发流程的添加顺序触发并执行第一个空闲流程，若该流程的处理速度跟不上图像的输出速率，软件将再次按照触发流程的添加顺序触发并运行一个空闲流程，以此实现对所有相机图像的直线查找。根据以上需求，可提炼方案的实现细节如下：

- 新建 3 个流程和 1 个数据队列模块。
- 在数据队列模块，设置队列可缓存的数据类型和需触发的流程，从而实现在不同流程之间传递图像数据且触发对应流程执行。
- 流程 1 中需包含图像源模块（用于获取图像）和发送数据模块（用于将流程 1 中的数据发送给数据队列）。
- 流程 2 和流程 3 中的配置需完全一致，需要包含接收数据模块（用于接收数据队列的数据）和直线查找模块（用于查找图像中的直线）。

说明

完成设置后，当流程 2 检测图像耗时较长时，数据队列模块将触发流程 3 执行，继续检测相机后续采集的图像。

不同流程之间图像处理逻辑如下图所示：

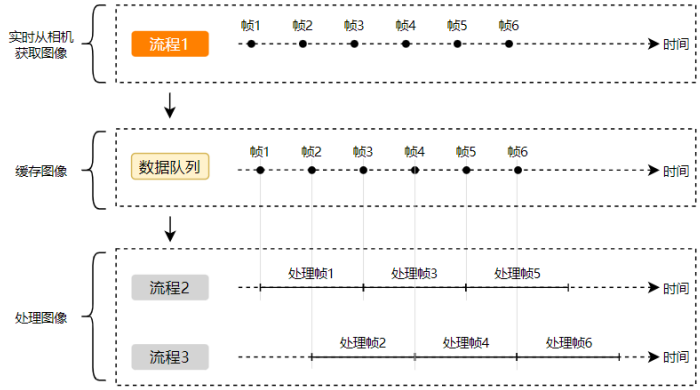


图 8-23 多流程间的图像处理逻辑

操作步骤

1. 新建流程 1、流程 2 和流程 3，涉及的模块和顺序如下图所示。

   说明

   流程 2 和流程 3 中的模块和顺序完全一致。

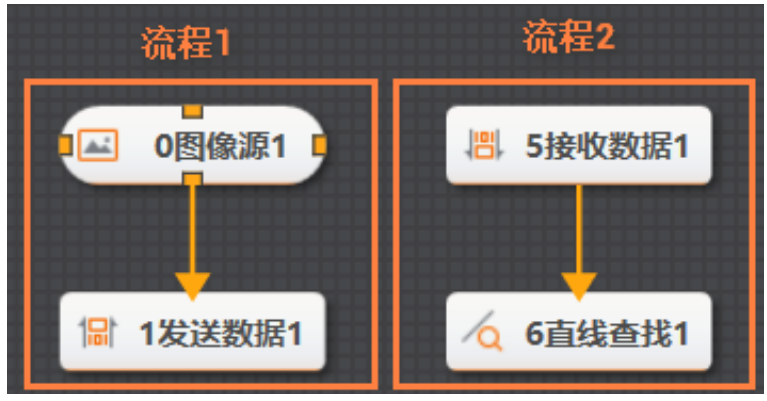


图 8-24 流程示意

2. 新建*数据队列*模块，并完成队列设置和触发流程设置。
   - 队列设置：添加一个数据队列，**数据类型**选择 IMAGE，**列名称**可自定义，如 img。
   - 触发流程：添加两个待触发流程，并订阅流程 2 和流程 3。您还可以选择是否开启**异步触发**。关于触发方式的详细介绍，请参见 *表 8-8*。



图 8-25 数据队列设置

3. 完成流程 1 中各模块的配置，并单击**确定**保存修改。
   - *图像源*：根据实际需求设置图像来源。
   - *发送数据*：在**输出配置**区域，选择输出至**数据队列**，并订阅步骤 2 新建的数据队列；在**输出数据**区域的**选择队列**字段，单击输入框右下角的三角按钮，下拉选择步骤 2 中新建的图像队列，并在**选择数据**字段订阅*图像源*模块的图像。



图 8-26 发送数据

4. 完成流程 2 和流程 3 中各模块的配置。

说明

流程 2 和流程 3 中的模块配置完全一致。

- *接收数据*：在**输入配置**区域数据源选项，选择**数据队列**，并订阅步骤 2 新建的数据队列；在**输入数据**区域的**数据**列，单击输入框右下角的三角按钮，并选择步骤 2 中新建的图像队列。



图 8-27 接收数据

- *直线查找*：在**图像输入**的**输入源**参数，订阅**接收数据**模块中的输入数据。其他参数根据实际需求设置。



图 8-28 直线查找

后续操作

完成方案搭建后，运行流程 1。当流程 1 获取到图像数据时，流程 2 会同时运行，并输出基于流程 1 采集到的图像的直线查找结果。若流程 2 来不及处理流程 1 的图像数据时，流程 3 也会同时运行，输出基于流程 1 采集到的图像的直线查找结果。

第 9 章 方案搭建-运行界面设计

运行界面用于直观操控和监控方案运行，业务现场的终端用户通过运行界面操控方案运行（甚至流水线作业）。本文为您介绍运行界面的功能模块说明，以及如何自定义配置运行界面。

9.1 设计窗口概览

运行界面设计窗口用于自定义配置运行界面，本文为您介绍设计窗口各区域功能及说明。

单击主界面工具条中的 可进入运行界面设计窗口。

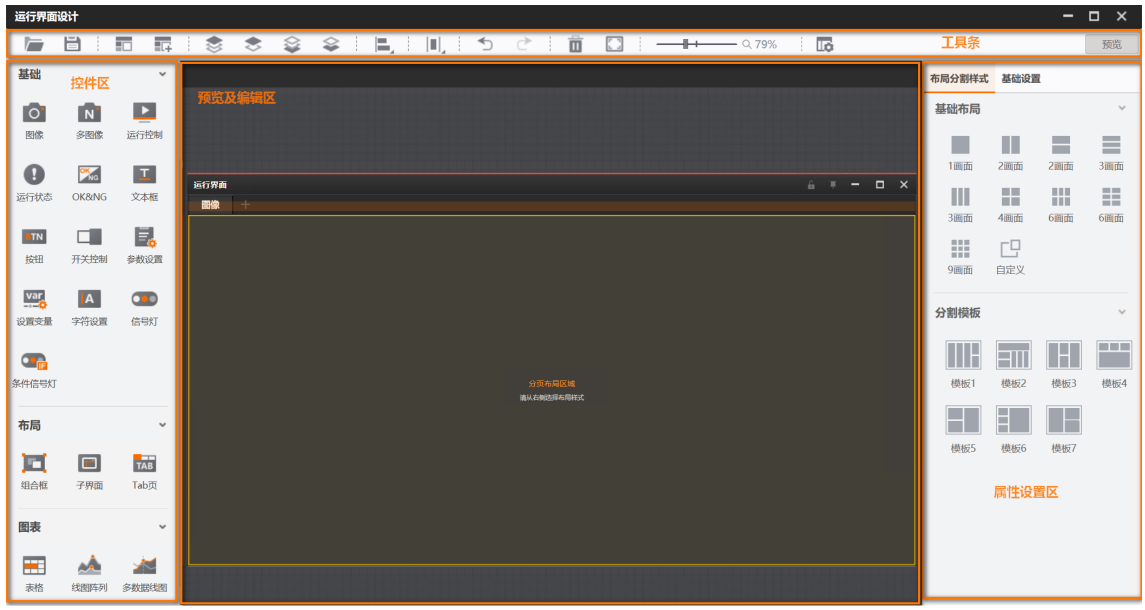


图 9-1 运行界面设计

该界面各区域的说明参见下表：

表 9-1 运行界面设计窗口介绍

名称	描述
工具条	提供运行界面保存、模板设置、图层设置、删除、固定栏设置、导出等功能，详情参见 *工具条介绍* 。
控件区	呈现可在运行界面上使用的控件，详情参见 *控件配置参考* 。

名称	描述
预览及编辑区	呈现当前设计中的运行界面。可绘制新的控件，或编辑当前已添加的控件，具体如何操作参见 *控件通用操作* 。
属性设置区	展示预览及编辑区选中画布、固定栏及控件的属性（例如布局、数据源、颜色、字体等），并进行设置。

工具条介绍

工具条可设置如下功能：

- ：加载已保存的运行界面。
- ：保存当前设计的运行界面。若为首次保存，需选择保存路径。
- ：可选择已保存的运行界面模板进行应用或删除。
- ：可将当前设计的运行界面作为模板保存。保存时，需填写**名称**、**创建人**、**推荐使用场景**和**描述**，并点击*确定*。
- ：可对预览及编辑区中选中的控件设置图层位置，分别对应顶层、上一层、底层、下一层。
- ：预览及编辑区域选中多个控件时，以选中控件组成的区域为基准对选中控件的对齐方式进行调整。可选左对齐、水平居中、右对齐、顶部对齐、垂直居中、底部对齐。
- ：预览及编辑区域选中多个控件时，以选中控件组成的区域为基准对选中控件的分布方式进行调整。可选水平分布和垂直分布。

  说明

  水平/垂直分布需选择 3 个以上的控件，才起作用。对应的效果分别为水平/垂直方向控件等间隔分布。其余选项选择 2 个以上控件即可。

- ：撤销最近一次操作。
- ：取消撤销操作，恢复至撤销前的状态。
- ：删除预览及编辑区域选中的控件。
- :可将*预览及编辑区*中的运行界面窗口相对于*预览及编辑区*按照合适比例进行大小调整。

  说明

  打开运行界面设计窗口时，默认调用一次该功能。后续可在调整窗体宽/高后，使用该功能。

- 84% ：可通过拖动滑块调整预览及编辑区域中运行界面的大小。
- ：可设置是否在运行界面增加顶部、底部、左侧和右侧固定栏。点击勾选即可，可同时选择多个位置的固定栏。
- **预览**：可预览当前设计的运行界面。
- **导出**：可将 VM 当前编辑的方案以及设计的运行界面以工程包的形式导出。详情参见 *导出运行界面并运行* 。

说明

- 其中删除、撤销、重做、图层位置功能也支持在选中控件后通过鼠标右键实现；
- 复制粘贴功能仅支持在选中控件后通过 Ctrl+C 和 Ctrl+V 实现。
- 、 和 功能仅型号中带 6210 或 7210 的加密狗支持使用。

9.2 设计运行界面

您可根据实际业务场景及需求设计运行界面，帮助终端用户在运行界面更方便、直观操作和监控方案，从而提高工作效率。

9.2.1 通用设计思路

运行界面默认为空白，需自行通过*运行界面设计*进行设计。

整体设计思路为：

1. 设置是否使用模板：在*运行界面设计*中单击 ，查看是否有可直接复用的运行界面模板。若有，直接选用即可，也可基于选用的模板自行调整；若无，则进行下一步。
2. 设置固定栏布局：单击 设置是否需要上/下/左/右固定栏。
3. 设置画布布局：选中画布，通过属性设置区设置画布的整体布局。
4. 添加并设置控件：将需要使用的控件拖拽到固定栏或画布中，并完成数据源等相关设置。

   说明

   设置控件时，可通过上方相关按钮对图层位置、对齐方式等进行设置。

9.2.2 控件通用操作

设计运行界面的核心之一是添加各个控件，并完成控件的参数设置。

在左侧控件区通过鼠标选中某个控件后，按住鼠标左键不动并移动到中间的预览及编辑区，在计划摆放控件的区域松开鼠标即可完成控件添加。添加控件后，可在预览及编辑区调整控件的位置和大小，还可通过右侧的属性设置区完成控件的属性设置。具体操作如下视频所示。

9.2.3 设计示例

本节以一实际方案为您介绍设计运行界面全过程。

前提条件

已完成方案搭建。

说明

本章节以流程如下图所示的方案作为示例方案搭建运行界面。

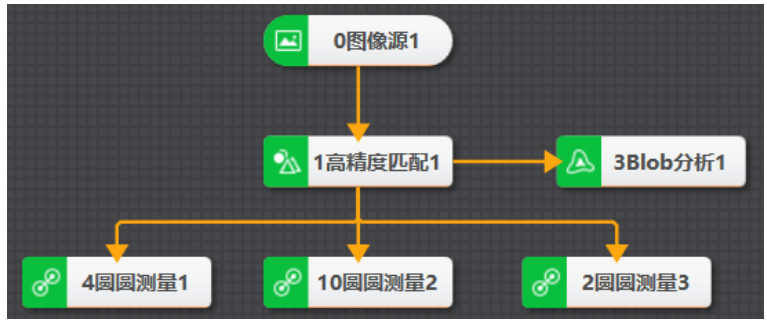


图 9-2 示例方案

操作步骤

1. 单击 进入*运行界面设计*。
2. 单击 勾选*顶部固定栏*和*底部固定栏*后*确定*。
3. 设置顶部固定栏。
   1) 选中顶部固定栏，设置**分割模板**，默认为“模板 1”。此处我们使用模板 1，不做调整。
   2) 选中***文本框***控件拖拽到顶部固定栏的第一个区域中，并完成相关参数设置。

      以下为相关参数设置情况，其余参数均为默认设置。

      - **字符信息**：AAA 铸件圆孔测距；
      - **自动填充**：启用；
      - **字体**：大小为 48，字形选择加粗。

   3) 选中*运行控制*控件拖拽到顶部固定栏的第二个区域中，并完成相关参数设置。

      说明

      该控件的参数均使用默认设置。

   4) 选中 *OK&NG* 控件拖拽到顶部固定栏的第二个区域中，放置在*运行控制*的右侧，并完成相关参数设置。

      说明

      数据源订阅“圆圆测量 2 的模块状态“，其余参数均使用默认值。

   5) 选中*按钮*控件拖拽到顶部固定栏的第二个区域中，放置在 *OK&NG* 的右侧，并完成相关参数设置。

      以下为相关参数设置情况，其余参数均为默认设置。

      - **数据源**：点击*配置*按钮进行设置，**触发命令类型**选择方案，**方法**选择保存；
      - **显示文本**：保存方案。

   以上子步骤对应的操作可参考以下视频。

4. 设置底部固定栏：选中底部固定栏，在右侧**选择列表**勾选需实时查看连接状态的设备，例如全局相机、TCP 客户端、VB2230 等。
5. 设置画布布局。
   1) 选中画布，并设置**布局分割样式**。此处选择分割模板的“模板 5”。
   2) 通过网格线微调每个区域的大小。
   3) 选中画布左上角的区域，设置**布局分割样式**。此处选择基础布局中的“2 画面”。
   4) 选中画布左下角的区域，设置**布局分割样式**。先选择基础布局中的“2 画面”。
   5) 依次将上一步分割的两个区域再分割分割成 2 个“2 画面”。
   6) 选中上一步分割后 4 画面的中间两个画面，并设置为“1 画面”。
   7) 微调左下角区域的网格线，使 3 个区域等分。

   以上子步骤对应的操作可参考以下视频。
6. 设置画布左侧区域，用于显示图像：选中*图像*控件分别拖拽到画布对应的 5 个区域中，完成相关参数设置。

   以下为相关参数设置情况，其余参数均为默认设置。

   - **数据源**：均选择“流程 1”的“图像源 1_输出图像”。
   - **图形配置**：左上角的图像控件，勾选 3 个*圆圆测量*模块中的所有图形；上方中间的图像控件，勾选 *Blob 分析*模块中的所有图形；左下方 3 个区域，分别依次勾选 3 个*圆圆测量*模块的所有图形。
   - **显示工具栏**：关闭。

   以上子步骤对应的操作可参考以下视频。
7. 设置画布右侧区域，用于查看实时数据：选中*表格*控件拖拽到画布右侧区域中，完成相关参数设置。

   以下为相关参数设置情况，其余参数均为默认设置。

   - **行数**：22；
   - **列数**：5；
   - **行抬头**：结果显示；
   - **数据源**：单击*配置*完成对每一列数据进行设置。以下为相关参数设置情况，其余参数均为默认设置。
     - **列名**：从上到下依次为 CenterX、CenterY、Runtime、Angle 和 Dist。
     - **数据类型**：均设置为 float；
     - **数据源**：从上到下依次订阅“流程 1 的圆圆测量 1”中的圆 1 圆心 X、圆 1 圆心 Y、耗时（ms）、角度、距离。

   以上子步骤对应的操作可参考以下视频。

后续处理

通过*运行模式*打开软件后，单击单次执行/连续执行，运行情况如下图所示。



图 9-3 运行界面

9.3 界面元素参考

运行界面由画布、固定栏和控件三类界面元素搭建，其中控件用于与客户端交互或呈现信息，画布和固定栏用于组织和放置控件或显示设备状态。

9.3.1 画布及固定栏配置参考

设计运行界面时，您可通过预览及编辑区和属性配置区配置画布（即运行界面的背景）和固定栏。

本节包含内容如下：

- *画布*
- *顶部/左侧/右侧固定栏*
- *底部固定栏*

画布

运行界面的画布可通过顶部固定栏下方新增或删除。点击最右侧画布的 可新增画布；选中已有画布，通过属性设置区的**内容**参数可修改画布名称，也可右键单击删除画布。
画布默认为单画面。选中画布后，可通过属性设置区进行基础配置和布局分割样式的配置。

- 布局分割样式：可根据需求从基础布局和分割模板中选择。

说明

- 该功能仅型号中带 6210 或 7120 的加密狗支持。
- 通过画布中的网格线，可调整各个区域的大小。
- 画布进行布局分割后，选中某一个或多个区域时支持继续进行布局分割。

基础布局

可选择画布的基础布局方式。其中*自定义*可自行设置行数和列数，范围均为 1 ~ 10。

分割模板

可从已有的模板中选择。

- 基础设置：

显示网格

可设置画面设置为多画面时，画面之间的框线是否显示。

网格颜色

背景

可分别设置画面之间的框线以及整个画布背景的颜色。

顶部/左侧/右侧固定栏

顶部/左侧/右侧固定栏和运行界面的中间区域类似，可放置各种控件。
顶部/左侧/右侧固定栏相关操作和 *画布* 类似，此处不再赘述。

说明

左侧/右侧固定栏还可通过基础设置进行**宽度**参数的设置。

底部固定栏

可实时查看相机、通信设备、控制器等设备的连接状态，最多支持同时查看 20 个设备的状态。
参数分为布局和选择列表两大类。

- 布局：

边距

可调整底部固定栏的位置。点击某个数据右侧的 □ 可使所有数据修改为该数据。

- **选择列表**:可选择需实时查看连接状态的设备。可选择的设备为方案中在 *相机管理* 和 *通信管理* 中添加的设备。

9.3.2 控件配置参考

运行界面由多个控件组合而成，不同类型的控件功能以及属性设置有所差别。

运行界面具体支持的控件如下：

基础 ： • *图像* • *多图像* • *运行控制* • *运行状态*	*布局* ： • *组合框* • *子界面* • *Tab 页*	*图表* ： • *表格* • *线图阵列* • *多数据线图* • *生产统计*	*其他* ： • *图片* • *世界时钟* • *界面日志*

• *OK/NG* • *文本框* • *按钮* • *开关控制* • *参数设置* • *设置变量* • *字符设置* • *信号灯* • *条件信号灯*		• *饼图* • *参数组合* • *文本框组合*	

基础

图像

图像控件可订阅流程中模块输出的图像，并做显示。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  可订阅显示图像的来源，点击 ... 后选择模块输出的图像即可。

  图形配置

  可勾选需在数据源基础上叠加显示的图形。

  说明

  设置该参数前，需先完成**数据源**的设置，可从加载的流程中选择多个模块的图形。否则，显示为空。

  叠加图层 1/2/3

  可基于数据源订阅的图像数据，叠加显示其他图像。操作方法和**数据源**相同，但只能从**数据源**订阅的流程中选择模块输出的图像。

  说明

  设置该参数前，需先完成**数据源**的设置。

  保留上一次结果

  可设置当前流程执行结束时若订阅结果无图像数据输出，是否保留上一次渲染结果显示。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

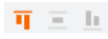
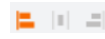
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

显示状态栏

可设置是否显示右下角的状态信息。信息主要为图像分辨率、鼠标当前所在位置的坐标信息和 RGB 分量或灰度值。

显示工具栏

可设置是否开放左上角的工具。工具对应功能分别为放大、缩小、1:1、全屏和锁定。

说明

1:1 功能主要用于在图像放大、缩小或移动后，快速恢复为在图像窗口自适应居中显示。

显示辅助线

可设置运行界面预览或运行时是否显示十字辅助线。

运行界面不运行方案或流程时，还可选中十字辅助线中心点，调整位置。

说明

编辑运行界面时无法查看效果，仅在运行界面预览或运行时显示。

历史图像列表

开启该功能后，触发流程执行后图像控件下方增加历史列表区域，可查看最近几次执行的渲染结果（包括**图像数据**、**图形数据**，但不包含**叠加图层**）。

说明

图像输出过快时，历史图像可能存在缺少中间某次执行图像的情况。

历史图像数量

可设置最多可显示的历史图像数量。

多图像

多图像控件功能与图像控件相同，但可订阅并显示多个图像。
参数分为数据源和布局两大类。

- 数据源：

  画面数量

  可设置多图像窗口显示的画面数量和画面的位置关系。

  数据源

  点击*配置*后对各个画面**画面名称**、**图像源**、**图形源**、**叠加图层 1/2/3**、**显示辅助线**、**保留上一次结果**等进行设置。操作方式和 *图像* 控件大同小异，此处不再赘述。

  说明

  推荐先完成**画面数量**设置。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

  说明

  若开启**自动填充**，**自适应**将无法开启。

  尺寸

  可调整控件的大小，也可通过预览及编辑区调整。

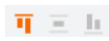
  边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

运行控制

运行控制控件可控制全部流程或单一流程的运行方式。
参数分为数据源、布局、外观三大类。

- 数据源：

数据源

可下拉选择当前加载方案中的流程或全流程。

> **说明**
> 全流程代表整个方案。

控制类型

可选单次执行、连续执行和同时显示。

单次执行

只生成单次执行的按钮。

连续执行

只生成连续执行的按钮。

同时显示

同时生成以上两个按钮。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

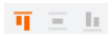
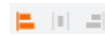
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

按钮颜色
边框颜色
图标颜色

可分别设置按钮、边框和图标的颜色。

运行状态

运行状态控件可显示全部流程或单一流程的运行状态。
参数分为数据源、布局、外观三大类。

- 数据源：

数据源

可下拉选择当前加载方案中的流程或全流程。

说明

- 若绑定单个流程，该流程开始运行则该控件显示“运行中”(忙碌状态)，反之显示“停止运行”（空闲状态）。
- 若绑定全流程，任一流程开始运行则该控件显示“运行中”(忙碌状态)，反之显示“停止运行”（空闲状态）。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

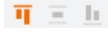
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

空闲-背景色
空闲-前景色
忙碌-背景色
忙碌-前景色

可分别设置图标空闲、忙碌状态各部分的颜色。

圆角半径

可设置图标四个角的圆角半径，控制图标形状。

OK/NG

OK/NG 控件可根据订阅的数据判断显示为 OK 或 NG，0 代表 NG，1 代表 OK。
参数分为数据源、布局、外观、OK、NG 五大类。

- 数据源：

数据源

可订阅方案中各模块中 int 型模块结果。推荐订阅各模块的模块状态或条件检测模块的结果(INT)。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

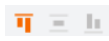
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

  圆角半径

  可设置图标四个角的圆角半径，控制图标形状。

- OK/NG：

  OK-内容
  NG-内容

  可分别设置图标 OK、NG 时显示的文字内容。

  OK-字体
  NG-字体

  可分别设置图标 OK、NG 时使用字体的大小及字形。

  OK-边框厚度
  NG-边框厚度

  可分别设置图标 OK、NG 时控件的边框厚度。

  OK-背景色
  NG-背景色
  OK-字体色
  NG-字体色
  OK-边框颜色
  NG-边框颜色

  可分别设置图标 OK、NG 时控件的背景、字体以及边框颜色。

文本框

文本框控件可在运行界面显示定义的文本信息。可自定义文本内容，可以订阅方案中的不同类型的数据显示。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据类型

  可设置文本框中显示数据的类型，可选 string（字符串）、int（整数型）和 float（浮点型）。

  数据源

  可订阅方案中各模块的模块结果、全局/局部变量或方案路径在文本框中显示。选择不同的数据类型，可订阅的数据有所差别。

说明

选择不同的**数据类型**，可订阅的数据有所差别。其中方案路径仅在**数据类型**为 string 时可以订阅。

数据绑定

绑定**数据源**后自动加载，显示为“订阅数据的描述 : <订阅的数据>”。其中“订阅数据的描述：”可自定义修改；“<订阅的数据>”为格式化语法内容，选择不同的数据，语法内容有所差别。

- **数据类型**选择 string 时，格式化语法内容对应为“%s”。
- **数据类型**选择 int 时，格式化语法默认为“%0d”。格式化语法为该默认内容时，控件输出数据源的 int 型数据。若有格式化需求，您可通过修改“%”后的阿拉伯数字，设置控件输出的 int 数据的最大位数。例如，若数据源输出的原数据为 600，格式化语法设置为“%5d”，那么方案运行时控件将输出 00600。
- **数据类型**选择 float 时，格式化语法内容对应为“%1.3f”。此时小数点后保留 3 位有效数字，可自定义修改。例如%1.6f 对应为小数点后保留 6 位有效数字。

说明

设置该参数时，需了解格式化语法相关要求。若不了解相关内容且需做修改，请联系技术同事。

字符信息

订阅数据源时，此处显示订阅后的信息；不订阅数据源时，可通过该参数设置文本框显示的信息。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

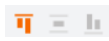
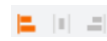
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

字体

可设置使用字体的大小及字形。

字体颜色

可设置使用字体的颜色，更改颜色实时生效。

按钮

若运行界面需要对软件部分功能进行设置或重置等操作，可使用按钮控件实现。
参数分为数据源、布局、外观三大类。

- 数据源：

数据源

点击*配置*进入配置窗口，单击  可添加触发命令。

每个触发命令通过**触发命令类型**、**方法**、**输入参数**进行设置。不同触发命令需设置的参数有所差别。

说明

一个按钮可配置多个触发命令，使用时根据设置的触发命令顺序每单击一次就执行一个命令。

触发命令类型分为界面链接、参数配置、方案、执行和重置这五种。

界面链接

单击按钮时，可快速打开选择的界面。

可选全局相机、全局通信、全局变量、界面日志和综合配置界面。

参数配置

单击按钮时，可快速打开选择流程或各模块的参数配置界面。

方案

单击按钮时，可保存或另存为方案。

执行

可选执行一次、开启连续执行、关闭连续执行、模块动作。

执行一次

可分别对选择的所有流程（即 AllProcess）、单个流程或单个模块运行一次。

开启连续运行
关闭连续运行

可分别对选择的所有流程（即 AllProcess）或单个流程开启/关闭连续运行。

模块动作

可对选择的模块执行相应动作。例如可重置***变量计算***模块的变量、清空 ***N 点标定*** 模块的标定点。

重置

单击按钮时，可重置选择的对象。

重置的对象可选变量计算或数据队列。

变量计算

可对所有流程中的变量计算模块（即 AllCalculators）或流程中的模块变量计算模块中的变量进行重置。

数据队列

可对所有数据队列（即 AllDataQueueModules）进行重置。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

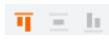
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

显示文本

可设置该按钮显示的文本信息。

字体

可设置使用字体的大小及字形。

字体颜色
背景
边框颜色
点击颜色

可分别设置按钮中字体、背景、边框以及点击时的颜色。

图像

点击 … 可选择本地图像在按钮上叠加显示。

拉伸模式

可设置**图像**在控件的显示方式，可选 None、Fill、Uniform 和 UniformToFill。

圆角半径

可设置图标四个角的圆角半径，控制图标形状。

开关控制

开关控制控件可订阅模块中 bool 型参数，从而通过运行界面运行方案时控制该参数是否启用。

参数分为数据源和布局两大类。

- 数据源：

  数据源

  可订阅各模块中的 bool 型参数。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

  说明

  若开启**自动填充**，**自适应**将无法开启。

  尺寸

  可调整控件的大小，也可通过预览及编辑区调整。

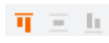
  边距

  可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

  该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

  假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

  对齐方式

   分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

参数设置

参数设置控件可订阅模块中 int 或 float 型参数，从而通过运行界面运行方案时也可设置该参数。

参数分为数据源、布局、外观三大类。

- 数据源：

参数类型

可选择订阅数据的类型，可选 int 或 float。

数据源

可订阅方案中 int 或 float 型参数。订阅的参数类型与**参数类型**有关。

参数名称

可自定义该参数的名称。

小数位数

参数类型选择 float 时，该参数决定显示参数的小数位数。

- 布局：

宽度

可调整参数控件的宽度，也可通过预览及编辑区实时调整。

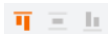
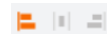
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

背景
字体颜色

可分别设置参数控件背景以及字体的颜色。

字体

可设置使用字体的大小及字形。

设置变量

设置变量控件可订阅变量计算模块中添加的变量，订阅表达式时会标红显示。运行界面可实时显示该数值，并支持修改。

参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  可订阅变量计算模块中添加的变量。

- 布局：

  宽度

  可调整参数控件的宽度，也可通过预览及编辑区实时调整。

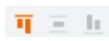
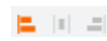
  边距

  可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

  该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

  假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

  对齐方式

   分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

  背景
  字体颜色

  可分别设置参数控件背景以及字体的颜色。

  字体

  可设置使用字体的大小及字形。

字符设置

字符设置控件可将订阅的字符串信息在控件中显示。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  可订阅全局/局部变量中 string 型的变量及各模块中 string 型的参数。

- 布局：

宽度

可调整参数控件的宽度，也可通过预览及编辑区实时调整。

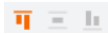
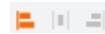
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

背景

字体颜色

可分别设置参数控件背景以及字体的颜色。

字体

可设置使用字体的大小及字形。

信号灯

信号灯控件可根据订阅的模块状态判断控件显示为绿色或红色，0 显示为红色，1 显示为绿色。

参数分为数据源、布局、外观三大类。

- 数据源：

数据源

可订阅各模块的模块状态。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

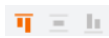
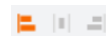
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

信号灯内容

可自定义信号灯控件显示的内容。

字体

可设置使用字体的大小及字形。

字体颜色

可设置使用字体的颜色，更改颜色实时生效。

条件信号灯

信号灯控件可根据订阅的数据以及设置的条件，判断控件显示何种颜色。

说明

当订阅的参数满足多个条件时，依次按照＞值、＜值、默认值顺序优先选择符合条件的第一个颜色显示。

参数分为数据源、布局、外观三大类。

- 数据源：

  参数类型

  可选择订阅的数据类型，可选 int 或 float。

  数据源

  可订阅各模块或全局变量的 int 或 float 型参数。订阅的参数类型，与**参数类型**有关。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

  说明

  若开启**自动填充**，**自适应**将无法开启。

  尺寸

  可调整控件的大小，也可通过预览及编辑区调整。

  边距

  可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

  该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

  假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

  对齐方式

   分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

  信号灯内容

可自定义信号灯控件显示的内容。

字体

可设置使用字体的大小及字形。

字体颜色

可设置使用字体的颜色，更改颜色实时生效。

圆角半径

可设置图标四个角的圆角半径，控制图标形状。

默认值背景颜色
默认值边框颜色
＞值背景颜色
＞值边框颜色
＜值背景颜色
＜值边框颜色

可分别设置订阅的数值为默认值、＞值、＜值时，条件信号灯的边框/背景颜色。

＞值阈值
＜值阈值

需分别大于和小于对应的判断数值。

说明

默认值无需设置，对应为同时不符合＞值以及＜值要求的数值。

布局

组合框

当运行界面有较多控件时，可使用组合框控件将多个控件组合到一起。
将控件移入组合框的操作方法：鼠标选中某个控件并拖动到组合框控件所在区域即可。

说明

移出方法与移入类似，此处不再赘述。

参数分为布局和外观两大类。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

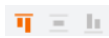
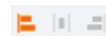
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置组合框控件左上角的标题名称。

字体

可设置使用字体的大小及字形。

标题背景
字体颜色
背景
边框颜色

可分别设置组合框控件标题背景、字体、背景及边框的颜色。

子界面

子界面控件可在运行界面中新增一个运行界面作为子界面呈现。点击子界面控件后弹出新的运行界面窗口。子界面中各个控件的使用与运行界面完全一致，此处不再赘述。

说明

运行界面最多支持嵌套 3 层子界面。

参数分为数据源、布局、外观三大类。

- 数据源：

  编辑子界面

  单击*配置*后，预览及编辑区域切换为子界面窗口。此时可通过*布局分割样式*和*基础设置*调整控件的布局、分割模板以及基础设置。完成配置后，关闭运行界面设计即可退出子界面编辑窗口。

  密码

  单击*配置*可设置打开子界面是否需要密码，以及具体的密码设置。默认无需密码。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

  说明

  若开启**自动填充**，**自适应**将无法开启。

  尺寸

  可调整控件的大小，也可通过预览及编辑区调整。

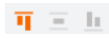
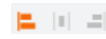
  边距

  可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

  该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

  假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

  对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

  字符信息

  可自定义子界面控件的名称。

  字体

  可设置使用字体的大小及字形。

  字体颜色
  背景
  边框颜色

  可分别设置子界面控件字体、背景及边框的颜色。

  圆角半径

  可设置图标四个角的圆角半径，控制图标形状。

  显示图标

  可设置子界面控件的字符信息前面是否显示  。

Tab 页

Tab 页控件可理解为多个组合框控件的组合，一个组合框对应一个 Tab 页。

说明

控件移入或移出 Tab 页的方法与 *组合框* 相同，此处不再赘述。

选中整个 Tab 页及单个 Tab 页名称时，均可设置相关参数。

- 选中整个 Tab 页时，参数分为数据源、布局、外观三大类。
  - 数据源：

    Tab 页数量

    可设置 Tab 页数量。
  - 布局：

    自动填充

    在运行界面中，可将控件自动填充满所在网格。

    自适应

    若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

> **说明**
> 若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

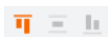
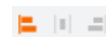
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

背景
边框颜色
字体颜色

可分别设置 Tab 页背景、边框及字体的颜色。

选中背景

设置选中单个页签时该页签标题的颜色。

显示标题栏

可设置是否显示 Tab 页的标题栏。

- 选中单个 Tab 页名称时，参数分为外观和布局两大类。
  - 外观：

Tab 页名称

可自定义选中 Tab 页的名称。

  - 布局：

尺寸

Tab 页名称的尺寸大小可通过预览及编辑区调整，此处可实时查看尺寸信息。

边距

可调整 Tab 页名称的位置（无法通过预览及编辑区调整）。点击某个边距数据右侧的 □ 可使所有边距数据修改为该数据。

图表

表格

表格控件可订阅需要呈现的数据，使订阅的数据按列滚动呈现最新数据。数据按照执行次数，倒叙呈现订阅历史结果。一般在需要同时查看多个流程的运行结果时使用。
参数分为数据源、布局、外观三大类。

- 数据源：

数据源

可设置每一列的**列名**、**列宽**、**数据类型**、**数据源**、和**颜色编辑**。

说明

推荐先完成**列数**的设置。

数据类型

可选 string、int 或 float。

数据源

可订阅各模块或全局/局部变量中符合**数据类型**要求的数据。

颜色编辑

可设置订阅的数据源满足**条件**和**比较值**要求时，数据显示对应的**颜色**。

说明

同一个数据满足多个**颜色编辑**设置的条件时，根据设置的条件的顺序执行第一个。

条件

包括**= =**（等于）、**! =**（不等于）、 **>**（大于）、**<**（小于）、 **> =**（大于等于）、**< =**（小于等于）。

说明

其中 string 型数据仅支持**= =**（等于）和**!=**（不等于）。

行数
列数

可分别设置表格的列数和行数。

行抬头

可设置表格左上角第一个单元格显示的内容。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

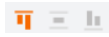
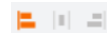
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置标题名称。

显示标题栏

可设置标题栏是否显示。

字体

可设置使用字体的大小及字形。

标题文本行
标题行
字体颜色
背景
列标题背景
行背景
交替行背景
边框颜色

可分别设置表格对应的颜色。

线图阵列

线图阵列控件可将订阅的数据以线图的方式呈现执行次数和当前执行次数对应值之间的线性关系。每个数据源单独一个线图呈现。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  点击*配置*进入配置窗口，单击 ⊞ 可添加数据。

  每条数据由以下参数组成。

  数据类型

  可选 int 或 float。

  数据源

  可从方案中订阅符合**数据类型**要求的数据。

  名称

  自定义该数据的名称。

  数组数据索引

  若订阅的**数据源**为数组数据，则通过该参数控制选择数组中的第几位数据使用。默认为 0，代表第 1 位数据。当**数据源**不是数组数据时，该参数应设置为 0。

  默认颜色

  控制该控件中该参数以设置的颜色做呈现。

  上下限设置

  单击 ✎ 进入配置窗口进行设置。需开启**显示上下界限**参数方可设置。

  完成设置后，线图中会以设置的颜色显示上下限值，方便查看数据是否在上下限范围内。

上限值
下限值

可分别设置上限/下限的数值以及对应的线条颜色。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

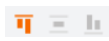
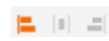
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置标题名称。

显示标题栏

可设置标题栏是否显示。

标题文本行
标题行

线图背景色
坐标轴颜色

可分别设置以上对应板块的颜色。

最新执行数据条数

可设置获取最近执行多少次的结果。

多数据线图

多数据线图可将订阅的数据以线图的方式呈现执行次数和当前执行次数对应值之间的线性关系。与*线图阵列*控件的差别为该控件在一个线图上同时呈现多个数据源。
多数据线图的参数设置和 *线图阵列* 一致，此处不再赘述。

生产统计

生产统计主要用于实时查看订阅参数的实时数值、最大值、最小值、平均值、极差和标准差等。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  点击*配置*进入配置窗口，单击 ⊞ 可添加数据。

  每条数据由以下参数组成。

  数据类型

  可选 int 或 float。

  数据源

  可从方案中订阅符合**数据类型**要求的数据。

  名称

  自定义该数据的名称。

  数组数据索引

  若订阅的**数据源**为数组数据，则通过该参数控制选择数组中的第几位数据使用。默认为 0，代表第 1 位数据。当**数据源**不是数组数据时，该参数应设置为 0。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

> **说明**
> 若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

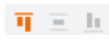
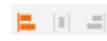
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置标题名称。

显示标题栏

可设置标题栏是否显示。

列标题显示

可选择显示的数据类型，可勾选当前值、最大值、最小值、平均值、极差和标准差。

标题文本行
标题行
列标题文本颜色
列标题背景
行背景
交替行背景
字体颜色

可分别设置以上对应板块的颜色。

饼图

饼图控件可将多个数据源的数值之和作为一个整体，在运行界面实时显示每次流程执行时各数据源数值在整体中的占比情况。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  点击*配置*进入配置窗口，单击 ⊞ 可添加数据。

  每条数据由以下参数组成。

  数据类型

  可选 int 或 float。

  数据源

  可从方案中订阅符合**数据类型**要求的数据。

  名称

  自定义该数据的名称。

  数组数据索引

  若订阅的**数据源**为数组数据，则通过该参数控制选择数组中的第几位数据使用。默认为 0，代表第 1 位数据。当**数据源**不是数组数据时，该参数应设置为 0。

  默认颜色

  控制该控件中该参数以设置的颜色做呈现。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

  > **说明**
  > 若开启**自动填充**，**自适应**将无法开启。

  尺寸

  可调整控件的大小，也可通过预览及编辑区调整。

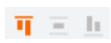
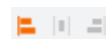
  边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置标题名称。

显示标题栏

可设置标题栏是否显示。

标题文本行
标题行
背景色

可分别设置以上对应板块的颜色。

参数组合

参数组合控件即将多个参数控件组合为一个控件。
参数分为数据源、布局、外观三大类。

- 数据源：

数据源

点击*配置*进入配置窗口，单击  可添加数据。

每条数据由以下参数组成。

参数名称

可自定义该参数的名称。

数据类型

可选择订阅数据的类型，可选 int 或 float。

数据源

可订阅方案中 int 或 float 型参数。订阅的参数类型与**数据类型**有关。

小数位数

数据类型选择 float 时，该参数决定显示参数的小数位数。

阵列方式

可选纵向阵列或横向阵列。

间距

可设置该控件中每个控件之间的间距。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

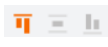
边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

 分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置标题名称。

显示标题栏

可设置标题栏是否显示。

字体

可设置使用字体的大小及字形。

标题文本行
标题行
字体颜色

可分别设置以上对应板块的颜色。

文本框组合

文本框组合控件即将多个文本框控件组合为一个控件。
参数分为数据源、布局、外观三大类。

- 数据源：

  数据源

  点击*配置*进入配置窗口，单击 ⊞ 可添加数据。

  每条数据由以下参数组成。

  字符信息

  订阅数据源时，此处显示订阅后的信息；不订阅数据源时，可通过该参数设置文本框显示的信息。

  数据绑定

  绑定**数据源**后自动加载，显示为“订阅数据的描述：<订阅的数据>”。其中“订阅数据的描述：”可自定义修改；“<订阅的数据>”为格式化语法内容，选择不同的数据，语法内容有所差别。

  - **数据类型**选择 string 时，格式化语法内容对应为“%s”。
  - **数据类型**选择 int 时，格式化语法默认为“%0d”。格式化语法为该默认内容时，控件输出数据源的 int 型数据。若有格式化需求，您可通过修改“%”后的阿拉伯数字，设置控件输出的 int 数据的最大位数。例如，若数据源输出的原数据为 600，格式化语法设置为“%5d”，那么方案运行时控件将输出 00600。
  - **数据类型**选择 float 时，格式化语法内容对应为“%1.3f”。此时小数点后保留 3 位有效数字，可自定义修改。例如%1.6f 对应为小数点后保留 6 位有效数字。

说明

设置该参数时，需了解格式化语法相关要求。若不了解相关内容且需做修改，请联系技术同事。

数据类型

可设置文本框中显示数据的类型，可选 string（字符串）、int（整数型）和 float（浮点型）。

数据源

可订阅方案中各模块的模块结果、全局/局部变量或方案路径在文本框中显示。选择不同的数据类型，可订阅的数据有所差别。

说明

选择不同的**数据类型**，可订阅的数据有所差别。其中方案路径仅在**数据类型**为 string 时可以订阅。

颜色编辑

可设置订阅的数据源满足**条件**和**比较值**要求时，数据显示对应的**颜色**。

说明

同一个数据满足多个**颜色编辑**设置的条件时，根据设置的条件的顺序执行第一个。

条件

包括**= =**（等于）、**! =**（不等于）、 **>**（大于）、**<**（小于）、 **> =**（大于等于）、**< =**（小于等于）。

说明

其中 string 型数据仅支持**= =**（等于）和**!=**（不等于）。

阵列方式

可选纵向阵列或横向阵列。

间距

可设置该控件中每个控件之间的间距。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

> **说明**
> 若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

分别对应垂直方向的顶部对齐、居中对齐和底部对齐；分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

标题

可自定义设置标题名称。

显示标题栏

可设置标题栏是否显示。

字体

可设置使用字体的大小及字形。

标题文本行
标题行
字体颜色

可分别设置以上对应板块的颜色。

其他

图片

图片控件可添加本地图片在运行界面上显示。

参数分为数据源、布局、外观三大类。

- 数据源：

图像

可加载本地图像在运行界面显示。

拉伸模式

可设置**图像**在控件的显示方式，可选 None、Fill、Uniform 和 UniformToFill。

- 布局：

自动填充

在运行界面中，可将控件自动填充满所在网格。

自适应

若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

说明

若开启**自动填充**，**自适应**将无法开启。

尺寸

可调整控件的大小，也可通过预览及编辑区调整。

边距

可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

对齐方式

分别对应垂直方向的顶部对齐、居中对齐和底部对齐；分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

背景

边框颜色

可分别设置图片背景和边框的颜色。

边框厚度

可设置图片的边框厚度。

说明

点击某个数据右侧的 □ 可使所有数据修改为该数据。

世界时钟

世界时钟控件可在运行界面实时显示所选时区的当前时间。
参数分为时间设置、布局、外观三大类。

- 时间设置：

  格式

  可设置显示时间的格式类型。

  时区

  可设置世界时钟的所属时区。

- 布局：

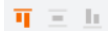
  边距

  可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

  该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

  假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

  对齐方式

   分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

  字体

  可设置使用字体的大小及字形。

  字体颜色

  可设置使用字体的颜色，更改颜色实时生效。

界面日志

界面日志控件可在运行界面上实时显示软件的日志信息（即*系统*的*日志*）。
参数分为布局和外观两大类。

- 布局：

  自动填充

  在运行界面中，可将控件自动填充满所在网格。

  自适应

  若开启，运行界面实际使用时，当前控件的大小和坐标将随运行界面大小调整而等比例缩放。

  说明

  若开启**自动填充**，**自适应**将无法开启。

  尺寸

  可调整控件的大小，也可通过预览及编辑区调整。

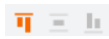
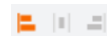
  边距

  可调整控件的位置，也可通过预览及编辑区调整。边距由 T、B、L、R 4 个参数的数值组成，分别对应控件距离顶部、底部、左侧、右侧的像素数。单击 T、B、L、R 中任一参数右侧的 □，可使所有参数的数值同步为被单击参数的数值。

  该参数的 T 和 B、R 和 L 分别仅生效其中一个。具体生效哪两个，由**对齐方式**选择的垂直方向对齐方式及水平方向对齐方式决定（即以**对齐方式**对应的点作为原点，显示控件在水平方向和垂直方向到原点的距离）。

  假设**对齐方式**选择的是顶部对齐和左对齐，则**边距**参数中生效的为 T 和 L 参数，分别显示控件顶部及左侧到运行界面顶部及左侧的距离。此时 B 和 R 无效，数值显示为 0，即使手动修改 B 和 R 的数值也不起作用。

  对齐方式

   分别对应垂直方向的顶部对齐、居中对齐和底部对齐；  分别对应水平方向的左对齐、居中对齐和右对齐。

- 外观：

  标题

  可自定义设置标题名称。

  显示标题栏

  可设置标题栏是否显示。

  标题文本行

标题行
列标题文本颜色
列标题背景
行背景
交替行背景
字体颜色

可分别设置以上对应板块的颜色。

第 10 章 方案调试与运行

在最终向您的客户交付方案之前，您需先在模拟环境中完成方案的试运行与初步调试。之后还需在业务现场运行并调试，直至运行效果满足业务现场的检测需求。

10.1 调试方案

若方案试运行结果不理想，您可以调试模块基本参数、运行参数、模型配置、多流程配置、全局配置和设备配置等。

您可根据主界面 ***图像显示区域*** 和 ***结果显示区域*** 所显示的图像渲染和运行结果信息，调试模块参数和方案设计。您也可通过 ***综合配置*** 将各个流程的运行参数、图像数据和输出参数整合到同一个配置界面，方便调试和查看调试效果。

调试方案时，您需要重点关注以下事项：

1. 确认相机能够正常取图，且取图质量满足业务场景需求。

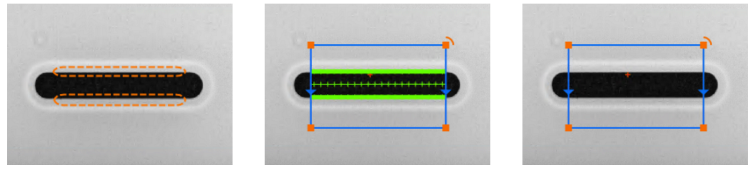


亮度过低（曝光不足）　亮度正常（曝光合适）　亮度过高（曝光过度）

2. 方案的检测精度满足业务需求。您可以检查模块 ROI 设置、运行参数取值以及模型精度。

> **说明**
>
> 如需调试运行参数，您可参考 ***模块使用参考*** 中对应模块的 *参数配置* 章节，调整参数选项或参数取值。

以 ***平行线查找*** 模块的**边缘阈值**参数为例。参数值过大，可能导致模块过滤目标边缘，导致查找失败。



目标平行线
(橙色框框选的边缘)　边缘阈值设置合理　边缘阈值设置不合理

3. 确保 VM 能正常与视觉系统中的设备通信。

10.1.1 图像显示区域

图像显示区域展示两类信息：单个模块输出的图像或者流程输出的图像；模块的输出结果。

图像显示区域在主界面的位置如下图所示。

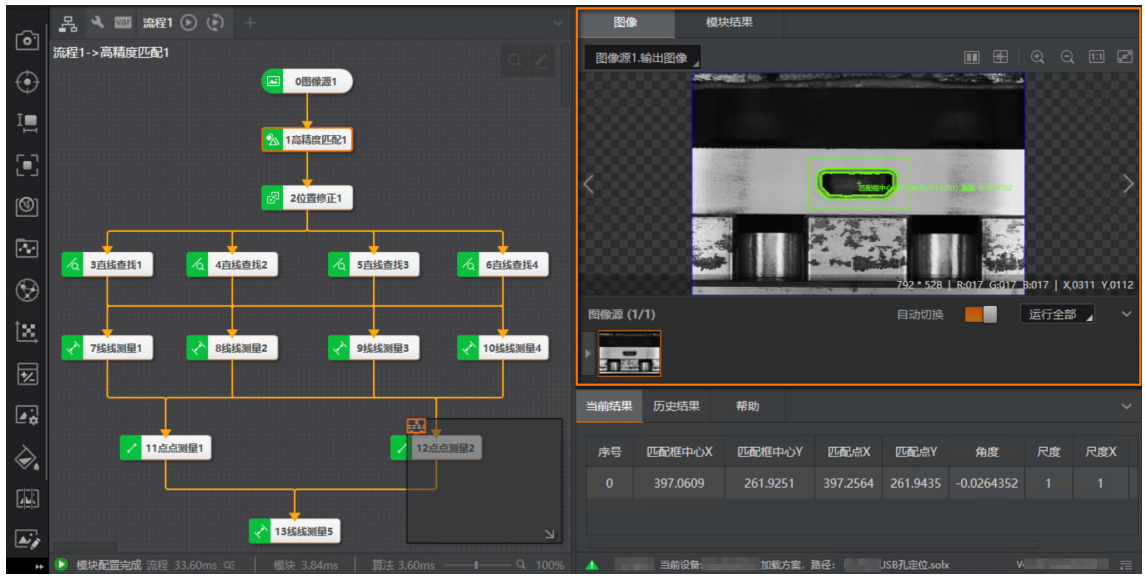


图 10-1 图像显示区域

查看模块图像

模块图像指流程中某个模块输入或输出的图像，支持在主界面的图像显示区域查看。

前提条件

已完成流程配置，包括模块排布连接和各模块参数的配置。

操作步骤

1. 在主界面快捷工具条区域，单击 ▷ 或 ▷ 。

   您也可以在主界面流程栏，单击 ▷ 或 ▷ ，执行流程。

2. 在主界面流程编辑区，单击目标模块。

   执行成功的模块，可在右侧的图像显示区域查看该模块输出的图像。

   图像显示区域支持图像切换和图像叠加，规则如下。

   图像切换

   图像显示区域可切换图像数据类型，在图像显示区域左上角下拉框中可选择目标图像数据类型进行切换。具体可切换的图像数据类型，取决于当前选中的模块在流程拓扑中的

位置，具体规则为：选中某个模块后，可在图像窗口切换的图像类型包括“当前选中模块所输出的图像”和“所有前序模块输出的图像”，如下图所示。



图 10-2 图像切换

图像叠加

不同图像数据类型可叠加显示，在图像切换下拉框中勾选目标图像数据类型前的 ☐，即可叠加显示当前勾选的图像数据。

模块图像相关功能

在图像显示区域还可以对图像进行进行单/双画面切换、添加十字辅助线、本地图配置等操作。

模块图像功能概览

本节提供图像显示区域的概览性内容。

图像显示区域如下图所示。

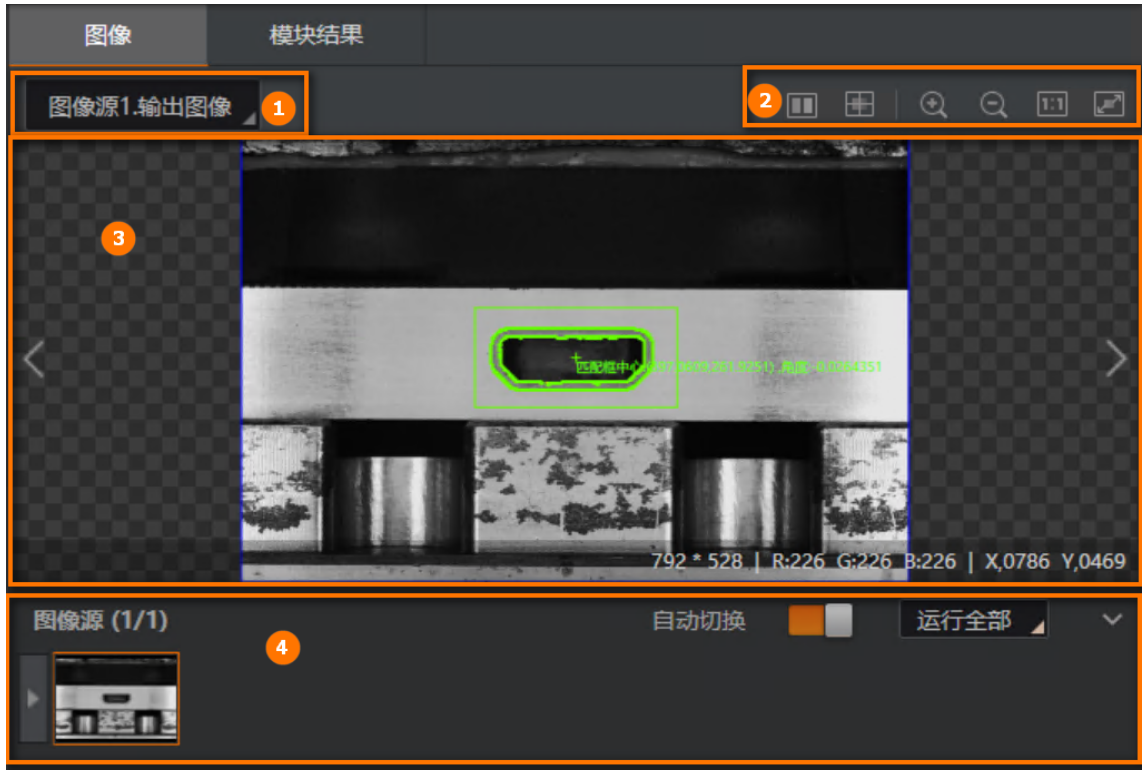


图 10-3 图像显示区域

上图中各区域的具体功能说明如下：

表 10-1 图像显示区域功能说明

区域序号	说明
①	单击选择在区域③（预览窗口）显示的图像数据。 • 单画面显示模式下，可选择当前模块所有输入和输出的图像，并可多选进行叠加显示。 • 双画面显示模式下，左侧的预览窗口仅可选择当前模块输入源订阅的图像数据，右侧预览窗口与单画面模式下支持显示的功能一致。
②	可对预览窗口进行预览相关操作，从左至右的图标依次分别表示“切换画面数量”（切换单/双画面模式）、“*十字辅助线配置*”、“放大”、“缩小”、“自适应”和“全屏显示”。

区域序号	说明
③	预览窗口，显示区域①处选择的图像数据。如图像源为本地图且已在区域④添加多张本地图，可单击该区域左右两侧的箭头图标切换图像。
④	当图像源模块选择图像源为本地图时，可通过该区域添加本地图或局域网内共享的图像并进行相关操作配置，详情请参见 *本地图像配置* 。

十字辅助线配置

单击区域②中的 打开十字辅助线窗口，可在该窗口开启并设置两个互不干涉的十字辅助线。



图 10-4 添加十字辅助线

各参数说明如下：

- **十字辅助线 1/2**：启用后，图像预览窗口显示对应的十字辅助线。
- **坐标 X/Y**：十字辅助线交叉点在图像中对应的 x 轴和 y 轴坐标。
- **粗细**：设置十字辅助线的粗细程度。数值越大，十字辅助线越粗。
- **颜色**：设置十字辅助线的颜色。
- **显示**：单击*居中*可将十字辅助线的交叉点移动至图像的中心点。
- **锁定**：开启后，将无法直接在预览画面上拖动十字辅助线。

说明

如果未开启**锁定**，可单击十字辅助线的交叉点激活拖动功能，然后直接拖动调整其位置（如以下动图所示）。

拖动过程中，十字辅助线的**坐标 X** 和**坐标 Y** 数值将同步更新。

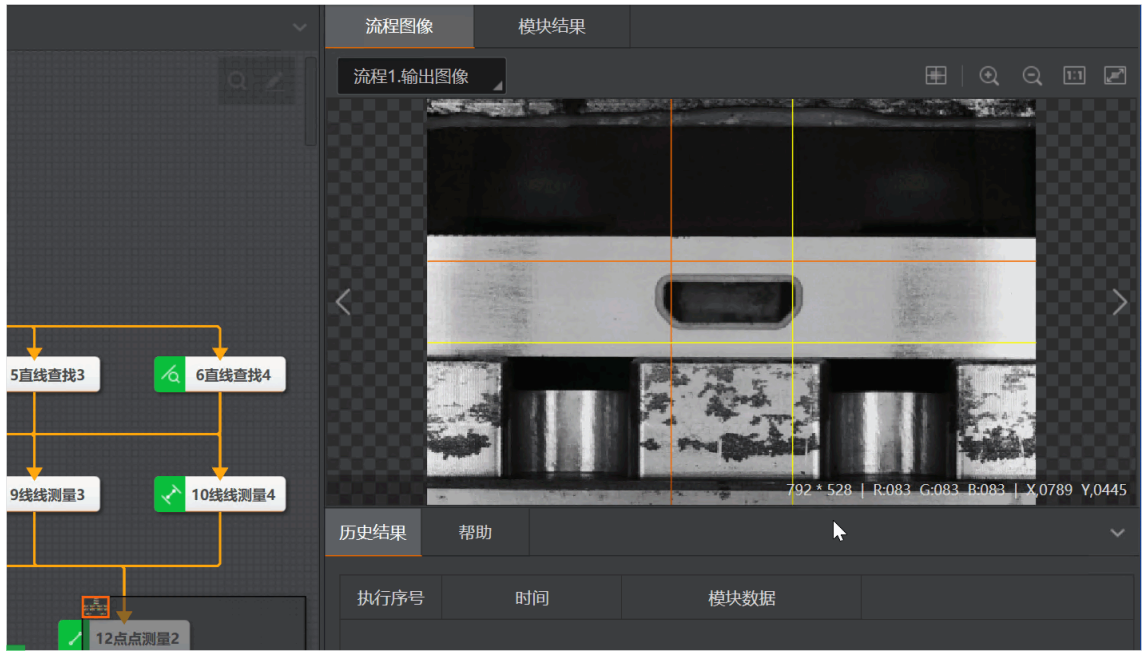


图 10-5 拖动十字辅助线

本地图像配置

当在*图像源*模块中选择本地图作为图像源时，区域④将显示。区域④的各界面功能说明如下：

- ：单击添加本机或局域网内共享的图像。
- ：单击添加本地文件夹或局域网内共享文件夹内的所有图像。
- ：单击将已添加的图像全部删除。
- ：单击显示已添加图像中的第一张图像。
- **自动切换**：一般与**运行全部/运行选中**配合使用。
  - 若开启**自动切换**且开启**运行全部**，单击快捷工具条的 连续执行流程时，可自动切换所有已添加的本地图，并对各图连续执行流程。
  - 若开启**自动切换**且开启**运行选中**，进行连续执行时，仅自动切换选中的本地图，并对各选中的本地图连续执行流程。
  - 若关闭**自动切换**，单击 时仅对单张本地图连续执行流程。
- **运行全部/运行选中**：具体说明参见**自动切换**。

查看流程图像

流程图像指多流程模式下，某个流程被成功执行后最终输出的图像。

前提条件

已完成流程配置，包括模块的排布连接和各模块参数的配置。

操作步骤

1. 在主界面的流程栏，单击 。
2. 配置流程图像。
   1) 在所有流程页面，单击目标流程小窗口上的 。

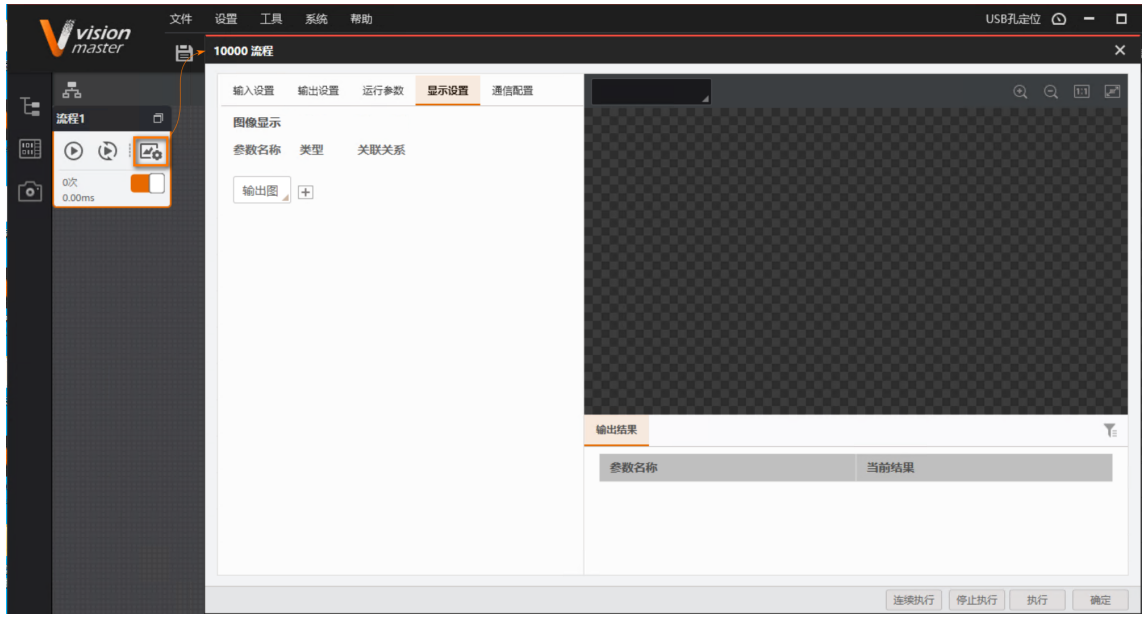


图 10-6 流程设置

   2) 在*流程*窗口，单击*显示设置*。
   3) 在*显示设置*页签的*参数名称*列，选择 *输出图像* → 。

说明

如果您需要同时显示模块图像和数据，*参数名称*选择**多选**，单击 后在*批量选择*窗口批量勾选目标图像和数据，示例图如下。



图 10-7 批量选择

4) 在*关联关系*列，单击 。



图 10-8 订阅关联关系

5) 在订阅列表中，双击选择流程最终输出的图像。

6) 在*流程*窗口右下角，单击*确定*。

即可完成流程图像配置，您也可以单击*执行*或*连续执行*，在*流程*窗口预览该流程最终输出的图像。

3. 在所有流程页面，单击目标流程小窗口上的 或 。

在图像显示区域即可查看流程图像，如下图所示。



图 10-9 查看流程图像

还可对图像进行添加十字辅助线、缩放、全屏显示等操作，详情请参见 *模块图像相关功能*。

查看模块结果

图像显示区域的*模块结果*页签显示所选模块或所选流程的输出数据。

- 在*模块结果*页签选中该模块某个输出参数后，下方将显示 SDK 中与该输出参数对应的信息，如下图所示。其中 VmSolution.Instance 表示方案实例，BlobNum 即 blob 个数。可单击 复制该信息，用于二次开发的结果获取。



图 10-10 SDK 中的对应等信息

- 可单击**变量管理**列的 选择 *全局/局部变量*，将其与对应的结果数据绑定。

10.1.2 结果显示区域

结果显示区域呈现流程中的模块对输入数据的处理结果。此外，该区域还附带其他功能，如配置全局相机的常用参数、查看所选模块的帮助信息（简要的模块使用说明）以及一键清空或保存结果信息。

该区域显示的内容，根据流程编辑区域中选中的实体类型有所区别，具体如下：

选中某个模块时

- **选中采集或逻辑工具模块时**
  执行流程时，如果选中流程中的 *采集模块* 或 *逻辑工具模块*，可在结果显示区查看模块数据处理的历史结果。

  说明

  “采集”类别模块的当前结果即模块输出的图像，已在预览窗口实时展示，无需实时关注的额外信息，因此结果显示区域仅展示这些模块的历史结果；“逻辑工具”类别模块同理，该类别模块的输出为流程的走向或判断结果，无需实时关注，因此不在此处展示这些信息。

  以下动图以*图像源*模块为例展示其历史结果。历史结果信息包括执行序号、执行时间和模块数据（即模块输出的数据）。



图 10-11 仅历史结果

- **选中其他模块时**

执行流程时，如果选中流程中除采集和逻辑工具以外类型的模块，可在结果显示区查看模块数据处理的当前结果和历史结果。以下动图以 *Blob 分析*模块的结果显示为例。

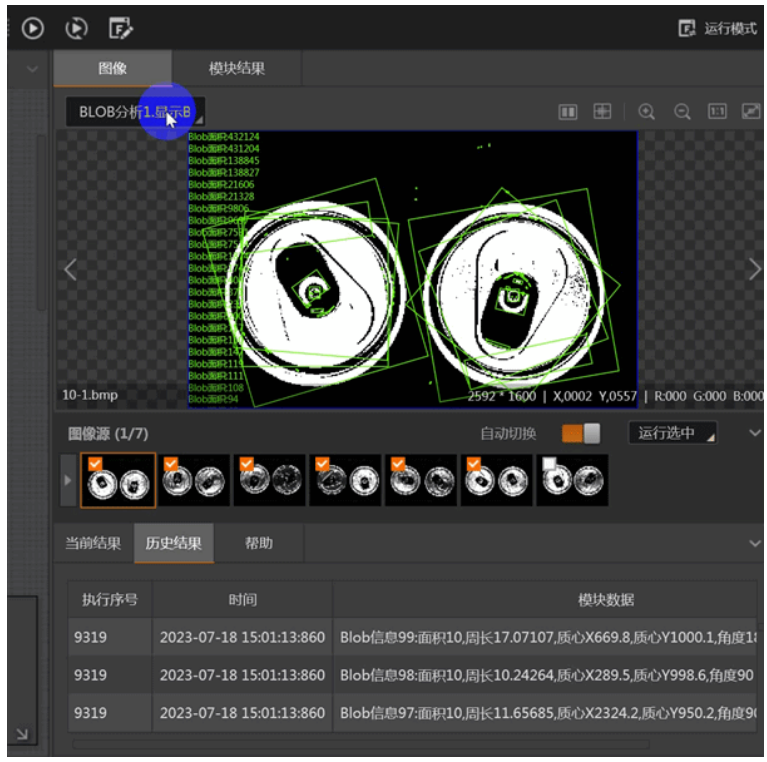


图 10-12 当前结果和历史结果

选中全局相机时

当在流程编辑区选中某个全局相机时（见如下动图），可在结果显示区配置全局相机的常用参数。

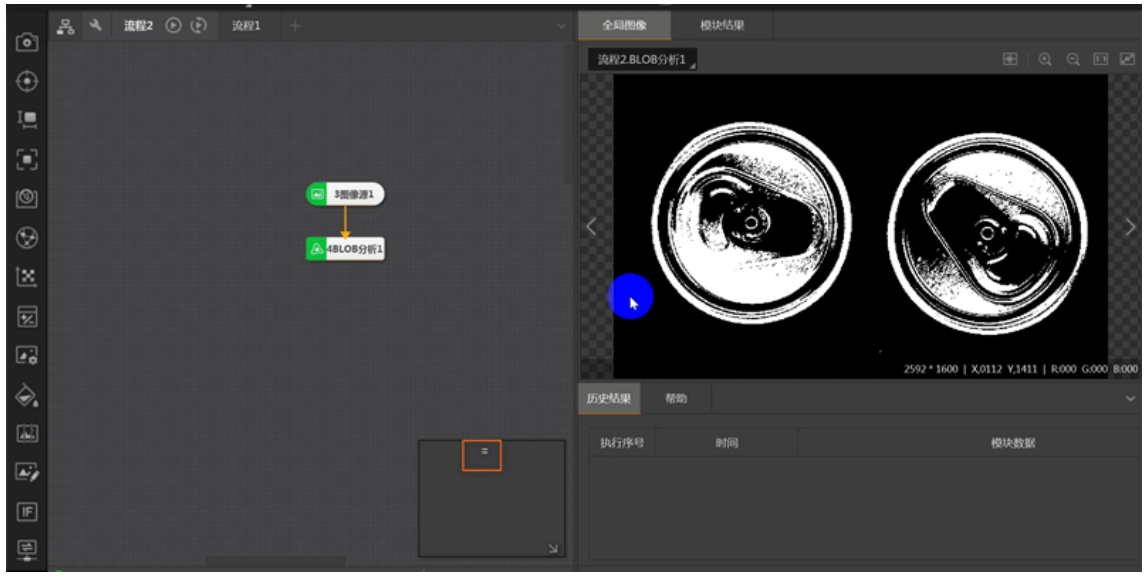


图 10-13 全局相机常用参数展示

选中某个数据队列时

当在流程编辑区选中某个已配置完成的数据队列时（见如下动图），可在结果显示区查看数据队列接收到的各队列数据。

说明

数据队列相关配置及其使用示例，请参见 *数据队列* 及其子章节。

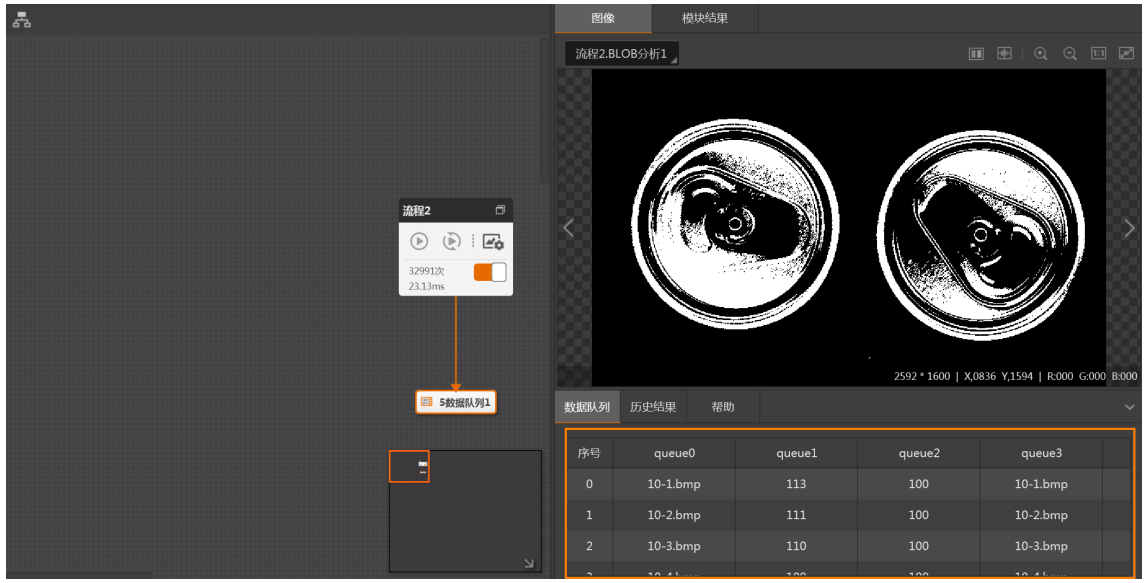


图 10-14 数据队列信息

帮助信息

在流程编辑区选中某个模块、全局相机或数据队列，均可查看对应的简要帮助信息。

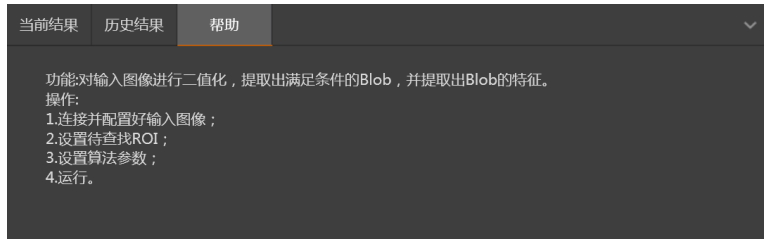


图 10-15 模块的帮助信息（以 Blob 分析为例）

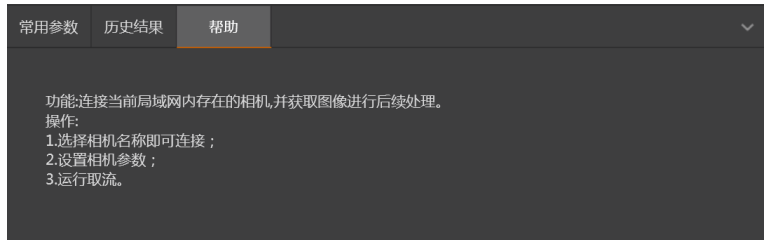


图 10-16 全局相机的帮助信息



图 10-17 数据队列的帮助信息

快捷操作

结果显示区域还支持一键清空或保存所有结果信息。右键单击该区域任意位置并选择*清空*或*保存*，即可实现对应操作。

10.1.3 综合配置

综合配置窗口可将各个流程的运行参数、图像数据和输出参数整合到一起，方便您快速查看关键参数调整后的效果。本节主要介绍综合配置窗口的各个配置区域。

单击快捷工具条的，屏幕将显示综合配置窗口。综合配置窗口如下图所示，分为 *综合参数设置*、*图像显示* 和 *输出结果* 三个区域。

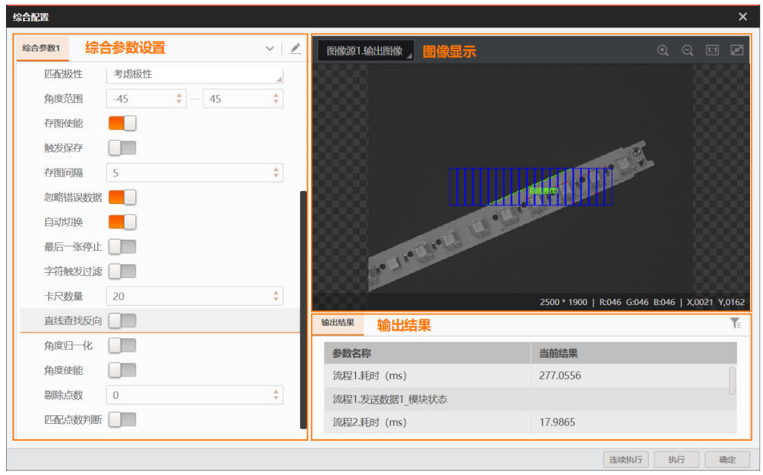


图 10-18 综合配置窗口

各区域的功能说明如下：

综合参数设置

在*综合参数*设置区域，您可以统一查看方案中各流程设置的运行参数并根据实际需要修改相应运行参数。修改操作将实时生效。

说明

*综合参数*区域仅支持显示已设置的自定义流程运行参数。如您未设置自定义流程运行参数，该区域将不显示任何内容。请确保在打开*综合配置*窗口前，您至少已完成一个流程的自定义运行参数设置。具体操作说明，请参见 *自定义各流程运行参数* 。

图像显示

在*图像显示*区域左上角，您可以下拉选择图像源数据，切换查看不同图像源数据的显示效果。当您在*综合参数*设置区域选中某个运行参数时，该区域可显示对应参数所在模块的图像。

输出结果

在*输出结果*区域，您可以实时查看方案中各个流程设置的输出参数。您还可以自定义显示一个或多个输出结果参数，单击 可自定义勾选**流程耗时**参数或方案中各流程设置的输出参数。

10.2 运行单个方案

单个方案的正式运行分为两种方式，即直接在运行界面运行和导出程序运行。

说明

可在菜单栏的 *设置* → *运行策略* 中设置方案运行时 CPU、GPU、内存和网络使用率资源如何合理安排，详情参见 *运行策略* 。

10.2.1 在运行界面运行

本节介绍如何在运行界面运行方案。根据是否已添加*运行控制*控件，运行方式存在些许差异。

已添加“运行控制”控件

若已在运行界面添加*运行控制*控件：

1. 单击主界面右上角的*运行模式*切换至运行界面。
2. 单击运行界面上的*运行控制*即可运行方案。

未添加“运行控制”控件

若未在运行界面添加*运行控制*控件：

1. 在主界面上单击 连续运行方案。
2. 单击主界面右上角的*运行模式*切换至运行界面。
   切换后，方案在运行界面连续运行。

10.2.2 导出运行界面并运行

运行界面设计中支持导出 C#开发语言的程序。导出程序后，可直接运行，也可直接通过二次开发自定义运行界面。通过该方式运行方案，可降低软件对计算机资源的利用率。

前提条件

若已在菜单栏的 *设置* → *权限设置* 配置项中启用加密，请确保您以管理员角色登录 VM。该情况下若以其他角色登录，将无法导出运行界面。

操作步骤

1. 单击*运行界面设计*窗口右上角的*导出*。
2. **名称**处输入自定义的程序名称。
3. **图标**处设置导出程序的图标，仅支持 ico 格式，默认为本公司软件 logo。
4. **选择类型**处选择导出程序的形式，可选 exe、vmCodeProject 和 exe+vmCodeProject 三种类型。
   - 选择 **exe** 时，仅导出生成的程序。
   - 选择 **vmCodeProject** 时，仅导出工程文件和对应的代码文件。
   - 选择 **exe+vmCodeProject** 时，以上程序和文件均会导出。
5. 选择**工程类型**，可选 wpf 或 winform。
6. **存储路径**处选择自定义存储的路径。
7. 单击**导出**即可生成程序。

说明

导出程序时，整体耗时较长，请耐心等待。

示例

假设以**名称**为 test、**选择类型**为 exe+vmCodeProject、**工程类型**为 wpf 方式导出程序，导出程序的文件夹下各文件具体含义如下图所示。

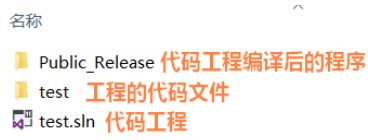


图 10-19 导出程序的文件说明

可通过 test.sln 文件对运行界面进行二次开发，生成后的 exe 在 public_release 文件夹下，双击运行即可。

后续处理

导出的运行界面工程文件在方案文件的基础上，增加了运行界面相关配置，提供了 SDK 的开发示例和基本结构。您可以在此运行界面工程文件基础上进行二次开发，通过运行界面工程文件修改控件属性或增删控件，相比从零搭建界面工程更简单，但限制也会更多。通过运行界面工程文件修改控件操作步骤如下。

a. 双击代码工程文件（例如 test.sln）打开工程。

b. 在“MainWindow.xaml”文件控件区修改控件属性（宽、高等）或增删控件。
在控件区调整代码即可调整已添加控件的属性或增删控件，新增 OK&NG 控件示例见下图。

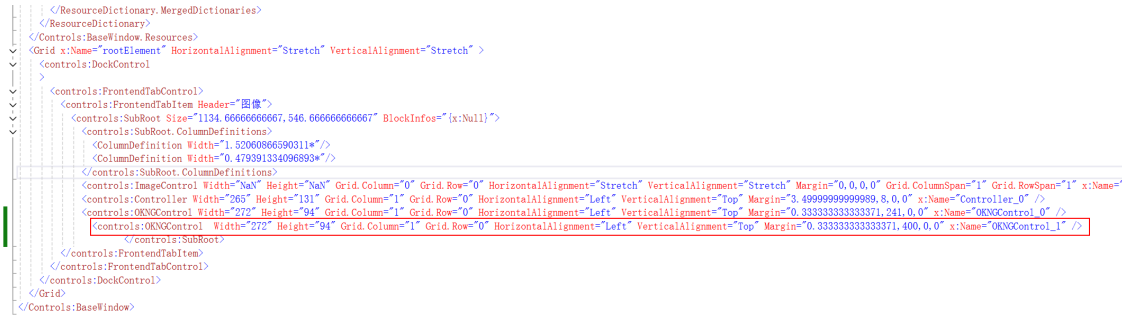


图 10-20 新增 OK&NG 控件示例

说明

在控件区输入<controls:即可在弹框中选择运行界面控件。您也可添加自定义 WPF 控件。

c. 如果新增控件需配置数据源，需在“MainWindow.xaml.cs”文件中为控件配置数据源。

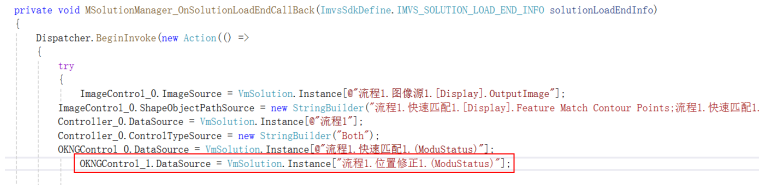


图 10-21 为控件配置数据源示例

数据源信息可在 VM *模块结果*处获取，详情请参见 *查看模块结果* 。

d. 单击*启动*，即可启动运行界面查看效果。

后续运行界面调用及后台代码逻辑的开发详细说明请查看 VM 算法开发平台 SDK 开发指南，文档可在“.\Development\V4.x\Documentations”路径下获取。

10.3 自动切换方案并运行

除了手动运行方案，VM 也支持自动切换并运行方案。该功能适用于“通过通信设备自动控制方案运行”的业务场景。

前提条件

已添加并配置通信设备，详情参见 *通信设备管理* 。

操作步骤

1. 进入方案自动切换配置界面。
   - **选项 1**：在主界面菜单栏进入 *设置* → *方案设置* 。
   - **选项 2**：在*运行界面设计*窗口单击右上角的 ，并选择*方案管理*，打开如下窗口。



图 10-22 方案切换配置窗口

2. 配置方案切换数据。
   1) 单击**方案路径**输入框右侧的 选择需加载的方案文件。

说明

若该方案设置了密码，则在**方案密码**输入框中输入正确的密码；若未设置密码，则无需填写。

2) 在**通信字符串**输入框中输入可触发方案切换的字符串内容。

注意

- 字符串内容支持阿拉伯数字、字母和汉字，但不支持符号。
- 请务必确保不同方案的字符串内容不一致。

3) 开启*通信切换*。

若不开启，配置不生效。换言之，VM 接收到字符串内容时，无法自动切换至对应的方案。

3. 可选操作: 单击 ⊞ 添加更多方案配置数据，并重复步骤 2 进行配置。

结果说明

VM 接收到来自于通信设备的指定字符串后，在运行界面自动切换方案。

注意

该方案切换配置与工控机绑定，在工控机 A 中设置的方案切换策略仅在工控机 A 中生效，不会随方案在工控机 B 中生效。若更换工控机运行方案，请重新配置。

说明

可在菜单栏的 *设置 → 运行策略* 中设置方案运行时 CPU、GPU、内存和网络使用率资源如何合理安排，详情参见 *运行策略* 。

第 11 章 软件通用设置

您可通过菜单栏的*设置*选项，调整可在 VM 软件全局生效的通用设置，包括权限、方案和运行策略等。

11.1 权限设置

权限设置可导入或导出权限信息，也可设置不同角色人员（包括管理员、技术员和操作员）的权限及密码。

权限导入导出

可对当前软件权限设置模块的配置以 txt 格式文档进行*导入*或*导出*。

启用加密

打开*启用加密*时，需设置管理员密码。设置后可启用管理员角色的权限管理，软件将以管理员角色登录并操作。点击**修改密码**处的*修改*，并输入管理员的旧密码和新密码后即可完成管理员密码的修改。

说明

设置密码时，数字或字母可任意组合，长度需为 3~15 位。技术员和管理员密码设置也需遵循该原则，后续不再赘述。

软件以管理员角色运行时，可对所有功能进行设置，且可设置技术员和操作员的权限。

技术员权限设置

技术员可配置管理员指定的模块参数和编辑前端运行界面，仅管理员可设置技术员权限相关内容。

- 技术员权限：开启后，可设置技术员密码和权限分配，软件以技术员角色登录并操作。
- 技术员密码：设置技术员角色的密码，密码设置原则与管理员密码一致。
- 权限分配：点击*配置*可对技术员支持操作的工具模块进行自定义权限分配。若勾选*开放所有工具*，则所有工具模块均可配置。



图 11-1 权限分配

操作员权限设置

操作员只能对前端运行界面的按钮点击操作，管理员和技术员均可设置操作员的权限。

- 操作员权限：开启后，可设置操作员密码和权限分配，软件以技术员角色登录并操作。
- 操作员密码：设置操作员角色的密码，密码设置原则与管理员密码一致。
- 操作员使能：启用该功能并以操作员身份登录运行界面时，运行界面内部控件可正常操作；否则，运行界面内部控件只可查看不能操作。

软件登录角色切换

完成各个角色的权限设置后，软件右上角会显示当前登录的角色信息。点击当前登录的角色，选择*切换角色*可切换登录的角色，此时输入对应角色的正确密码方可登录。



图 11-2 切换角色

11.2 软件设置

在软件设置，您可进行启动关闭设置、撤销重做设置以及其他高级设置。

启动关闭设置

可设置软件、运行界面、方案以及文件的自动加载操作。

开机软件自启动

该选项启用后，可设置 PC 开机时，软件将延时多少秒自动开启。在**延时启动**参数处，填写自定义的延时时间。

关闭软件密码

该选项启用后，关闭软件时，需输入设置的密码，方可关闭软件。您可以在弹窗中设置自定义的密码，并单击**确定**保存。

隐藏欢迎页

该选项启用后，打开软件时会跳过欢迎页，直接进入主界面。

说明

欢迎页勾选*不再显示*也可以开启此功能。

启动时最大化

该选项启用后，软件启动时，默认最大化显示软件界面。

启动时全屏

该选项启用后，软件启动时，默认全屏显示软件界面。

默认启动运行界面

该选项启用后，软件启动时，将默认不显示主界面，直接打开运行界面。

运行界面独立启动时最大化

该选项启用后，软件启动时，将最大化显示运行界面。

说明

该选项仅在您开启*默认启动运行界面*选项后才能生效。

自动加载指定方案

该选项启用后，打开软件时，软件将根据设置自动加载方案。启用该功能后，需设置以下参数：

载入路径

单击 选择需自动加载方案的路径。

方案密码

若加载的方案已设置加密，需输入正确的方案密码，否则无法打开。

启动状态

可选**非连续执行**或**连续执行**。选择**非连续执行**时，若同时开启高级设置中的***静默执行***，则默认加载方案后静默执行一次方案；若未开启***静默执行***，则不执行方案。选择**连续执行**时，加载方案后连续运行。

选择身份

选择可以查看自动加载方案的身份。可选管理员、技术员和操作员。

说明

仅当您在软件开启权限设置时，才能查看该参数。

自动加载最近打开方案

该选项启用后，软件打开时，会自动加载最近打开方案列表中的第一个方案。启用该功能后，需设置以下参数。

启动状态

可选方案打开时的启动状态。可选**非连续执行**或**连续执行**。选择**非连续执行**时，若同时开启高级设置中的***静默执行***，则默认加载方案后静默执行一次方案；若未开启***静默执行***，则不执行方案。选择**连续执行**时，加载方案后连续运行。

说明

自动加载指定方案和**自动加载最近打开方案**功能互斥，不可同时使用。请根据实际需要选择开启其中一个功能。

撤销重做设置

缓存步数设置

可设置快捷工具条中***撤销*** 和***重做*** 功能最多可缓存的步数，上限为 15 步。

模块自动连线间距设置

模块间距

可设置模块间自动连线的长度，上限为 80 px。设置后，当您在方案中的某一模块下方添加算法模块时，软件将根据设置的长度自动提示连线位置，将算法模块放置到橙色指示块处，模块之间的连线将自动创建。比如，将模块间距设置为 50 px，连线效果如下图所示。

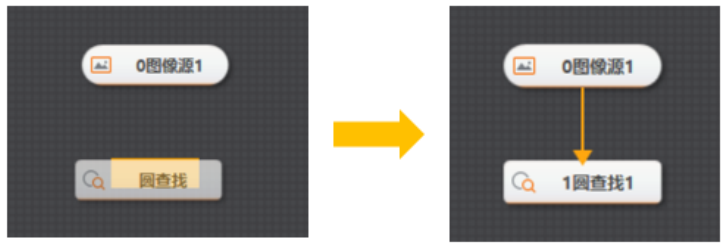


图 11-3 模块自动连线示意图

高级设置

模块最大数量

可设置在方案中允许拖入的模块最大数量。

说明

设置**模块最大数量**参数后，需要重启软件以使改动生效。

静默执行

开启后，可设置手动加载方案运行是否启运一次静默执行。

说明

设置*静默执行*后，需要重启软件以使改动生效。

静默执行模块选择

当您开启**静默执行**后，可从以下选项中选择需执行的模块。

- **深度学习和脚本模块**
- **所有模块**
- **深度学习模块**
- **脚本模块**

修改参数自执行

开启后，修改绘制的 ROI 或运行参数时可自动展示出该模块对应的图像效果，订阅的参数不支持此功能。

修改模型自执行

开启后，在模型参数配置界面修改对应参数时，模型参数会立即生效。

流程延时

可设置流程执行完一次，延时多长时间执行下一次。

图像高质量渲染

开启后，会关闭图像下采样，视觉效果更好，但也会增加渲染耗时和 CPU 占用。

图形数量上限

可按需设置单个渲染控件单次渲染图形数量的上限，用于防止图形数量过多导致渲染出现卡顿的情况。

深度学习库预加载

开启时，可降低深度学习模块首次执行的耗时。

GPU 锁频开启

可用于稳定 GPU 的耗时波动，并在一定程度降低 GPU 的运行耗时。开启后，可通过 **GPU 锁频频率**参数设置 GPU 固定工作的频率。例如，当该参数设置为 100%时，可锁定 GPU 工作频率为其支持的最高频率；当该参数设置为 80%时，可锁定 GPU 工作频率为其最高频率的 80%。

注意

- **GPU 锁频开启**功能仅适用于英伟达显卡，不支持在其他显卡上使用。
- 设置的百分比越高，GPU 的运行效率越高，但是这也会相应增加 GPU 的功耗。如 GPU 长时间以最高频率运行，则可能会导致显卡使用寿命降低。请您根据实际情况设置该参数。

ROI 图层是否偏移

可设置是否将图像中绘制 ROI 的图层整体偏移 0.5 个像素点。默认开启。

看门狗

针对特殊场景，一般无需开启。

执行前清空缓存

开启后，每个流程在执行前会清空一次历史结果数据。

11.3 方案设置

方案设置可进行方案自动切换、回调设置和自动保存设置。

方案自动切换

通过配置以下参数，您可以指定软件在收到特定字符串后，自动切换方案。点击 ⊞ 可设置多个方案的自动切换策略。

说明

方案自动切换配置仅在当前电脑生效。比如，在电脑 A 中设置的方案切换策略只在电脑 A 中生效，不会随方案在电脑 B 中生效。因此，如果您需要更换电脑运行方案，必须在新电脑重新配置方案切换策略。

方案路径

点击 选择需自动切换的方案。

方案密码

若选择的方案有设置密码，需输入正确的方案密码；否则，在方案切换时，软件无法正常加载方案。

通信字符串

设置通信字符串。软件收到指定的字符串后，将自动切换方案。

通信切换

该功能开启后，在通信设备发送指定字符串内容给软件时，方案会自动切换。

说明

使用方案自动切换功能时，需确保通信管理处已完成通信设备的添加和设置。通信设备管理详情，请参见 *通信设备管理* 章节。

回调设置

单击*全部打开*并在弹窗中点击*确定*，可通过回调获取整个方案中各个模块的模块结果信息。

说明

该功能按钮开启后，当您选中方案中的单个模块时，该功能会自动关闭，此时您通过回调仅能获取选中模块的模块结果信息。如需通过回调获取全部模块的模块结果信息，需再次开启该功能按钮。

自动保存设置

启用*自动保存设置*功能后，软件每 5 分钟检查模块运行参数是否有更新。若有改动，则将数据保存在方案文件的同级目录下。

自动保存设置（全量保存）

启用*自动保存设置（全量保存）*功能后，软件将根据设置的**自动保存间隔**时间自动保存方案。

11.4 运行策略

运行策略可设置软件运行时 CPU、GPU、内存和网络使用率资源如何合理安排等。

运行模式

可选**普通模式**和**诊断模式**。

- 普通模式：当 CPU、GPU、内存和网络使用率中任意一项过高时，软件不会告警提示；
- 诊断模式：当 CPU、GPU、内存和网络使用率中任意一项过高时，软件会进行告警提示。

策略模式

可选**默认**、**自定义**和**耗时稳定**。

- 默认：该模式下，所有流程集中调度，线程数量与 CPU 核数自动分配；
- 自定义：该模式下，可针对各个流程进行**线程配置**以及 **CPU 配置**，一般在流程耗时波动较大或 CPU 占用较高时使用；

  说明

  进行 CPU 配置时，可通过*全选*框，批量选中或取消选中所有 CPU 选项。

- 耗时稳定：该模式下，每个流程的任务在同一线程中以串行方式执行。

11.5 采集策略

通过*采集策略*，您可以统一管理多个*输出图像*模块的存图操作，避免因高频存图操作导致磁盘资源的使用率过高，从而引起存图失败。

存图设置

当您使用多个*输出图像*模块进行高频存图时，可能会导致磁盘使用率过高，从而引起存图失败问题。此时建议您在存图设置区域开启*大文件存图*功能，统一管理多个*输出图像*模块的存图操作。该功能开启后，当*输出图像*模块输出的图像达到固定数量时，软件将自动把待存储图像统一压缩成一个.dat 文件存到本地，降低数据的写入频率，优化存图效率。该功能需配套解压工具使用，普通图像文件可单击直接打开查看，.dat 文件需通过 DecompressionTool.exe 工具（即 *存图大数据解压工具*）进行查看。

大文件存图

点击*大文件存图*右侧按钮即可开启或关闭该功能。开启后，您需要配置以下参数：

大文件路径

设置.dat 文件的本地保存路径。

说明

设置大文件路径时，需注意以下几点：

- 确保所选路径下的存储空间足够。如存图过程中出现磁盘空间不足，将导致后续采集的图像无法存储至指定路径。
- 当您开启*大文件存图*功能后，图像会全部存储到该功能设置的存图路径下。此时，各*输出图像*模块下设置的存图路径将不生效。

第 12 章 工具介绍

VM 中集成多种辅助性工具，帮助您快速完成方案搭建和运行过程中相关的辅助工作，例如生成标定版、检测软件运行环境和切换软件版本。部分工具可通过主界面菜单栏的*工具*选项访问，部分工具需前往 VM 安装路径下的\Applications\Tools 文件夹访问。

12.1 标定配置引导

通过菜单栏的*工具*可打开*标定配置引导*。该工具通过步骤引导的方式，可快速完成一键标定的复杂操作流程。

根据实际需求选择**标定类型**后，点击*创建*即可进入标定的引导配置流程。

工具支持 3 种标定类型，具体如下：

- *静态标定*：通过静止的单相机和标定板确定图像坐标系和物理坐标系之间的转换关系。
- *动态标定*：通过特征点的移动确定图像坐标系和物理坐标系之间的转换关系以及旋转中心。
- *映射标定*：通过特征点确定两个相机之间图像坐标系的转换关系。

12.1.1 静态标定

静态标定主要通过*标定板标定*模块实现标定。

前提条件

通过*标定配置引导*工具选择**静态标定**并*创建*，从而进入引导配置界面。

操作步骤

1. 完成*通信配置*的设置。
   1) 点击**设备管理**处的*配置*可快速打开 *通信管理*，添加并配置外部通信设备。

      说明

      开始标定前需在通信管理中启用选择的设备。

   2) 在**设备选择**处选择需关联的通信设备。

      说明

      此处仅支持选择 TCP 或 UDP 类型的设备。

   3) 在**触发信号**处配置控制流程执行的字符串信息。
   4) 根据实际需求选择是否*使用结束符*。若使用，还需下拉选择结束符，可选\r（回车）、\n（换行）、\r\n（回车换行）。
2. 点击 *下一步*进入*图像来源*，并完成配置。
   1) 点击**相机管理**处的*配置*可快速打开 *相机管理*，添加并配置全局相机。
   2) 点击*图像源配置*对 *图像源* 模块进行配置。

3. 点击 *下一步*进入*标定方式*，并完成配置。
   1) 选择**标定方式**，当前仅支持标定板标定的方式。
   2) 点击*标定板标定*对 *标定板标定* 模块进行配置。
4. 完成前 3 步操作后，点击 流程模式 可查看该标定流程中的模块搭建及配置情况。
5. 点击 *下一步*进入*结果查看*，进行标定并生成标定文件。
   1) 勾选*开始标定*。
   2) 通过选择的外部设备发送特定的触发信号，此时开始标定并显示信号内容和结果状态。

   说明

   发送触发信号时，需确保通信管理的设备列表处已启用该设备，并通讯正常。

   3) 点击*生成标定文件*可另存为标定文件，*标定矩阵*处也会显示具体信息。

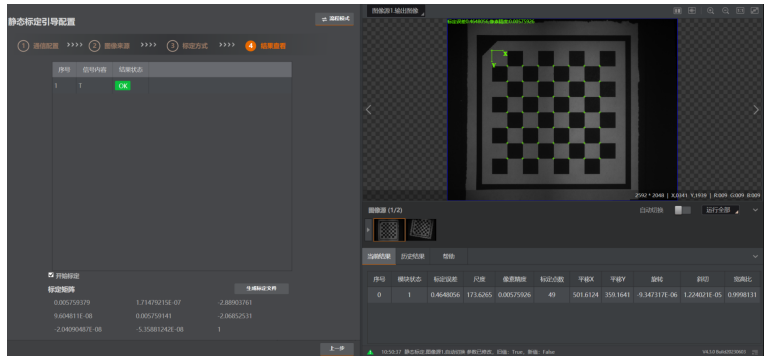


图 12-1 静态标定的结果查看

后续处理

完成标定后，可将生成的标定文件在标定等相关模块中进行加载使用。

12.1.2 动态标定

动态标定主要通过 *N 图像标定*或 *N 点标定*模块实现。

前提条件

通过*标定配置引导*工具选择**动态标定**并*创建*，从而进入引导配置界面。

操作步骤

1. 完成*通信配置*的设置。
   1) 点击**设备管理**处的*配置*可快速打开 *通信管理*，添加并配置外部通信设备。

   说明

   开始标定前需在通信管理中启用选择的设备。

   2) 在**设备选择**处选择需关联的通信设备。

说明

此处仅支持选择 TCP 或 UDP 类型的设备。

3) 在**通信信号**处分别配置**开始信号**、**标定信号**、**结束信号**和**示教信号**的**输入**。

说明

- **标定信号**处的 X、Y 和 R 分别为物理坐标点的横坐标、纵坐标和角度，根据实际情况输入即可。
- **OK 输出**和 **NG 输出**处分别呈现各个信号 OK 或 NG 时输出给外部通信设备的信号。

4) 根据实际需求选择是否*使用结束符*。若使用，还需下拉选择结束符，可选\r（回车）、\n（换行）、\r\n（回车换行）。

2. 点击 *下一步*进入*图像来源*，并完成配置。
   1) 点击**相机管理**处的*配置*可快速打开 *相机管理*，添加并配置全局相机。
   2) 点击*图像源配置*对 *图像源* 模块进行配置。
3. 点击 *下一步*进入*标定方式*，并完成配置。
   1) 选择**标定方式**，可选标定板标定或特征点标定。

      选择不同的标定方式，操作有所差别。

      - 选择标定板标定时，点击 *N 图标定*对 *N 图像标定* 模块进行配置。
      - 选择特征点标定时，往下执行子步骤。

   2) 点击**模块配置的**的 选择特征点由哪些模块提供，可添加多个。点击添加的模块，可对模块进行配置。
   3) 点击 *N 点标定*对 *N 点标定* 模块进行配置。
4. 完成前 3 步操作后，点击 流程模式 可查看该标定流程中的模块搭建及配置情况。

说明

点击右上角的 引导模式 可返回引导配置模式。

5. 点击 *下一步*进入*结果查看*，进行标定并生成标定文件。
   1) 勾选*开始标定*。
   2) 通过选择的外部设备发送特定的触发信号，此时开始标定并显示信号内容和结果状态。

      说明

      发送触发信号时，需确保通信管理的设备列表处已启用该设备，并通讯正常。

   3) 点击*生成标定文件*可另存为标定文件，*标定矩阵*处也会显示具体信息。

后续处理

完成标定后，可将生成的标定文件在标定等相关模块中进行加载使用。

12.1.3 映射标定

映射标定主要通过*映射标定*模块实现。

前提条件

通过*标定配置引导*工具选择**映射标定**并*创建*，从而进入引导配置界面。

操作步骤

1. 完成*通信配置*的设置。
   1) 点击**设备管理**处的*配置*可快速打开 *通信管理*，添加并配置外部通信设备。

      说明

      开始标定前需在通信管理中启用选择的设备。

   2) 在**设备选择**处选择需关联的通信设备。

      说明

      此处仅支持选择 TCP 或 UDP 类型的设备。

   3) 在**通信信号**处分别配置**主相机触发信号**、**从相机触发信号**和**示教信号**。
   4) 根据实际需求选择是否*使用结束符*。若使用，还需下拉选择结束符，可选\r（回车）、\n（换行）、\r\n（回车换行）。
2. 点击 *下一步*进入*图像来源*，并完成配置。
   1) 点击**相机管理**处的*配置*可快速打开 *相机管理*，添加并配置全局相机。
   2) 点击*主相机图像源配置*和*从相机图像源配置*分别对主从相机的 *图像源* 模块进行配置。
3. 点击 *下一步*进入*标定方式*，并完成配置。
   1) 选择**标定方式**，当前仅支持映射标定的方式。
   2) 点击*映射标定*对 *映射标定* 模块进行配置。
4. 完成前 3 步操作后，点击 流程模式 可查看该标定流程中的模块搭建及配置情况。

   说明

   点击右上角的 引导模式 可返回引导配置模式。

5. 点击 *下一步*进入*结果查看*，进行标定并生成标定文件。
   1) 勾选*开始标定*。
   2) 通过选择的外部设备发送特定的触发信号，此时开始标定并显示信号内容和结果状态。

      说明

      发送触发信号时，需确保通信管理的设备列表处已启用该设备，并通讯正常。

   3) 点击*生成标定文件*可另存为标定文件，*标定矩阵*处也会显示具体信息。

后续处理

完成标定后，可将生成的标定文件在标定等相关模块中进行加载使用。

12.2 标定板生成

您可通过*标定板生成工具*生成标定板轮廓图纸文件。在获得图纸文件后，您需根据实际场景需求填充和打印标定板。

前提条件

在使用该工具前，建议您先确认现场中的相机工作距离、项目精度要求、标定板材质等信息，以确保生成的标定板满足需求。

打开软件，通过菜单栏选择 ***工具 → 标定板生成*** 进入工具窗口。

操作步骤

1. 选择标定板类型并按下回车键。
   - 海康 I 型标定板和海康 II 型标定板的区别，请参见 ***标定板生成*** 模块的相关介绍。
   - 常规型与扩展型标定板支持的行列数不同。常规型标定板支持行列数最大到 255，扩展型标定板支持最大 1000 行和列。
2. 根据界面提示，依次输入标定板相关参数。

   理想情况下，标定板需覆盖相机视野。

   说明

   标定板相关参数受到项目精度、标定板加工精度、加工材质、相机视野等多因素影响。
3. 配置完成后，按下回车键，生成标定板预览图像（.bmp 格式文件）以及标定板轮廓图纸（.dxf 格式文件）。

   注意

   预览图像仅在行列数都在 50 以下时生成。该文件供预览标定板显示效果，请勿用于打印标定板。

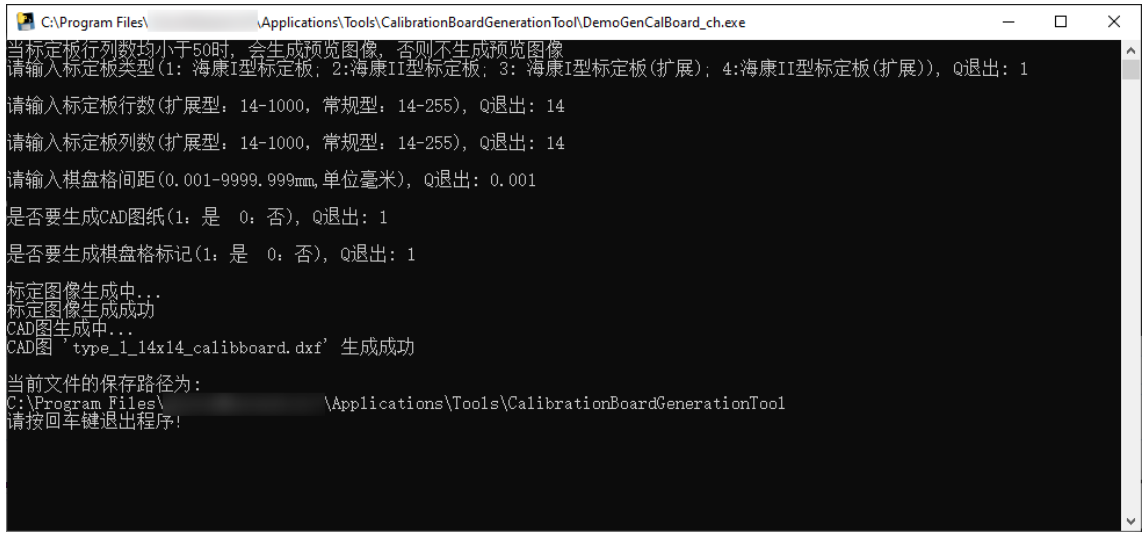


图 12-2 标定板生成工具

4. 前往 VM 安装路径获取生成的文件。

   相对路径：.\Applications\Tools\CalibrationBoardGenerationTool。

说明

该文件无法直接用于标定板加工，您需向棋盘格填充颜色后使用。

后续处理

在获得轮廓图纸后，您还需手动填充棋盘格颜色并加工标定板。

说明

- 填充棋盘格时，您需根据场景中使用的光源、标定板材料等相关因素，选择合适的填充颜色。
- 在加工标定板时，您需确保图纸的缩放比例保持不变、选择正确的单位且未栅格化图纸。同时建议您在加工后测量标定板数据，以确保加工的标定板可用于您的方案。

12.3 异常信息收集

异常信息收集可收集电脑配置信息、VM 环境信息、算子 SDK 日志和二次开发相关日志等，主要用于程序使用出现问题时收集相关信息并提供给技术人员进行问题排查。

菜单栏选择 *工具 → 异常信息收集* 即可打开工具。打开后默认进行 *VM 环境*检测。

工具可进行如下功能。

系统匹配

单击*系统匹配*时，工具自动检测**本机配置**，并提供**最低配置**和**推荐配置**。

VM 环境

单击 *VM 环境*时，工具自动对软件、计算机用户权限、Python 版本、加密驱动、相机驱动、软件运行依赖库、二次开发环境、算子包等进行检测。
可单击右上角的*一键修复*解决发现的问题，也可在各个检测项的右侧进行相关修复操作。

说明

在软件检测项的右侧，点击*打开版本切换工具*可进行版本切换，具体介绍参见 *<u>版本切换工具</u>* 。

SDK 引用

当使用 SDK 进行二次开发时，若遇到问题无法定位到原因，可参考 *SDK 引用*中的排查步骤进行排查。若在已安装软件版本的基础上有重新安装软件的补丁包，建议单击*确认更新*对项目的引用进行更新。

说明

界面上方会显示 **VM 当前版本**。

- 软件为 4.2 及以上版本时，需先通过*立即检测*检测 VMSDK 项目是否包含 GAC 文件，再进行*确认更新*。
- 软件版本低于 4.2 时，需通过 **SDK 项目路径**选择 VM SDK 项目 exe 生成路径并点击*确认更新*进行 VM SDK 引用更新。

日志调整

*日志调整*可设置*日志收集*所生成日志的等级。
通过**请选择日志等级**勾选对应等级并单击*切换等级*即可。
日志等级相关介绍参见下表。

表 12-1 日志等级说明

日志等级	说明
Error 级别	错误日志，日志信息最简单。
Info 级别	信息类日志，默认为该等级。
Debug 级别	用于开发 Debug 调试。一般程序出现异常且容易复现时，推荐使用该级别进行定位，收集结束后调回 Info 级别。
Trace 级别	日志打印最详细，有助于排查故障的问题原因

说明

- 当无法定位问题原因时，可调整日志等级，收集更详细的日志信息。当前版本只适用于 VM4.2 及以上版本的日志等级调整，其它版本需在 *SDK 引用*中手动增加项目 exe 的生成路径。
- 单击*还原设置*可将日志等级恢复为默认设置，即 Info 级别。

日志收集

*日志收集*可收集软件或基于 SDK 二次开发程序的日志信息。
勾选需收集的日志，单击*开始收集*即可。收集的日志存储在软件安装路径下，相对路径：.\Applications\Tools\VMCollector。

- VM 应用问题日志：对应软件的日志，已自动填写软件所在路径，无需修改。
- VM SDK 问题日志：对应基于 SDK 二次开发程序的日志，需自行选择项目 exe 的生成路径。

说明

算子的日志路径为 C:\Windows\Temp\MVDSDKLog。

12.4 参数查找工具

参数查找工具可查看方案中各模块输入/输出/运行参数的名称、对应算子参数名称以及当前值。

说明

该功能仅型号中带 6210 或 7120 的加密狗支持。

通过菜单栏中的***工具***选择***参数查找工具***即可打开该工具。打开工具后，左上角选择具体流程后，左侧显示该流程下当前使用的模块。选择具体的模块后，即可通过右侧查看该模块的输入参数、输出参数和运行参数。

模块参数名称处显示软件二次开发中的参数名称，**算子参数名称**处显示对应算子包 SDK 中的参数名称，**当前值**处显示当前参数的数值。

说明

仅部分**模块参数**有对应的**算子参数**，具体请以实际显示为准。

选中具体参数，工具下方会显示该参数分别在软件和算子参数二次开发中对应的信息，如下图所示。二次开发时，可通过复制这串代码在编码中获取对应参数值。



图 12-3 参数查找工具

- 软件中的调用路径：
  VmSolution.Instance 表示方案实例，*IMVSBlobFindModuCs* 为对应模块的命名空间，*IMVSBlobFindModuTool* 为对应模块类名称，*ModuParams* 为对应模块所属类，*Polarity* 为对应的参数名称。
- 算子参数的调用路径：
  AlgorithmTool 表示算法工具实例，*GetRunParam* 为算法获取参数信息的方法，*CircleFindMode* 为对应参数名称（Polarity），*ref* 为参数输出标记，*paramValue* 为输出参数值。

12.5 自定义模块生成

自定义模板生成可通过相关配置自定义生成新模块的 xml、dll 等文件。

前提条件

确保已经打开自定义模块生成工具。
如果您未打开该工具，可在.\Applications\Tools\AlgorithmXMLGenerator 路径下，找到 AlgorithmXMLGenerator.exe 文件并双击该文件，即可打开自定义模块生成工具。

操作步骤

1. 在 **VM 版本**处，下拉选择一个 *VM 版本*。

   说明

   请根据您使用的 VM 软件版本选择对应版本，其中 **V3.X** 对应软件 V3.0.0 ~ V3.4.0 版本，**VM4.X** 对应软件 V4.0.0 ~ V4.3.0 版本，**VM431** 对应软件 V4.3.1 及之后版本。

2. 在**基本输入**处填写自定义模块的**模块名称**，并根据模块的基本参数确定是否需要支持**输入图像源**或**位置修正信息**相关设置，勾选需要支持的功能即可。

   说明

   模块名称处只能填写英文。

3. 在**基本输出**处根据模块的输出信息确定是否需要**模块状态**或**输出图像**相关设置，勾选需要支持的功能即可。
4. 点击**自定义输入输出**处的*添加*，在下方表格中设置参数相关内容，如下图所示。
   - 参数名称：设置参数的底层名称，只能为英文。
   - 显示名称：设置参数上层显示的名称。
   - 参数类型：设置参数的类型，可选 float、int、string、POINT、LINE 和 ROIBOX。
   - 输入/输出：设置参数为输入还是输出，可选 Input（输入）或 Output（输出）。选择 Input 时，对应模块中的基本参数；选择 Output 时，对应模块的模块结果。



图 12-4 算法模块输入输出 XML 配置

说明

自定义输入输出处，还可进行其他操作：

- 点击*删除*，可将当前选中的参数行删除。
- 点击*插入*，可在当前选中参数上方插入一行参数。
- 点击*上移*，可将当前选中的参数上移一行。
- 点击*下移*，可将当前选中的参数下移一行。
- 点击*导出*，可将当前自定义输入输出中显示的参数信息以固定模板的 iofml 格式文件导出。
- 点击*导入*，可导入 iofml 格式文件中的相关参数。导入时，要求 iofml 文件的内容需按照导出的 iofml 文件模板格式进行填写。完成导入后，当前已添加的参数会清空，仅保留导入文件中的参数。

5. 点击**结果显示**处的*更新自定义输出*，则自定义输入输出处设置为 Output 的参数会自动显示在结果显示的各 Tab 页。
6. 根据实际需求在结果显示的**当前结果**、**历史结果**以及**文本显示** Tab 页中设置需要显示的参数。

说明

三个模块互相独立，需分别设置。

7. 点击*下一步*，进入第二步（算法模块界面 XML 配置）界面。
8. 在**基本参数**处根据实际情况勾选需要的 ROI 类型和屏蔽区。

说明

基本参数中的输入图像源和位置修正、以及结果显示的文本显示已在第一步完成设置，此处仅加载设置情况，不可修改。若需修改，可通过点击右下角的*上一步*返回上一步进行相关设置。

9. 点击**自定义运行参数处**的*添加*，在下方表格中设置参数相关内容，如下图所示。
   - 参数类型：设置参数的类型，可选 float、int、enumeration、string、bool、floatBetween、intBetween。
   - 参数名称：设置参数的底层名称，只能为英文。
   - 显示名称：设置参数上层显示的名称。
   - 编辑：对参数的相关内容进行设置。不同类型的参数需设置的内容有所不同。
     - float、int：需设置参数的最大值、最小值和默认值；
     - enumeration：需设置显示名称和枚举值；
     - string：需设置最大长度和默认值；
     - bool：需设置默认为 True 还是 False；
     - floatBetween、intBetween：需分别设置参数上限值和下限值的参数名称、显示名称、最小值、最大值和默认值。
   - 编辑状态：编辑处完成相关设置后显示为 Yes，若未设置则显示为 No。

说明

生成 XML 和工程文件前，必须完成所有参数的编辑，使编辑状态均显示为 Yes。



图 12-5 算法模块界面 XML 配置

说明

删除、插入、上移、下移、导入和导出与自定义输入输出处的相关功能类似，具体请查看第 4 步的说明，此处不再赘述。

10. 根据实际需求确认是否需要模板配置界面。若需要，则勾选*模板配置*，并点击*单击编辑*对相关参数进行配置。

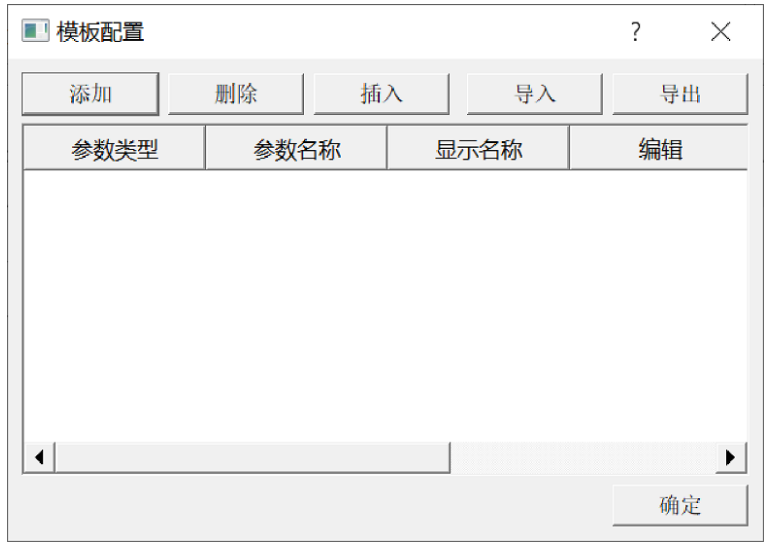


图 12-6 模板配置

说明

参数配置与自定义运行参数处相关功能一致，具体请查看第 9 步的介绍，此处不再赘述。

11. 点击右下角的*生成 XML/ 生成 C++工程/ 生成 C#工程*，选择路径后会将生成的相关文件保存在所选路径下。
    - 生成 XML：生成以自定义模块名称命名的文件夹，主要为相关配置的 xml 文件。
    - 生成 C++工程：生成以“Proj_模块名称”命名的文件夹，主要为 C++相关的工程文件。
    - 生成 C#工程：生成以“CsProj_模块名称”命名的文件夹，主要为 C#相关的工程文件。

说明

推荐将生成的 xml 和工程文件放在同一路径下。此时工程文件编译后的 dll 和 xml 文件也在同一路径下。

12. 对 C++工程文件和 C#工程文件进行编译，生成相应的 dll 文件。
13. 将 dll、xml 和 png 整合到一个文件夹下，并放置到软件安装路径\Module(sp)\x64 的对应类别下。

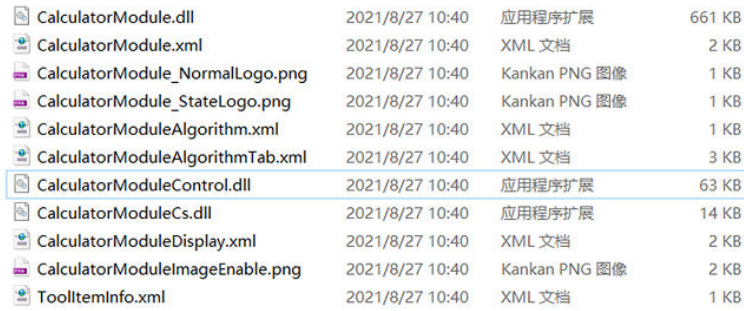


图 12-7 模块文件组成

说明

单个视觉模块需由 10 到 11 个文件组成，如下图所示。其中 CalculatorModuleControl.dll 对应模板配置功能。若模块不需要模板配置功能，该文件可缺少，但其他文件不可缺少。

重新打开软件，对应类别下将新增自定义创建的模块，如下图所示。

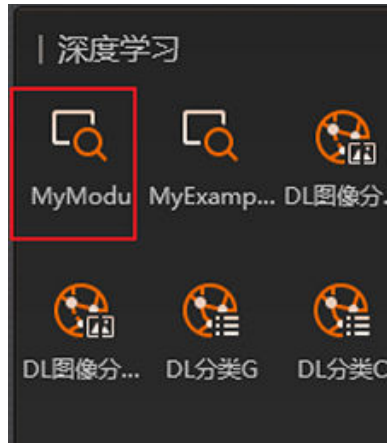


图 12-8 自定义模块呈现效果

12.6 版本切换工具

VMV4.2 版本新增版本切换工具，支持 4.2 及以上版本与其他版本间进行切换。当 PC 上同时安装该软件的多个版本时，可使用该工具切换当前可使用软件的版本。

前提条件

说明

自 VMV4.4.0 的维护版本（2025.04）起，版本切换工具将不再更新，推荐您使用 *激活工具* 切换当前可使用的软件版本。

确保已关闭 VM 和 Visual Studio 软件的所有相关进程。

打开版本切换工具有以下两种方式：

- 在 VM 安装路径下找到 VersionSwitchingAssistant.exe 程序，并双击打开。该程序的相对路径：.\Applications\Tools\VersionSwitchAssistant。
- 通过 *异常信息收集* 的 **VM 环境**面板点击*打开版本切换工具*。

操作步骤

1. 启动工具后会根据注册表自动获取电脑上已安装的软件版本，工具左上角会显示当前版本号。



图 12-9 版本切换工具

说明

版本切换工具右上角按钮功能如下：

- 点击*清除当前*可清除当前运行版本的配置信息，点击*清除所有*可清除所有已安装版本的配置信息。
- 点击*环境检测*会弹出环境检测工具，可检测当前的环境配置信息情况。
- 默认勾选*附带算法切换*，可在版本切换时将附带的算法进行同步切换。

2. 点击对应版本进行切换，等待操作完成即可。



图 12-10 版本切换完成

说明

若切换完成后运行方案不成功，可通过“算子包安装程序”手动安装对应版本的算子包。“算子包安装程序”位于 VM 安装路径下，相对路径为：.\Drivers\MVDAlgorithmSDK_STD.exe；若需使用深度学习相关功能，您还需双击该路径下的 MVDAlgorithmSDK_Patch.exe，安装深度学习算子。

12.7 加密狗写密码工具

若授权方式为本机加密狗或远程加密狗，您可通过“加密狗写密码工具”为加密狗设置密码，对“方案相关成果物文件”进行私有化加密，保护您的知识产权。

前提条件

已在工控机上插入加密狗。

说明

- “方案相关成果物文件”包括方案、流程、Group 以及各模块的模型（不包含深度学习模块加载的模型文件）。
- 通过“加密狗写密码工具”设置加密狗密码之后，上述文件仅可在满足如下两个条件的环境中打开或加载。
  - 工控机已插入加密狗。
  - 加密狗的密码，与通过“加密狗写密码工具”设置的密码相同。

操作步骤

1. 在软件安装路径下找到“加密狗写密码工具”（SetDonglePassword.exe），并双击打开。

   该工具相对路径：.\Applications\Tools\SetDonglePassword。



图 12-11 加密狗写密码工具

2. 在*设置密码*处输入密码，并确认密码。

   密码需包含 6 至 16 为字符，且仅可包含大小写字母和数字。

注意

当您完成加密狗密码设置后，VM 在保存方案、导出流程及导出 Group 文件时默认用该密码进行加密。该密码无法找回，请务必牢记设置的密码。

3. **可选操作**: 按需执行如下可选操作。

   修改密码　通过该工具输入旧密码、新密码即可。

   取消密码　单击工具界面左下角的*恢复出厂设置*，并输入当前密码即可。

结果说明

注意

若使用已设置密码的加密狗，保存方案时请谨慎考虑是否设置方案密码（详见 *菜单栏* 中对 *文件 → 保存方案* 选项的说明）。因为一旦设置方案密码，打开方案必须同时满足如下 2 个条件。为以防万一（例如遗忘方案密码），建议在设置方案密码之前备份原始方案。

- 条件 1：工控机上插入的加密狗的密码，与通过“加密狗写密码工具”设置的密码相同。
- 条件 2：输入的方案密码正确。

12.8 读取加密狗序列号工具

VM 安装路径下自带可读取加密狗序列号的工具 ReadDongleSerialNumber.exe，可到..\VisionMaster4.4.0\Applications\Tools\ReadDongleSerialNumber 路径下获取。

确保 PC 已连接加密狗的情况下，双击打开工具即可获取加密狗的序列号。

说明

加密狗分为 4.0 和 5.0 版本，工具会识别加密狗版本并在对应版本处显示序列号（SN）信息。序列号为 9 位字母和数字的组合，如下图所示。

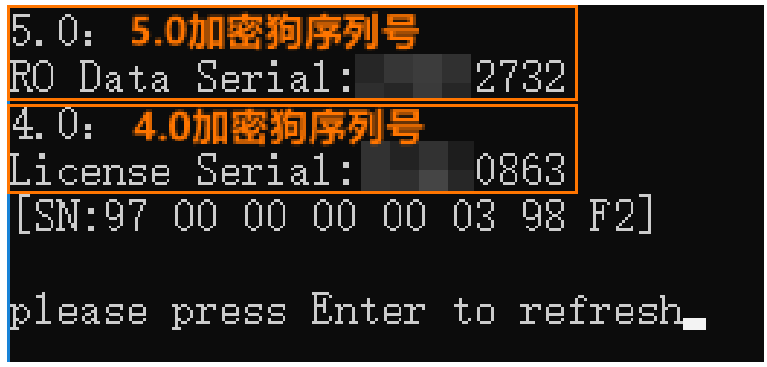


图 12-12 读取加密狗序列号

12.9 存图大数据解压工具

存图大数据解压工具即 DecompressionTool.exe 工具可从 VM 安装路径下获取，相对路径为.\Applications\Tools\SaveImageBigDataDecompression。

该工具主要用于解压 *大文件存图* 存储的.dat 文件，从而得到 *输出图像* 模块存储的图像。

操作方法有以下 2 种：

- 解压单个文件
  - 单击 *导入文件* → *导入文件* 选择单个.dat 文件，并单击*解压*。
  - 将.dat 文件直接拖拽到工具对应区域即可完成解压。
- 批量解压文件
  单击 *导入文件* → *导入文件夹* 选择包含多个.dat 文件的文件夹，并单击*解压*。

导入文件或文件夹后，您可以在图像预览区域预览当前文件内的图像。解压文件后，图像数据将存放在.dat 文件所在路径。若勾选**是否生成和解压文件同名的文件夹**，则会在.dat 文件所在路径下创建和.dat 文件同名的文件夹存放图像；否则，直接将解压后的图像存放在.dat 文件所在路径下。

12.10 激活工具

VMV4.4.0 维护版本新增激活工具，可用于自动激活当前工具所在目录对应的 VM 版本。当 PC 上同时安装 VM 的多个版本时，推荐使用该工具切换对应的 VM 版本。

说明

仅 2025 年 4 月及之后发布的 VMV4.4.0 维护版本支持使用激活工具。若需激活其他 VM 版本，需使用 *版本切换工具* 。

操作方法：

1. 在 VM 的安装路径下找到 ActivationTool.exe 程序, 并双击打开。该程序的相对路径如下: .\Applications\Tools\ActivationTool。
2. 启动后，该程序将自动激活当前版本，等待该工具提示“激活成功”即可。

第 13 章 模块使用参考

软件支持适用于多种检测场景的视觉功能模块，均在主界面左侧的模块箱中。将模块从模块箱拖动到流程编辑区域，并完成模块间的连接和参数配置，即可完成视觉流程搭建。

13.1 采集

采集分类下的模块可设置图像输入源以及如何存图，包括图像源、多图采集、输出图像、缓存图像和光源模块。

13.1.1 图像源

图像源模块可为流程中其他模块提供初始图像，支持三种输入类型，分别为本地图像、相机和 SDK 三种。在方案搭建阶段，您可通过该模块载入本地图像作为图像源，从而对方案的运行效果进行快速调试；现场调试阶段或方案正式运行时，为采集业务现场的实时图像，满足实际业务需求，您可将相机或二次开发接口（SDK）设置为图像源。

使用方法

在流程中，**图像源**模块的前后序模块介绍如下：

- 前序模块：无前序模块，**图像源**模块通常就是整个流程中的第一个模块，负责图像数据的输入。
- 后序模块：**图像源**对后序模块无特定要求，只需满足实际场景的需求即可。但一般情况下，该模块后序不连接 *多图融合* 模块。

注意事项

在流程中，后序模块会对**图像源**模块输出的图像数据进行进一步处理。如果图像出现过曝、过暗、模糊等问题，将可能导致视觉检测方案的精度下降，从而影响最终结果的准确性。

以下将介绍对于图像数据采集过程中的注意事项，以帮助您获取高质量的图像。这些注意事项旨在确保输入的图像数据具有良好的清晰度、对比度和细节呈现，从而显著提升方案的效果和精度。

降低光照影响

VM 在一定程度上能够抵抗环境光照变化的影响。当环境光变化较小时，方案仍可以有效工作。然而，当环境光发生较大变化时，图像中目标物体的细节可能会丢失，从而影响方案的整体效果。

实际业务场景中，阳光直射、照明不足、相机类型不合适以及相机参数未调节恰当等因素，都会对图像质量产生影响。此处列举了几种应对方法，您可参照这些方式来降低环境光对于图像质量的影响。

- 若遇到阳光直射等情况导致的图像过曝，可通过遮光等方式降低阳光对图像质量的影响。
- 若遇到无照明、低照明或强照明等情况，可通过增加或减少现场光源，并在流程中调用 ***光源*** 模块，调节和控制现场照明条件。光源设备的建立和连接方式具体请见 ***控制器管理*** 章节。
- 根据实际需求选择合适的相机类型，也可适当减少环境光对图像质量的一些负面影响。
  - 若仅需使用定位、测量或分析图像中的某些边缘和特征，则仅需使用黑白相机，例如方案中仅需调用 ***定位*** 、 ***测量*** 、 ***缺陷检测*** 等模块时；
  - 若同时还需要对彩色图像进行处理，则需要使用彩色相机，例如方案中需调用 ***颜色处理*** 等模块时。
- 当**图像源**选择**相机**时，可通过调整相机的曝光模式，使采集到的图像与实际工作环境保持一致。

说明

相机的曝光模式具体介绍以及如何调整请见 ***配置全局相机*** 。

保证图像清晰

在流程中，输入图像的清晰度对方案的整体效果有着较大的影响。图像越清晰，越容易识别出图像中的目标物体，减少错误和不确定性，从而保证方案的准确性和效果。

- 当**图像源**选择**本地图像**，可通过以下方式提高本地图像的质量。
  - 调节图像对比度。手动调整对比度参数，根据图像实际情况，适度增加或降低对比度，使图像亮部更亮、暗部更暗，突出原本不明显的细节。
  - 调节图像锐化。设置合适的锐化强度，避免过度锐化导致图像出现噪点或失真，在保留原有画质的基础上提升清晰度。
  - 调节图像增益。针对曝光不足的图像，提高增益可提升亮度。但增益数值越高，图像噪声也会随之增加，从而影响图像质量，因此需要根据实际情况适当提高增益。
- 当**图像源**选择**相机**或**二次开发**时，可通过以下方式提高实时采集的图像质量。
  - 灵活调节相机的工作距离、镜头焦距，使图像细节更清晰。
  - 调整相机曝光模式，不同模式下图像清晰度有所不同。自动曝光模式下，相机可根据实际场景自动匹配曝光参数；手动曝光模式下，可对曝光参数进行精细调整，从而减少噪点、提升对比度，获得清晰的图像。
    曝光程度不同，图像清晰度也会有所差异。曝光不足、正常曝光以及过度曝光三种状态下的图像清晰度对比如下图所示。

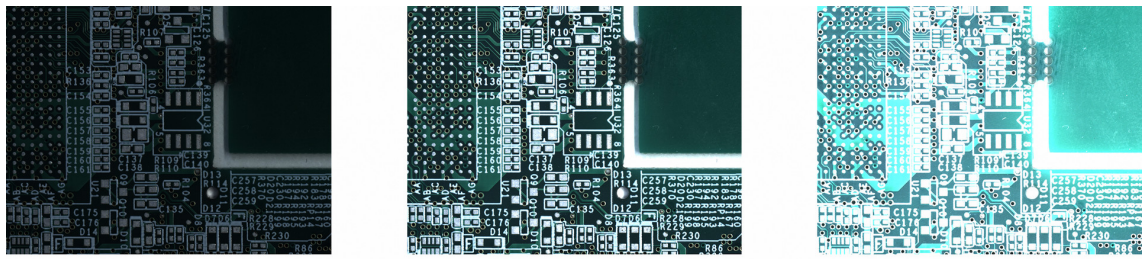


图 13-1 曝光效果对比图

参数配置

根据选择的**图像源**不同，配置的参数也会有所变化。**图像源**可选 、 或 三种。

本地图像

图像源选择**本地图像**时，可从本地图像中加载图像，也可通过图像显示区域设置加载的图像并进行相关操作。
添加的本地图像应尽可能模拟业务现场的实时图像，涵盖待检测目标可能出现的各种位姿与状态。
如何通过图像显示区域加载本地图像具体参见如下视频，详细介绍参见 *模块图像相关功能* 。
此时需设置以下参数：

基本参数

像素格式

可设置像素格式为 MONO8 或 RGB24。当导入的本地图像为黑白图像时，需选择 MONO8；当导入的本地图像为彩色图像时，需选择 RGB24。

方案存图

开启后，保存方案时将一同保存导入的本地图像，下次打开该方案且**图像源**选择**本地图像**时，将自动加载已保存的本地图像，无需重新导入。

若不开启，则保存方案时只保存本地图像的导入路径，下次打开该方案时将不再有本地图像，需要重新导入。

说明

方案存图最多可保存 120 张图像。

显示图像名称

开启后，在图像显示区域的左下角会显示出图像的名称。

SN 初始值

可通过 *模块结果* 中的 **SN 码**查看当前图像的帧号，一般作为结果标记使用。

- 使用自定义数值时，设置的数值为运行时首张图像的帧号值，后续图像的帧号以步进 1 逐步累加。
- 使用订阅时，每次运行都会读取订阅的数值。故显示结果为订阅数值的基础上加 1。

说明

一般情况下，不推荐使用该功能。当需要外部逻辑控制时，可考虑通过订阅全局变量实现。

图像缓存

用于设置加载本地图像时，缓存在内存中的图像数量，最多可缓存 5 张图像。

输出 Mono8

当**像素格式**选择 **RGB24** 时，可设置该参数。开启后，可在输出彩色图像的同时输出一张灰度图像。

拼接使能

若单张图像无法完全展示被测物体时，可导入多张包含被测物的图像，并启用**拼接使能**参数。

启用时，可根据设置的参数将上一张缓存图像裁剪后，再将两张图像（上一张缓存图像和本次执行的图像）在 Y 方向上依次拼接后输出。当没有上一张缓存图像或上一张缓存图像与当前图像的像素格式不一致时，会自动补黑填充。启用后，需设置以下参数：

起始高度

可设置上一张缓存图像在 Y 轴向下方向的裁剪起点，取值范围为 0~20000。

说明

起始高度与**拼接高度**之和需小于等于上一张缓存图像的原始图像高度。

拼接高度

可设置上一张缓存图像拼接时的图像高度，取值范围为 0~20000。

说明

起始高度与**拼接高度**之和需小于等于上一张缓存图像的原始图像高度。

以原始图像高度为 1024，**起始高度**设置为 100，**拼接高度**设置为 800 为例。上一张缓存图像裁剪后的高度为 800，拼接后的图像高度为 1824，拼接原理如下图所示。

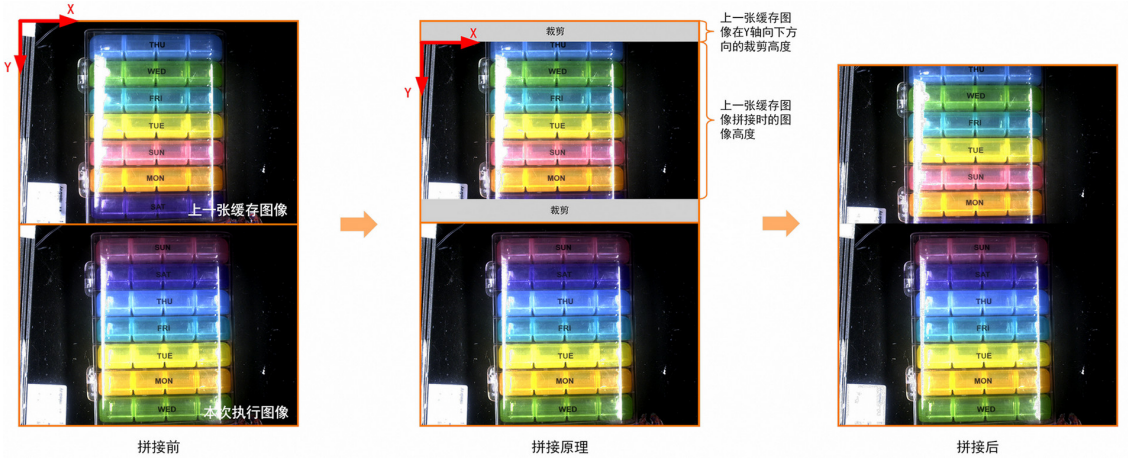


图 13-2 拼接原理

触发清空

启用后，当订阅的**触发变量**非 0，则清空上一张图像的缓存。

触发变量

一般订阅全局变量或局部变量。多流程场景中，也可订阅前序流程输出的数据。

说明

全局变量和局部变量相关详情，请参见 *全局/局部变量* 。

触发设置

自动切换

开启后，流程每次重新运行时，会自动切换到下一张图像。

最后一张停止

开启后，当使用本地图像连续运行方案时，运行完最后一张图像，方案会自动停止运行。

字符触发过滤

开启后，可通过外部通信控制图像源模块是否运行。此时需设置以下参数：

输入字符

选择输入字符的来源。

触发字符

未设置字符时传输任意字符均可触发图像源模块，设置字符后传输相应字符即可触发图像源模块。传输进来的字符与设置的字符不一致时，模块不被触发。

订阅图片文件夹

订阅文件夹

启用该功能时，从订阅的文件夹路径下获取图像。还需设置以下参数。

图像路径

设置订阅图像的文件夹路径。

更新缩略图

开启该功能后，当订阅文件夹中图像有增加或删除时，该模块会自动更新图像显示区域的缩略图。

触发流程执行

开启该功能后，当订阅文件夹中图像有增加或删除时，流程会自动执行。

相机

图像源选择**相机**时，可从已配置的相机中获取图像。此时需设置以下参数：

关联相机

选择关联的相机，配置步骤可参考下图，具体步骤请参考 *添加相机* 。



图 13-3 关联相机配置

控制曝光

该功能在配合脚本使用时，可按需实现用某种逻辑控制曝光值；除绑定脚本输出，也可绑定其他模块数据输出，但必须为 float 型数据。

控制增益

操作方式与**控制曝光**参数一致，但控制对象为相机的增益。

SN 初始值

可设置运行时首张图像的帧号值，后续图像的帧号以步进 1 逐步累加。可通过模块结果中的 SN 码查看当前图像的帧号。

输出 Mono8

开启后，可在输出彩色图像的同时输出一张灰度图像。

拼接使能

若单张图像无法完全展示被测物体时，可导入多张包含被测物的图像，并启用**拼接使能**参数。

启用时，可根据设置的参数将上一张缓存图像裁剪后，再将两张图像（上一张缓存图像和本次执行的图像）在 Y 方向上依次拼接后输出。当没有上一张缓存图像或上一张缓存图像与当前图像的像素格式不一致时，会自动补黑填充。启用后，需设置以下参数：

起始高度

可设置上一张缓存图像在 Y 轴向下方向的裁剪起点，取值范围为 0~20000。

说明

起始高度与**拼接高度**之和需小于等于上一张缓存图像的原始图像高度。

拼接高度

可设置上一张缓存图像拼接时的图像高度，取值范围为 0~20000。

说明

起始高度与**拼接高度**之和需小于等于上一张缓存图像的原始图像高度。

以原始图像高度为 1024，**起始高度**设置为 100，**拼接高度**设置为 800 为例。上一张缓存图像裁剪后的高度为 800，拼接后的图像高度为 1824，拼接原理如下图所示。

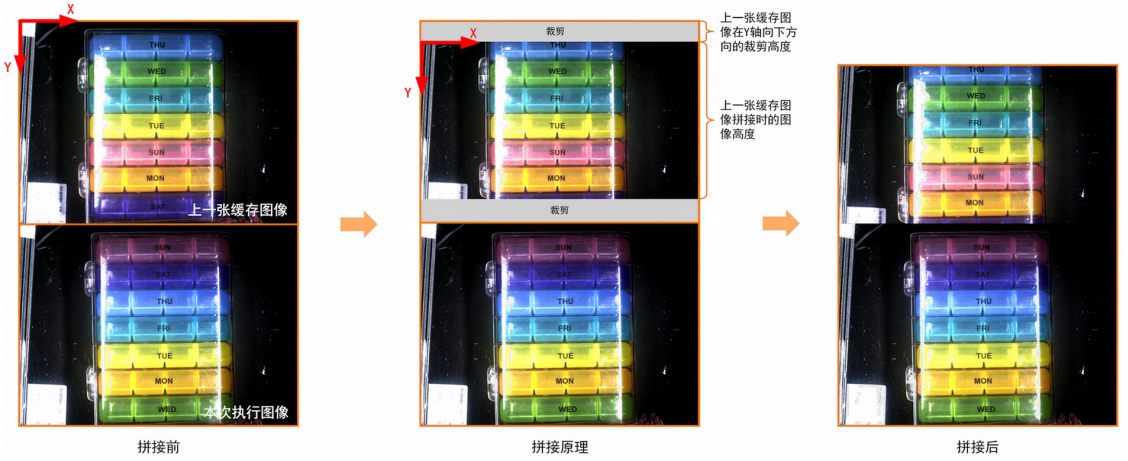


图 13-4 拼接原理

触发清空

启用后，当订阅的**触发变量**非 0，则清空上一张图像的缓存。

触发变量

一般订阅全局变量或局部变量。多流程场景中，也可订阅前序流程输出的数据。

说明

全局变量和局部变量相关详情，请参见 *全局/局部变量* 。

字符触发过滤

开启后，可通过外部通信控制图像源模块是否运行。此时需设置以下参数：

输入字符

选择输入字符的来源。

触发字符

未设置字符时传输任意字符均可触发图像源模块，设置字符后传输相应字符即可触发图像源模块。传输进来的字符与设置的字符不一致时，模块不被触发。

说明

连接相机前，建议先使用相机配套客户端完成图像调试，确保图像达到要求。

SDK

图像源选择 **SDK** 时，可通过二次开发获取图像。此时需设置以下参数：

SN 初始值

可设置运行时首张图像的帧号值，后续图像的帧号以步进 1 逐步累加。可通过模块结果中的 SN 码查看当前图像的帧号。

输出 Mono8

开启后，可在输出彩色图像的同时输出一张灰度图像。

拼接使能

若单张图像无法完全展示被测物体时，可导入多张包含被测物的图像，并启用**拼接使能**参数。

启用时，可根据设置的参数将上一张缓存图像裁剪后，再将两张图像（上一张缓存图像和本次执行的图像）在 Y 方向上依次拼接后输出。当没有上一张缓存图像或上一张缓存图像与当前图像的像素格式不一致时，会自动补黑填充。启用后，需设置以下参数：

起始高度

可设置上一张缓存图像在 Y 轴向下方向的裁剪起点，取值范围为 0~20000。

说明

起始高度与**拼接高度**之和需小于等于上一张缓存图像的原始图像高度。

拼接高度

可设置上一张缓存图像拼接时的图像高度，取值范围为 0~20000。

说明

起始高度与**拼接高度**之和需小于等于上一张缓存图像的原始图像高度。

以原始图像高度为 1024，**起始高度**设置为 100，**拼接高度**设置为 800 为例。上一张缓存图像裁剪后的高度为 800，拼接后的图像高度为 1824，拼接原理如下图所示。

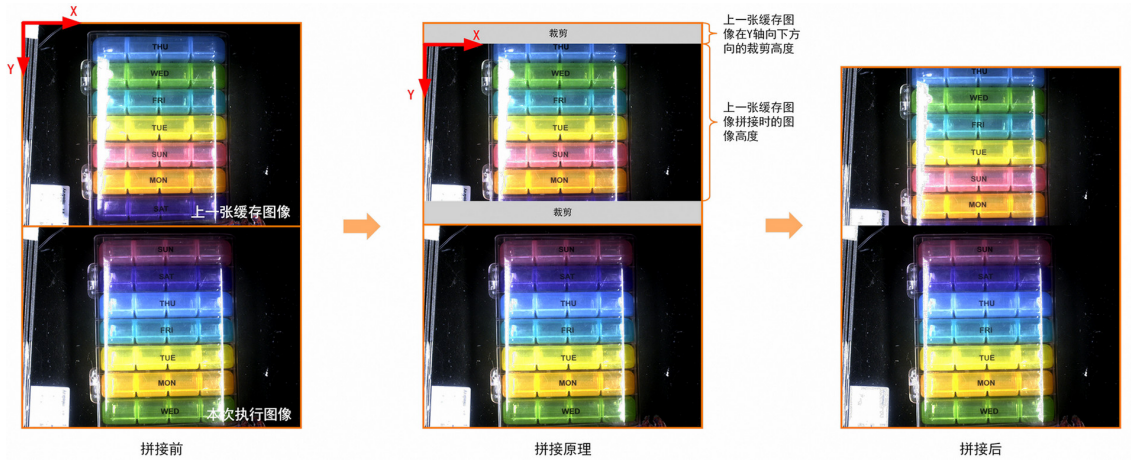


图 13-5 拼接原理

触发清空

启用后，当订阅的**触发变量**非 0，则清空上一张图像的缓存。

触发变量

一般订阅全局变量或局部变量。多流程场景中，也可订阅前序流程输出的数据。

说明

全局变量和局部变量相关详情，请参见 *全局/局部变量* 。

字符触发过滤

开启后，可通过外部通信控制图像源模块是否运行。此时需设置以下参数：

输入字符

选择输入字符的来源。

触发字符

未设置字符时传输任意字符均可触发图像源模块，设置字符后传输相应字符即可触发图像源模块。传输进来的字符与设置的字符不一致时，模块不被触发。

模块结果

*图像源*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

图像

图像数据

image 型，代表当前图像的数据。

图像宽度

int 型，代表当前图像的图像宽度。

图像高度

int 型，代表当前图像的图像高度。

图像像素格式

int 型，代表当前图像的像素格式的枚举值（Mono8：17301505；RGB24：35127316）。

图像*（*代表 1~3）

image 型，代表相机采集到的图像。该参数与*相机管理*中的**分时频闪拆图**参数相关。开启*分时频闪拆图*功能后，相机将依照设置的拆图数量输出对应图像。

说明

当*图像源*开启**输出 Mono8** 使能时，仅**图像**参数输出 Mono 8 格式图像，图像*（*代表 1~3）不输出。

图像数据

image 型，代表当前图像的数据。

图像宽度

int 型，代表当前图像的宽度。

图像高度

int 型，代表当前图像的高度。

图像像素格式

int 型，代表当前图像像素格式对应的枚举值。其中，“17301505”对应 Mono 8 格式；“35127316”对应 RGB 24 格式。

灰度图像

灰度图像数据

image 型，代表当前图像的灰度图像数据。

灰度图像宽度

int 型，代表当前图像的灰度图像宽度。

灰度图像高度

int 型，代表当前图像的灰度图像高度。

灰度图像像素格式

int 型，代表当前图像的灰度图像像素格式的枚举值。

当前图像路径

string 型，代表当前图像的路径。

当前图像名称

string 型，代表当前图像的名称。

帧号

int 型，代表相机本次采图时当前图像对应的帧序号。

丢帧数

int 型，代表相机本次采图时丢弃的图像帧数。

丢包数

int 型，代表相机本次采图时丢弃的图像数据包数。

相机获取失败次数

int 型，代表连接相机时获取失败的次数。

SN 码

int 型，代表设置运行后首张图像的帧号，后续图像帧号步进 1 累加。

13.1.2 多图采集

多图采集模块可用于获取多张本地图像数据，或通过相机和光源拍摄多张不同角度、不同亮度的图像。

使用方法

以下对该模块的前后序模块和采集流程进行介绍。

多图采集模块通常是整个流程中的第一个模块，负责多张图像数据的输入。该模块对后序模块无特定要求，但主要搭配 ***多图融合*** 模块使用。

该模块通过调节相机以及光源的参数，或改变光源的位置，可实现多张不同角度、不同亮度的图像的拍摄。根据设置的图像配置参数进行输出图像，该模块的多图流程如下图所示。

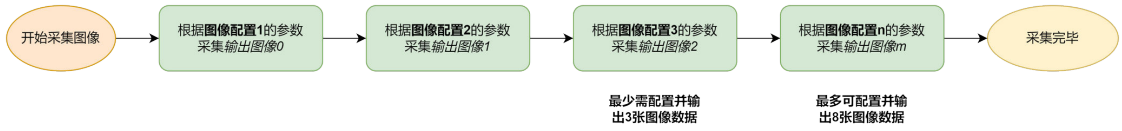


图 13-6 多图采集功能示意图

注意事项

在流程中，**多图融合**模块会对**多图采集**模块输出的 3 至 8 张不同角度光线的图像数据进行融合。如果图像出现过曝、过暗、模糊等问题，将可能导致视觉检测方案的精度下降，从而影响最终结果的准确性。

该模块对于图像数据采集过程中的注意事项与**图像源**模块一致，详情参见 ***注意事项*** 。

参数配置

根据选择的**图像源**不同，配置的参数也会有所变化。**图像源**可选 或 。

本地图像

图像源选择**本地图像**时，需选择图像路径并导入本地图像。此时需设置以下参数：

输入配置

取图数量

可设置最终输出的图像数量，最少需输出 3 张图像数据，最多可输出 8 张图像数据。

图像路径

可选择图像文件的路径。

图像配置 n

可通过改变光源的位置，输出在不同角度打光下的多张图像数据。**图像配置**的数量与**取图数量**相关，若**取图数量**设置为 3，则需设置 3 组**图像配置**参数。

分布角

指在俯视视角下，图像中心指向右侧的方向与投影到水平成像平面中的光线方向之间的夹角，可帮助确定光源相对于图像中心的位置。

- 当分布角为 0°时，表示光线是从图像的右侧方向射入的；
- 当分布角为 90°时，表示光线是从图像的顶部方向射入的；
- 当分布角为 180°时，表示光线是从图像的左侧方向射入的。

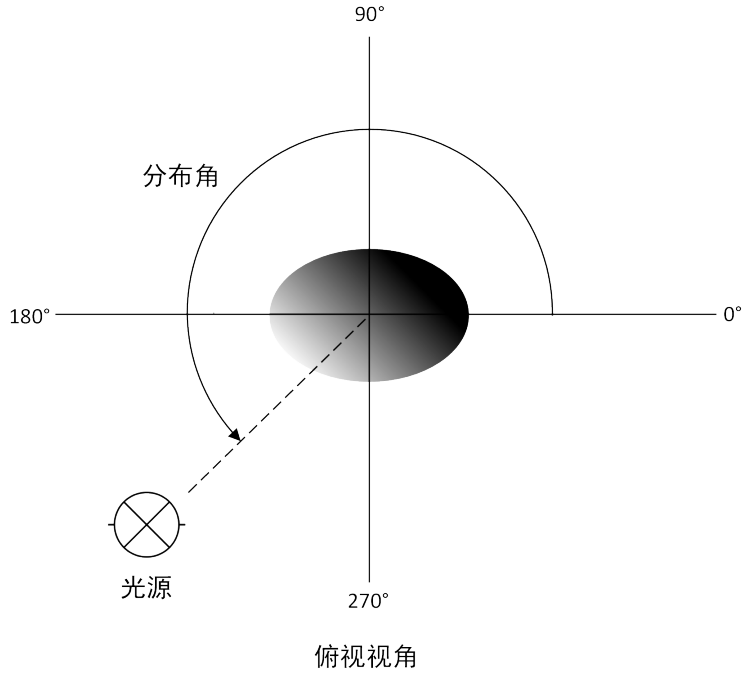


图 13-7 分布角

照射角

指在侧视视角下，相机的光轴与入射光线之间的夹角，相机的光轴是指从相机镜头中心到成像面中心的一条直线。

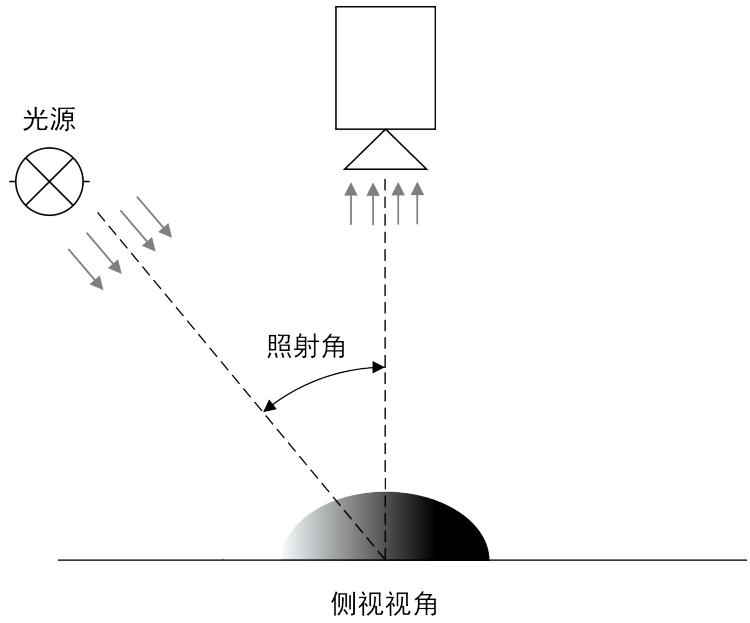


图 13-8 照射角

相机

图像源选择**相机**时，需通过快捷工具条中相机管理创建全局相机，详情参见 *相机管理* 章节。此时需设置以下参数：

输入配置

关联相机

可选择关联对应的全局相机。

取图间隔

可设置相邻两次取图的时间间隔，单位为 ms。

图像配置 n

可通过调节相机以及光源的参数，或改变光源的位置，输出在不同角度打光、不同亮度下的多张图像数据。输出的图像数量与设置的**图像配置**参数的组数相关，若设置 3 组**图像配置**参数，则输出 3 张图像。

相机曝光

可通过增加曝光值来提高图像亮度。

实际业务场景中，若需调整图像亮度，建议优先调节**相机曝光**。

相机增益

实际业务场景中，当调节**相机曝光**后仍无法满足需求，可考虑在不增加曝光值的情况下，通过调节**相机增益**来调整图像亮度。

随着增益数值越高，图像亮度也越高，但同时图像噪声也会增加，对图像质量有所影响。

光源设备

关联控制管理器中已添加的光源设备。

触发时间

可设置光源触发后的持续点亮时间。

光源通道

可选择对应的光源通道。

光源亮度

可控制对应通道的对应光源亮。

分布角

指在俯视视角下，图像中心指向右侧的方向与投影到水平成像平面中的光线方向之间的夹角，可帮助确定光源相对于图像中心的位置。

- 当分布角为 0°时，表示光线是从图像的右侧方向射入的；
- 当分布角为 90°时，表示光线是从图像的顶部方向射入的；
- 当分布角为 180°时，表示光线是从图像的左侧方向射入的。

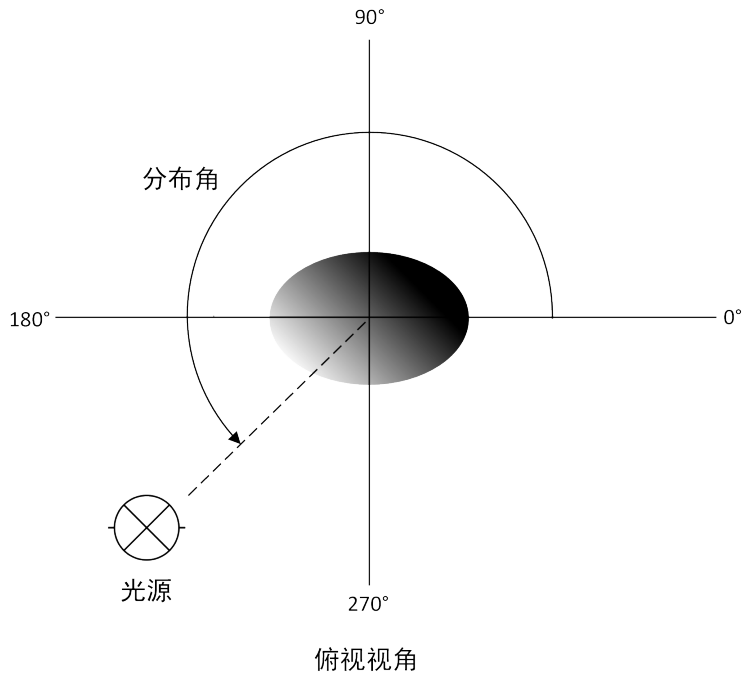


图 13-9 分布角

照射角

指在侧视视角下，相机的光轴与入射光线之间的夹角，相机的光轴是指从相机镜头中心到成像面中心的一条直线。

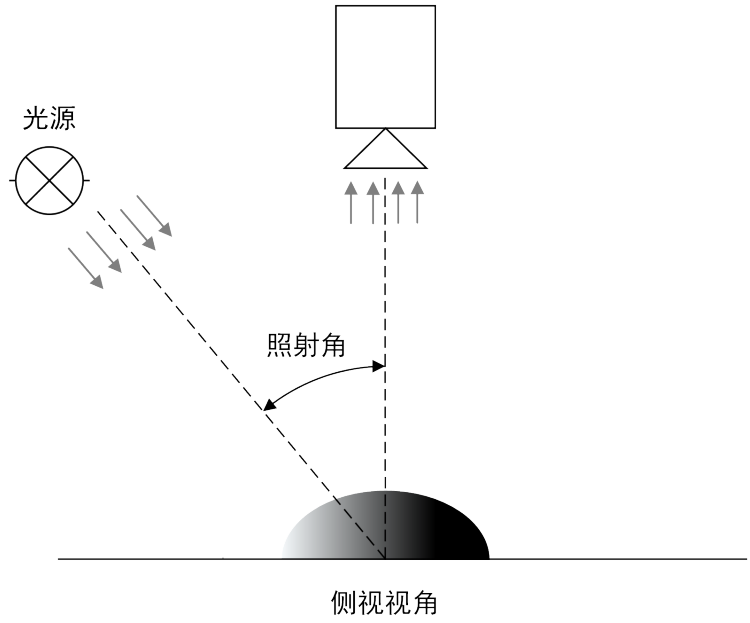


图 13-10 照射角

模块结果

*多图采集*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

输出图像数据

图像数量

int 型，代表多图采集单次执行输出图像数量（3-8 张）。

输出图像*（*代表 0~7）

输出图像数据

image 型，代表相机输出的图像数据。

图像数量

int 型，代表多图采集单次执行输出图像的数量（3~8 张）。

输出图像*（*代表 0~7）

image 型，代表相机输出的图像。

输出图像*数据

image 型，代表输出图像的图像数据。

输出图像*宽度

int 型，代表输出图像的宽度。

输出图像*高度

int 型，代表输出图像的高度。

输出图像*格式

int 型，代表输出图像的像素格式。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出图像*分布角

float 型，代表每张输出图像的分布角。

输出图像*照射角

float 型，代表每张输出图像的照射角。

输出图像*宽度

int 型，代表输出图像的宽度。

输出图像*高度

int 型，代表输出图像的高度。

输出图像*格式

int 型，代表输出图像的像素格式。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出图像*分布角（*代表 0-7）

float 型，代表每张图片拍摄时光源在水平方向上的分布角度。

输出图像*照射角（*代表 0-7）

float 型，代表每张图片拍摄时光源在垂直方向上的角度。

帧号

int 型，代表相机本次采图时当前图像对应的帧序号。

丢帧数

int 型，代表相机本次采图时丢弃的图像帧数。

丢包数

int 型，代表相机本次采图时丢弃的图像数据包数。

相机获取失败次数

int 型，代表连接相机时获取失败的次数。

13.1.3 输出图像

输出图像模块可对全局相机图像、本地图像或者图像处理模块处理过的图像进行存储。

说明

图像处理下各模块具体介绍请见 *图像处理* 。

参数配置

该模块的参数分为**基本参数**、**渲染参数**和**渲染设置**。

基本参数

输入源

选择存图的来源，可选择方案中的*图像源*图像或其它模块处理后的图像。

输出使能

开启时，输出图像；关闭时，不输出图像，可节省耗时。

存图使能

默认关闭状态，此时*输出图像*模块仅实现输出图像的功能；启用该功能后，可通过设置如下参数，使该模块输出图像的同时存储图像。

触发保存

默认关闭状态，此时，只需开启**保存渲染图/原图**，无需触发条件即可自动保存图像；启用该功能后，不仅需要开启**保存渲染图/原图**，还需设置如下参数，仅当如下触发条件满足时才可进行存图。

触发变量

一般绑定条件检测结果，配合**保存条件**进行存图。

保存条件

可选**全部保存**、**OK 时保存**、**NG 时保存**和**不保存**。

- **全部保存**：存储全部图像。
- **OK 时保存**：当**触发变量**参数结果为 OK 时触发保存。保存时，渲染图和（或）原图文件名中会包含“OK”字符。
- **NG 时保存**：当**触发变量**参数结果为 NG 时触发保存。保存时，渲染图和（或）原图文件名中会包含“NG”字符。
- **不保存**：即不保存图像。

说明

即使**保存渲染图/原图**功能开启，也不保存图像。

生成目录

使能后可根据月份以及日期创建文件夹，图片保存在创建的日期文件夹中。

存图间隔

可设置触发存图时缓存的图像张数，不能大于图像缓存队列上限，最多可设置 10 张图像。

保存渲染图/原图

使能后，可在**渲染图/原图路径**下保存渲染图/原图。

渲染图/原图路径

可查看或设置存储渲染图/原图的路径。仅在关闭**触发保存**，并开启**保存渲染图/原图**时可设置，即将所有渲染图保存在同一路径下，将所有原图保存在同一路径下。

说明

Windows 下默认系统路径的限制长度为 260，渲染图/原图的路径长度最大不超过 255。若超过最大长度，则图片会保存失败。

渲染 OK/NG 路径

可查看或设置**触发变量**参数结果为 OK/NG 时渲染图的存储路径。

仅在开启**触发保存**和**保存渲染图**时可设置，即将所有包含“OK”字符的渲染图保存在同一路径下，将所有包含“NG”字符的渲染图保存在同一路径下。

说明

Windows 下默认系统路径的限制长度为 260，渲染 OK/NG 的路径长度最大不超过 255。若超过最大长度，则图片会保存失败。

原图 OK/NG 路径

可查看或设置**触发变量**参数结果为 OK/NG 时原图的存储路径。

仅在开启**触发保存**和**保存原图**时可设置，即将所有包含“OK”字符的原图保存在同一路径下，将所有包含“NG”字符的原图保存在同一路径下。

说明

Windows 下默认系统路径的限制长度为 260，原图 OK/NG 的路径长度最大不超过 255。若超过最大长度，则图片会保存失败。

渲染图/原图命名

可设置渲染图/原图的命名规则，仅在开启**保存渲染图/原图**时可设置。文件名前半部分可自定义或从方案中订阅，后半部分可选**序号**、**时间**和**无**。

说明

保存渲染图或原图时，文件名前半部分必须设置，否则图像将无法保存。

渲染图/原图缓存

可设置存图时缓存队列容纳的最大图片数量，最多可缓存 20 张渲染图/原图。

存储方式

设置达到最大存储数量或所在磁盘空间不足时对图片处理的方式，可选择**覆盖存储**、**停止存储**两种存储方式。

磁盘剩余空间

设置存图目标磁盘的剩余空间。若磁盘剩余空间达到设置的数值，则会按照设置的**存储方式**进行存储 ；若磁盘剩余空间小于设置的数值，且已配置**存储方式**为**覆盖存储**，则会尝试删除旧的存图文件以释放磁盘空间。

存储单位

可选 MB 或 GB。

最大保存天数

设置自动删除存储图像的时间，存储图像最多可保留 500 天 。

保存格式

可选 BMP、JPEG 或 PNG 格式。保存格式为 JPEG 时，可以修改**图片压缩质量**。

图片压缩质量

反映图片压缩后的质量好坏。该参数值设置越大，压缩后的图片品质越好；反之，参数值越小，压缩后的图片品质越低。

启用 FTP

启用后，可将图像保存在设置的 FTP 服务器上。

需根据实际情况设置**服务器 IP**、**服务器端口**、**用户名**、**密码**、**编码格式**和 **FTP 路径**，并启用**连接**。

FTP 连接成功后，图像将保存在设置的 **FTP 路径**中。若本地路径存在缓存图像，这些图像将被上传到 **FTP 路径**中，并从本地路径中删除。若 FTP 连接失败，图像将直接保存在本地，并且不会显示上传失败的提示。

说明

- 若 **FTP 路径**不存在，则自动创建该路径。
- 必须至少启用一个**保存渲染图**或**保存原图**，并设置**渲染图/原图命名**才可保存图像。
- 单击**连接**参数右侧的 可校验 FTP 服务器的连接状态。
- 若连接失败，可从两方面排查：(1) 以上参数是否和 FTP 服务器对应参数保持一致；(2) FTP 服务器能否正常通讯。

像素格式

可设置像素格式为 RGB24 或 MONO8。

渲染参数

图形倍率类型

- **原图尺寸**：图像、字体以及线宽按照原图尺寸、原线宽以及原字体大小存储。
- **界面尺寸**：图像保持原始尺寸不变，线宽和字体大小则根据原图宽度除以 500 个像素点的倍率进行等比缩放并存储。
- **自定义倍率**：存图时根据设置的**线宽倍率**和**字宽倍率**调整线宽和字宽。

线型

可选三种类型：**4 连通线**、**8 连通线**和**抗锯齿线**。

渲染设置

内容

可选择存储在硬盘中图像上渲染的文本信息。

颜色

可更改渲染信息的颜色。

字号

可更改渲染信息字体的大小。

位置 X/Y

可更改渲染信息横纵坐标的位置。

前项存储设置

可将前序模块的输出结果显示在最终的输出图像上。

通过单击 添加。添加后，单击 订阅前序模块，再单击 设置是否将前序模块中的输出结果显示在最终的输出图像上。 表示显示该结果信息， 表示不显示该结果信息。

说明

- 可订阅多个模块，并将多个输出结果信息显示在最终的输出图像上。
- 订阅的前序模块需支持设置结果显示参数，结果显示参数详情请参见 *结果显示* 。

模块结果

*输出图像*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

本地原图保存路径

string 型，代表原图在本地工控机上的保存路径。

原图名称

string 型，代表原图的文件名称。

本地渲染图保存路径

string 型，代表渲染图在本地工控机上的保存路径。

渲染图名称

string 型，代表渲染图的文件名称。

大文件路径

string 型，代表大文件所存储的路径。在开启大文件存图时可用。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.1.4 缓存图像

缓存图像模块可用于方案功能调试，特别是在某些样本图像出现误判时，通过缓存图像功能，可将图像临时存储，以便在后续调试过程中快速绑定和重新处理，而不需要每次都重新加载原始图像。

模块原理

该功能使能时流程运行一次可缓存一张图像，最多缓存 15 张。按照时间顺序，缓存区存满后，最新缓存图像从下方进入缓存区，并将缓存区内最早保存的图像覆盖掉，其余图像位置上升一位。以缓存 5 张图为例，覆盖逻辑如下图所示。后续处理模块的数据源可以绑定 15 张缓存图像的任意一张，便于追溯方案调试过程。



图 13-11 缓存图像覆盖逻辑

参数配置

输入源

需要订阅前序模块输出的图像数据。

缓存使能

0 代表不使能，1 代表使能。数据订阅仅限 int 型数据，当没有数据绑定时默认使能。

缓存数量

缓存图片的数量，可直接自定义输入也可绑定，仅限 int 型数据。

模块结果

*缓存图像*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像*（*代表 0~14）

输出图像*数据

image 型，代表输出图像的图像数据。

输出图像*宽度

int 型，代表输出图像的宽度。

输出图像*高度

int 型，代表输出图像的高度。

输出图像*格式

int 型，代表输出图像的像素格式。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.1.5 光源

光源模块可控制和调节现场照明条件，例如控制光源常亮或光源频闪等。

使用方法

如何建立和连接光源设备可参照 *控制器管理* 章节，**光源**模块创建成功后可直接在流程中进行调用，且对前后序模块无特定要求。

说明

VC3000、*VC3000(IO 扩展板)*、*VT2000* 和 *VT3000* 品牌不支持接入*光源*模块。

参数配置

光源设备

选择需关联的光源设备。

输出类型

设置前序模块触发光源时的状态条件。可选择 **OK 时输出**或 **NG 时输出**，即当前序模块状态为 OK 或 NG 时触发光源。

触发字符

设置前序模块触发光源的状态条件，需根据选择的**输出类型**手动输入对应的 OK 或 NG。

说明

目前仅支持手动输入 OK 或 NG 字符，且需和**输出类型**一致。若输入或订阅其他字符，光源均无法触发。

触发时间(ms)

用于设置触发后光源常亮的时间，范围为-1 ~ 65535 ms。设置为-1 时光源常亮，设置大于 0 时光源频闪。

说明

对光源模块设置**触发时间**后，触发一次后在所设置的触发时间内不要触发第二次，否则会导致光源提前关闭。

输入类型

可选*输入*或*订阅*。

通道*使能

开启对应通道光源使能。

说明

通道数量由对应光源设备的光源参数决定，可在控制器管理处查看，详情请参见 ***光源参数*** 。

通道*亮度

设置对应通道光源亮度，0 表示光源关闭，值越大，光源越亮 。

光源状态*

设置对应通道光源状态，包含常亮和常灭两种状态，常亮时可调节光源亮度，常灭时光源始终处于熄灭状态。光源设置触发时，常亮状态下触发可灭，常灭状态下触发可亮。

说明

光源设备选择 VC2000、VC3000(Light 扩展板)、VC3000H/X、VC4000、VB2200 和 VB2230 时，可设置此参数。

通道*沿定义

设置对应通道触发光源的信号类型。可选**上升沿**和**下降沿**。

说明

光源设备选择 VC3000(Light 扩展板)、VC3000H/X 和 VC4000 时，可设置此参数。

持续时间(ms)

设置对应通道光源触发后亮/灭的持续时长。

说明

光源设备选择 VC3000(Light 扩展板)、VC3000H/X 和 VC4000 时，可设置此参数。

模块结果

*光源*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

通道*使能（*代表 1~6）

bool 型，true 代表通道使能，false 代表不使能。

通道*亮度（*代表 1~6）

int 型，代表光源通道的亮度，根据不同类型范围不同，视觉控制器范围为 0~100，光源控制器范围为 0~255。

通道*光源状态（*代表 1~4）

int 型，1 代表处于常灭状态的光源经过触发后亮起，2 代表处于常亮状态的光源经过触发后熄灭。

通道*沿定义（*代表 1~4）

int 型，1 代表上升沿，2 代表下降沿。

通道*持续时间（*代表 1~4）

int 型，代表光源触发后的持续时间（0~65535）。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.2 定位

定位相关模块主要用于定位或检测图像中的某些特征。

不同定位模块的适用场景有所差异，以下仅介绍*轮廓匹配*、*高精度匹配*、*快速匹配*和*灰度匹配*模块的适用场景和模块特点。

表 13-1 模块适用场景

模块	特点	适用场景
轮廓匹配	建模时提取物体的较精细轮廓，**匹配效率与定位精度均较好**	• 图像中待匹配目标的轮廓点数量较充足且稳定 • 对定位精度和匹配效率要求均较高

模块	特点	适用场景
		说明 *轮廓匹配*为新一代的模板匹配，相比*高精度匹配*和*快速匹配*，*轮廓匹配*兼具匹配效率与定位精度。建议优先使用**轮廓匹配**。
高精度匹配	建模时提取物体的精细轮廓，**定位精度较高，但匹配效率较低**	• 图像中待匹配目标的轮廓点数量充足且稳定 • 对定位精度要求较高但对匹配效率要求不高
快速匹配	建模时提取物体的粗糙轮廓，定位精度较低。**对存在形变的目标，匹配效果更好**	• 图像中待匹配目标的轮廓点数量较充足且稳定 • **匹配目标存在一定形变** • 对定位精度要求不高 例如有无判断、方向判断、产品计数等场景
灰度匹配	建模时提取物体在图像中的灰度值分布	图像中的待匹配目标无明显且稳定的轮廓点，但灰度值分布相对稳定。例如对比度较低和受随机噪声干扰的场景

13.2.1 轮廓匹配

*轮廓匹配*可提取 ROI 中的边缘点特征，并结合特征信息创建模板，在图像中搜索与模板相似的目标，实现定位、计数和判断有无等业务需求。

模块原理

*轮廓匹配*基于边缘点获取目标特征模板，并对目标物体进行查找和匹配。

算法工作流

该模块的算法工作流大致包括以下步骤：

1. 算法从输入图像中提取目标的边缘特征。
2. 算法基于提取到的边缘轮廓创建特征模板。
3. 算法基于目标特征模板在图像中搜索与模板匹配的目标。

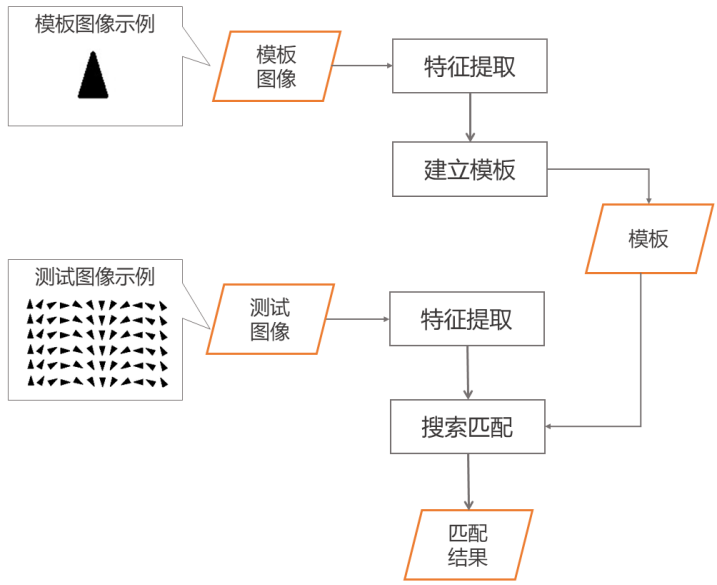


图 13-12 模板匹配算法工作流

算法公式

基于边缘点的模板匹配算法中涉及的关键公式如下：

$$\underset{x,y,\theta,scaleX,scaleY}{\operatorname{argmax}} \quad S(F_m, F_s; x, y, \theta, scaleX, scaleY)$$

图 13-13 基于边缘点的模板匹配算法公式

该算法公式中涉及的参数如下：

- x：模板在 x 轴的平移量
- y：模板在 y 轴的平移量
- θ：模板的旋转角度
- scaleX：模板在 x 轴的伸缩尺度
- scaleY：模板在 y 轴的伸缩尺度

在上述五个参数的共同作用下，模板会缩放、旋转并移动至待匹配图像中的某一位置。此时，待匹配图像的特征为 F_s，模板特征为 F_m，而函数 $S=(F_m, F_s)$则度量了模板特征与待匹配图像特征的相似度得分。算法通过每一组特定的(x , y , θ , scaleX , scaleY)均能计算出一个相似度得分。遍历(x , y , θ , scaleX, scaleY)的所有取值后，算法会选取既满足相似度得分阈值且得分最高的 n 个结果作为输出。

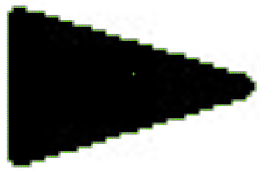
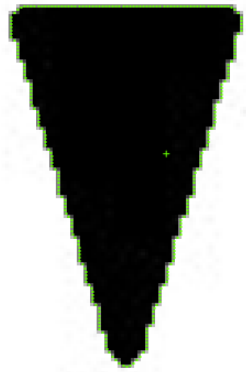
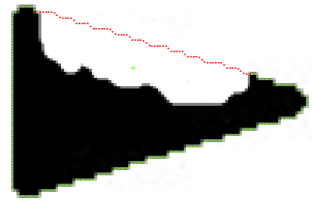
说明

“n”为允许查找到的最大目标数量。

F_m、F_s 和 $S=(F_m, F_s)$可分别做如下简化理解：

- F_m：模板图像中的显著边缘轮廓点(绿色的点)
- F_s：匹配图像对应区域的显著边缘轮廓点
- $S=(F_m, F_s)$：模板上的特征轮廓点经过特定的拉伸、旋转、平移后，与目标图像中的边缘轮廓点相匹配的比例。该比例值越高，相似度得分越高。
  以下表中的图像为例，在相似度得分高的图像中，所有模板特征点均在匹配目标中找到了对应边缘特征点，故其得分为 1 分；在相似度得分较低的图像中，模板特征点中约有 30% 未在匹配目标中找到对应边缘特征点，故其得分为 0.71 分。

表 13-2 相似度得分示例

模板图像	高相似度图像（得分：1）	较低相似度图像（得分：0.71）
		

匹配加速机制

在模块实际运行过程中，算法会对待匹配的图像做如下处理，以加速匹配过程。

1. 对待匹配图像建立图像金字塔。
2. 在图像金字塔中，至顶向下逐层进行目标搜索，直至找到满足条件的目标。

下图以在含有 R 和 P 字样的图像中搜索字母 R 的过程为例，介绍算法实现匹配加速的过程。



图 13-14 图像金字塔的示例

该示例中的匹配加速流程如下：

1. 建立图像金字塔。
2. 在图像金字塔的顶层找到 R 和 P 这两个候选目标。
3. 在图像金字塔的下一层中对 R 和 P 进行进一步的判定并调整其位姿。
4. 删除错误的候选目标 P。
5. 在图像金字塔的底层对候选结果 R 进行最终的位姿调整，并输出查找结果。

算法在创建特征模板的过程中，需使用**速度尺度**和**特征尺度**参数，定义图像金字塔大小。

- **速度尺度**：定义图像金字塔顶层的大小，用于调节匹配搜索的速率。可在配置特征模板时调整该参数取值。具体参数定义参见*特征模板*中的 ***速度尺度*** 。
- **特征尺度**：定义图像金字塔底层的大小，用于调节匹配算法的抗形变能力。可在配置特征模板时调整该参数取值。具体参数定义参见*特征模板*中的 ***特征尺度*** 。

使用方法

前后序模块

在流程中，*轮廓匹配*通常需要和 *位置修正* 、 *条件检测* 、*脚本*、 *测量* 类模块等其他模块搭配使用，可满足部件定位、方向识别、角度检测等多种检测场景的需求。

使用示例

下图展示了如何通过*轮廓匹配*获取图像点坐标，并使用 *N 点标定*模块实现 12 点标定（9 次平移，3 次旋转）的过程，可用于实现工件定位。

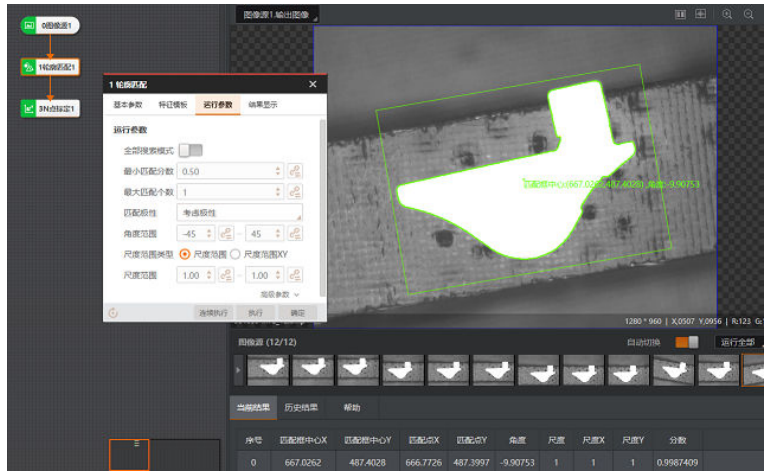


图 13-15 基于轮廓匹配进行 N 点标定

特征模板

*轮廓匹配*模块基于特征模板进行目标的定位和匹配。运行该模块前，需先配置特征模板。

操作步骤

1. 双击*轮廓匹配*模块，打开模块配置窗口。
2. 选择*特征模板*页签。



3. 单击*创建*，打开*模板配置*窗口。

说明

最多仅支持创建 36 个特征模板。

如果 *图像源* 模块已正常运行，则该窗口默认显示当前*图像源*模块采集的图像。如需基于其他图像进行模板配置，可单击*选择其他图像*。

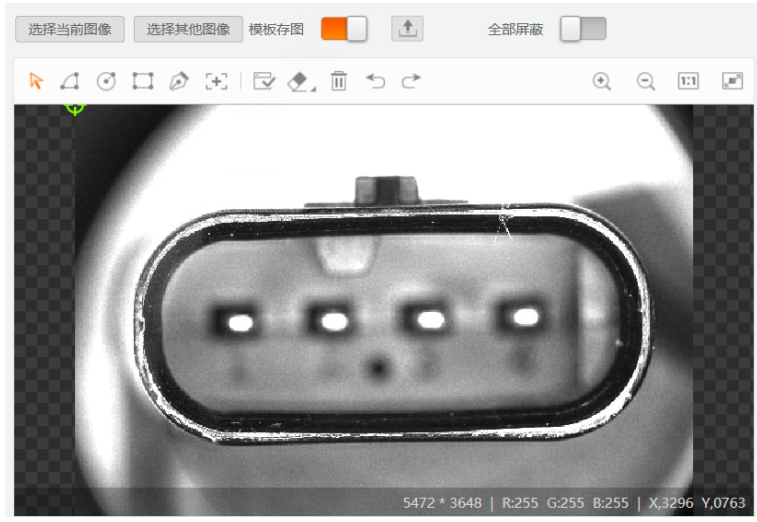


图 13-16 模板配置窗口

4. 在图像上特征明显的区域绘制掩膜，算法将在掩膜区域内提取图像特征。

说明

- 具体如何绘制掩膜区域参见 *模块中模型配置的掩膜区域* 章节。
- 支持绘制多个不同形状的掩膜。

5. **可选操作**: 进行如下可选操作：

删除掩膜	右击掩膜，并在弹出的右键菜单上单击*删除*。
复制掩膜	右击掩膜，并在弹出的右键菜单上单击*复制*。
调整掩膜位置	单击选中掩膜，并拖拽调整其位置。
选择匹配中心	单击 ，并在图像上的合适位置单击，将其设置为模板的匹配中心。
锁定匹配中心	选择匹配中心后，可在*模板配置*窗口的下方勾选*锁定*，锁定匹配中心。锁定后，该窗口工具栏上不再显示“选择匹配中心”图标 。
移动图像	单击 后，将光标移动至图像上并拖拽图像进行移动。

缩放图像	单击 / 缩放图像；或将光标置于图像区域，并滚动鼠标滚轴缩放图像。
清空掩膜	单击 清空掩膜。
撤销上一步操作	单击 撤销上一步操作。
恢复至撤销前	单击 取消撤销，恢复至撤销前的状态。
图像自适应	单击 将图像大小设置为自适应模式。
全屏显示	单击 将图像全屏显示。
绘制 ROI	开启***全部屏蔽***，将已绘制的掩膜全部屏蔽和匹配中心。此时可单击 ，并在图像通过“刷”的方式绘制 ROI（具体见下图）。



图 13-17 刷出 ROI

6. 单击 生成模板。



图 13-18 轮廓匹配模板效果示例

7. **可选操作**: 如果模板中提取了不必要的轮廓点，单击 ，并将光标悬浮至这些轮廓点将其擦除。

   可单击 右下角的三角形图标，并在弹出的浮窗滑动滑块调整“橡皮擦”的大小。

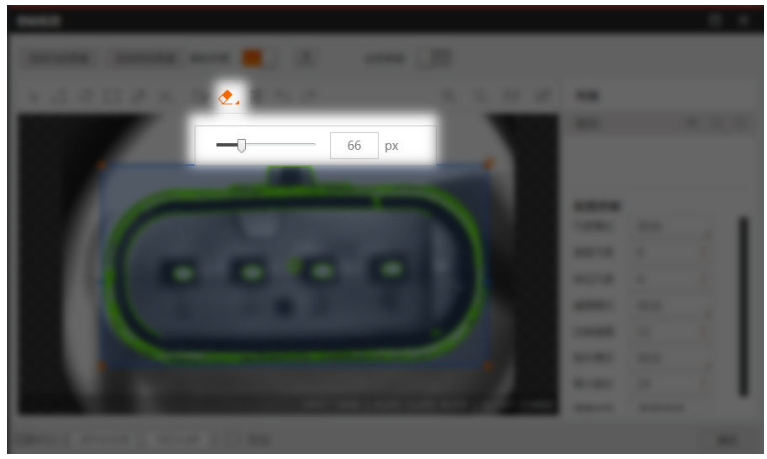


图 13-19 调整“橡皮擦”大小

8. **可选操作**: 调整模板的配置参数，并重新生成模板，直至模板效果满足需求。

   模板配置参数如下：

   尺度模式

   设置**速度尺度**的配置模式，可选**自动**或**手动**模式。默认为**自动**。如果**自动**模式能满足特征提取需求则无需调节；如果不能满足要求，可切换为**手动**模式并手动调整**速度尺度**。

   速度尺度

   设置特征匹配速度。该值越大表示图像金字塔顶层的图像越小，相应提取的边缘点就越稀疏，进而加快特征匹配速度，但是过大的取值可能会导致漏检。取值范围为 1~20。

特征尺度

表示图像金字塔底层的大小，调整该值可调节匹配算法的抗形变能力。“抗形变能力”指算法在匹配目标存在一定形变的情况下也能匹配到目标的能力。

该值越大表示图像金字塔底层的搜索图像越小，提取的特征点越稀疏，但抗形变能力越强，匹配速度越快。取值过大可能导致误检，定位精度也会有一定程度下降。

说明

该参数取值不能大于**速度尺度**取值。

当取值为 1 时最精细，一般调节后会使轮廓点数量发生较大变化，如下图所示



图 13-20 特征尺度配置效果

阈值模式

设置**边缘阈值**的调整模式，可选**自动**或**手动**，默认为**自动**。如果**自动**模式能满足需求则无需手动调整**边缘阈值**；如果不能满足，可切换至**手动**模式并调整**边缘阈值**。

边缘阈值

此处“边缘阈值”指边缘特征点和背景的灰度值差距。边缘阈值越大，被淘汰的特征点越多。取值范围为 1~255。

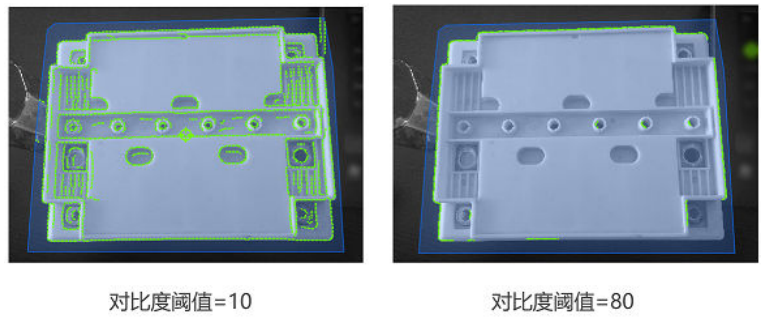


图 13-21 边缘阈值配置效果

链长模式

可选自动和手动两种模式，主要是对最小链长进行调整。原则是自动模式能满足需求则不进行调节，自动模式不能满足要求再切换至手动模式并调整最小链长。

最小链长

设置建模时，由特征点组成的特征链的最小长度。只有链长超过最小链长值时，特征链才会被保留。

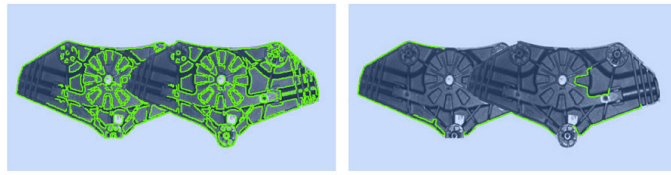


图 13-22 最小链长配置效果

建模类型

设置模板创建类型。可选**原图建模**和**掩膜图建模**。

9. **可选操作**: 在掩膜列表进行如下可选操作。

隐藏/显示掩膜	单击 / 隐藏或显示掩膜。
屏蔽掩膜	单击 右下角的三角形图标，并单击 屏蔽对应掩膜，再单击 重新生成模板，可屏蔽该掩膜中的特征点提取。
增强匹配	（仅*高精度匹配*支持） 如果对特定掩膜内的匹配精度有较高要求，可单击 右下角的三角形图标，并单击 开启**增强匹配**。在相同的**最小匹配分数**下，开启**增强匹配**的掩膜对于模板和检测目标的相似度要求更高，因此具备更高的匹配精度。 **说明** **最小匹配分数**相关详情，请参见 *参数配置* 中对该参数的说明。
删除掩膜	单击 删除对应的掩膜。



图 13-23 轮廓匹配掩膜列表

10. 单击 将模板保存至本地路径。

    模板图像默认保存至模板文件中。如无需保存，可关闭*模板存图*。

11. 单击*模板配置*窗口右下角的*确定*，完成模板配置。界面将自动跳转至*特征模板*页签，并展示已创建的特征模板。



图 13-24 轮廓匹配特征模板列表

12. 可选操作: 在*特征模板*页签进行如下可选操作。

模板排序　单击 ↑ / ↓ 对模板进行排序。

在未开启**全部搜索模式**时，算法将优先采用排序更靠前的模板与目标图像进行匹配。如果找到符合**最小匹配分数**要求的目标，则停止匹配。

说明

全部搜索模式和**最小匹配分数**相关更多详情，请参见 *参数配置* 中对该两个参数的说明。

勾选模板　可勾选使用的模板。勾选多个模板时，按照勾选后模板的先后顺序执行。

编辑模板　单击*编辑模板*继续编辑模板。

清空模板　单击*删除所有模板*清空模板。

载入模板　单击*载入*从本地路径载入模板。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 特征模块参数详情，请参见 *轮廓匹配的模板配置参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

全部搜索模式

开启后，算法将逐个使用所有勾选的模板对目标进行匹配，输出最优的匹配结果。该参数默认关闭。

*轮廓匹配*模块开启该参数时，还需设置**多模板重叠率**参数。

多模板重叠率

不同模板检测的目标重叠时，若重叠率低于该数值，则目标均输出；若高于该数值，则仅输出匹配分数高的目标。

说明

重叠率的计算方式为“两个重叠目标中重叠部分的面积”/“两个重叠目标的总面积”。

最小匹配分数

设置模板与待匹配图像中目标的相似度阈值。算法仅在相似度达到该阈值时，才会将该搜索到的目标判定为有效匹配结果。该参数最大可设置为 1，表示模板的轮廓点与待匹配图像的完全契合。默认为 0.5。

最大匹配个数

设置允许查找到的最大目标数量，默认值为 1，有效取值范围为 1~1000。

匹配极性

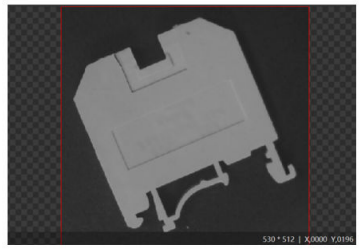
设置极性表示图形边缘到背景的颜色过渡情况（从黑到白或者从白到黑）。

如果查找目标的极性和模板的极性不一致时，仍要保证目标被查找到，则需将该参数设置为**不考虑极性**。如不需要则可设置为**考虑极性**，缩短查找时间。

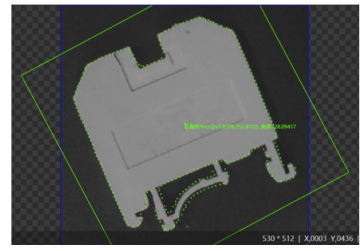
角度范围

设置模板图像可相较水平方向的旋转角度范围（顺时针为正，逆时针为负）。默认值为-45~45，表示旋转角度范围为-45°至 45°。

搜索有旋转变化的目标时，需根据目标的旋转角度范围设置该参数（取值范围-180~180），否则可能导致匹配失败。



角度范围：0°~45°
（未匹配成功）



角度范围：-45°~45°
（匹配成功）

图 13-25 效果示例

尺度范围类型

设置待匹配目标相对于已创建模板的缩放尺度范围类型。搜索存在尺度缩放的目标时，需设置该参数，可从整体或分别从 X、Y 轴的缩放尺度范围进行匹配，所以可选择如下两种类型。

尺度范围

从整体设置待匹配目标相对于已创建模板的缩放尺度范围。

尺度范围 XY

分别从 X、Y 轴设置待匹配目标相对于已创建模板的缩放尺度范围，选择此类型，需设置 **X 尺度范围**、**Y 尺度范围**和**横切角度范围**。

横切角度范围为待匹配目标相对于已创建模板沿着横坐标轴旋转的角度变换范围，默认值为 0~0（即仅搜索无旋转的目标）；有效值范围为-45~45，表示旋转角度范围为-45°至 45°。

说明

默认的尺度范围为 1.00~1.00（即仅搜索无尺度缩放的目标），有效值范围为 0.1~10.0。因此搜索存在尺度缩放的目标时，请务必根据实际的缩放程度调整该参数取值，否则将导致匹配失败（如下图示例所示）。

最大重叠率

设置同一个模板检测到有多个目标且其中两个匹配目标彼此重合时，两个匹配框的最大重叠比例。该参数取值越大表示两个目标可重叠的程度越大。取值范围 0~100，默认 50。当两个目标的重叠率大于所设置的参数值时，算法将舍弃相似度得分较低的目标，保留得分较高的目标。

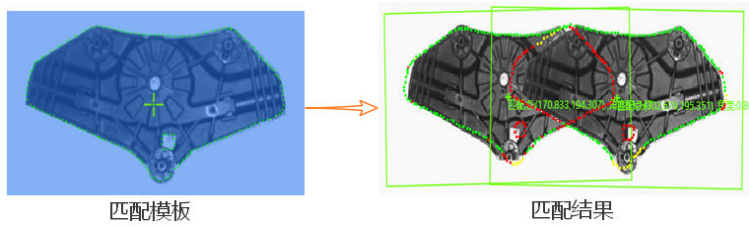


图 13-26 效果示例

排序类型

设置同一个模板检测到有多个目标时在*当前结果*的排序依据。

下图为按分数降序排序的示例。

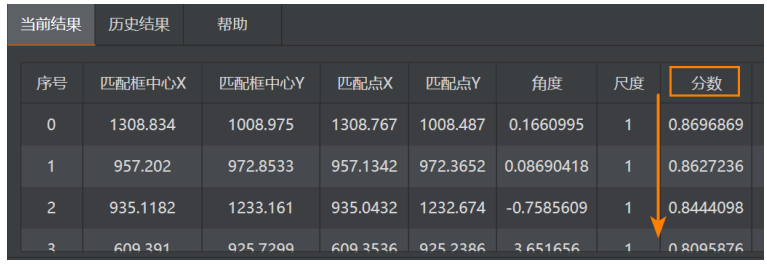


图 13-27 排序示例

按分数降序排序

按照特征匹配的得分降序排列。

按角度降序排序

按照当前结果里面相对角度偏移降序排列。

按 X 由小到大排序

按照匹配框中心的 X 轴坐标由小到大排序。

按 Y 由小到大排序

按照匹配框中心的 Y 轴坐标由小到大排序。

X 由小到大，Y 由小到大

按照匹配框中心点 X 轴坐标由小到大排序，当 X 坐标整数化后值相同时再按照 Y 轴坐标从小到大排序。

Y 由小到大，X 由小到大

按照匹配框中心点 Y 轴坐标由小到大排序，当 Y 坐标整数化后值相同时再按照 X 轴坐标从小到大排序。

阈值类型

自动阈值

根据目标图像自动生成边缘对比度阈值参数进行边缘轮廓点提取。

模板阈值

将模板的边缘对比度阈值经过内部转换后，作为匹配阶段的边缘对比度阈值进行边缘轮廓点的提取。

手动阈值

以设定的边缘对比度阈值提取边缘轮廓点。

是否考虑噪点

噪点是指匹配图中不在边缘链上的特征点。开启该参数后，算法会考虑噪点特征，若特征存在毛刺，则评分降低。默认关闭。

如下图所示，是否考虑噪点会对匹配分值产生影响。（左：匹配模板，中：考虑噪点，右：不考虑噪点）。



图 13-28 效果示例

延拓阈值

该参数取值表示“目标物体的特征在图像边缘显示不全时，缺失特征相对于完整特征的比例”。当目标物体被图像边缘截断时，设置延拓阈值可保证目标物体被查找到。

下图示例中，延拓阈值大于 40，即可保证被图像右侧边缘截断的目标物体被查找到。

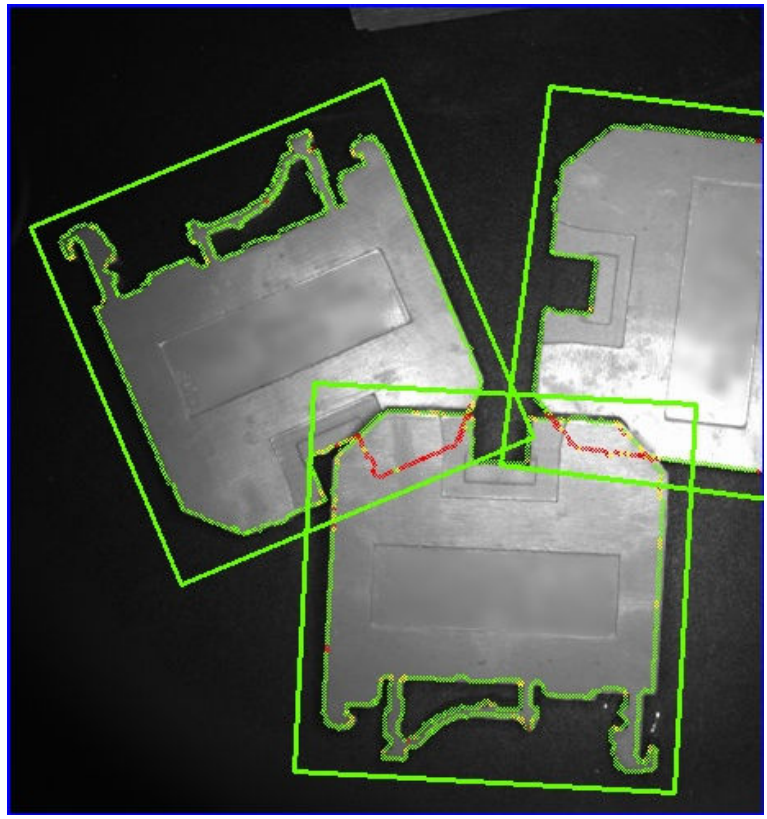


图 13-29 效果示例

超时控制

设置搜索超时的时长。搜索时间达到超时时长算法将停止搜索，不返回任何搜索结果。取值范围：0~10000，单位：ms。如设置为 0，则关闭超时控制功能。

是否获取轮廓

开启则显示模板轮廓特征点。

关闭则不显示特征点，只显示匹配框，可降低匹配耗时。

是否输出轮廓

打开则在模块结果中输出轮廓；关闭则在模块结果中不输出轮廓。

模块结果

*轮廓匹配*模块的模块结果如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

匹配个数

int 型，代表图像中与模板相同或相似对象的个数。

匹配模板编号

int 型，代表所匹配模板对应的编号。

匹配模板名称

string 型，代表所匹配模板的名称。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

匹配点

float 型，代表每个匹配点。

匹配点 X

float 型，代表每个匹配点的 X 坐标。

匹配点 Y

float 型，代表每个匹配点的 Y 坐标。

尺度

float 型，代表匹配对象相对于已创建模板的等比例缩放倍数。

尺度 X

float 型，代表匹配对象在 X 轴方向上的缩放尺度。

尺度 Y

float 型，代表匹配对象在 Y 轴方向上的缩放尺度。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

横切值

float 型，代表匹配对象 Y 轴向 X 轴的旋转角度。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

匹配轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不显示。

匹配轮廓点个数

int 型，代表图像中与模板相同或相似图像的轮廓点个数。

轮廓点集

轮廓点 X

float 型，代表各个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表各个轮廓点的 Y 坐标。

轮廓点序号

int 型，代表轮廓点集中各个轮廓点的编号。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.2 高精度匹配

*高精度匹配*模块适用于需要定位目标、识别目标方向、检测目标偏移角度等对定位精度有较高要求的业务场景。

模块原理

*高精度匹配*模块的原理和*轮廓匹配*模块的原理一致，详情请参见 *模块原理* 。*高精度匹配*模块相比于*快速匹配*模块在匹配精度上有所提升。前者在执行基于图像金字塔的匹配加速时使用了图像金字塔中的多层图像搜索目标，而后者仅在图像金字塔的顶层搜索目标。

使用方法

前后序模块

在流程中，*高精度匹配*模块需要和 *位置修正* 、 *N 点标定* 、 *测量* 相关模块等其他模块配合使用。配合使用的目的是将*高精度匹配*输出的物体位置等信息传递至其他模块。

使用示例

在二维码识别场景中，可根据以下思路搭建流程：

1. 通过*高精度匹配*将二维码位置信息传递至 *位置修正* 。
2. *位置修正*将修正的二维码位置信息传递至 *二维码识别* ，进而保证*二维码识别*的 ROI 随部件位置变化而变化。



图 13-30 高精度匹配应用示例

特征模板

*高精度匹配*模块的特征模板设置过程与*轮廓匹配*一致，详情请参见 *特征模板* 。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- *高精度匹配*和*轮廓匹配*的模板配置参数基本一致，更多模板配置参数详情，请参见 *特征模板* 。
- 结果显示参数详情，请参见 *结果显示* 。

全部搜索模式

开启后，算法将逐个使用所有勾选的模板对目标进行匹配，输出最优的匹配结果。该参数默认关闭。

*轮廓匹配*模块开启该参数时，还需设置**多模板重叠率**参数。

多模板重叠率

不同模板检测的目标重叠时，若重叠率低于该数值，则目标均输出；若高于该数值，则仅输出匹配分数高的目标。

说明

重叠率的计算方式为“两个重叠目标中重叠部分的面积”/“两个重叠目标的总面积”。

最小匹配分数

设置模板与待匹配图像中目标的相似度阈值。算法仅在相似度达到该阈值时，才会将该搜索到的目标判定为有效匹配结果。该参数最大可设置为 1，表示模板的轮廓点与待匹配图像的完全契合。默认为 0.5。

最大匹配个数

设置允许查找到的最大目标数量，默认值为 1，有效取值范围为 1~1000。

匹配极性

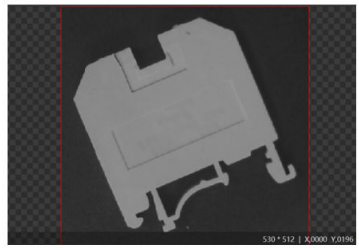
设置极性表示图形边缘到背景的颜色过渡情况（从黑到白或者从白到黑）。

如果查找目标的极性和模板的极性不一致时，仍要保证目标被查找到，则需将该参数设置为**不考虑极性**。如不需要则可设置为**考虑极性**，缩短查找时间。

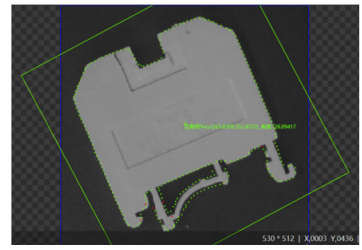
角度范围

设置模板图像可相较水平方向的旋转角度范围（顺时针为正，逆时针为负）。默认值为-45~45，表示旋转角度范围为-45°至 45°。

搜索有旋转变化的目标时，需根据目标的旋转角度范围设置该参数（取值范围-180~180），否则可能导致匹配失败。



角度范围：0°~45°
（未匹配成功）



角度范围：-45°~45°
（匹配成功）

图 13-31 效果示例

尺度范围类型

设置待匹配目标相对于已创建模板的缩放尺度范围类型。搜索存在尺度缩放的目标时，需设置该参数，可从整体或分别从 X、Y 轴的缩放尺度范围进行匹配，所以可选择如下两种类型。

尺度范围

从整体设置待匹配目标相对于已创建模板的缩放尺度范围。

尺度范围 XY

分别从 X、Y 轴设置待匹配目标相对于已创建模板的缩放尺度范围，选择此类型，需设置 **X 尺度范围**和 **Y 尺度范围**。

说明

默认的尺度范围为 1.00~1.00（即仅搜索无尺度缩放的目标），有效值范围为 0.1~10.0。因此搜索存在尺度缩放的目标时，请务必根据实际的缩放程度调整该参数取值，否则将导致匹配失败。

最大重叠率

设置同一个模板检测到有多个目标且其中两个匹配目标彼此重合时，两个匹配框的最大重叠比例。该参数取值越大表示两个目标可重叠的程度越大。取值范围 0~100，默认 50。当两个目标的重叠率大于所设置的参数值时，算法将舍弃相似度得分较低的目标，保留得分较高的目标。

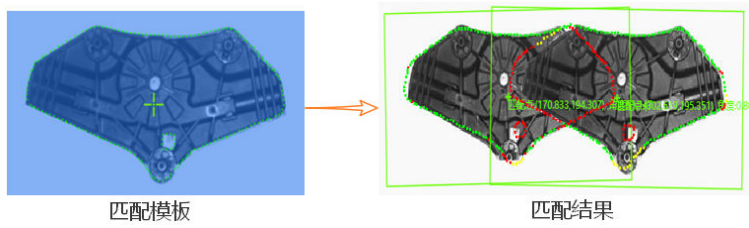


图 13-32 效果示例

排序类型

设置同一个模板检测到有多个目标时在*当前结果*的排序依据。

下图为按分数降序排序的示例。

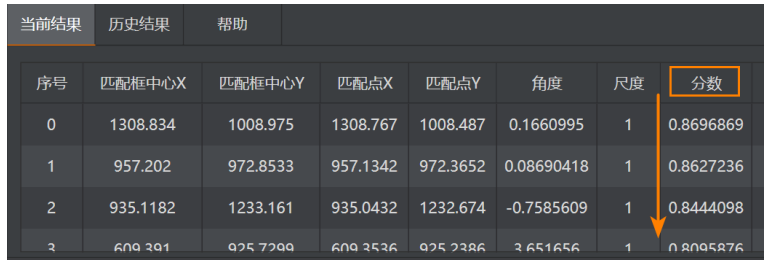


图 13-33 排序示例

按分数降序排序

按照特征匹配的得分降序排列。

按角度降序排序

按照当前结果里面相对角度偏移降序排列。

按 X 由小到大排序

按照匹配框中心的 X 轴坐标由小到大排序。

按 Y 由小到大排序

按照匹配框中心的 Y 轴坐标由小到大排序。

X 由小到大，Y 由小到大

按照匹配框中心点 X 轴坐标由小到大排序，当 X 坐标整数化后值相同时再按照 Y 轴坐标从小到大排序。

Y 由小到大，X 由小到大

按照匹配框中心点 Y 轴坐标由小到大排序，当 Y 坐标整数化后值相同时再按照 X 轴坐标从小到大排序。

阈值类型

自动阈值

根据目标图像自动生成边缘对比度阈值参数进行边缘轮廓点提取。

模板阈值

将模板的边缘对比度阈值经过内部转换后，作为匹配阶段的边缘对比度阈值进行边缘轮廓点的提取。

手动阈值

以设定的边缘对比度阈值提取边缘轮廓点。

是否考虑噪点

噪点是指匹配图中不在边缘链上的特征点。开启该参数后，算法会考虑噪点特征，若特征存在毛刺，则评分降低。默认关闭。

如下图所示，是否考虑噪点会对匹配分值产生影响。（左：匹配模板，中：考虑噪点，右：不考虑噪点）。



图 13-34 效果示例

延拓阈值

该参数取值表示“目标物体的特征在图像边缘显示不全时，缺失特征相对于完整特征的比例”。当目标物体被图像边缘截断时，设置延拓阈值可保证目标物体被查找到。

下图示例中，延拓阈值大于 40，即可保证被图像右侧边缘截断的目标物体被查找到。

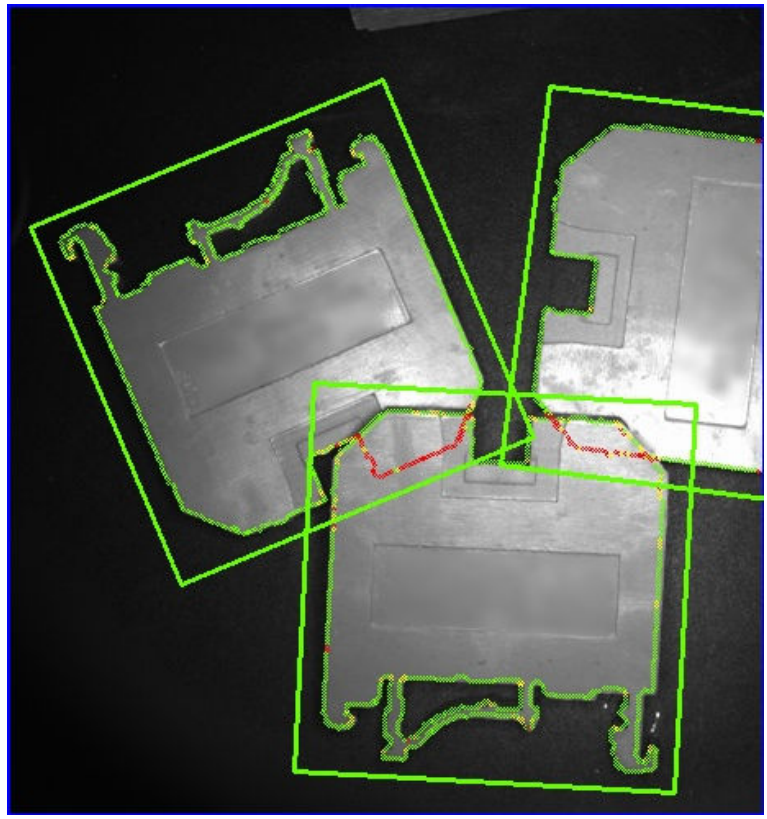


图 13-35 效果示例

超时控制

设置搜索超时的时长。搜索时间达到超时时长算法将停止搜索，不返回任何搜索结果。取值范围：0~10000，单位：ms。如设置为 0，则关闭超时控制功能。

是否获取轮廓

开启则显示模板轮廓特征点。

关闭则不显示特征点，只显示匹配框，可降低匹配耗时。

模块结果

*高精度匹配*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

匹配个数

int 型，代表图像中与模板相同或相似对象的个数。

匹配模板编号

int 型，代表所匹配模板对应的编号。

匹配模板名称

string 型，代表所匹配模板的名称。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

匹配点

float 型，代表每个匹配点。

匹配点 X

float 型，代表每个匹配点的 X 坐标。

匹配点 Y

float 型，代表每个匹配点的 Y 坐标。

尺度

float 型，代表匹配对象相对于已创建模板的等比例缩放倍数。

尺度 X

float 型，代表匹配对象在 X 轴方向上的缩放尺度。

尺度 Y

float 型，代表匹配对象在 Y 轴方向上的缩放尺度。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

匹配轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不显示。

匹配轮廓点个数

int 型，代表图像中与模板相同或相似图像的轮廓点个数。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.3 快速匹配

*快速匹配*模块相较于*高精度匹配*模块，通常拥有更快的匹配速度，且对存在形变的目标有着更好的匹配效果，但匹配精度不如*高精度匹配*模块。该模块适用于基于定位且对定位精度要求不高的业务场景，例如物体的有无判断、方向判断、计数等。

模块原理

*快速匹配*模块的模块原理和*轮廓匹配*的模块原理一致，详情请参见 *模块原理* 。

使用方法

前后序模块

在流程中，*快速匹配*模块通常需要和 *位置修正* 、 *条件检测* 、 *脚本* 、 *测量* 相关模块等其他模块配合使用。配合使用的目的是将*快速匹配*输出的物体位置和目标数量信息传递给需要这些数据的模块。

使用示例

缺针检测场景下，可在流程中调用*快速匹配*模块和其他模块，检测物理接口的针脚缺失问题。

搭建流程的核心思路为——当针脚缺失的时候，图像对应的位置就不会出现亮斑。基于此，具体思路如下：

- 通过*快速匹配*匹配针脚特征。
- 通过 *位置修正* 规避产品位置变化对检测的影响。
- 通过 *亮度测量* 测量 ROI 范围内图像的亮度值。
- 通过 *条件检测* 判断亮度值符合要求（OK）或不符合要求（NG）。
- 通过 *格式化* 将 OK 和 NG 数据格式化。
- 通过 *发送数据* 将格式化后的数据发送出去。

根据以上思路搭建的流程见下图。

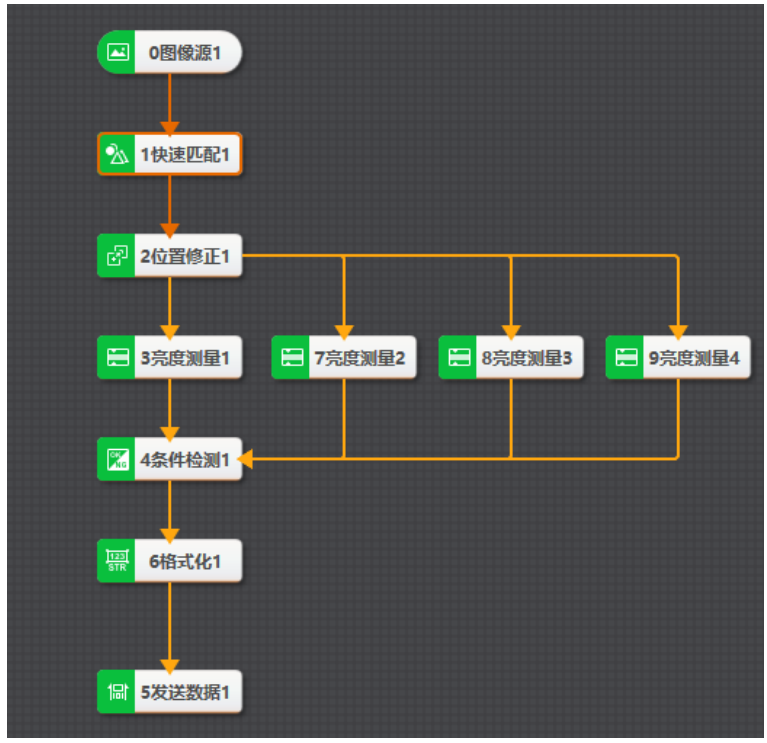
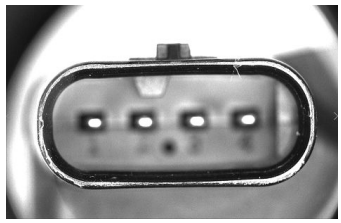
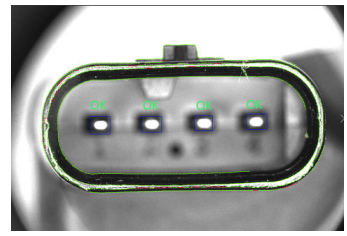


图 13-36 断针检测示例流程

表 13-3 检测效果

产品图	检测效果图
	

特征模板

*快速匹配*的特征模板配置过程和*轮廓匹配*一致。操作详情请参见 *特征模板* 。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- *快速匹配*的模板配置参数定义均可参考*轮廓匹配*中的对应参数。更多特征模板参数详情，请参见 *特征模板* 。
- 结果显示参数详情，请参见 *结果显示* 。

全部搜索模式

开启后，算法将逐个使用所有勾选的模板对目标进行匹配，输出最优的匹配结果。该参数默认关闭。

*轮廓匹配*模块开启该参数时，还需设置**多模板重叠率**参数。

多模板重叠率

不同模板检测的目标重叠时，若重叠率低于该数值，则目标均输出；若高于该数值，则仅输出匹配分数高的目标。

说明

重叠率的计算方式为“两个重叠目标中重叠部分的面积”/“两个重叠目标的总面积”。

最小匹配分数

设置模板与待匹配图像中目标的相似度阈值。算法仅在相似度达到该阈值时，才会将该搜索到的目标判定为有效匹配结果。该参数最大可设置为 1，表示模板的轮廓点与待匹配图像的完全契合。默认为 0.5。

最大匹配个数

设置允许查找到的最大目标数量，默认值为 1，有效取值范围为 1~1000。

匹配极性

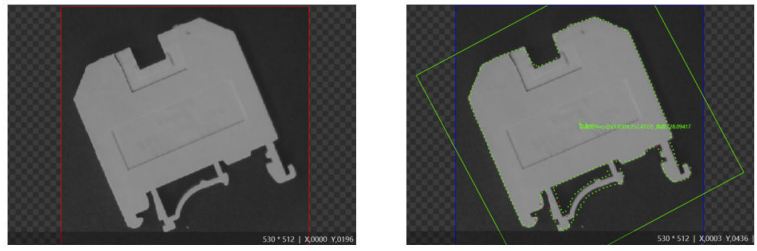
设置极性表示图形边缘到背景的颜色过渡情况（从黑到白或者从白到黑）。

如果查找目标的极性和模板的极性不一致时，仍要保证目标被查找到，则需将该参数设置为**不考虑极性**。如不需要则可设置为**考虑极性**，缩短查找时间。

角度范围

设置模板图像可相较水平方向的旋转角度范围（顺时针为正，逆时针为负）。默认值为-45~45，表示旋转角度范围为-45°至 45°。

搜索有旋转变化的目标时，需根据目标的旋转角度范围设置该参数（取值范围-180~180），否则可能导致匹配失败。



角度范围：0°~45°
（未匹配成功）

角度范围：-45°~45°
（匹配成功）

图 13-37 效果示例

尺度范围

设置待匹配目标相对于已创建模板的缩放尺度范围。搜索存在尺度缩放的目标时，需设置该参数，有效值范围为 0.1~10.0。

说明

默认的**尺度范围**为 1.00~1.00（即仅搜索无尺度缩放的目标）。因此搜索存在尺度缩放的目标时，请务必根据实际的缩放程度调整该参数取值，否则将导致匹配失败（如下图示例所示）。

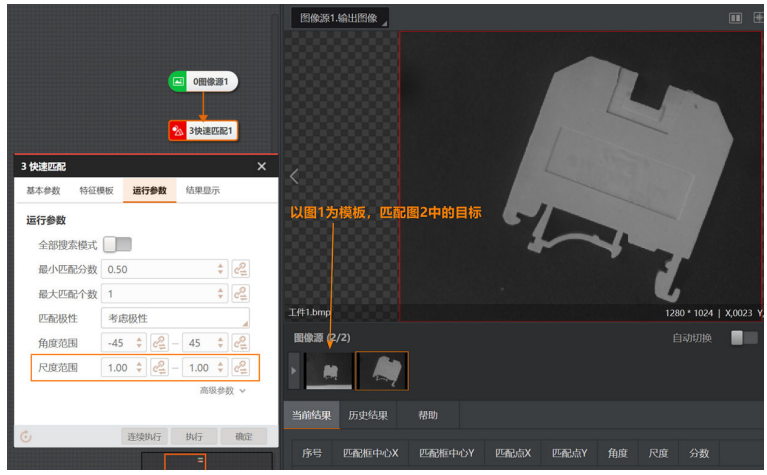


图 13-38 匹配失败（目标存在缩放但尺度范围未调整）

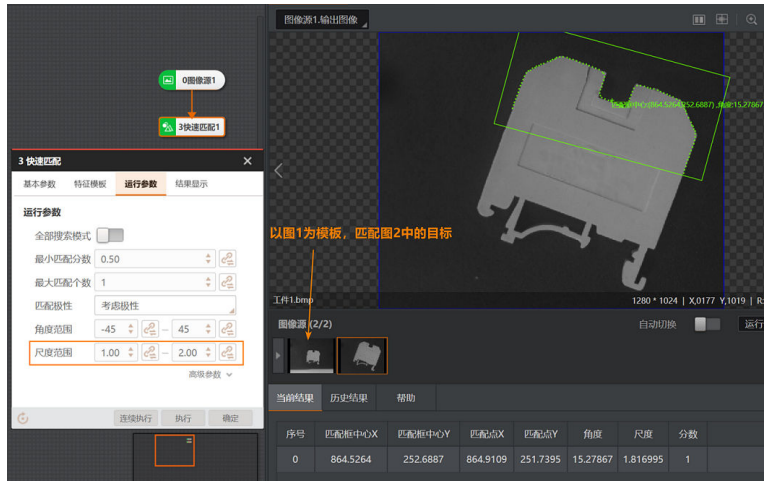


图 13-39 匹配成功（尺度范围根据目标缩放做调整）

最大重叠率

设置同一个模板检测到有多个目标且其中两个匹配目标彼此重合时，两个匹配框的最大重叠比例。该参数取值越大表示两个目标可重叠的程度越大。取值范围 0~100，默认 50。当两个目标的重叠率大于所设置的参数值时，算法将舍弃相似度得分较低的目标，保留得分较高的目标。

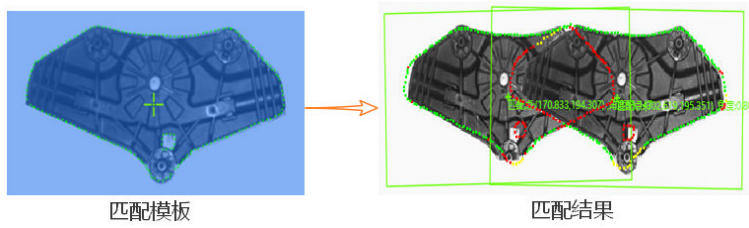


图 13-40 效果示例

排序类型

设置同一个模板检测到有多个目标时在*当前结果*的排序依据。

下图为按分数降序排序的示例。

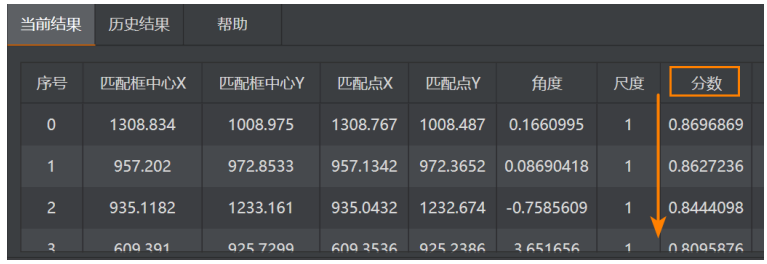


图 13-41 排序示例

按分数降序排序

按照特征匹配的得分降序排列。

按角度降序排序

按照当前结果里面相对角度偏移降序排列。

按 X 由小到大排序

按照匹配框中心的 X 轴坐标由小到大排序。

按 Y 由小到大排序

按照匹配框中心的 Y 轴坐标由小到大排序。

X 由小到大，Y 由小到大

按照匹配框中心点 X 轴坐标由小到大排序，当 X 坐标整数化后值相同时再按照 Y 轴坐标从小到大排序。

Y 由小到大，X 由小到大

按照匹配框中心点 Y 轴坐标由小到大排序，当 Y 坐标整数化后值相同时再按照 X 轴坐标从小到大排序。

阈值类型

自动阈值

根据目标图像自动生成边缘对比度阈值参数进行边缘轮廓点提取。

模板阈值

将模板的边缘对比度阈值经过内部转换后，作为匹配阶段的边缘对比度阈值进行边缘轮廓点的提取。

手动阈值

以设定的边缘对比度阈值提取边缘轮廓点。

是否考虑噪点

噪点是指匹配图中不在边缘链上的特征点。开启该参数后，算法会考虑噪点特征，若特征存在毛刺，则评分降低。默认关闭。

如下图所示，是否考虑噪点会对匹配分值产生影响。（左：匹配模板，中：考虑噪点，右：不考虑噪点）。



图 13-42 效果示例

延拓阈值

该参数取值表示“目标物体的特征在图像边缘显示不全时，缺失特征相对于完整特征的比例”。当目标物体被图像边缘截断时，设置延拓阈值可保证目标物体被查找到。

下图示例中，延拓阈值大于 40，即可保证被图像右侧边缘截断的目标物体被查找到。

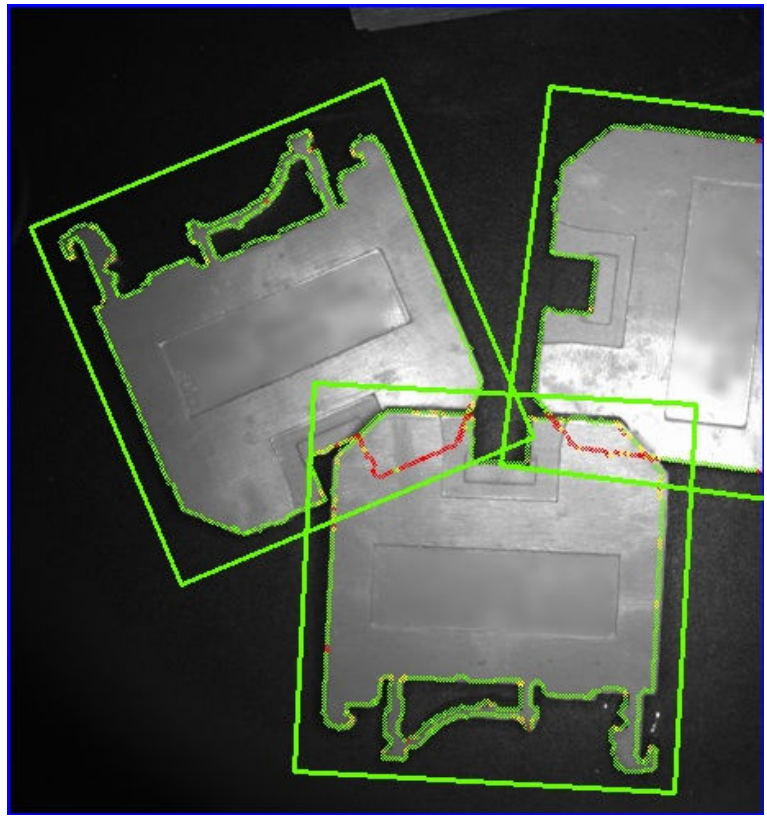


图 13-43 效果示例

超时控制

设置搜索超时的时长。搜索时间达到超时时长算法将停止搜索，不返回任何搜索结果。取值范围：0~10000，单位：ms。如设置为 0，则关闭超时控制功能。

是否获取轮廓

开启则显示模板轮廓特征点。

关闭则不显示特征点，只显示匹配框，可降低匹配耗时。

模块结果

*快速匹配*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

匹配个数

int 型，代表图像中与模板相同或相似对象的个数。

匹配模板编号

int 型，代表所匹配模板对应的编号。

匹配模板名称

string 型，代表所匹配模板的名称。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

匹配点

float 型，代表每个匹配点。

匹配点 X

float 型，代表每个匹配点的 X 坐标。

匹配点 Y

float 型，代表每个匹配点的 Y 坐标。

尺度

float 型，代表匹配对象相对于已创建模板的等比例缩放倍数。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

匹配轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不显示。

匹配轮廓点个数

int 型，代表图像中与模板相同或相似图像的轮廓点个数。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.4 灰度匹配

*灰度匹配*模块基于物体灰度特征进行特征匹配。由于不使用边缘点作为特征，该模块适用于在边缘点不明显的场景中进行物体定位、物体方向识别、物体角度检测等检测工作。

模块原理

*灰度匹配*模块基于目标在图像中的灰度值分布匹配对应目标。

算法工作流

基于灰度值的模板匹配的算法工作流与基于边缘点的基本相同。简言之，即计算模板图像所有可能的旋转平移结果与待匹配图像的相似性，取相似度分数达到阈值且最高的 n 个结果作为输出。

不同之处在于两者提取的特征。基于灰度值时，以模板图像和待匹配图像的灰度值分布作为特征，使用“归一化互相关性”（Normalized Cross Correlation，NCC）来评价两者灰度值分布的相似性。

算法公式

该算法涉及的关键公式如下：

$$\mathrm{ncc}(r,c)=\frac{1}{n}\sum_{(u,v)\in R}\frac{t(u,v)-m_t}{\sqrt{s_t^2}}\cdot\frac{i(r+u,c+v)-m_i(r,c)}{\sqrt{s_i^2(r,c)}}$$

图 13-44 算法公式（基于灰度值）

- r 和 c：分别代表模板图像的平移量，等同于轮廓匹配算法公式中 *图 13-13* 的 x 和 y
- t：模板图像
- i：待匹配图像
- mt：mi(r, c)：模板图像的灰度均值
- mi(r, c)：待匹配图像特定区域的灰度均值
- s2t：模板图像的灰度值方差
- s2i（r, c）：待匹配图像在特定区域的灰度值方差

如果当前模板图像和待匹配图像某个局部的灰度值线性正相关（两者灰度分布的趋势相同，仅存在整体上的数值偏移或者存在某种比例关系），则 NCC 值为 1；如果线性无关，则 NCC 值为 0；如果线性负相关，则 NCC 值为-1。

匹配加速机制

基于灰度值的模板匹配算法使用图像金字塔加速匹配过程。该算法中与图像金字塔相关的参数为**金字塔层数**和**匹配结束层**，这两个参数的定义有所差异。

- **金字塔层数**：相当于“基于轮廓点的匹配算法”的匹配加速机制中的**速度尺度**。可在配置灰度模板时配置该参数，具体定义参见*灰度模板*中的 *金字塔层数* 。
- **匹配结束层**：相当于“基于轮廓点的匹配算法”的匹配加速机制中的**特征尺度**。可在*灰度匹配*模块的*运行参数*页签配置该参数，具体定义参见*参数配置*中的 *匹配结束层* 。

使用方法

前后序模块

在流程中，*灰度匹配*通常需要和 *位置修正* 、 *条件检测* 、 *脚本* 、 *测量* 相关模块等其他模块配合使用。配合使用的目的是将*灰度匹配*输出的物体位置和个数信息传递给其他需要这些数据的模块。

使用示例

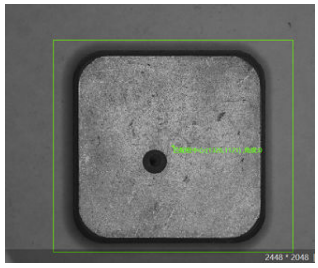
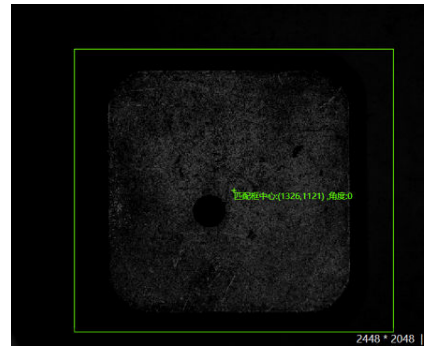
如下流程所示，*灰度匹配*用于定位和计数图像中的包装盒。 *脚本* 模块订阅*灰度匹配*定位的包装盒数量，并用于相应的脚本处理。最后，流程通过*发送数据*模块，将包装盒数量发送至通信设备。

表 13-4 灰度匹配计数示例

匹配效果	输出计数数据
	

该示例中使用*灰度匹配*，而非*高精度匹配*和*快速匹配*，是由于包装表面的轮廓信息不明显、轮廓一致性欠佳，使用*高精度匹配*和*快速匹配*都存在匹配遗漏或匹配错误的风险。
*灰度匹配*相较于“基于轮廓点信息”的*高精度匹配*、*快速匹配*和*轮廓匹配*，在光源发生变化或者相机曝光时间发生变化的情况下，依然可以匹配到目标物体（见下图）。

表 13-5 抗曝光时长变化

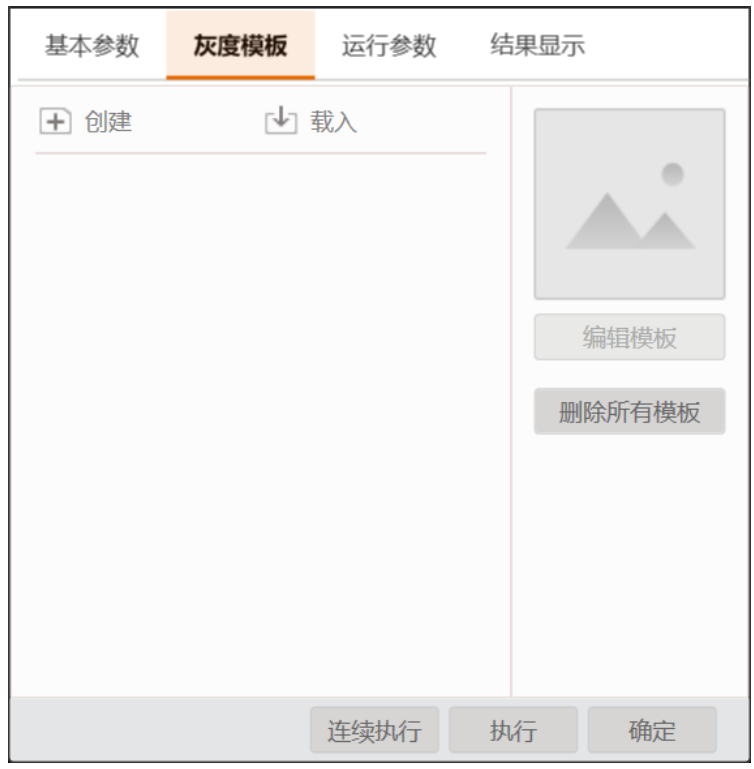
较充足曝光时长	曝光时长过短
	

灰度模型

*灰度匹配*模块运行过程中的特征提取基于灰度模板进行。

操作步骤

1. 在流程中双击模块打开模块配置窗口，并选择*灰度模板*页签。



2. 单击*创建*打开*模板配置*窗口。
   - 如果*图像源*模块已正常运行，则该窗口默认显示当前*图像源*模块采集的图像。如需基于其他图像进行模板配置，可单击*选择其他图像*。
   - *模板存图*按钮可用于保存模板文件。仅在该功能开启时，模板支持编辑功能。

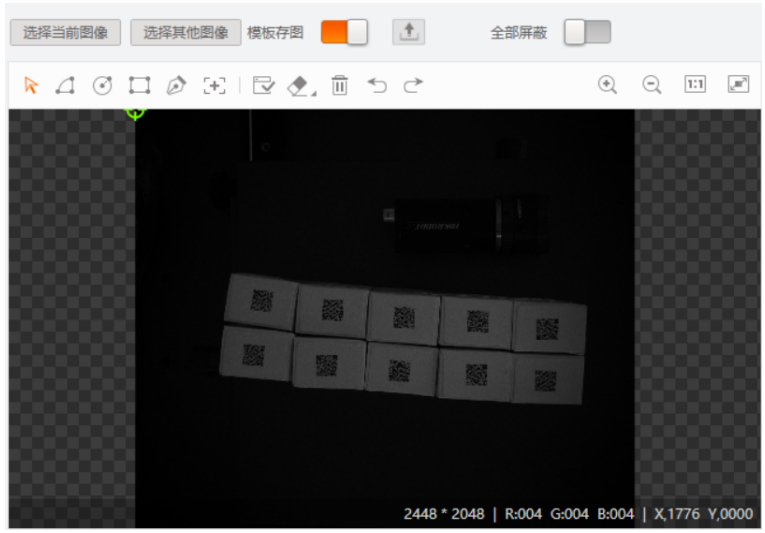


图 13-45 模板配置

3. 在图像上特征明显的区域绘制掩膜。算法将在掩膜区域内提取图像特征。

说明

- 具体如何绘制掩膜区域参见 *模块中模型配置的掩膜区域* 章节。
- 仅支持绘制单个矩形掩膜。

4. **可选操作**: 进行如下可选操作。

删除掩膜	右击掩膜，并在弹出的右键菜单上单击*删除*。
复制掩膜	右击掩膜，并在弹出的右键菜单上单击*复制*。
调整掩膜位置	单击选中掩膜，并拖拽调整其位置。
选择匹配中心	单击 ，并在图像上的合适位置单击，将其设置为模板的匹配中心。
锁定匹配中心	选择匹配中心后，可在*模板配置*窗口的下方勾选*锁定*，锁定匹配中心。锁定后，该窗口工具栏上不再显示“选择匹配中心”图标 。
移动图像	单击 后，将光标移动至图像上并拖拽图像进行移动。
缩放图像	单击 / 缩放图像；或将光标置于图像区域，并滚动鼠标滚轴缩放图像。
清空掩膜	单击 清空掩膜。
撤销上一步操作	单击 撤销上一步操作。
恢复至撤销前	单击 取消撤销，恢复至撤销前的状态。
图像自适应	单击 将图像大小设置为自适应模式。
全屏显示	单击 将图像全屏显示。
绘制 ROI	开启*全部屏蔽*，将已绘制的掩膜全部屏蔽和匹配中心。此时可单击 ，并在图像通过“刷”的方式绘制 ROI（具体见下图）。



图 13-46 刷出 ROI

5. 单击 生成模板。

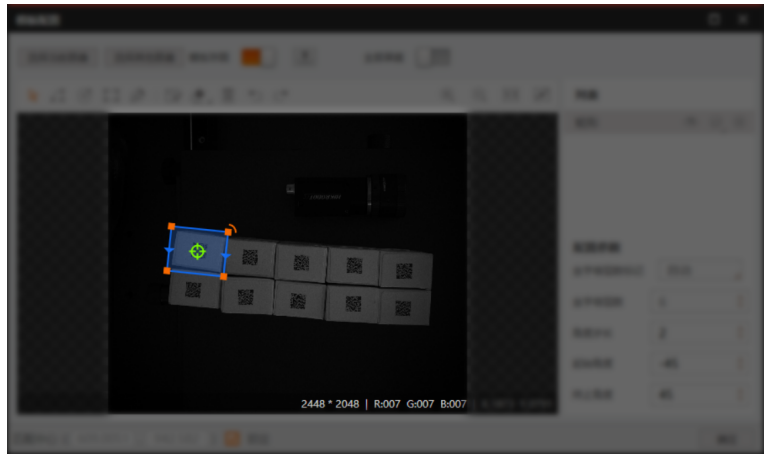


图 13-47 灰度模板效果示例

6. **可选操作**: 如果模板中提取了不必要的轮廓点，单击 ，并将光标悬浮至这些轮廓点将其擦除。

   可单击 右下角的三角形图标，并在弹出的浮窗滑动滑块调整“橡皮擦”的大小。

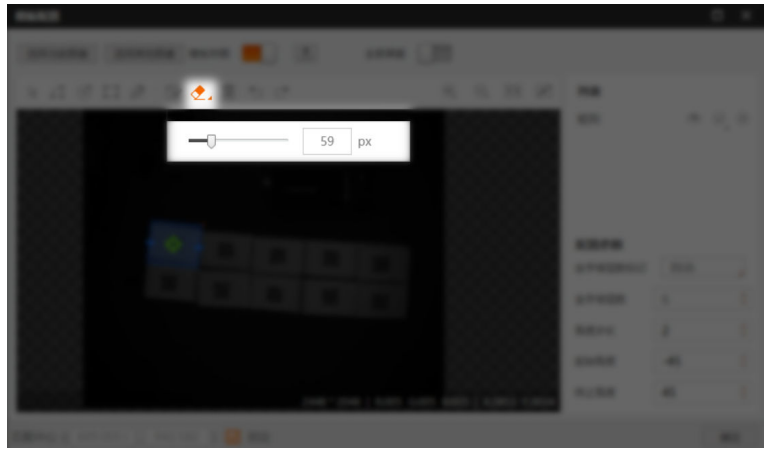


图 13-48 调整“橡皮擦”大小

7. **可选操作**: 调整模板的配置参数，并重新生成模板，直至模板效果满足需求。

   灰度匹配的模板配置参数如下：

   金字塔层数标记

   可设置图像金字塔的最大层数是自动还是手动配置。

   金字塔层数

   可查看或设置图像金字塔的最大层数（在必要情况下，算法将根据输入的模板图像适当减少层数）。金字塔层数越多则匹配效率越高，但匹配精度可能下降；越少则匹配效率越低，但匹配精度更高。

   说明

   仅在**金字塔层数标记**参数选择手动时，可设置该参数。

   角度步长

   模板图像旋转的角度间隔。角度步长越小则匹配结果的角度越精确，但耗时越长，且模型容量越大；角度步长越大则匹配结果的角度精度越低，但匹配效率越高，且模型容量越小。

   起始角度
   终止角度

   建模过程中，模板图像可旋转的角度范围。

8. **可选操作**: 在掩膜列表进行如下可选操作。

   隐藏/显示掩膜 单击 / 隐藏或显示掩膜。

   屏蔽掩膜 单击 右下角的三角形图标，并单击 屏蔽对应掩膜，再单击 重新生成模板，可屏蔽该掩膜中的特征点提取。

删除掩膜　　单击 ⊗ 删除对应的掩膜。

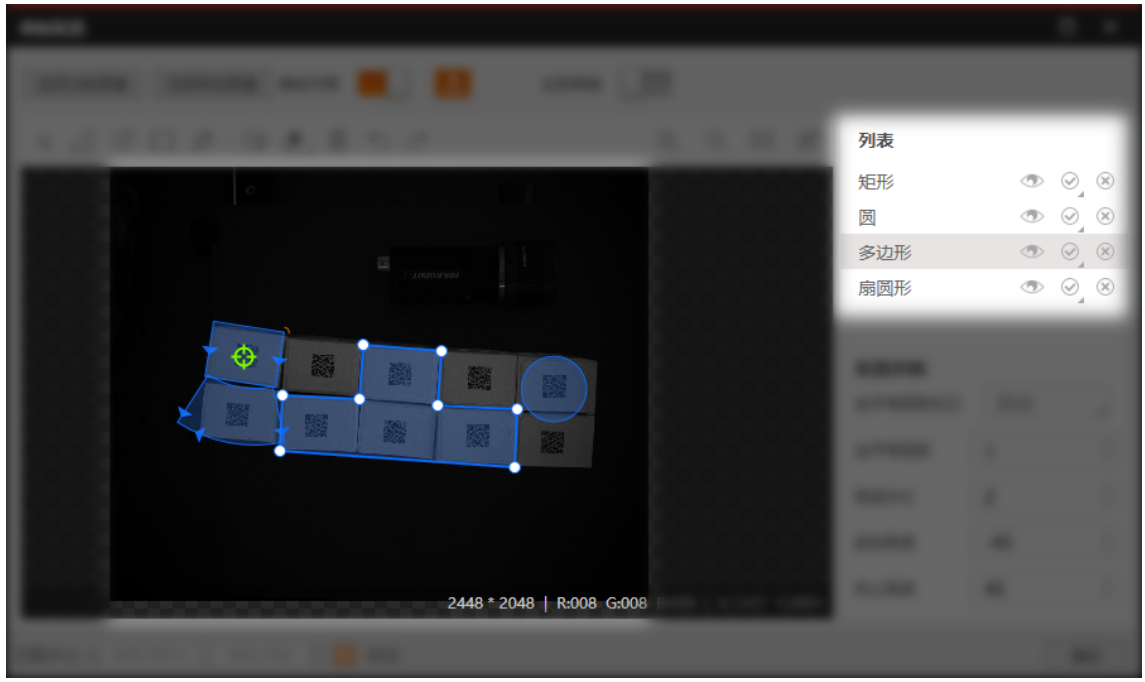


图 13-49 掩膜列表

9. 单击 将模板保存至本地路径。

   模板图像默认保存至模板文件中。如无需保存，可关闭*模板存图*。

10. 单击*模板配置*窗口右下角的*确定*，完成模板配置。界面将自动跳转至*灰度模板*页签，并展示已创建的灰度模板。



图 13-50 灰度模板列表

11. 可选操作: 在*灰度模板*页签进行如下可选操作。

模板排序　单击 ↑ / ↓ 对模板进行排序。

在未开启**全部搜索模式**时，算法将优先采用排序更靠前的模板与目标图像进行匹配。如果找到符合**最小匹配分数**要求的目标，则停止匹配。

说明

全部搜索模式和**最小匹配分数**相关更多详情，请参见 *快速匹配* 中对该两个参数的说明。

勾选模板　可勾选使用的模板。勾选多个模板时，按照勾选后模板的先后顺序执行。

编辑模板　单击*编辑模板*继续编辑模板。

清空模板　单击*删除所有模板*清空模板。

载入模板　单击*载入*从本地路径载入模板。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 灰度模板详情，请参见 *灰度匹配的模板配置参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

全部搜索模式

开启后，算法将逐个使用所有勾选的模板对目标进行匹配，输出最优的匹配结果。该参数默认关闭。

*轮廓匹配*模块开启该参数时，还需设置**多模板重叠率**参数。

多模板重叠率

不同模板检测的目标重叠时，若重叠率低于该数值，则目标均输出；若高于该数值，则仅输出匹配分数高的目标。

说明

重叠率的计算方式为“两个重叠目标中重叠部分的面积”/“两个重叠目标的总面积”。

最小匹配分数

设置模板与待匹配图像中目标的相似度阈值。算法仅在相似度达到该阈值时，才会将该搜索到的目标判定为有效匹配结果。该参数最大可设置为 1，表示模板的轮廓点与待匹配图像的完全契合。默认为 0.5。

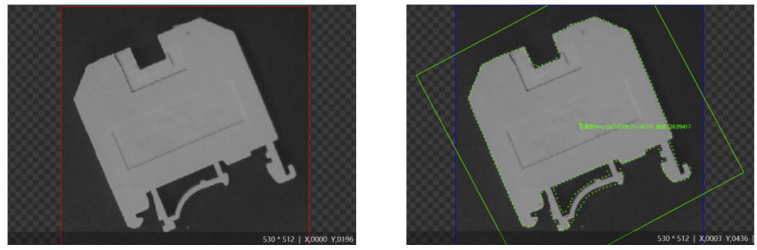
最大匹配个数

设置允许查找到的最大目标数量，默认值为 1，有效取值范围为 1~1000。

角度范围

设置模板图像可相较水平方向的旋转角度范围（顺时针为正，逆时针为负）。默认值为-45~45，表示旋转角度范围为-45°至 45°。

搜索有旋转变化的目标时，需根据目标的旋转角度范围设置该参数（取值范围-180~180），否则可能导致匹配失败。



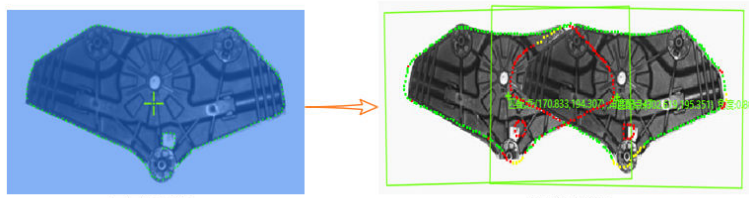
角度范围：0°~45°
（未匹配成功）

角度范围：-45°~45°
（匹配成功）

图 13-51 效果示例

最大重叠率

设置同一个模板检测到有多个目标且其中两个匹配目标彼此重合时，两个匹配框的最大重叠比例。该参数取值越大表示两个目标可重叠的程度越大。取值范围 0~100，默认 50。当两个目标的重叠率大于所设置的参数值时，算法将舍弃相似度得分较低的目标，保留得分较高的目标。



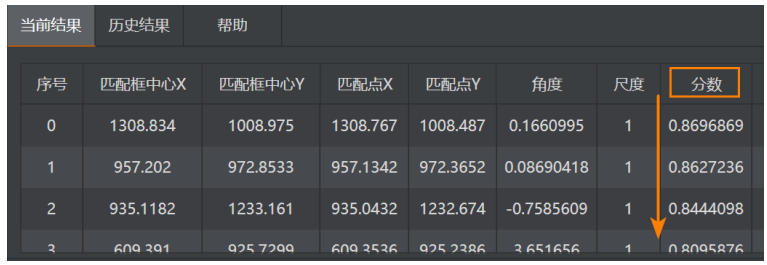
匹配模板　　匹配结果

图 13-52 效果示例

排序类型

设置同一个模板检测到有多个目标时在*当前结果*的排序依据。

下图为按分数降序排序的示例。



当前结果　历史结果　帮助

序号	匹配框中心X	匹配框中心Y	匹配点X	匹配点Y	角度	尺度	分数
0	1308.834	1008.975	1308.767	1008.487	0.1660995	1	0.8696869
1	957.202	972.8533	957.1342	972.3652	0.08690418	1	0.8627236
2	935.1182	1233.161	935.0432	1232.674	-0.7585609	1	0.8444098
3	609.391	925.7299	609.3536	925.2386	3.6551656	1	0.8095876

图 13-53 排序示例

按分数降序排序

按照特征匹配的得分降序排列。

按角度降序排序

按照当前结果里面相对角度偏移降序排列。

按 X 由小到大排序

按照匹配框中心的 X 轴坐标由小到大排序。

按 Y 由小到大排序

按照匹配框中心的 Y 轴坐标由小到大排序。

X 由小到大，Y 由小到大

按照匹配框中心点 X 轴坐标由小到大排序，当 X 坐标整数化后值相同时再按照 Y 轴坐标从小到大排序。

Y 由小到大，X 由小到大

按照匹配框中心点 Y 轴坐标由小到大排序，当 Y 坐标整数化后值相同时再按照 X 轴坐标从小到大排序。

匹配极性

设置极性表示图形边缘到背景的颜色过渡情况（从黑到白或者从白到黑）。

如果查找目标的极性和模板的极性不一致时，仍要保证目标被查找到，则需将该参数设置为**不考虑极性**。如不需要则可设置为**考虑极性**，缩短查找时间。

超时控制

设置搜索超时的时长。搜索时间达到超时时长算法将停止搜索，不返回任何搜索结果。取值范围：0~10000，单位：ms。如设置为 0，则关闭超时控制功能。

匹配结束层

结束匹配时所用的金字塔层数，默认值为 0。该参数取值越大，匹配速度越快，且对匹配目标的抗变形能力越强，但定位精度可能降低。

模块结果

*灰度匹配*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

匹配个数

int 型，代表图像中与模板相同或相似对象的个数。

匹配模板编号

int 型，代表所匹配模板对应的编号。

匹配模板名称

string 型，代表所匹配模板的名称。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

匹配点

float 型，代表每个匹配点。

匹配点 X

float 型，代表每个匹配点的 X 坐标。

匹配点 Y

float 型，代表每个匹配点的 Y 坐标。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.5 图形定位

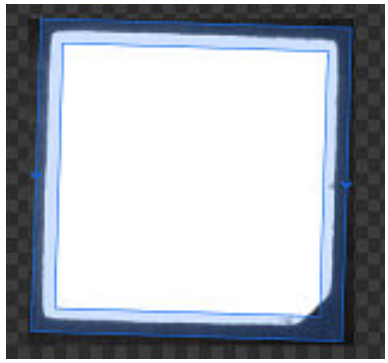
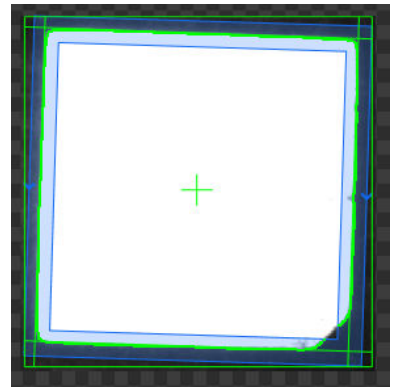
*图形定位*模块可判断图像指定区域内是否存在指定图形（回字形和十字形），并输出指定图形的中心点、角度、尺度等信息。该模块常用于目标物体的粗定位，也可用于产品计数、判断有无等其他基于定位的业务场景。

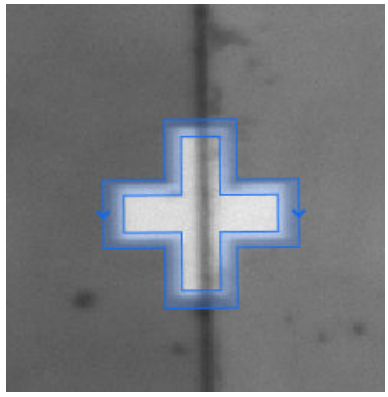
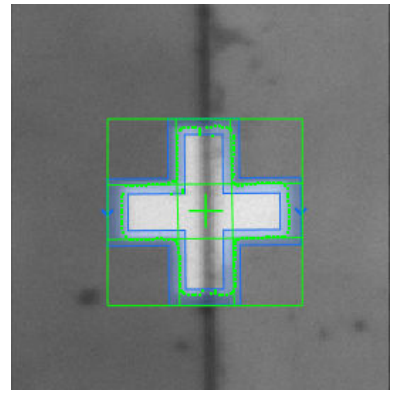
本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*
- *特征模型*

模块原理

*图形定位*模块的算法原理与 *基于轮廓点的模板匹配算法* 基本相同，不同之处在于掩膜的形状。该模块支持回字形和十字形掩膜，分别创建回字形和十字形的模板。完成建模后，算法会利用创建的模板对目标图像进行匹配定位 。

回字形掩膜	建模结果
	

十字形掩膜	建模结果
	

使用方法

在流程中，*图形定位*的前序模块为 *图像源*，两者配合可初步定位指定特征。*图形定位*的后序模块包括 *位置修正* 、 *圆查找* 、 *直线查找* 和 *卡尺工具* 等，配合使用可修正 ROI，辅助精定位。

下图中展示了通过*图像源*、*图形定位*、*位置修正*和 *边缘交点* 模块组建流程，实现边缘交点检测的案例。

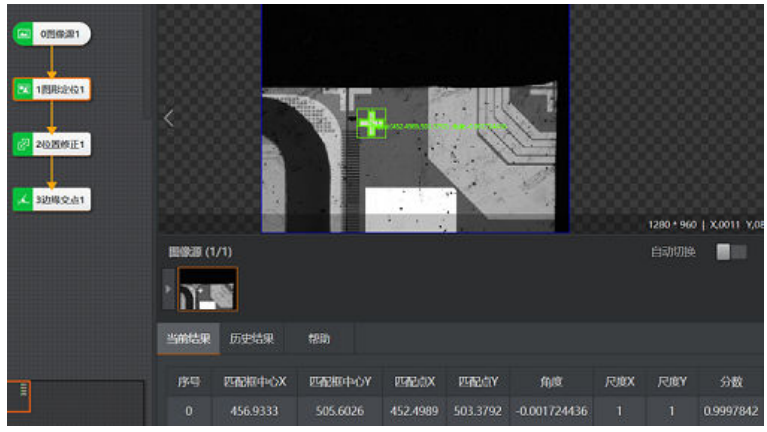


图 13-54 图形定位使用示例

特征模型

*图形定位*模块运行过程中的特征提取基于特征模型进行。运行该模块之前，需先完成特征模型配置。

请参照如下步骤配置特征模型。

操作步骤

1. 在流程中双击模块打开模块配置窗口，并选择*特征模型*页签。



图 13-55 特征模型页签

2. 单击*+新建模型*打开*模型配置*窗口。

   如果 *图像源* 模块已正常运行，则该窗口默认显示当前*图像源*模块采集的图像。如需基于其他图像进行模型配置，可单击*选择其他图像*。

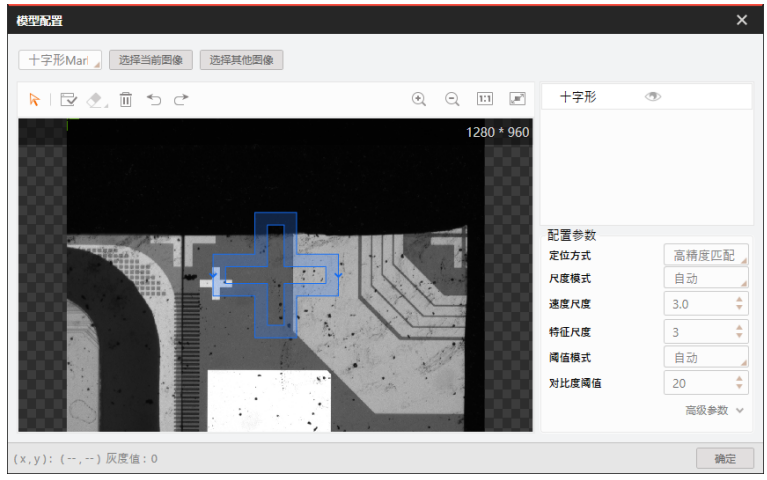


图 13-56 模型配置窗口

3. 选择掩膜类型（如下图所示）。

   十字形 Mark 点即前文*模块原理*中提及的十字形掩膜，**矩形 Mark 点**即回字形掩膜。



图 13-57 选择掩膜类型

4. 将光标悬浮至掩膜，显示掩膜节点，并在图像上拖拽掩膜节点调整掩膜大小和形状，直至掩膜覆盖目标特征区域。

   调整掩膜过程中，可滚动鼠标滚轴，按需缩放图像。

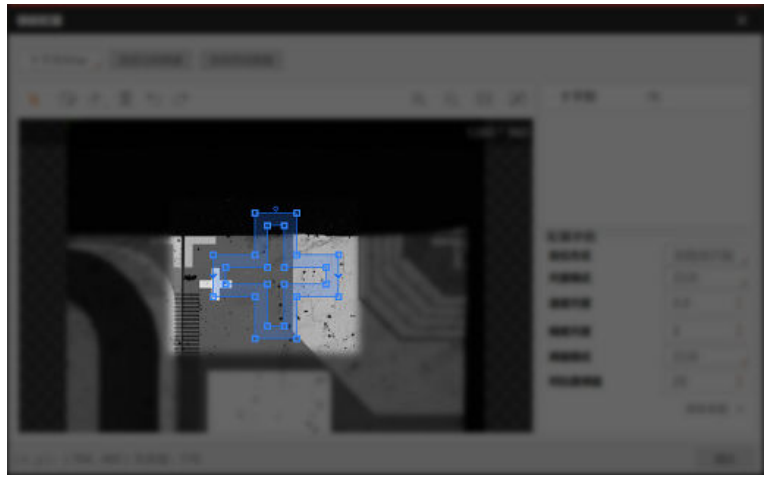


图 13-58 掩膜节点

5. **可选操作**: 进行如下可选操作。

图像另存为	右键单击模型图像，并在弹出的右键菜单上单击*另存为*，将模型图像另存至本地路径。
ROI 锁定	右键单击模型图像，并在弹出的右键菜单上单击 *ROI 锁定*。锁定后，无法调整掩膜形状和大小。
ROI 解锁	右键单击模型图像，并在弹出的右键菜单上单击 *ROI 解锁*。解锁后，可调整掩膜的形状和大小。
移动图像	单击 后，将光标移动至图像上并拖拽图像进行移动。
缩放图像	单击 / 缩放图像；或将光标置于图像区域，并滚动鼠标滚轴缩放图像。
清空掩膜	单击 清空掩膜。
撤销上一步操作	单击 撤销上一步操作。
恢复至撤销前	单击 取消撤销，恢复至撤销前的状态。
图像自适应	单击 将图像大小设置为自适应模式。
全屏显示	单击 将图像全屏显示。

6. 单击 生成模型。

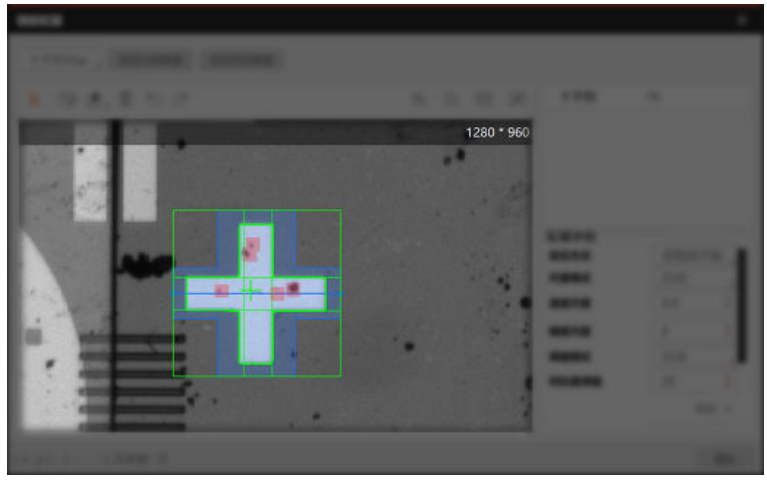


图 13-59 模型效果示例

7. 如果模型中提取了不必要的轮廓点，单击 ，并将光标悬浮至这些轮廓点将其擦除。
   可单击 右下角的三角形图标，并在弹出的浮窗滑动滑块调整“橡皮擦”的大小。



图 13-60 调整“橡皮擦”大小

8. **可选操作**: 调整模型的配置参数，并重新生成模型，直至模型效果满足要求。

   定位方式

   设置定位方式，可选 *快速匹配* 或 *高精度匹配* 。

   尺度模式

   可选**自动**和**手动**两种模式，默认为**自动**。如果**自动**模式能满足需求则无需调节；如果不能满足要求，可切换为**手动**模式并调整**特征尺度**

   速度尺度

设置特征匹配速度。该值越大表示图像金字塔顶层的图像越小，相应提取的边缘点就越稀疏，进而加快特征匹配速度。取值范围为 1~20。

特征尺度

仅在*定位方式*设置为**高精度匹配**时可配置，表示提取边缘点的精细程度。取值需小于或等于**速度尺度**，且仅可为整数。取值为 *1* 时提取边缘点数量最多，精度最高 。

阈值模式

设置对比度调整模式，可选**自动**或**手动**，默认为**自动**。如果**自动**模式能满足需求则无需手动调整对比度；如果不能满足，可切换至**手动**模式并调整**对比度阈值**。

对比度阈值

此处“对比度”指特征点和背景的灰度值差距。对比度阈值越大，被淘汰的特征点越多。取值范围为 1~255。

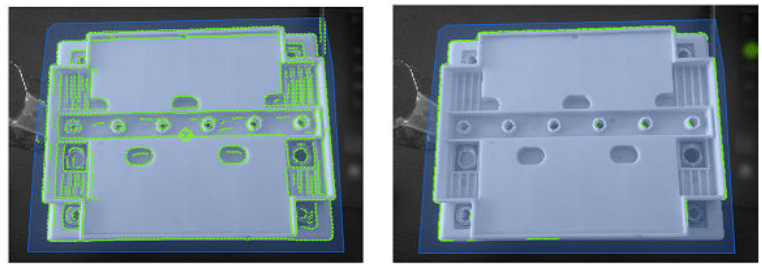


图 13-61 对比度阈值配置效果

旋转角度

掩膜与待检测图片之间的角度阈值。实际检测中，两者之间的角度超过该阈值时无法匹配。建模时若没有成功匹配到结果建议增大该参数取值。

旋转步长

取值范围为 0.1~1，建议使用默认值 0.5

投影区间

取值范围为 1~10，建模时若检测到的匹配点数量较少，建议增大该参数取值。

Mark 类型

设置掩膜的的细分类型。如果上文步骤中掩膜类型选择为**十字形 Mark 点**，此处可设置为**实心十字**或**空心十字**。

9. **可选操作**: 在掩膜列表单击 / 隐藏或显示掩膜。

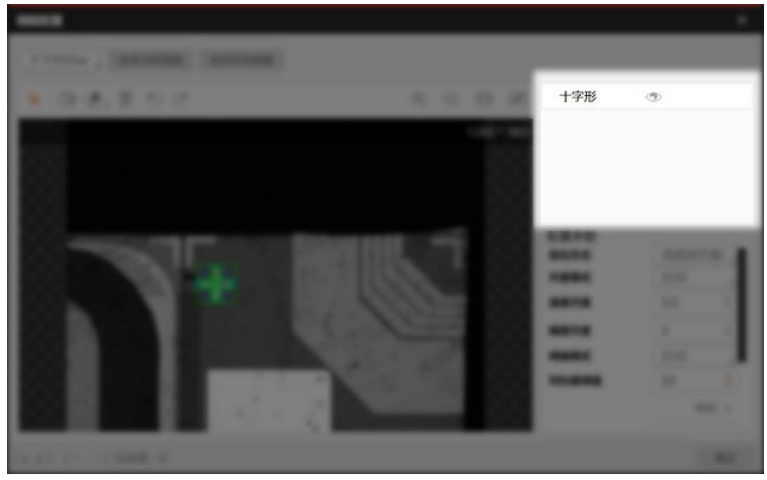


图 13-62 掩膜列表

10. 单击*模型配置*窗口右下角的*确定*，完成模型配置，并返回*特征模型*页签。

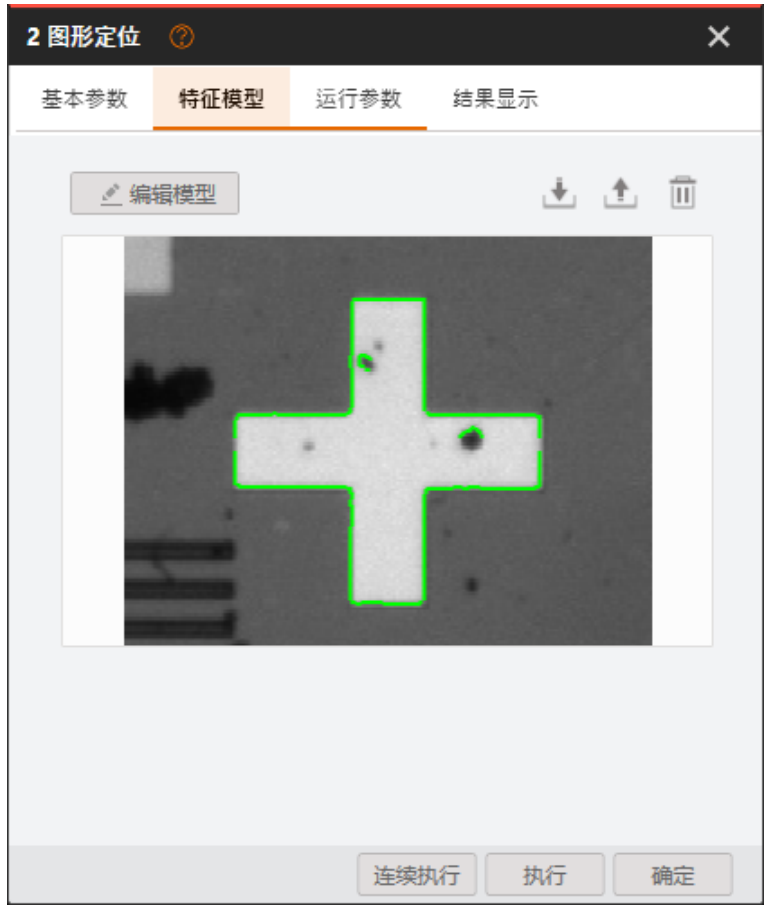


图 13-63 特征模型页签

11. **可选操作**: 在*特征模型*页签进行如下可选操作。

编辑模型　单击*编辑模型*继续编辑模型。

删除模型　单击 删除模型。

导出模型　单击 将当前模型导出至本地路径。

载入模型　单击 从本地路径载入模型。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 特征模板参数详情，请参见 *特征模型* 。
- 结果显示参数详情，请参见 *结果显示* 。

最小匹配分数

设置 *特征模型* 与待匹配图像中目标的相似度阈值。算法仅在相似度达到该阈值时，才会将该搜索到的目标判定为有效匹配结果。该参数最大可设置为 1，表示完全契合。默认为 0.5。

最大匹配个数

设置允许查找到的最大目标数量，默认值为 1，有效取值范围为 1~1000。

匹配极性

设置极性表示图形边缘到背景的颜色过渡情况（从黑到白或者从白到黑）。

如果查找目标的极性和 *特征模型* 的极性不一致时，仍要保证目标被查找到，则需将该参数设置为**不考虑极性**。如不需要则可设置为**考虑极性**，缩短查找时间。

角度范围

设置模板图像可相较水平方向的旋转角度范围（顺时针为正，逆时针为负）。默认值为-45~45，表示旋转角度范围为-45°至 45°。

搜索有旋转变化的目标时，需根据目标的旋转角度范围设置该参数（取值范围-180~180），否则可能导致匹配失败。

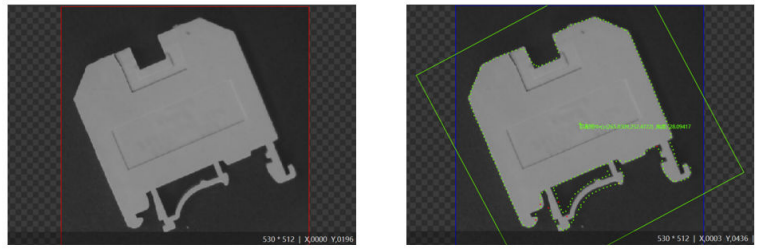


图 13-64 效果示例

定位类型

可以选择**直接映射**以及**二次修正**两种定位类型。**直接映射耗**时短，精度较低。**二次修正**耗时长，精度高。

直接映射

直接使用模板匹配的定位结果作为最终输出的图形定位。

二次修正

基于模板进行粗定位后，再对图形进行一次精定位，获取更精准的定位结果。

最大重叠率

设置搜索多个目标且其中两个匹配目标彼此重合时，两个匹配框的最大重叠比例。该参数取值越大表示两个目标可重叠的程度越大。取值范围 0~100，默认 50。当两个目标的重叠率大于所设置的参数值时，算法将舍弃相似度得分较低的目标，保留得分较高的目标。

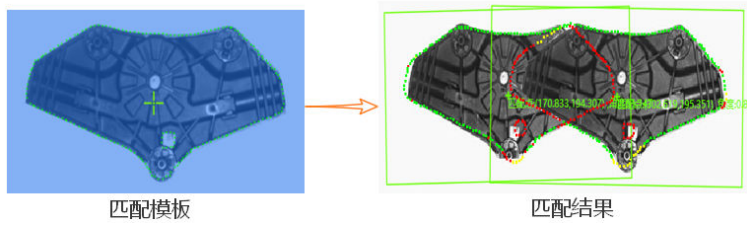


图 13-65 效果示例

排序类型

设置匹配结果的排序依据，如按分数降序排序、按角度降序排序、按 X 由小到大排序等。下图为按分数降序排序的示例。

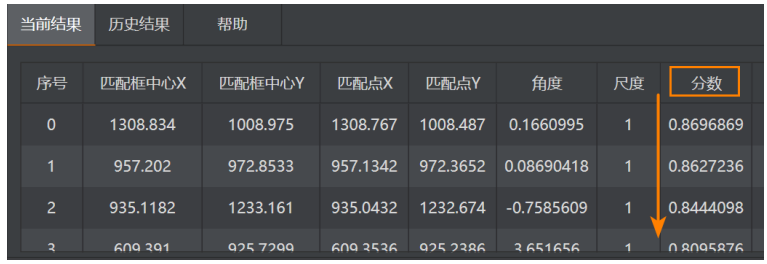


图 13-66 排序示例

按分数降序排序

按照特征匹配的得分降序排列。

按角度降序排序

按照当前结果里面相对角度偏移降序排列。

按 X 由小到大排序

按照匹配框中心的 X 轴坐标由小到大排序。

按 Y 由小到大排序

按照匹配框中心的 Y 轴坐标由小到大排序。

X 由小到大，Y 由小到大

按照匹配框中心点 X 轴坐标由小到大排序，当 X 坐标整数化后值相同时再按照 Y 轴坐标从小到大排序。

Y 由小到大，X 由小到大

按照匹配框中心点 Y 轴坐标由小到大排序，当 Y 坐标整数化后值相同时再按照 X 轴坐标从小到大排序。

阈值类型

设置边缘阈值的类型。边缘阈值指边缘对比度阈值，即轮廓点和背景的灰度值的差距阈值。仅大于该阈值的轮廓点可被提取。

自动阈值

根据目标图像自适应边缘阈值。

模板阈值

以模板的边缘阈值经过内部转换后作为匹配阶段的边缘阈值。

手动阈值

以手动设定的***边缘阈值***作为边缘阈值。

是否考虑杂斑

开启后，算法会考虑杂斑特征。如果特征存在毛刺，则评分降低。默认关闭。

如下图所示，是否考虑噪点会对匹配分值产生影响。（左：匹配模板，中：考虑噪点，右：不考虑噪点）。



图 13-67 效果示例

延拓阈值

延拓阈值为特征在图像边缘显示不全时，特征缺失的部分相对于完整的特征的比例。当被查找的目标被图像边缘截断时，设置延拓阈值可保证目标被找到。

下图中只要设置延拓阈值大于 40 时就可以保证图像上方被截断的目标被查找到。

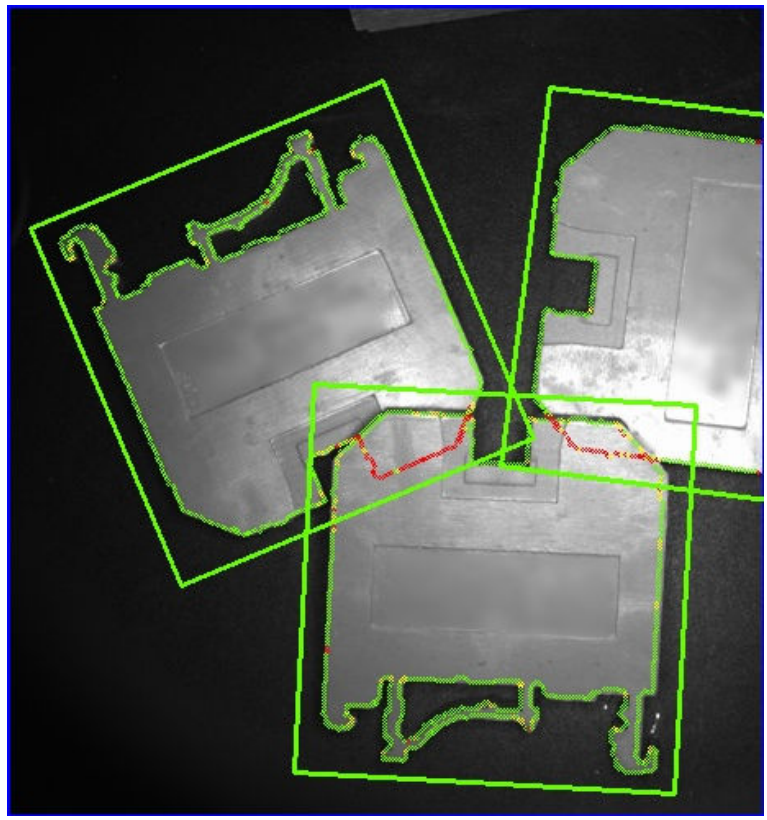


图 13-68 效果示例

超时控制

设置搜索超时的时长。搜索时间达到超时时长算法将停止搜索，不返回任何搜索结果。取值范围：0~10000，单位：ms。如设置为 0，则关闭超时控制功能。

轮廓使能

勾选后显示模板轮廓特征点。

不勾选则不显示特征点，只显示匹配框，可降低匹配耗时。

模块结果

*图形定位*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

直线状态

int 型，1 代表匹配成功，0 代表匹配失败。

匹配个数

int 型，代表图像中与模板相同或相似对象的个数。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

匹配点

float 型，代表每个匹配点。

匹配点 X

float 型，代表每个匹配点的 X 坐标。

匹配点 Y

float 型，代表每个匹配点的 Y 坐标。

输出直线

line 型，代表模块输出的直线。

直线起点

point 型，代表输出直线的起点坐标。

直线起点 X

float 型，代表输出直线起点的 X 坐标。

直线起点 Y

float 型，代表输出直线起点的 Y 坐标。

直线终点

point 型，代表输出直线的终点坐标。

直线终点 X

float 型，代表输出直线终点的 X 坐标。

直线终点 Y

float 型，代表输出直线终点的 Y 坐标。

尺度 X

float 型，代表匹配对象在 X 轴方向上的缩放尺度。

尺度 Y

float 型，代表匹配对象在 Y 轴方向上的缩放尺度。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

匹配轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不显示。

匹配轮廓点个数

int 型，代表图像中与模板相同或相似图像的轮廓点个数。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.6 位置修正

*位置修正*模块可根据模板匹配结果实时修正后序模块的 ROI。该模块常用于修正目标的运动偏移，实现精确定位。

模块原理

该模块的算法工作流程包括以下三个主要步骤：

1. 获取 *模板匹配* 输入的位置信息。
2. 基于输入的位置信息，在基准图像中创建基准点。
   例如在以下左图中选取右侧五角星上方的橙色点作为基准点。
3. 计算待修正图像中相对于基准点的偏移和旋转角度等信息。
   a. 在待修正图像中标注基准点。
      待修正图像中的基准点与基准图像中的基准点坐标相同。
   b. 基于 *模板匹配* 在待修正图像中匹配到的目标，在待修正图像中找出另一个点（以下称为“运行点”）。运行点相对目标的位置，与基准图像中基准点相对于目标的位置一致。
      例如以下右图中的黑点，其相对于五角星的位置与左图中的橙点一致。
   c. 在待修正图像中，对比运行点和基准点的位置，即可计算出待修正图像中的目标相对于基准图像的位姿变化信息。

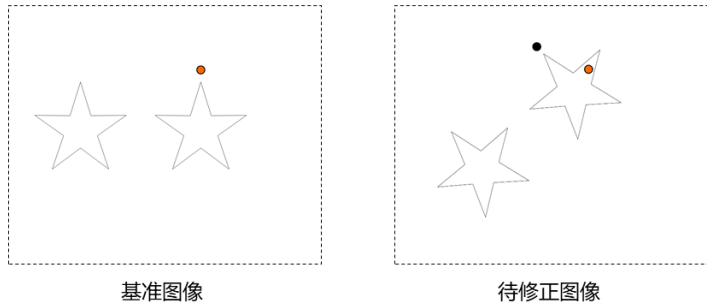


图 13-69 位置修正工作原理

使用方法

前序/后序模块

在流程中，*位置修正*模块的前序模块通常为 *图像源* 模块和 *轮廓匹配* 相关模块，后序模块通常为 *圆查找* 、*直线查找* 、*卡尺工具* 或 *图像修正* 等模块。*位置修正*可为这些后序模块修正 ROI，实现精确定位查找目标对象。

配置流程

*位置修正*模块的配置流程如下：

1. 单次执行流程以获取或更新前序模块输出至*位置修正*的数据。
2. 在*基本参数*页签选择相应的输入信息。
3. 单击*创建基准*。



图 13-70 配置步骤

应用示例

下图示例中，*位置修正*对*轮廓匹配*输出的 ROI 的位置进行修正后，*圆查找*通过修正后的 ROI 实现了对部件上的圆孔的精确查找。

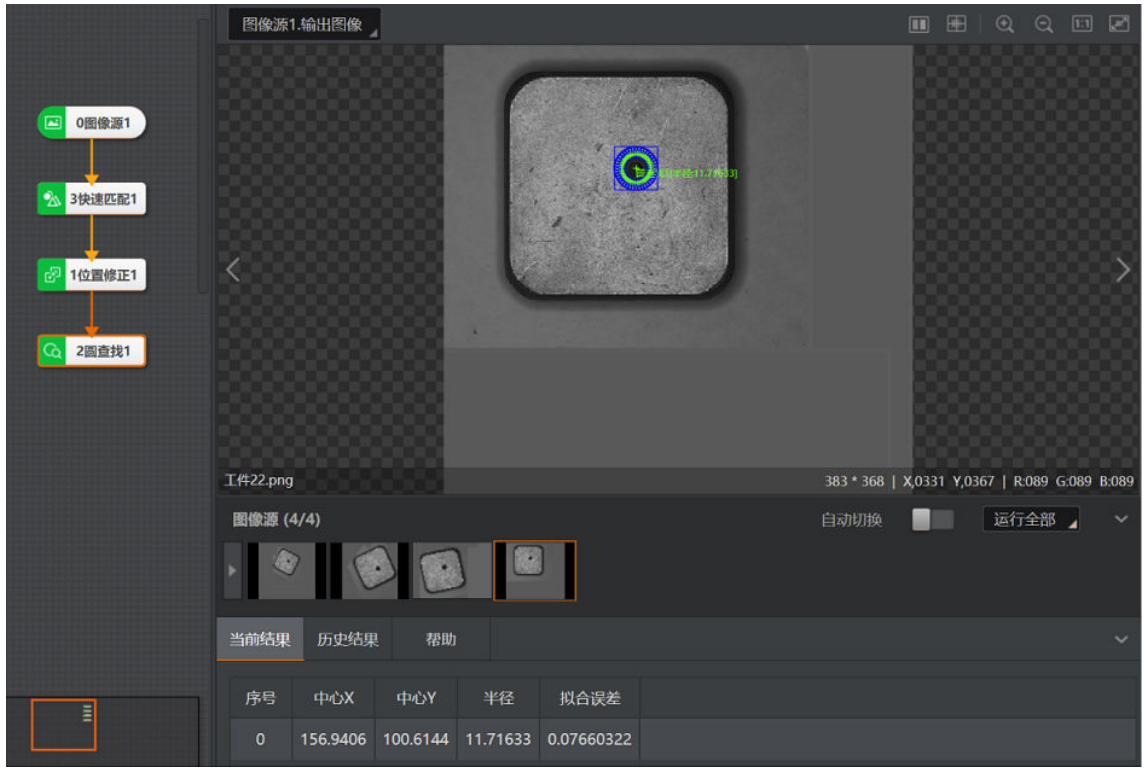


图 13-71 示例：精定位找圆

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

结果显示参数详情，请参见 *结果显示* 。

表 13-6 选择方式

子参数	次级子参数	描述
按点	原点	选择或输入基准点的位置。
	角度	原点基于水平线的角度。
	X 方向尺度	图像在 X 轴方向上的伸缩尺度。
	Y 方向尺度	图像在 Y 轴方向上的伸缩尺度。
按坐标	原点 X/Y	选择或输入基准点的 X/Y 轴坐标。

子参数	次级子参数	描述
	角度	原点基于水平线的角度。
	X 方向尺度	图像在 X 轴方向上的伸缩尺度。
	Y 方向尺度	图像在 Y 轴方向上的伸缩尺度。

注意事项

- 如果前序模块为 *快速匹配* 或 *灰度匹配*，上表中的 **X 方向尺度**和 **Y 方向尺度**无需订阅，此情况下无法实现同分辨率图像中不同大小目标的位置修正。
- 如果前序模块为 *轮廓匹配* 或 *高精度匹配*，*位置修正*模块自动订阅前者输出的尺度数据，实现同分辨率图像中不同大小目标的位置修正。

模块结果

*位置修正*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

位置修正信息

基准点

基准点 X

float 型，代表设置的基准点 X 坐标。

基准点 Y

float 型，代表设置的基准点 Y 坐标。

基准角度

float 型，代表基准点相对于水平线的角度，顺时针为正，逆时针为负。

基准尺度 X

float 型，代表基准点在 X 轴方向上的缩放尺度。

基准尺度 Y

float 型，代表基准点在 Y 轴方向上的缩放尺度。

运行点

运行点 X

float 型，代表查找到的运行点的 X 坐标。

运行点 Y

float 型，代表查找到的运行点的 Y 坐标。

运行角度

float 型，代表运行点相对于水平线的角度，顺时针为正，逆时针为负。

运行尺度 X

float 型，代表运行点在 X 轴方向上的缩放尺度。

运行尺度 Y

float 型，代表运行点在 Y 轴方向上的缩放尺度。

13.2.7 Blob 分析

*Blob 分析*模块可在二值图像中定位或分析目标物体。二值图像指仅包含纯黑（灰度值 0）和纯白（灰度值 255）两种颜色的图像。该模块可输出目标物体的存在性、数量、位置、形状、方向以及目标物体间的拓扑关系等信息。

本节内容包含：

- *模块原理*
- *使用场景*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

开始使用 *Blob 分析*模块前，建议先了解该模块的算法工作原理，以便更合理地配置该模块的参数。

本节内容包含：

- *什么是 Blob*
- *Blob 分析流程*

什么是 Blob

Blob，即图像上由互相连通的像素点构成的区域。此处的“连通”分为如下两种类型：

- **4 连通**：两个像素点有共同的边缘，即一个像素点在另一个像素的上方、下方、左侧或右侧。
- **8 连通**：在 **4 连通**的基础上，将对角线上的相邻像素也纳入为“互相连通”的像素。

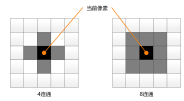


图 13-72 4 连通和 8 连通

Blob 分析流程

Blob 分析的基本流程如下图所示。

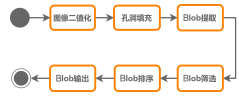


图 13-73 Blob 分析

图像二值化

图像二值化指将图像中的像素点转换为纯黑（灰度值 0）或纯白（灰度值 255）。转换后，图像被分割为黑白两部分。转换过程中，灰度值转换为 0 和 255 的像素点具体由预设的像素阈值（即特定的灰度值）决定。具体的转换机制由下文*参数配置*中的 ***阈值方式*** 定义。

合理设置阈值可让 Blob 分析得到更好的分析结果。

孔洞填充

此处的“孔洞”指被 Blob 包围且与 Blob 颜色相反的部分。Blob 为白色，孔洞为黑色。孔洞填充在 Blob 分析中是否执行、如何执行，由下文*参数配置*中提及的 ***填充面积阈值*** 决定。

Blob 提取

算法根据连通域特性提取 Blob，连通特性包括上文已提及的 4 连通或 8 连通，可通过下文*参数配置*中提及的 ***连通性*** 配置。

Blob 筛选

算法根据预设的条件筛选 Blob。Blob 分析过程中具体如何进行筛选，由下文 ***参数配置*** 中提及的各项“使能”参数决定。合理配置这些参数，可获取更为精准的目标 Blob。

Blob 排序

算法对上一步筛选出的 Blob 按照特定特征进行排序。Blob 分析过程中具体如何进行排序，由下文*参数配置*中提及的 ***排序特征*** 和 ***排序方式*** 决定。

Blob 输出

算法返回如下信息：

- 二值化图像
- Blob 图像

- 符合预设筛选条件的 Blob 数量
- 符合预设筛选条件的每个 Blob 的基本特征信息以及特征加权得分。

使用场景

*Blob 分析*模块仅适用于特定的业务场景。

如果业务场景满足如下所述的适用情况，建议在流程中调用该模块进行 Blob 分析。

适用情况	不适用情况
• 仅需提取检测对象的二维特征 • 检测图像的对比度高 • 需判断检测对象是否存在 • 检测图像的缩放尺度固定且不存在图像旋转	• 检测图像的对比度低 • 仅使用两个灰度级不能反映待检测对象的基本属性

使用方法

在流程中，*Blob 分析*模块的前后序模块详情如下。

表 13-7 前后续模块

前后序模块	描述
前序模块	*图像源*、*表面缺陷滤波*以及*深度学习*分类下的*DL 快速图像分割*和*DL 实例分割*等。该两种“分割”模块，输出图像一般都是概率图。概率图中，概率值越高的地方图像越亮，概率值越低的地方，图像越暗。因此通常需要将这些模块输出图像进行 Blob 分析处理，获取目标位置信息。
后序模块	可处理 Blob 分析输出数据的模块，如*格式化*、*条件检测*以及*测量*分类下的模块。

下图所示的计数场景中，*格式化*模块将 *Blob 分析*处理后的数据做格式化处理，屏蔽无关数据，最终仅输出图中的 Blob 数量。

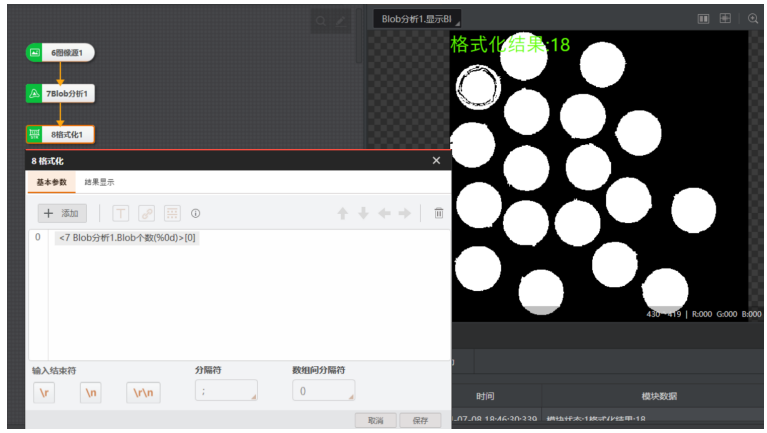


图 13-74 示例 1：格式化 Blob 数据

下图的胶囊计数检测场景中，*条件检测*模块通过 *Blob 分析*模块获取胶囊个数，并判断胶囊个数是否满足预设的条件，最终输出判断结果。

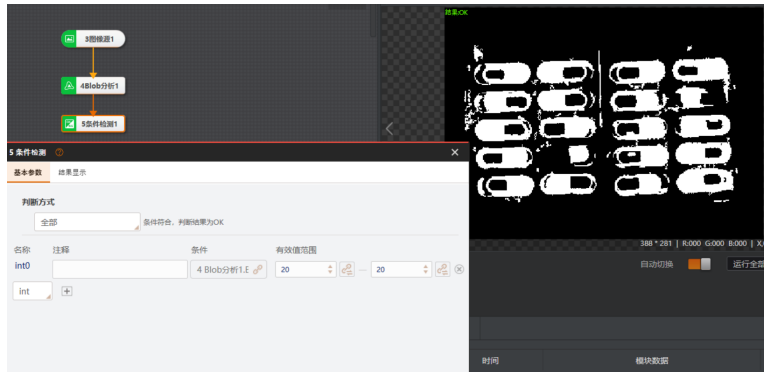


图 13-75 示例 2：检测胶囊个数是否达标

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数*。
- 结果显示参数详情，请参见 *结果显示*。

阈值方式

定义图像二值化的阈值设置方式，包括**不进行二值化**、**单阈值**、**双阈值**、**自动阈值**等。图像二值化的结果将直接影响 Blob 分析结果，请根据实际场景谨慎设置**阈值方式**。

不进行二值化

适用于输入图像本身已为二值图的场景。

单阈值

通过单阈值算法进行图像二值化，需设置**低阈值**。该算法适用于对算法效率要求较高或需对较大检测目标进行二值化的场景。该算法二值化处理速度快，但存在空间量化误差，相对精度较低。

不同**极性**设置下，二值化效果有所差异。**极性**用于指定在二值化之前，灰度图中前景的灰度是否亮于背景。

- **极性**设置为**暗于背景**时，灰度值小于**低阈值**的像素点被分割为前景，其余像素点为背景。
- **极性**设置为**亮于背景**时，灰度值大于或等于**低阈值**的像素点被分割为前景，其余像素点为背景。

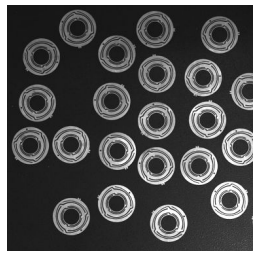


图 13-76 原图

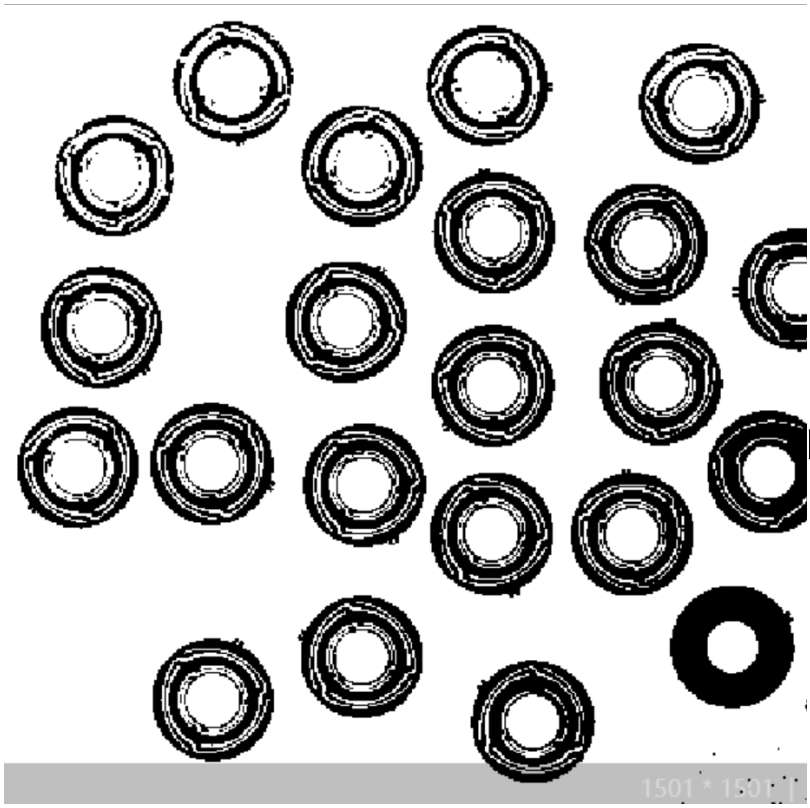


图 13-77 单阈值且极性为暗于背景

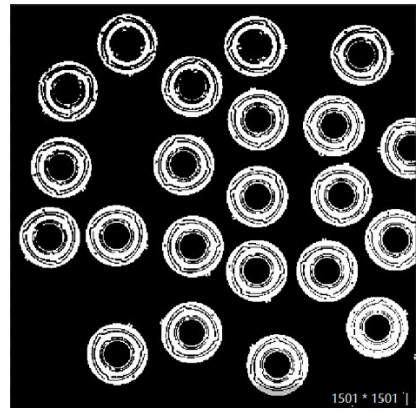


图 13-78 单阈值极性为亮于背景

双阈值

通过双阈值算法进行图像二值化。需设置两个阈值(**低阈值**和**高阈值**)。该算法适用于对算法效率要求较高或需对较大检测目标进行二值化的场景。该方式二值化处理速度快，但存在空间量化误差，相对精度较低。

- 当**低阈值**小于**高阈值**时，灰度值介于两个阈值之间的像素点将被分割为前景，其余像素点为背景。
- 当**低阈值**大于**高阈值**时，灰度值在两个阈值之外的像素点将被分割为前景，其余像素点为背景。

自动阈值

算法根据图像自动配置阈值进行图像二值化，适用于需要根据输入图像动态计算阈值的场景。不同**极性**设置下，二值化效果有所差异。**极性**用于指定在二值化之前，灰度图中前景的灰度是否亮于背景。

软阈值（固定）

软阈值算法是相对于硬阈值算法的概念，分为固定软阈值算法和相对软阈值算法。软阈值算法可降低由空间量化误差（Spatial Quantization Error，SQE）导致的 Blob 大小、轮廓长度和位置的不精确问题。

说明

硬阈值算法，即上文已提及的**单阈值**、**双阈值**和**自动阈值**。

下图展示了硬阈值算法与软阈值算法的核心区别。为了降低空间量化误差对算法输出结果的影响，软阈值算法给位于感兴趣区域内的每个像素点分配一个像素权重。像素权重为 0，表明该像素是一个背景像素；像素权重为 1，表明该像素是一个前景像素；像素权重在 0 和 1 之间，表明该像素位于前景和背景区域的边界。

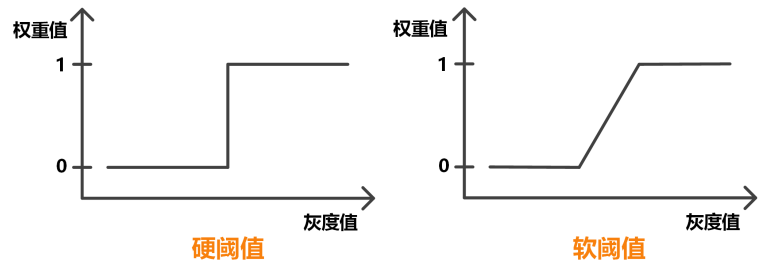


图 13-79 软阈值和硬阈值的区别

软阈值（固定）即固定软阈值算法，适用于目标和背景之间可进行简单二值划分的场景。该算法耗时相对较长。

通过该算法进行图像二值化，需设置**阈值范围**和**软阈值柔和度**。不同**极性**设置下，二值化效果有所差异。**极性**用于指定在二值化之前，灰度图中前景的灰度是否亮于背景。

- **极性**为**暗于背景**时，灰度值小于等于低阈值的像素权重为 1（为前景）；大于等于高阈值的像素权重为 0（为背景）；介于两阈值之间的像素权重在 0 和 1 之间，即为前景和背景区域的边界。
- **极性**为**亮于背景**时，灰度值小于等于低阈值的像素点的权重为 0（为背景）；大于等于高阈值的像素点的权重为 1（为前景）；介于两阈值之间的像素权重在 0 和 1 之间，即为前景和背景区域的边界。

软阈值柔和度

设置像素权重值 0 和 1 之间的阶梯个数（如下图所示）。该值为 0 时，软阈值处理结果等价于**硬阈值**。



图 13-80 柔和度原理

柔和度越高，则 Blob 边缘越柔和，毛刺感越弱（如下图所示）。

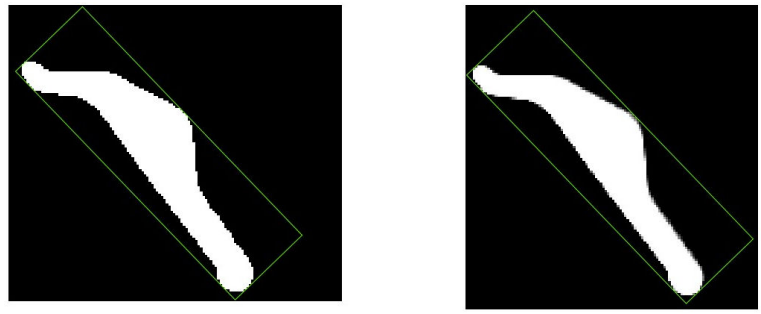


图 13-81 柔和度比对效果

软阈值（相对）

通过相对软阈值算法进行图像二值化。该二值化方法适用于照明环境不稳定的场景。该算法耗时相对较长。

选用该算法，需设置**低尾部比例**、**高尾部比例**、**低阈值比例**、**高阈值比例**和上文已提及的**软阈值柔和度**。

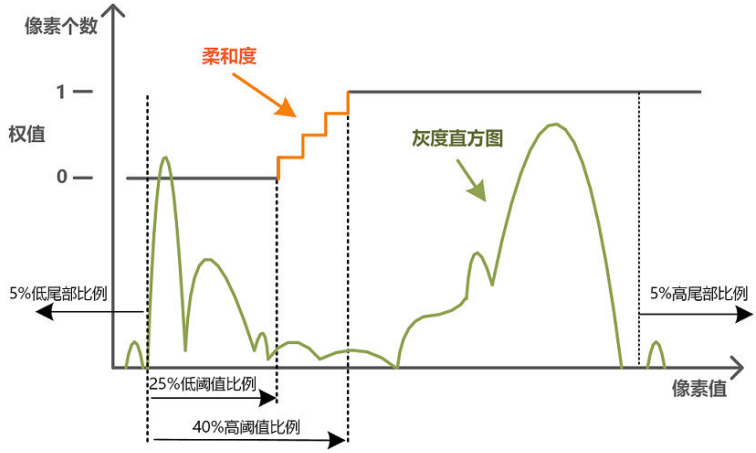


图 13-82 相对软阈值算法原理

相对软阈值算法的工作流主要包括两个步骤。以下结合上图对该两个步骤进行说明。

1. 通过**高/低尾部比例**排除具有极端灰度值的若干像素点。如果图像的像素总数为 N，**高/低尾部比例**分别为 H%和 L%，则图像中灰度值最低的 N×L%个像素和灰度值最高的 N*H%个像素分别赋予 0 和 1 的权重值。
2. 在剩余像素点的灰度值范围内划分高/低阈值。如果剩余像素点的灰度值分布在[A, B]的范围内，**低阈值比例**为 TL%，**高阈值比例**为 TH%，则低阈值为（B-A）* TL% + A，高阈值为（B-A）* TH% + A。
   - **极性**为**亮于背景**时，灰度值小于等于低阈值的像素权重为 0（为背景），大于等于高阈值的像素权重为 1（为前景），介于两阈值之间的像素权重在 0 和 1 之间。
   - **极性**为**暗于背景**时，灰度值小于等于低阈值的像素权重为 1（为前景），大于等于高阈值的像素权重为 0（为背景），介于两阈值之间的像素权重在 0 和 1 之间。

低尾部比例

起始于 0 灰度值的较低灰度值像素点相对于像素点总数的比例。这些较低灰度值像素点将在图像二值化中忽略。

高尾部比例

终止于 255 灰度值的较高灰度值像素点相对于像素点总数的比例。该部分像素点将在图像二值化中忽略。

低阈值比例

去除**低尾部比例**和**高尾部比例**对应的像素点后，灰度值低于低阈值的像素点相对所有剩余像素点的比例。

高阈值比例

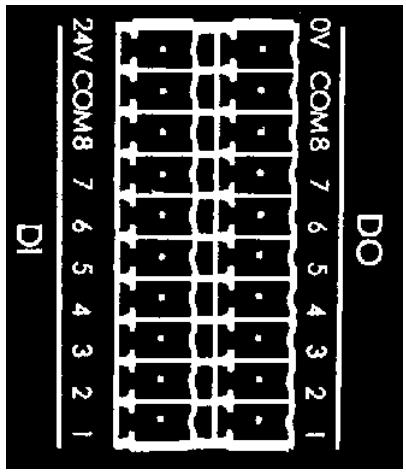
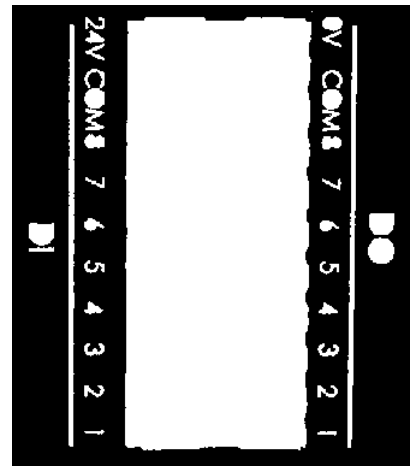
去除**低尾部比例**和**高尾部比例**对应的像素点后，灰度值在最低灰度值至高阈值之间的像素点相对所有剩余像素点的比例。

查找个数

设置最终输出的 Blob 的最大数量。

填充面积阈值

Blob 区域内的最小非 Blob 区域面积（可理解为是 Blob 中的“孔洞”）。如果“孔洞”面积小于或等于该值，则将其与外围的 Blob 合并。

孔洞填充前	孔洞填充后
	

面积使能

开启后，算法仅筛选出面积在**面积范围**内的 Blob。

输出轮廓类型

选择输出轮廓类型。可选项说明如下。

不输出轮廓

图像上不显示 Blob 的轮廓。

输出无序轮廓

选择后，模块将输出 Blob 的无序轮廓。所有轮廓点之间无固定顺序关系，且无法形成连续路径。

输出有序轮廓

选择后，模块将输出 Blob 的有序轮廓。所有轮廓点按空间相邻顺序进行排列，可形成闭合或开放的连续路径。该类型适用于对轮廓点前后顺序有特定要求的检测场景，如胶路检测。

Blob 图像输出

关闭后，算法将不输出 Blob 分析后的图像。

二值化图输出

关闭后，算法将不输出 Blob 分析后的二值化图像。

高级参数

直方图信息输出

开启后，可输出亮度分布直方图信息。

角度使能

开启后，算法仅筛选出角度在**角度范围**内的 Blob。此处的角度指 Blob 的“最小外接矩形”的长轴与图像 x 轴方向的夹角（顺时针为正，逆时针为负的）。

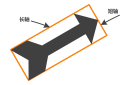


图 13-83 最小外接矩形

RECT 宽使能

开启后，算法仅筛选出 Blob 矩形框宽度在 **RECT 宽范围**内的 Blob。

说明

此处的“矩形框”与 *图 13-83* 不同，特指宽高均与图像宽高平行的矩形（如下图所示）。

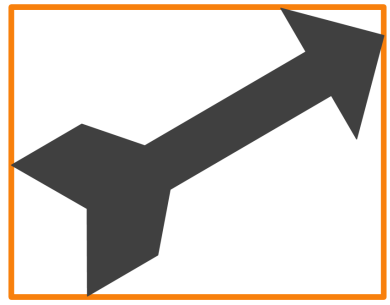


图 13-84 Blob 矩形框

RECT 高使能

开启后，算法仅筛选出矩形框高度在 **RECT 高范围**内的 Blob。

说明

此处的“矩形框”与 *图 13-83* 不同，特指宽高均与图像宽高平行的矩形。

周长使能

开启后，算法仅筛选出轮廓长度在**周长范围**内的 Blob。

短轴使能

开启后，算法仅筛选出 *图 13-83* 短轴长在**短轴范围**内的 Blob。

长轴使能

开启后，算法仅筛选出 *图 13-83* 长轴长度在**长轴范围**内的 Blob。

圆形度使能

开启后，算法仅筛选出圆形度在**圆形度范围**内的 Blob。

$$circularity = \frac{S}{\pi d^2}$$

图 13-85 圆形度计算公式

该计算公式中，circularity 为圆形度，S 为 Blob 面积（即连通域面积），d 为 Blob 质心到 Blob 轮廓点的最大距离。圆形度范围为 0~1，圆的圆形度为 1。圆形度越小，Blob 形状越不规则。

说明

质心为 Blob 内的所有像素点的坐标平均值所对应的点。



图 13-86 圆形度

矩形度使能

开启后，算法仅筛选出矩形度在**矩形度范围**内的 Blob。

$$R = \frac{S_1}{S_2}$$

图 13-87 矩形度计算公式

该计算公式中， R 为矩形度， S1 为 Blob 面积（即连通域面积）， S2 为 Blob 的 *图 13-83* 的面积。矩形度范围是 0~1，矩形的矩形度为 1。矩形度越小，Blob 形状越不规则。

质心偏移使能

开启后，算法仅筛选出质心偏移在**质心偏移范围**内的 Blob。

质心为 Blob 内的所有像素点的坐标平均值所对应的点，如下图中蓝点所示；矩形中心是 Blob 的 *图 13-83* 的几何中心，如下图中红点所示。两点之间的距离即为质心相对于矩形中心的偏移量。

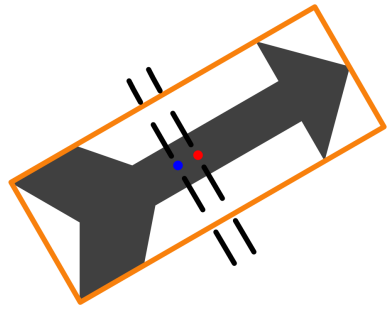


图 13-88 质心偏移

轴比范围

开启后，算法仅筛选出 *图 13-83* 短轴长度和长轴长度比值在**轴比范围**内的 Blob。

排序特征

设置输出的 Blob 在*当前结果*页签内的排序依据。可依据面积、周长、圆形度、矩形度、连通域中心 x、连通域中心 y、box 角度、box 宽、box 高、矩形左上顶点 x、矩形左上顶点 y、二阶中心距主轴角度、轴比等特征排序。

实际配置效果与下文的**排序方式**相关。

排序方式

按升序或降序对输出的 Blob 在在*当前结果*页签内进行排序。也可设置为**不排序**。

连通性

设置判断图像上像素点是否互相连通的标准，包括 **4 连通**和 **8 连通**。通常设置为 **8 连通**可比设置为 **4 连通**获得更大的 Blob。

以下示例中，如果采用 **4 连通**定义“连通”，则下图白色像素点构成的特征图像包含两个“连通区域”，即如下左图中两个橙框中的由白色像素点构成的区域；如果采用 **8 连通**定义，则仅包含一个“连通区域”，即如下右图中所有由白色像素点构成的区域。

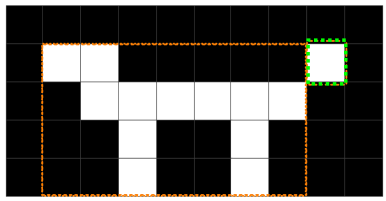


图 13-89 两个连通区域

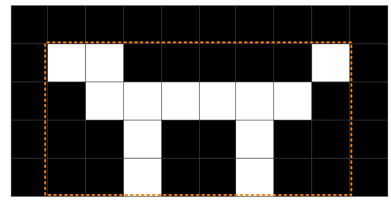


图 13-90 单个连通区域

最大内接矩形输出数量

可设置计算多少个 Blob 的最大内接矩形。

像素超界使能

开启后，如果 Blob 中与图像边缘、ROI 边缘或掩膜边缘重合的像素点数量大于**最大超界像素**，那么该 Blob 将被算法删除。

如果将**最大超界像素**设置为 0，所有与图像边缘、ROI 边缘或掩膜边缘相交的 Blob 都会被删除。

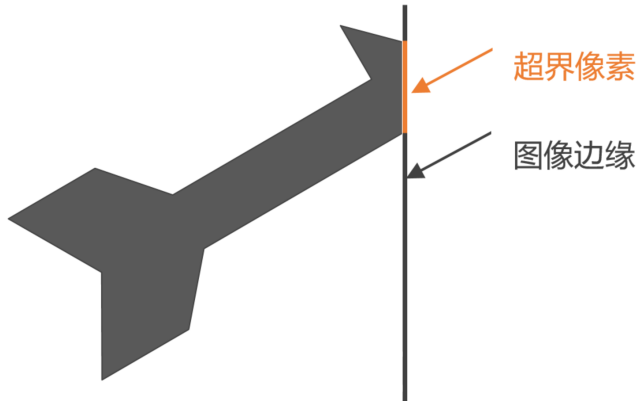


图 13-91 像素超界

模块结果

*Blob 分析*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

单体状态

int 型，代表各个 Blob 的状态。1 表示 Blob 识别成功，且出现 1 的次数表示检测出的 Blob 个数。

Blob 个数

int 型，代表图像中检测出的 Blob 个数。

面积

float 型，代表各个 Blob 的面积大小。

总面积

float 型，代表所有 Blob 加起来的总面积。

质心点

point 型，代表 Blob 质心。

质心 X

float 型，代表 Blob 质心的 X 坐标。

质心 Y

float 型，代表 Blob 质心的 Y 坐标。

最小外接矩形

box 型，表示能完整包含一个图形或目标区域的最小矩形。

中心点

point 型，表示最小外接矩形的中心点。

中心 X

float 型，代表最小外接矩形中心点的 X 坐标。

中心 Y

float 型，代表最小外接矩形中心点的 Y 坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

最小灰度值

float 型，代表灰度值的最小值。

最大灰度值

float 型，代表灰度值的最大值。

对比度

float 型。使用自动二值化将区域分为较亮和较暗的两个部分，分别计算其灰度均值。两个灰度均值差即为对比度。

最大内接矩形

box 型，代表表示能被一个图形或目标区域包含的最大矩形。

最大内接矩形中心点

point 型，代表最大内接矩形的中心点。

最大内接矩形中心点 X

float 型，代表最大内接矩形中心点的 X 坐标。

最大内接矩形中心点 Y

float 型，代表最大内接矩形中心点的 Y 坐标。

最大内接矩形宽度

float 型，代表最大内接矩形的宽度。

最大内接矩形高度

float 型，代表最大内接矩形的高度。

最大内接矩形角度

float 型，代表最大内接矩形长边基于水平线的旋转角度，顺时针为正，逆时针为负。

Blob 矩形框

矩形框中心点

Blob 矩形框 X

float 型，代表矩形框中心点的 X 轴坐标。

Blob 矩形框 Y

float 型，代表矩形框中心点的 Y 轴坐标。

Blob 矩形框宽度

float 型，代表矩形框的宽度。

Blob 矩形框高度

float 型，代表矩形框的高度。

Blob 矩形左上点 X

float 型，代表 Blob 矩形框左上点的 X 轴坐标。

Blob 矩形左上点 Y

float 型，代表 Blob 矩形框左上点的 Y 轴坐标。

Blob 矩形右下点 X

float 型，代表 Blob 矩形框右下点的 X 轴坐标。

Blob 矩形右下点 Y

float 型，代表 Blob 矩形框右下点的 Y 轴坐标。

主轴角度

float 型，代表 Blob 矩形框基于水平线旋转的角度。顺时针旋转为正，逆时针旋转为负。

周长

float 型，代表各个 Blob 的轮廓周长。

长轴

float 型，代表各个 Blob 的最小外接矩形中较长边的长度。

最小外接矩形定义参见 *图 13-83* 。

短轴

float 型，代表各个 Blob 的最小外接矩形中较短边的长度。

圆形度

float 型，代表检测出的各个 Blob 的圆形度。

圆形度定义参见 *圆形度使能* 。

矩形度

float 型，代表各个 Blob 的面积与其相对应的最小外接矩形面积的比值。

分数

float 型，代表 Blob 分析的得分。

二值化图像数据

IMAGE 型，表示经过二值化的图像数据。

二值化图像数据

image 型，代表经过二值化的图像数据。该类型的图像数据为二进制形式，故模块结果处不显示任何数据。

二值化图像宽度

int 型，代表二值化图像的宽度。

二值化图像高度

int 型，代表二值化图像的高度。

二值化图像像素格式

int 型，代表二值化图像的像素格式。其中 17301505 对应 Mono8 格式。

Blob 图像数据

IMAGE 型，代表模块输出的 Blob 图像。

Blob 图像数据

image 型，代表通过 Blob 分析后输出的图像。该类型的图像数据为二进制形式，故模块结果处不显示任何数据，您可在输出图像中查看该图像。

Blob 图像宽度

int 型，代表输出 Blob 图像的宽度。

Blob 图像高度

int 型，代表输出 Blob 图像的高度。

Blob 图像像素格式

int 型，代表输出 Blob 图像的像素格式。其中，“17301505”对应 Mono 8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

轮廓点集

point 型，代表匹配对象的轮廓点的集合。

轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不在界面显示。您可以通过二次开发中的相关接口获取轮廓点集数据。

轮廓点个数

int 型，代表图像中匹配对象的轮廓点个数。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.8 Blob 分析组合

*Blob 分析组合*模块可识别和分析二值图像中的连通区域（Blob）并且输出该区域的数量、面积、位置、周长以及灰度直方图等信息。相较于 *Blob 分析*模块，*Blob 分析组合*模块进一步强化了运行参数配置功能，支持独立设置多个 ROI 的运行参数。

说明

该模块的原理、使用场景及使用方法与 *Blob 分析*基本一致。唯一区别在于该模块支持为每个 ROI 独立设置运行参数，从而实现不同区域的差异化检测。有关模块原理、应用场景及使用方法的详细说明，请参见 *Blob 分析* 。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模块运行参数

按需开启图像输出使能并选择是否将 *每 ROI 运行参数*区域配置的参数统一应用于所有 ROI。

Blob 图像输出

开启后，模块将对输入图像和输入掩膜执行与运算，并输出运算后的图像。

说明

开启该参数前，需开启 *基本参数* 下的**输入掩膜使能**，并选择输入掩膜图像。

二值化图像输出

开启后，模块将输出 Blob 分析后的二值化图像。

第一组 ROI 运行参数使能

开启后，所有 ROI 将统一应用当前 *每 ROI 运行参数*区域设置的运行参数；关闭后，可在**ROI 序号**选择目标 ROI，并在 *每 ROI 运行参数*区域逐一配置当前 ROI 对应的运行参数。

ROI 序号

用于指定需要配置独立运行参数的目标 ROI 序号。

可选序号对应 *Blob 分析组合*模块通过*绘制*方式创建的 ROI 数量。当 **Blob 分析组合**模块的 *ROI 创建方式*为**继承**时，实际生效的 ROI 数量遵循“取少原则”：

- 继承数量 > 绘制数量：仅前 N 个（N=绘制的 ROI 数量）继承的 ROI 支持配置运行参数，且仅前 N 个 ROI 生效。
- 继承数量 = 绘制数量：所有继承的 ROI 均支持配置运行参数，且全部 ROI 生效。
- 继承数量 < 绘制数量：N 个（N=绘制的 ROI 数量）ROI 支持配置运行参数，但仅前 M 个（M=继承的 ROI 数量）继承的 ROI 生效。

每 ROI 运行参数

该模块的运行参数与 *Blob 分析*基本一致，仅不支持设置**最大内接矩形输出数量**，且**输出轮廓类型**中不包含**有序轮廓**选项。更多关于运行参数的介绍，请参见 *Blob 分析* 下的 *参数配置* 。

模块结果

*Blob 分析组合*模块的模块结果如下。

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

ROI 个数

int 型，代表图像上绘制的 ROI 的数量。

单个 ROI 状态

int 型，代表图像上各个 ROI 的状态。1 表示该 ROI 中已检测到 Blob；0 表示该 ROI 中未检测到 Blob。

Blob 个数

int 型，代表图像中检测出的 Blob 个数。

单个 ROI 的 Blob 个数

int 型，代表各 ROI 中检测到的 Blob 的数量。

Blob 对应的 ROI 序号

int 型，代表各个 Blob 所对应的 ROI 的编号。

单体状态

int 型，代表各个 Blob 的状态。1 表示 Blob 识别成功，且出现 1 的次数表示检测出的 Blob 个数。

面积

float 型，代表各个 Blob 的面积大小。

总面积

float 型，代表所有 Blob 加起来的总面积。

单个 ROI 的总面积

float 型，代表各 ROI 中 Blob 的总面积。

面积占比

float 型，代表各 ROI 中检测到的每个 Blob 的面积与该 Blob 所在 ROI 面积的比值。此处输出的面积占比个数与当前图像上检测到的 Blob 总数一致。

总面积占比

float 型，代表面积占比的总和。

单个 ROI 的总面积占比

float 型，代表各 ROI 中 Blob 的总面积与 ROI 总面积的比值。此处输出的面积占比个数与当前图像上绘制的 ROI 总数一致。

ROI 面积

float 型，代表当前图像上所有 ROI 的总面积。

质心点

point 型，代表 Blob 质心。

质心 X

float 型，代表 Blob 质心的 X 坐标。

质心 Y

float 型，代表 Blob 质心的 Y 坐标。

最小外接矩形

box 型，表示能完整包含一个图形或目标区域的最小矩形。

中心点

point 型，表示最小外接矩形的中心点。

中心 X

float 型，代表最小外接矩形中心点的 X 坐标。

中心 Y

float 型，代表最小外接矩形中心点的 Y 坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

最小灰度值

float 型，代表检测区域图像的最小灰度值。

最大灰度值

float 型，代表检测区域图像的最大灰度值。

对比度

float 型。使用自动二值化将区域分为较亮和较暗的两个部分，分别计算其灰度均值。两个灰度均值差即为对比度。

平均灰度值

float 型，代表每个检测到的 Blob 区域内所有像素灰度值的平均值。

像素个数

float 型，代表每个检测到的 Blob 区域内的像素个数。

灰度中值

float 型，代表每个检测到的 Blob 区域内所有像素灰度值的中位数。

标准差

float 型，代表每个检测到的 Blob 区域内所有像素灰度值的标准差。

灰度众数

float 型，代表每个检测到的 Blob 区域内所有像素灰度值的众数。

直方图

int 型，此为长度是 256 的数组，代表在各个 Blob 中每个灰度值所对应的像素个数。

累积直方图

int 型，代表 256 定长数组，检测区域内从 0 开始递增累计的像素点数。

Blob 矩形框

Rect 型，代表检测到的 Blob 矩形框。

矩形框原点

point 型，代表各个 Blob 矩形框的原点坐标。

Blob 矩形框原点 X

float 型，代表各个矩形框原点的 X 轴坐标。

Blob 矩形框原点 Y

float 型，代表各个矩形框原点的 Y 轴坐标。

Blob 矩形框宽度

float 型，代表各个矩形框的宽度。

Blob 矩形框高度

float 型，代表各个矩形框的高度。

Blob 矩形左上点 X

float 型，代表各个 Blob 矩形框左上点的 X 轴坐标。

Blob 矩形左上点 Y

float 型，代表各个 Blob 矩形框左上点的 Y 轴坐标。

Blob 矩形右上点 X

float 型，代表各个 Blob 矩形框右上点的 X 轴坐标。

Blob 矩形右上点 Y

float 型，代表各个 Blob 矩形框右上点的 Y 轴坐标。

Blob 矩形左下点 X

float 型，代表各个 Blob 矩形框左下点的 X 轴坐标。

Blob 矩形左下点 Y

float 型，代表各个 Blob 矩形框左下点的 Y 轴坐标。

Blob 矩形右下点 X

float 型，代表各个 Blob 矩形框右下点的 X 轴坐标。

Blob 矩形右下点 Y

float 型，代表各个 Blob 矩形框右下点的 Y 轴坐标。

主轴角度

float 型，代表 Blob 矩形框基于水平线旋转的角度。顺时针旋转为正，逆时针旋转为负。

周长

float 型，代表各个 Blob 的轮廓周长。

长轴

float 型，代表各个 Blob 的最小外接矩形中较长边的长度。

最小外接矩形定义参见 *图 13-83* 。

短轴

float 型，代表各个 Blob 的最小外接矩形中较短边的长度。

圆形度

float 型，代表检测出的各个 Blob 的圆形度。

圆形度定义参见 *圆形度使能* 。

矩形度

float 型，代表各个 Blob 的面积与其相对应的最小外接矩形面积的比值。

分数

float 型，代表 Blob 分析的得分。

二值化图像数据

IMAGE 型，表示经过二值化的图像数据。

二值化图像数据

image 型，代表经过二值化的图像数据。该类型的图像数据为二进制形式，故模块结果处不显示任何数据。

二值化图像宽度

int 型，代表二值化图像的宽度。

二值化图像高度

int 型，代表二值化图像的高度。

二值化图像像素格式

int 型，代表二值化图像的像素格式。其中 17301505 对应 Mono8 格式。

Blob 图像数据

IMAGE 型，代表模块输出的 Blob 图像。

Blob 图像数据

image 型，代表通过 Blob 分析后输出的图像。该类型的图像数据为二进制形式，故模块结果处不显示任何数据，您可在输出图像中查看该图像。

Blob 图像宽度

int 型，代表输出 Blob 图像的宽度。

Blob 图像高度

int 型，代表输出 Blob 图像的高度。

Blob 图像像素格式

int 型，代表输出 Blob 图像的像素格式。其中，“17301505”对应 Mono 8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

轮廓点集

point 型，代表匹配对象的轮廓点的集合。

轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不在界面显示。您可以通过二次开发中的相关接口获取轮廓点集数据。

轮廓点个数

int 型，代表图像中匹配对象的轮廓点个数。

轮廓点

point 型，代表输出图像上的所有轮廓点坐标。

轮廓点 X

float 型，代表每个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表每个轮廓点的 Y 坐标。

轮廓点分数

float 型，代表轮廓点集中各个轮廓点的分数。

轮廓点序号

float 型，代表轮廓点集中各个轮廓点的编号。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形

polygon 型，代表输出图像上的多边形。

多边形点数

int 型，代表多边形的顶点个数。

多边形点

point 型，代表多边形各个顶点的集合。

多边形点 X

float 型，代表多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表多边形各个顶点的 Y 坐标。

屏蔽区

string 型，代表绘制的屏蔽区位置信息，主要用于后序模块继承该屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.2.9 Blob 标签分析

*Blob 标签分析*模块可在其输出图像中为检测出对的 Blob 标注类别标签。该模块的使用场景与 *Blob 分析*有所差异，典型场景为搭配基于深度学习的 *DL 图像分割*模块使用进行缺陷检测，输出缺陷类别标签。

模块原理

*Blob 标签分析*模块的算法工作原理基于 *Blob 分析*，两者在算法上主要区别如下：

区别点	Blob 分析	Blob 标签分析
处理逻辑	在转化的二值图上处理	直接在灰度图上处理
类别标签和标签值	无该两项数据。 所有 Blob 是由前景像素组成，在二值图上的灰度值为 255	有该两项数据。 如果若干个像素点区域的平均灰度值相同，且互不连通，那么算法将判定这些像素点区域为多个 Blob，但类别标签的标签值相同（即这些 Blob 为相同标签的 Blob） 如果若干像素点区域的平均灰度值不同，且互不连通，那么算法将判定这些像素点区域为多个 Blob，但类别标签的标签值不同（即这些 Blob 为不同标签的 Blob） • 此处提及的“灰度值”，可通过下文*参数配置*中的 *灰度值* 参数指定 • 此处提及的“类别标签”，可通过下文*参数配置*中的 *类别名称* 参数指定

使用方法

前后序模块

*Blob 标签分析*对其在流程中的前后序模块的要求如下。

表 13-8 前后续模块

前后序模块	描述
前序模块	前序模块需为 *Blob 标签分析*提供 string 数组和 int 数组这两类可订阅的数据，且这两类数据的数量需一致。 • string 数组，作为该模块的**类别名称**参数的取值。 • int 数组，作为该模块的**灰度值**参数的取值。 例如， *DL 快速图像分割* 和 *DL 实例分割* 模块可为 *Blob 标签分析*提供这两类数据。
后序模块	后序模块无特殊要求。

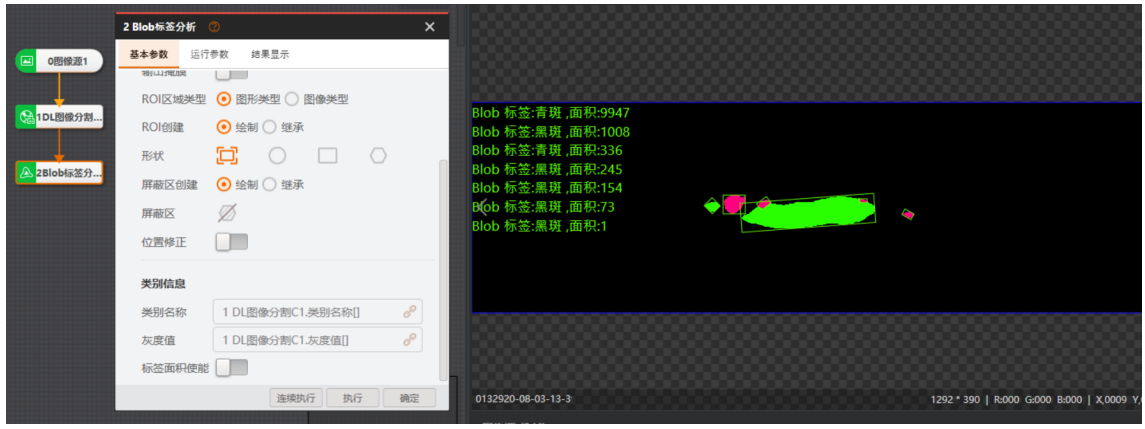


图 13-92 效果示例

参数配置

- 该模块基本参数中的类别信息参数详情如下。其他基本参数详情，请参见 *基本参数* 。

类别名称

订阅前序模块输出的 string 型数据。

例如，如果前序模块中包含 **DL 图像分割**，那么可将该参数取值订阅为 **DL 图像分割**输出的**类别名称**。

灰度值

订阅前序模块输出的 int 型数据。

例如，如果前序模块中包含 *DL 图像分割*，那么可将该参数取值订阅为 **DL 图像分割**输出的**灰度值**。

说明

由于**类别名称**和**灰度值**一对一映射，因此在为两者分别订阅数据时，需确保两者的数据数量保持一致。

标签面积使能

说明

该参数与该模块运行参数中的**全局面积使能**互斥。

开启后，该模块仅输出符合筛选规则的 Blob。具体筛选规则由下图所示列表中的**灰度值**、**面积范围**和**反选**三个元素定义。该列表各行分别对应一个 Blob 类别标签，各列的配置项定义如下：

灰度值

设置当前 Blob 类别标签所对应的灰度值。

面积范围

设置当前类别标签 Blob 的有效面积范围。

反选

如果不勾选（默认），该模块最终仅输出大小在指定**面积范围**以内的当前标签 Blob。

如果勾选，该模块最终仅输出大小在指定**面积范围**以外的当前标签 Blob。

类别名称使能

勾选该参数后，*灰度值*转换为*类别名称*，可设置当前 Blob 类别标签所对应的类别名称。

- 以下仅介绍该模块运行参数中的**全局面积使能**。该模块其他运行参数定义与 *Blob 分析*的同名参数相同，详情参见 *Blob 分析模块的运行参数* 。

  全局面积使能

  与 *Blob 分析*模块中的 *面积使能* 相同，即最终仅筛选出面积大小在指定范围内的 Blob。
- 该模块的结果显示参数详情，请参见 *结果显示* 。

模块结果

*Blob 标签分析*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

单体状态

int 型，代表各个 Blob 的状态。1 表示 Blob 识别成功，且出现 1 的次数表示检测出的 Blob 个数。

Blob 个数

int 型，代表图像中检测出的 Blob 个数。

面积

float 型，代表各个 Blob 的面积大小。

总面积

float 型，代表所有 Blob 加起来的总面积。

质心点

point 型，代表 Blob 质心。

质心 X

float 型，代表 Blob 质心的 X 坐标。

质心 Y

float 型，代表 Blob 质心的 Y 坐标。

最小外接矩形

box 型，表示能完整包含一个图形或目标区域的最小矩形。

中心点

point 型，表示最小外接矩形的中心点。

中心 X

float 型，代表最小外接矩形中心点的 X 坐标。

中心 Y

float 型，代表最小外接矩形中心点的 Y 坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

主轴角度

float 型，代表 Blob 矩形框基于水平线旋转的角度。顺时针旋转为正，逆时针旋转为负。

周长

float 型，代表各个 Blob 的轮廓周长。

长轴

float 型，代表各个 Blob 的最小外接矩形中较长边的长度。

最小外接矩形定义参见 *图 13-83* 。

短轴

float 型，代表各个 Blob 的最小外接矩形中较短边的长度。

圆形度

float 型，代表检测出的各个 Blob 的圆形度。

圆形度定义参见 *圆形度使能* 。

矩形度

float 型，代表各个 Blob 的面积与其相对应的最小外接矩形面积的比值。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

二值化图像数据

IMAGE 型，表示经过二值化的图像数据。

二值化图像数据

image 型，代表经过二值化的图像数据。该类型的图像数据为二进制形式，故模块结果处不显示任何数据。

二值化图像宽度

int 型，代表二值化图像的宽度。

二值化图像高度

int 型，代表二值化图像的高度。

二值化图像像素格式

int 型，代表二值化图像的像素格式。其中 17301505 对应 Mono8 格式。

Blob 图像

Blob 图像数据

image 型，代表通过 Blob 分析后输出的图像。该类型的图像数据为二进制形式，故模块结果处不显示任何数据，您可在输出图像中查看该图像。

Blob 图像宽度

int 型，代表输出 Blob 图像的宽度。

Blob 图像高度

int 型，代表输出 Blob 图像的高度。

Blob 图像像素格式

int 型，代表输出 Blob 图像的像素格式。其中，“17301505”对应 Mono 8 格式。

Blob 图像宽度

int 型，代表输出 Blob 图像的宽度。

Blob 图像高度

int 型，代表输出 Blob 图像的高度。

Blob 图像像素格式

int 型，代表输出 Blob 图像的像素格式。其中，“17301505”对应 Mono 8 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

轮廓点集

point 型，代表匹配对象的轮廓点的集合。

轮廓信息

pointset 型，代表轮廓的点集信息，该输出结果不在界面显示。您可以通过二次开发中的相关接口获取轮廓点集数据。

轮廓点个数

int 型，代表图像中匹配对象的轮廓点个数。

类别标签

string 型，代表 Blob 的类别标签。

灰度值

int 型，代表 Blob 的灰度值。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

13.2.10 圆查找

*圆查找*模块用于查找图像中指定区域内符合特定要求的圆，并输出圆相关数据，如圆是否存在、圆中心点坐标、圆半径、拟合误差等。

模块原理

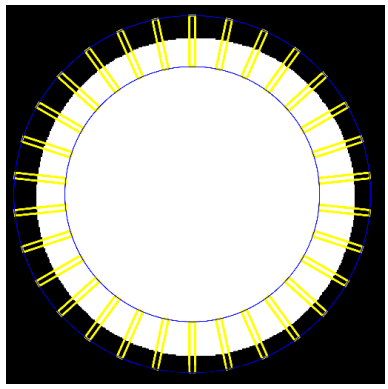
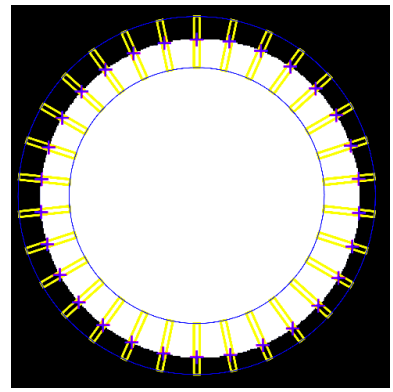
该模块的算法运行主要包含以下 3 个步骤：

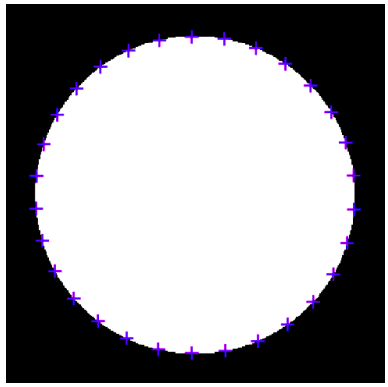
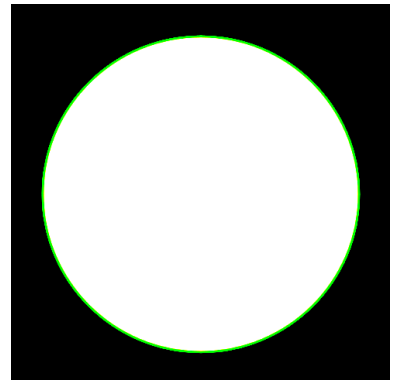
1. 在圆环形的 ROI 内自动设置卡尺区域。如下图 1 和图 2 中蓝色线条围合的环形区域即 ROI，黄色矩形即卡尺区域。

说明

卡尺的具体工作原理，请参见 *卡尺工具工作原理* 。

2. 在卡尺区域内提取目标的边缘点。如下图 2 和图 3 中紫色 X 形所示的点。
3. 将提取到的边缘点集拟合为圆，即下图 4 中所示的绿色圆。

ROI 内设置卡尺区域	在卡尺区域内提取边缘点
	

提取到的边缘点集	基于边缘点集进行圆拟合
	

使用限制

单个*圆查找*模块只能查找一个圆。如果要查找多个圆，建议配合 *Group* 模块的循环功能使用。如果需查找的多个圆有规律排列，可调用 *阵列圆查找* 模块。

使用方法

在流程中，*圆查找*模块的：

- 前序模块通常为 *图像源* 。*图像源*为*圆查找*提供图像输入。
- 后序模块可为逻辑工具模块（如 *脚本* 和 *数据集合* ）；也可为其他模块，如图形生成模块中的 *圆拟合* 和测量模块中的 *点点测量* 。*圆查找*为后序模块提供定位到的圆信息。

在流程中调用*圆查找*模块后，该模块的主要配置步骤如下：

1. 执行一次流程，使*图像源*将图像数据输出至*圆查找*。
2. 在*基本参数*页签选择 ROI 类型完成 ROI 绘制，如何绘制参见 *绘制 ROI 区域* 章节。
3. 根据业务需求设置*运行参数*页签中的**卡尺数量**、**扇环半径**、**滤波尺寸**等参数。

使用示例

以下视频展示了测量同心度的应用示例。具体为：在流程中组合调用*圆查找*和 *脚本* 、 *数据集合* 、 *圆拟合* 和 *点点测量* 等其他模块，测量汽车刹车盘上的 10 个小圆孔的同心度。

该应用示例中， *脚本* 模块的脚本代码如下。其中所调用的脚本接口详情，请参见 *脚本开放接口* 。

```
using System;
using System.Text;
using System.Windows.Forms;
using Script.Methods;
/**************************************
Shell Module default code: using .NET Framwwork 4.6.1
**************************************/
public partial class UserScript:ScriptMethods,IProcessMethods
{
    //the count of process
    //执行次数计数
    int processCount ;

    /// <summary>
    /// Initialize the field's value when compiling
    /// 预编译时变量初始化
    /// </summary>
    public void Init()
    {
        //You can add other global fields here
        //变量初始化，其余变量可在该函数中添加
        processCount = 0;

    }

    /// <summary>
    /// Enter the process function when running code once
    /// 流程执行一次进入Process函数
    /// </summary>
    /// <returns></returns>
    public bool Process()
    {
        //You can add your codes here, for realizing your desired function
        //每次执行将进入该函数，此处添加所需的逻辑流程处理
        float[] x = new float[9];
        float[] y = new float[9];
        for(int i=1;i<10;i++)
        {
            x[i-1] = (float)((x1-x0)*Math.Cos(36*i*Math.PI/180) - (y1-y0)*Math.Sin(36*i*Math.PI/
```

```
180) + x0);
            y[i-1] = (float)((x1-x0)*Math.Sin(36*i*Math.PI/180) + (y1-y0)*Math.Cos(36*i*Math.PI/
180) + y0);
        }
        SetFloatArrayValue("x2",x,0,9);
        SetFloatArrayValue("y2",y,0,9);
        return true;
    }
}
```

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 ***基本参数*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

扇环半径

ROI 扇环区域半径，建议内、外径与实际查找圆半径接近，可提高查找精度。

边缘类型

设置待查找的边缘类型。

最强

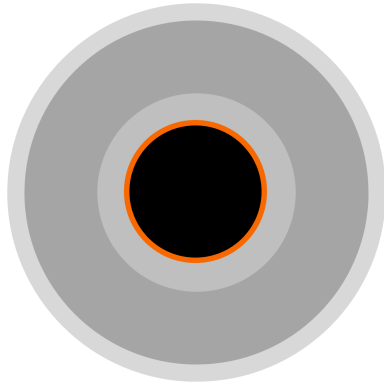
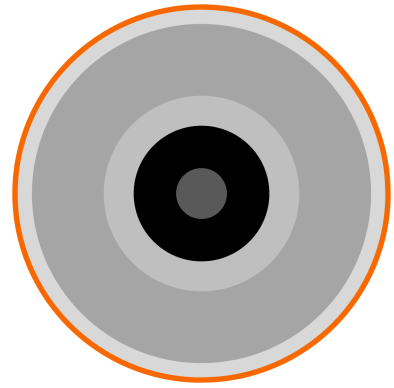
只提取扫描范围内梯度最大的边缘点集合并拟合成圆。

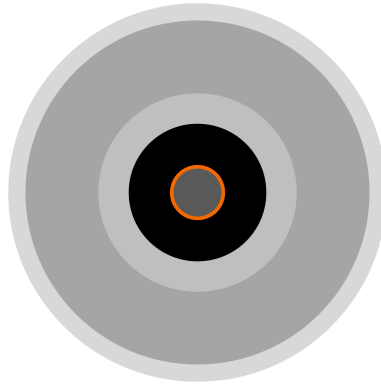
最后一条

只提取扫描范围内与圆心距离最大的边缘点集合并拟合成圆。

第一条

只提取扫描范围内与圆心距离最小的边缘点集合并拟合成圆

最强边缘	最后一条
	

第一条边缘


边缘极性

黑到白

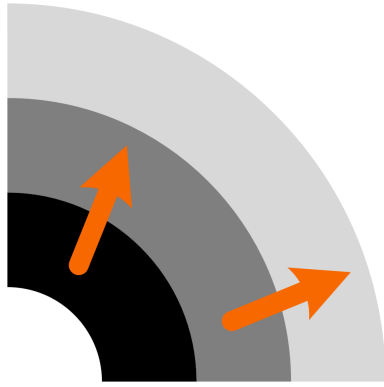
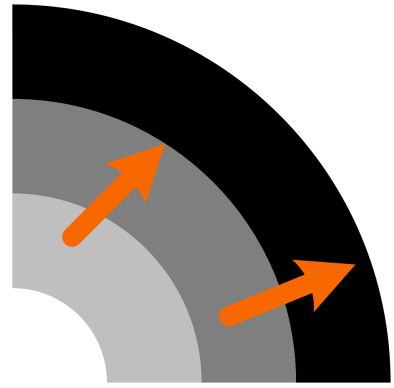
仅检测边缘极性为“黑到白”的边缘。从圆心向外看，灰度值低的区域过渡到灰度值高的区域的边缘，极性为黑到白。

白到黑

仅检测边缘极性为“白到黑”的边缘。从圆心向外看，灰度值高的区域过渡到灰度值低的区域的边缘，极性为白到黑。

任意

上述两种边缘均被检测。

黑到白	白到黑
	

边缘阈值

边缘阈值即梯度阈值，取值范围为 0~255。仅边缘梯度阈值大于该值的边缘点能被算法检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至可能导致目标边缘点被筛除

说明

边缘梯度是衡量卡尺区域内图像源的灰度变化率的指标。

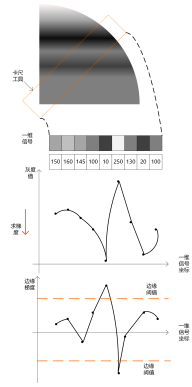


图 13-93 边缘阈值原理

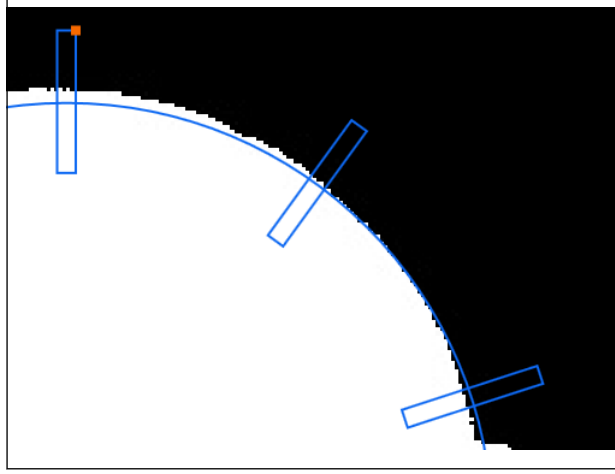
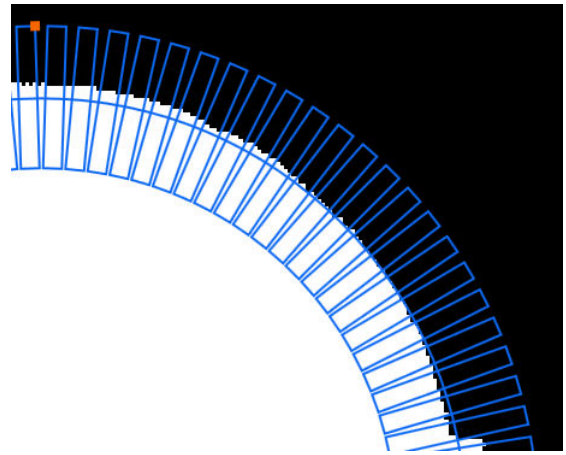
滤波尺寸

指滤波器大小。可用于增强边缘并抑制噪声，最小取值为 1。当边缘模糊或有噪声干扰时，增大该值可使检测结果更加稳定。但如果边缘与边缘之间距离小于滤波尺寸时，反而会影响边缘位置的精度甚至丢失边缘

边缘过渡带较小时建议使用较小滤波尺寸，边缘过渡带较大时使用较大滤波尺寸

卡尺数量

用于扫描边缘点的 ROI（即上文 *模块原理* 中提及的卡尺）的数量。当该数量设置为 10 和 100 时，相同弧度范围内的卡尺分布见下图

卡尺数量 10	卡尺数量 100
	

卡尺宽度
卡尺高度

分别用于设置扫描边缘点 ROI（卡尺）的宽度和高度。ROI 创建方式差异会导致参数是否可配置和生效：

- *ROI 创建*选择**绘制**：绘制 ROI 形状选择**全屏**或**矩形**时，这两个参数不生效；绘制 ROI 形状选择**圆卡尺**或**圆弧卡尺**时，此参数支持调整并与*基本参数*页签 ROI 参数中的**卡尺宽度**和**卡尺高度**保持同步。
- *ROI 创建*选择**继承**：当**继承方式**选择**按圆形区域**或**按圆形参数**时，因卡尺高度固定，只支持配置**卡尺宽度**。

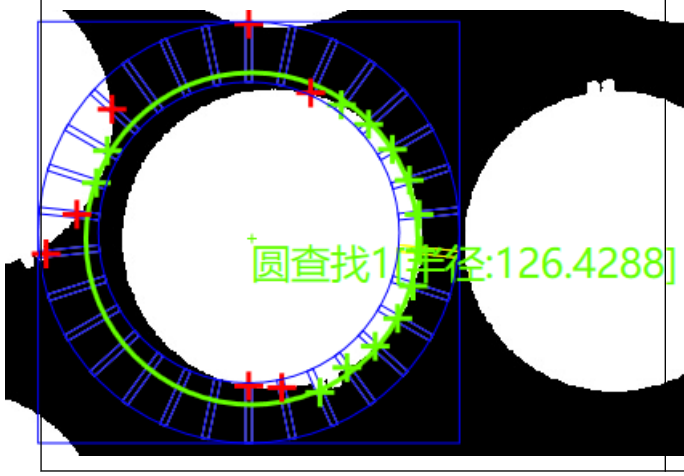
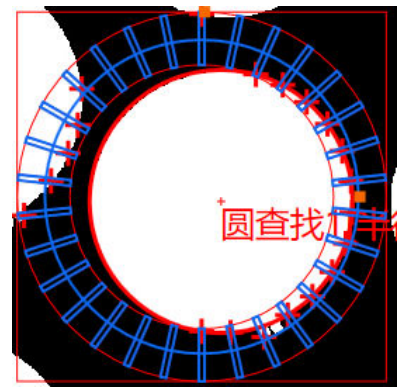
> **说明**
> 其他**继承方式**，不支持配置这两个参数。

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大。为获取更佳查找效果，建议与下文的**剔除距离**结合使用。

如下左图中红色点为剔除的边缘点。

当剔除点数设置过大时，可能会导致拟合失败，无法查找到圆（见如下右图）。

剔除点示意	剔除点数过大
	

初定位

开启**初定位**后，需配置下文提及的**圆定位敏感度**和**下采样系数**。**初定位**可大致判定 ROI 区域内更接近圆的区域中心作为初始圆中心，便于后续精细圆查找。

若关闭**初定位**，则默认 ROI 中心为初始圆中心。如果*圆查找*模块在流程中的前一模块为*位置修正*，建议关闭**初定位**。

下采样参数

下采样也叫降采样，即是采样点数减少。如果其降采样系数为 k，则表示在原图中每行每列每隔 k 个点取一个点组成一幅图像。因此下采样系数越大，轮廓点越稀疏，轮廓越不精细，该值不宜设置过大。

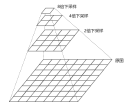


图 13-94 下采样

圆定位敏感度

排除干扰点。该参数取值越大，排除噪声干扰的能力越强，但也越容易导致圆的初定位失败。

剔除距离

允许离群点到拟合圆的最大像素距离，值越小，排除点越多。

投影宽度

在环形 ROI 中分布的若干个卡尺的宽度。在一定范围内增大该值可以获取更加稳定的边缘点，具体可参见 *直线查找* 。

初始拟合

设置圆的初始拟合模式。

局部

按照局部的特征点拟合圆。穷举遍历边缘点集中的部分点来圆拟合，并将最优圆作为初始拟合圆，如果局部特征更加准确反映圆所在位置，则采用局部最优，否则采用全局最优。

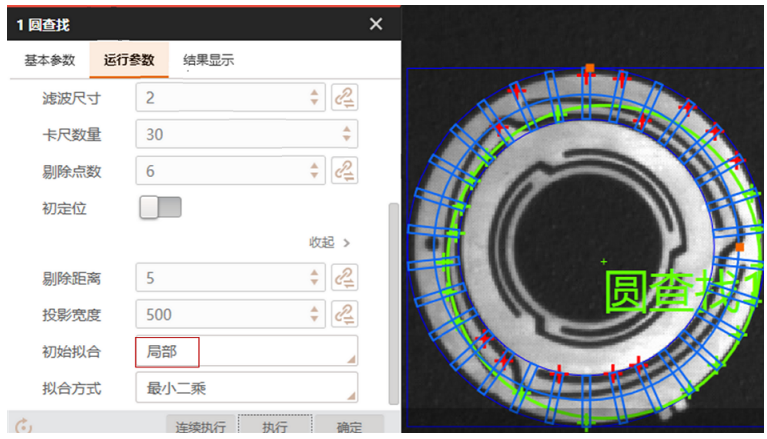


图 13-95 局部拟合

全局

按照查找到的全局特征点进行圆拟合。全局最优特征更加准确反映圆所在位置时采用该模式。

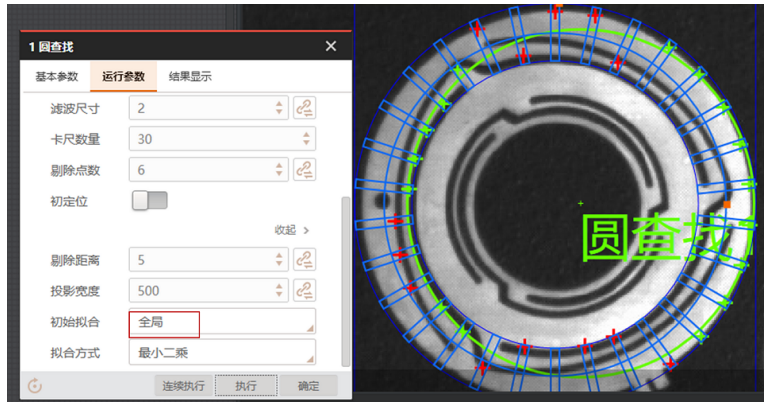


图 13-96 全局拟合

拟合方式

设置圆的拟合方式，包括**最小二乘**、**huber** 和 **tukey** 三种。三种拟合方式仅在权重的计算方式上有所差异。随着离群点数量增多以及离群距离增大，可逐次使用**最小二乘**、**huber** 和 **tukey**。

模块结果

*圆查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出圆环

ANNULUS 型，代表模块查找到的圆，记录了该圆的圆心、内径和角度等信息。

圆心

point 型，表示查找到的圆的圆心坐标。

圆心 X

float 型，代表圆心的 X 坐标。

圆心 Y

float 型，代表圆心的 Y 坐标。

内径

float 型，代表圆的内径。

圆半径

float 型，代表圆的半径。

圆起始角度

float 型，代表圆起始边与水平方向的夹角。

圆角度范围

float 型，代表圆起始边与终止边跨过的角度范围。

拟合误差

float 型，代表实际图形与拟合图形之间的误差。

圆轮廓点

point 型，记录了圆上各个轮廓点的坐标信息。

轮廓点 X

float 型，代表各个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表各个轮廓点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

卡尺框

box 型，表示用于定位目标对象所使用的卡尺框，记录了该卡尺框的中心点、宽度、高度和角度信息。

卡尺中心点

point 型，表示卡尺框的中心点，记录了卡尺框中心点的坐标信息。

卡尺中心 X

float 型，代表卡尺框中心点的 X 坐标。

卡尺中心 Y

float 型，代表卡尺框中心点的 Y 坐标。

卡尺宽度

float 型，代表每个卡尺框的宽度，一般相等。

卡尺高度

float 型，代表每个卡尺框的高度，一般相等。

卡尺角度

float 型, 代表每个卡尺框与水平方向的夹角。一般对应位置角度相差的大小为检测圆弧的角度范围。

卡尺框检测区

box 型，表示可完整包含所有卡尺框的检测区域，记录了该检测区域的中心点、宽度、高度和角度信息。

卡尺框检测中心点

point 型，表示卡尺框检测区域的中心点，记录了该检测区域的中心点坐标。

卡尺框检测中心 X

float 型，代表卡尺框检测区中心点的 X 坐标。

卡尺框检测中心 Y

float 型，代表卡尺框检测区中心点的 Y 坐标。

卡尺框检测区宽度

float 型，代表每个卡尺框检测区的宽度，一般相等。

卡尺框检测区高度

float 型，代表每个卡尺框检测区的高度，一般相等。

卡尺框检测区角度

float 型，代表每个卡尺框检测区与水平方向的夹角。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

13.2.11 椭圆查找

*椭圆查找*模块用于在图像指定区域中查找符合特定要求的椭圆形轮廓，并输出椭圆信息，包括椭圆是否存在、椭圆中心点、椭圆长半轴和短半轴、拟合误差等。

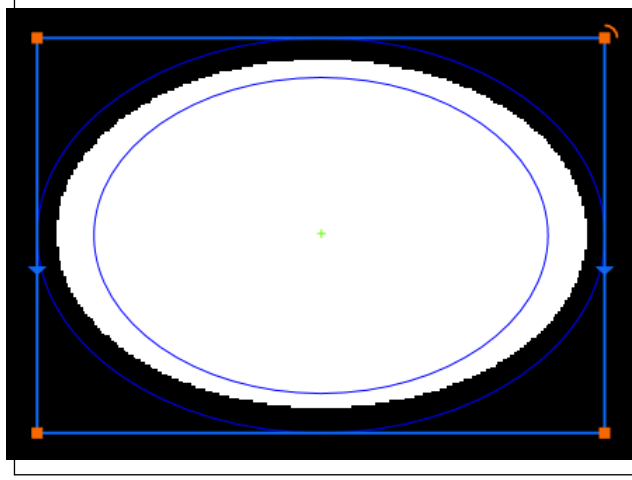
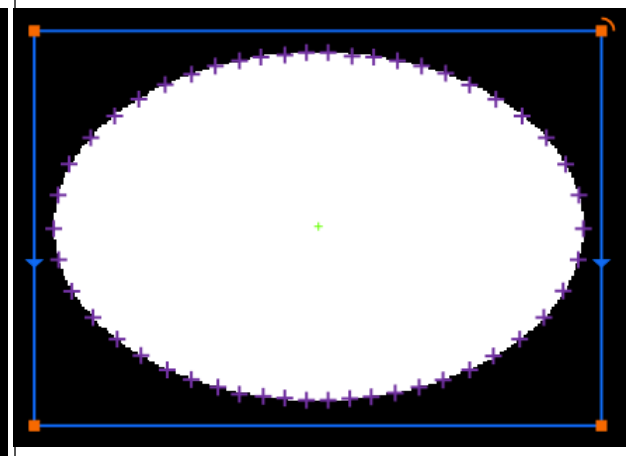
模块原理

该模块的算法工作流程可分为如下两个主要步骤：

1. 算法在椭圆环形状的 ROI 内，基于该 ROI 内置的卡尺提取边缘点。

说明

- 如下左图中两个深蓝色椭圆构成的椭圆环由该矩形 ROI（蓝色矩形）生成
- 卡尺的具体工作原理，请参见 *卡尺工具工作原理* 。

椭圆环 ROI	提取到的边缘点
	

2. 算法将提取到的边缘点拟合为椭圆。

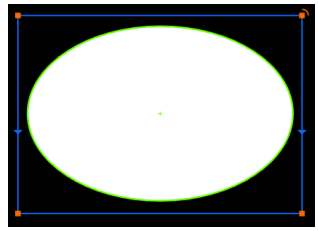


图 13-97 椭圆拟合结果

使用限制

*椭圆查找*模块一次仅能查找单个椭圆。如需查找多个椭圆，建议配合 *Group* 模块的循环功能使用。

使用方法

在流程中，*椭圆查找*的：

- 前序模块通常为 *图像源* 。*图像源*为*椭圆查找*提供图像输入。
- 后序模块可为逻辑工具模块（如 *脚本* 和 *数据集合* ）；也可为其他模块，如测量模块中的 *点点测量* 、 *点圆测量* 、 *点线测量* 等。*椭圆查找*为后序模块提供定位到的圆信息。

在流程中调用*椭圆查找*模块后，该模块的主要配置步骤如下：

1. 执行一次流程，使*图像源*将图像数据输出至*椭圆查找*。
2. 在*基本参数*页签单击 □ ，并将光标移动至图像区域绘制 ROI。

   说明

   绘制的矩形 ROI 的四边需接近目标椭圆的四个顶点（但需避免与椭圆相切），以确保定位精度。

3. 根据实际情况调整 ROI 参数。
4. 根据业务需求配置下文*参数配置*中提及的**边缘极性**、**卡尺数量**、**滤波尺寸**等运行参数。

使用示例

如下示意图展示了检测椭圆环部件宽度的应用示例。该应用示例中组合调用了*椭圆查找*、*圆环展开*和*间距检测*。

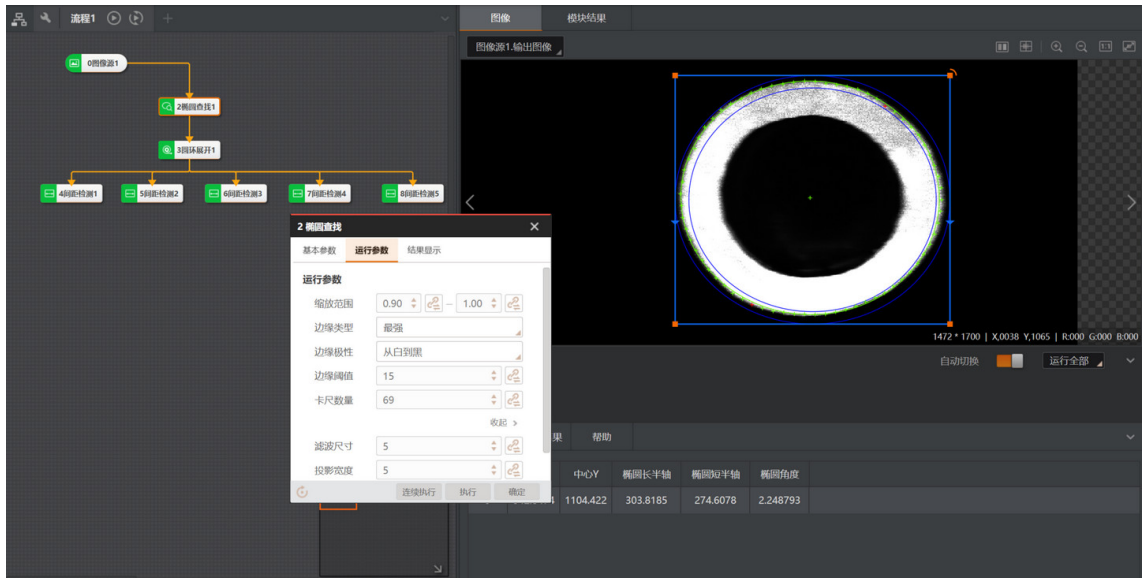


图 13-98 椭圆查找输出椭圆信息

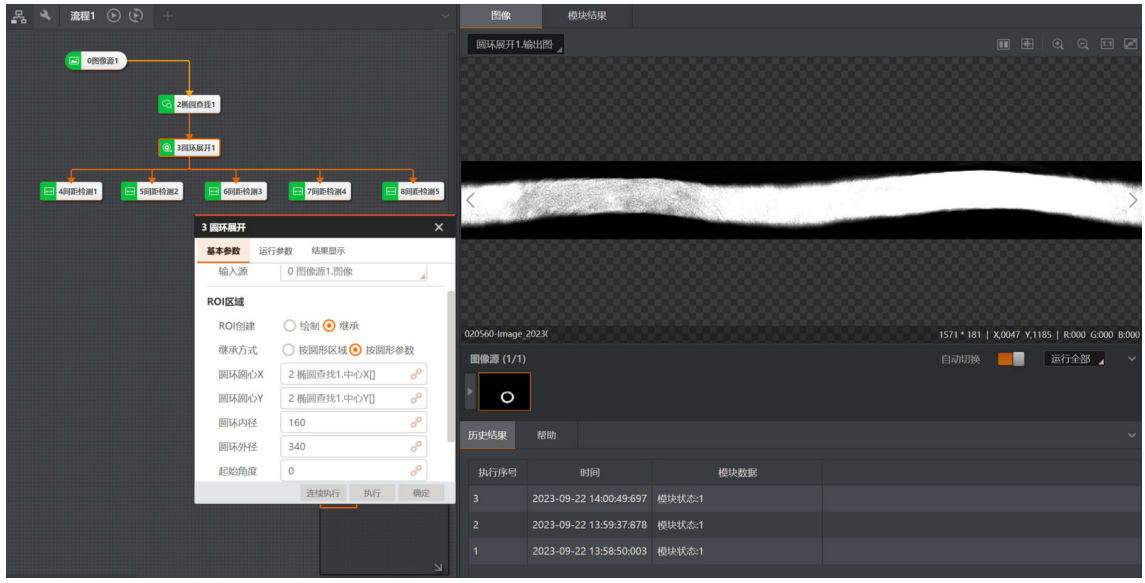


图 13-99 圆环展开继承椭圆查找的 ROI

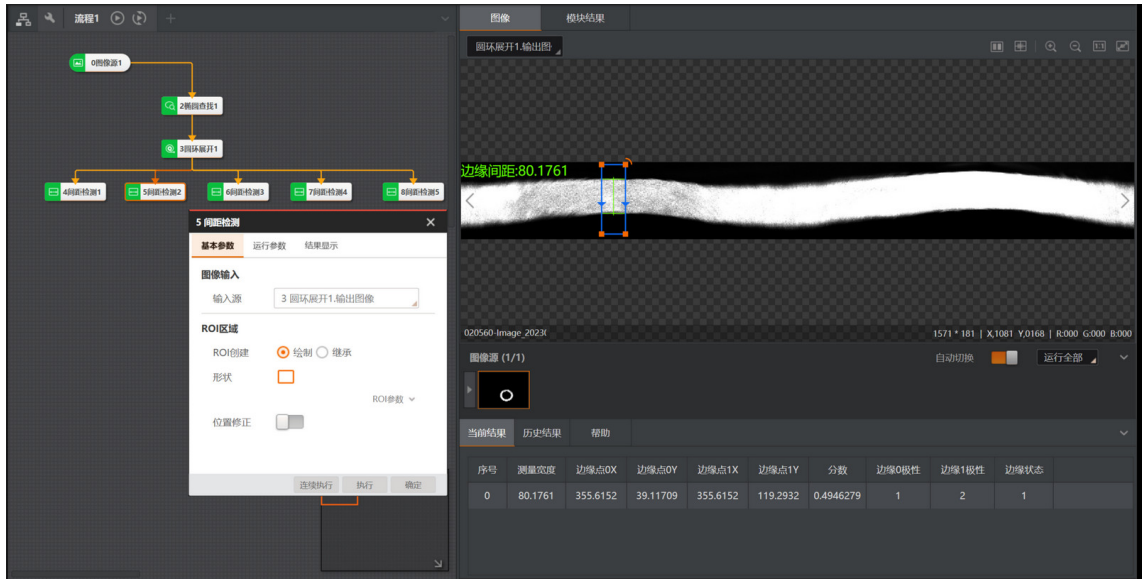


图 13-100 测量边缘间距

参数配置

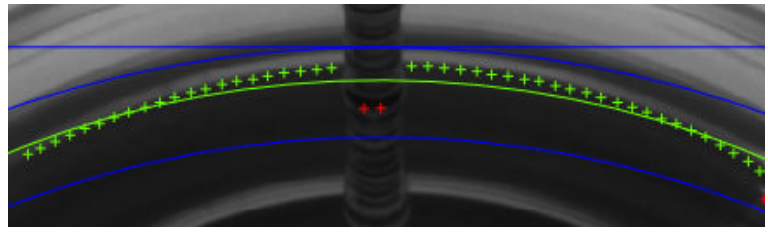
以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

表 13-9 运行参数

参数		描述
缩放范围	内缩放系数	根据用户设置的 ROI 区域生成椭圆环，椭圆环内椭圆的长短轴与 ROI 宽高之间的比例关系。
	外缩放系数	根据用户设置的 ROI 区域生成椭圆环，椭圆环外椭圆的长短轴与 ROI 宽高之间的比例关系。
边缘类型	最强	只提取扫描范围内梯度最大的边缘点集合并拟合成椭圆。
	最后一条	只提取扫描范围内与椭圆圆心距离最大的边缘点集合并拟合成椭圆。
	第一条	只提取扫描范围内与椭圆圆心距离最小的边缘点集合并拟合成椭圆。

参数		描述
边缘极性	黑到白	从椭圆圆心向外看，灰度值低的区域过渡到灰度值高的区域的边缘，极性为黑到白。
	白到黑	从椭圆圆心向外看，灰度值高的区域过渡到灰度值低的区域的边缘，极性为白到黑。
	任意	上述两种边缘均被检测。
卡尺数量		用于查找边缘点的 ROI 的数量，下图中的绿色“+”字形即卡尺。 
滤波尺寸		用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间距离小于滤波尺寸时反而会影响边缘位置的精度甚至丢失边缘。 更多详情可参见 *圆查找* 中的**滤波尺寸**。
投影宽度		在 ROI 中环形分布若干个边缘点查找 ROI，该值描述扫描边缘点查找 ROI 的区域宽度。在一定范围内增大该值可以获取更加稳定的边缘点。
误差容忍度		在进行椭圆拟合时会计算每个参与拟合的边缘点的拟合误差，该参数值越小时，拟合时删除掉的边缘点越多，拟合结果更加准确，但该值设置的过于小时，有可能造成过拟合。

模块结果

*椭圆查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

椭圆轮廓点

记录了检出椭圆上各个轮廓点的坐标信息。

轮廓点 X

float 型，代表各个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表各个轮廓点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

输出椭圆

记录了有关输出椭圆的圆心点、长半轴、短半轴和角度信息。

圆心点

point 型，表示查找到的椭圆的圆心点。

中心 X

float 型，代表圆心点的 X 坐标。

中心 Y

float 型，代表圆心点的 Y 坐标。

椭圆长半轴

float 型，代表椭圆的长半轴长度。

椭圆短半轴

float 型，代表椭圆的短半轴长度。

椭圆角度

float 型，代表椭圆的角度。

椭圆环

记录了有关检出椭圆环的圆心点、长半轴、短半轴和角度信息。

圆心点

point 型，表示查找到的椭圆的圆心点。

中心 X

float 型，代表圆心点的 X 坐标。

中心 Y

float 型，代表圆心点的 Y 坐标。

椭圆长半轴

float 型，代表椭圆的长半轴长度。

椭圆短半轴

float 型，代表椭圆的短半轴长度。

椭圆角度

float 型，代表椭圆的角度。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

13.2.12 阵列圆查找

*阵列圆查找*模块用于查找指定图像区域内符合特征要求且呈阵列排布（例如等距排列为 4 行 3 列）的多个圆，并以数组形式输出圆信息，包括圆是否存在、圆中心点坐标、圆半径、拟合误差等。

本节内容包含：

- *模块原理*
- *使用方法*
- *使用示例*
- *参数配置*
- *模块结果*

模块原理

*阵列圆查找*的算法原理基于*圆查找*，但可查找阵列排布的多个圆。算法原理详情参见 *圆查找的算法原理* 。

使用方法

在流程中，*阵列圆查找*的：

- 前序模块通常为 *图像源* 。*图像源*为*阵列圆查找*提供图像输入。
- 后序模块可为逻辑工具模块（如 *脚本* 和 *数据集合*）；也可为其他模块，如标定模块中的 *N 点标定*，图形生成模块中的 *圆拟合*，测量模块中的 *圆圆测量* 等。*阵列圆查找*为后序模块提供定位的圆信息。

在流程中调用*圆查找*模块后，该模块的主要配置步骤如下：

1. 执行一次流程，使*图像源*将图像数据输出至*阵列圆查找*。
2. 在*基本参数*页签通过 ⌑ 绘制 ROI 区域，如何绘制参见 *绘制 ROI 区域* 章节。
3. 根据业务需求配置*运行参数*页签中的**卡尺数量**、**扇环半径**、**滤波尺寸**等运行参数。运行参数详情参见下文*参数配置*。

使用示例

如下示意图展示圆心距离测量的应用示例。该应用示例中组合调用了*阵列圆查找*和 *圆圆测量*，前者为后者提供圆信息输入。

图 13-101 示意图中的检测出的圆标注了序号，其中标注为“1”和“3”的圆分别对应 *图 13-102* 示意图中*基本参数*页签中的**圆输入 1** 和**圆输入 2**。

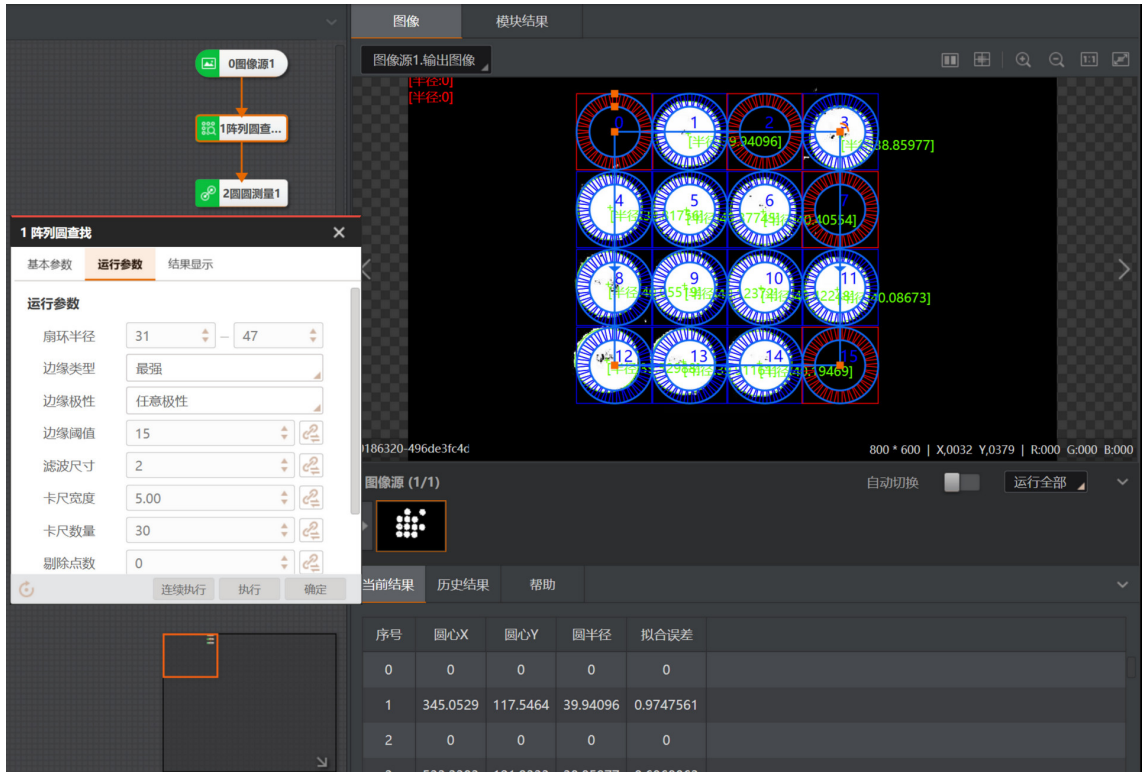


图 13-101 阵列圆序号

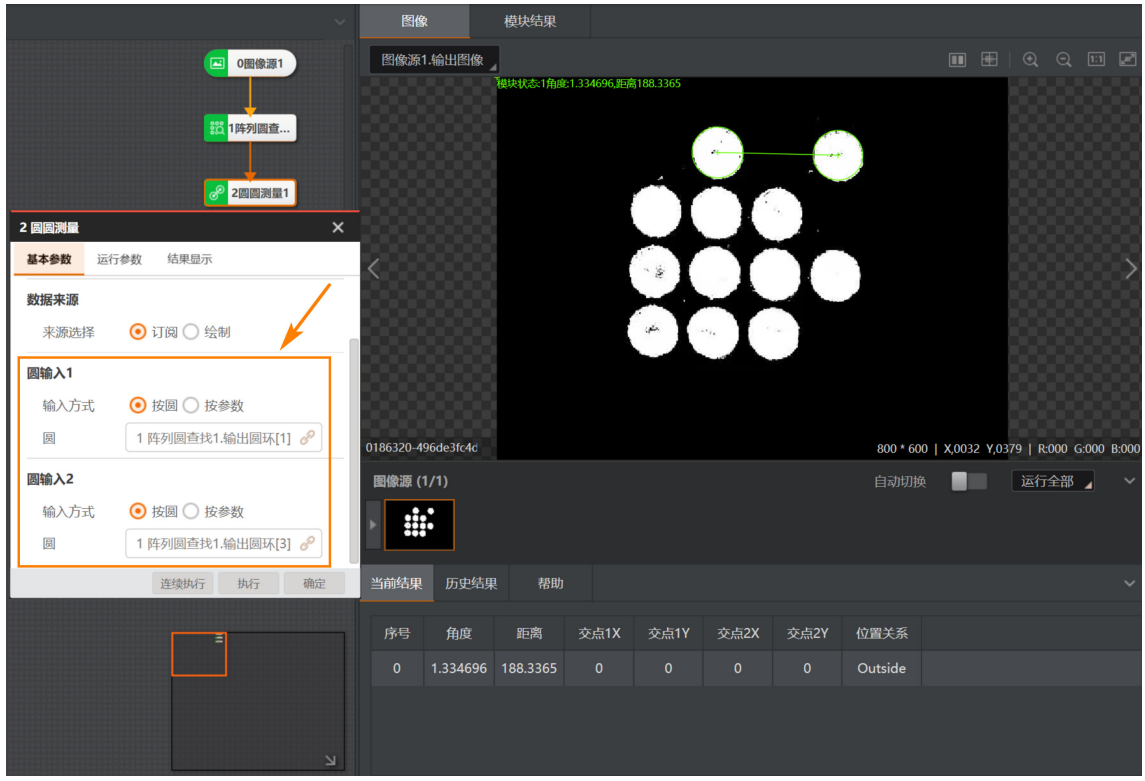


图 13-102 圆圆测量的圆输入配置

参数配置

- 基本参数详情，请参见 *基本参数* 。
- 该模块的运行参数和*当前结果*参数与*圆查找*的相同，详情参见 *圆查找模块的参数配置* 。
- 结果显示参数详情，请参见 *结果显示* 。

模块结果

*阵列圆查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

单个圆查找状态

int 型，代表阵列中每个圆的查找状态，1 代表查找到圆，0 代表未查找到圆。

输出圆环个数

int 型，代表阵列中找到的圆的个数。

输出圆环

ANNULUS 型，代表模块查找到的圆，记录了该圆的圆心、内径和角度等信息。

圆心

point 型，表示查找到的圆的圆心坐标。

圆心 X

float 型，代表圆心的 X 坐标。

圆心 Y

float 型，代表圆心的 Y 坐标。

内径

float 型，代表圆的内径。

圆半径

float 型，代表圆的半径。

圆起始角度

float 型，代表圆起始边与水平方向的夹角。

圆角度范围

float 型，代表圆起始边与终止边跨过的角度范围。

拟合误差

float 型，代表实际图形与拟合图形之间的误差。

圆轮廓点

point 型，记录了圆上各个轮廓点的坐标信息。

轮廓点 X

float 型，代表各个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表各个轮廓点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

卡尺框个数

int 型，代表用于查找阵列圆中各个圆所使用的卡尺数量。

卡尺框

box 型，表示用于定位目标对象所使用的卡尺框，记录了该卡尺框的中心点、宽度、高度和角度信息。

卡尺中心点

point 型，表示卡尺框的中心点，记录了卡尺框中心点的坐标信息。

卡尺中心 X

float 型，代表卡尺框中心点的 X 坐标。

卡尺中心 Y

float 型，代表卡尺框中心点的 Y 坐标。

卡尺宽度

float 型，代表每个卡尺框的宽度，一般相等。

卡尺高度

float 型，代表每个卡尺框的高度，一般相等。

卡尺角度

float 型, 代表每个卡尺框与水平方向的夹角。一般对应位置角度相差的大小为检测圆弧的角度范围。

卡尺框检测区

box 型，表示可完整包含所有卡尺框的检测区域，记录了该检测区域的中心点、宽度、高度和角度信息。

卡尺框检测中心点

point 型，表示卡尺框检测区域的中心点，记录了该检测区域的中心点坐标。

卡尺框检测中心 X

float 型，代表卡尺框检测区中心点的 X 坐标。

卡尺框检测中心 Y

float 型，代表卡尺框检测区中心点的 Y 坐标。

卡尺框检测区宽度

float 型，代表每个卡尺框检测区的宽度，一般相等。

卡尺框检测区高度

float 型，代表每个卡尺框检测区的高度，一般相等。

卡尺框检测区角度

float 型，代表每个卡尺框检测区与水平方向的夹角。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.13 直线查找

*直线查找*模块可查找目标的直线边，常用于测量、定位、缺陷检测场景的目标角度获取。

模块原理

该模块的算法工作流程可分为如下两个主要步骤：

1. 算法在预设的卡尺区域（下图所示黄框）内提取边缘特征点。

   说明

   卡尺的具体工作原理，请参见 *卡尺工具工作原理* 。

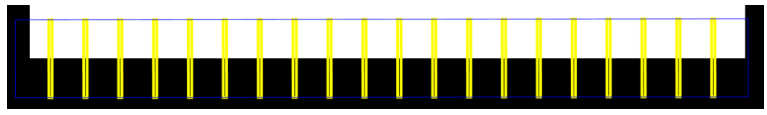


图 13-103 卡尺区域

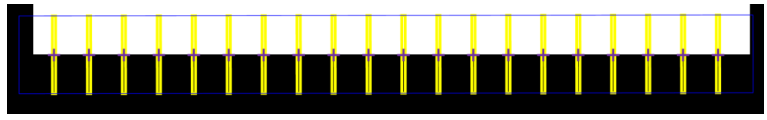


图 13-104 卡尺区域内提取到的边缘点

2. 算法将提取到的边缘特征点（下图中的紫色点）拟合为直线，最终输出的直线被界定在 ROI 内。



图 13-105 边缘点示意



图 13-106 拟合结果示意

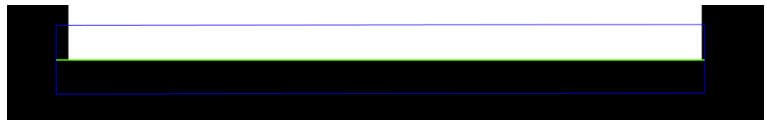


图 13-107 最终输出直线

使用方法

在流程中，***直线查找***的前序模块一般为 ***图像源***，对后序模块无特殊要求。
在流程中调用***直线查找***模块后，该模块的主要配置步骤如下：

1. 执行一次流程，使***图像源***将图像数据输出至***直线查找***。
2. 根据业务需求指定下文*参数配置*中提及的**卡尺数量**、**投影宽度**、**滤波尺寸**等运行参数。
3. 在***基本参数***页签，选择 ROI 形状，并在图像上圆所在区域绘制 ROI。

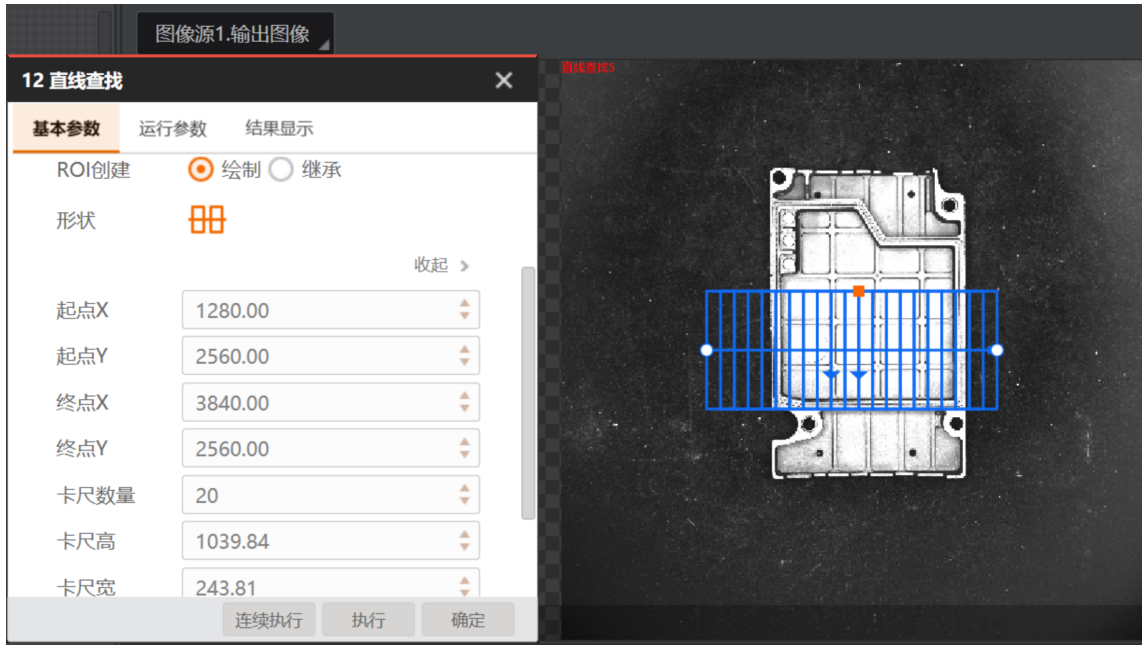


图 13-108 绘制 ROI

使用示例

下图展示了在流程中组合调用*直线查找*、*线线测量*、*点点测量*，对部件尺寸规格进行检测的应用示例。

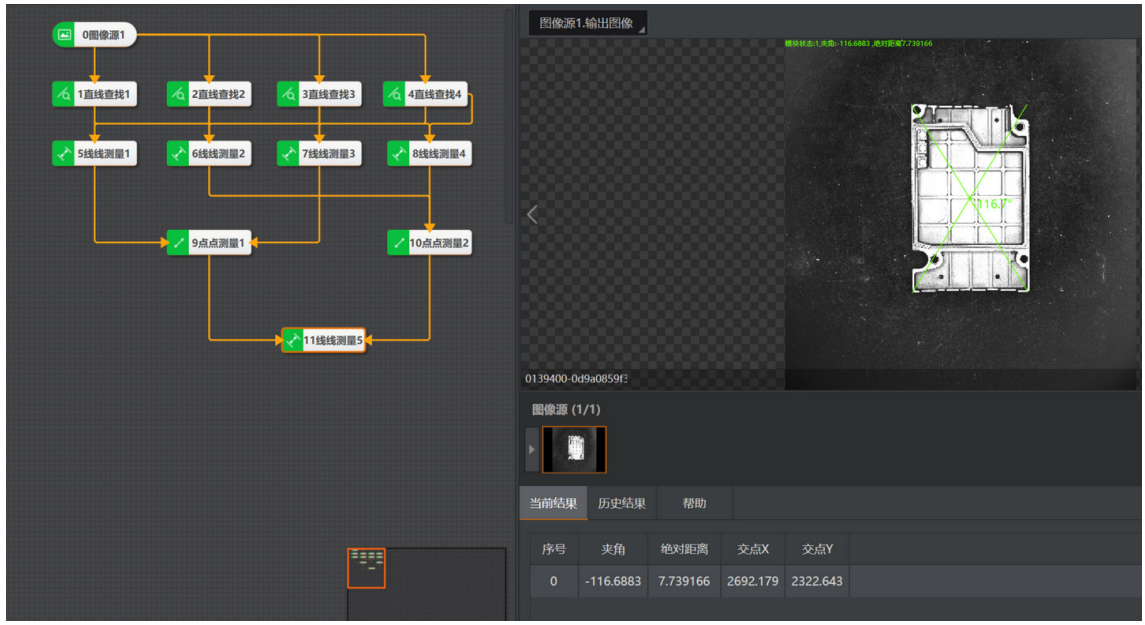


图 13-109 与测量模块配置使用的示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

边缘类型

可选最强、第一条、最后一条或接近中线。

最强

查找梯度阈值最大的边缘点集合，然后拟合成直线。

第一条
最后一条

查找满足条件的第一条/最后一条直线。

接近中线

查找最接近区域中线且满足条件的直线。

边缘极性

可选从白到黑、从黑到白、任意极性。

边缘极性分为两种，即白到黑和黑到白，如下图所示。绿色箭头所示为白到黑，红色箭头所示为黑到白，这里说的“黑”和“白”是相对的概念。



图 13-110 边缘极性

边缘阈值

边缘阈值即梯度阈值，范围 0~255，只有边缘梯度阈值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

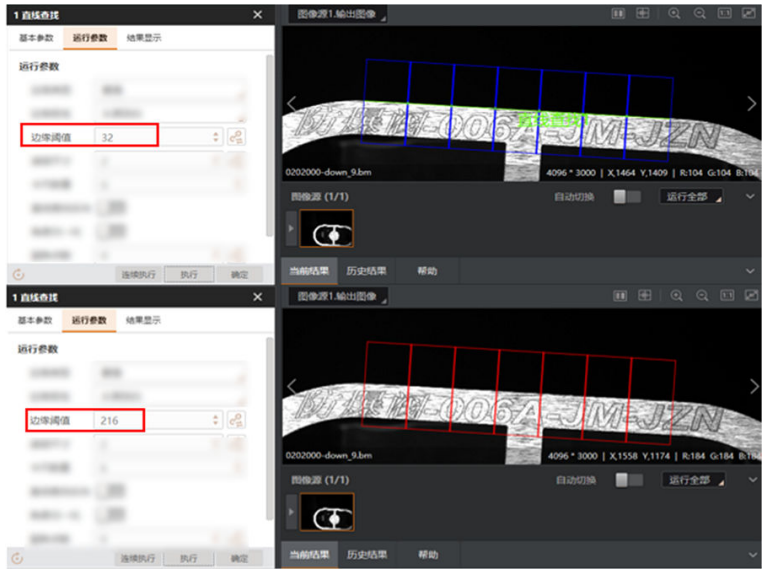


图 13-111 边缘阈值

滤波尺寸

定义目标边缘的清晰程度，最小值为 1。值越小，表示边缘越清晰且过滤带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间挨得太近（距离小于滤波尺寸）时反而会影响边缘位置的精度甚至丢失边缘。具体原理参考 *圆查找*。

卡尺数量

定义卡尺的数量，ROI 越大，建议设置越多的卡尺数量。

直线查找反向

开启后可将直线起点和终点的位置信息互换。

角度归一化

开启后，输出的直线角度在-90° ~ 90°；未开启时，输出的直线角度为-180° ~ 180°。

角度使能

开启后，根据**期望直线角度**和**角度旋转容忍**设置的数值判断直线的角度是否符合要求。

假设**期望直线角度**设为 10，**角度旋转容忍**设为 20，则直线角度在-10° ~ 30°之间为符合要求。

剔除点数

差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，为获取更佳查找效果，建议与**剔除距离**结合使用。

剔除距离

允许离群点到拟合直线的最大像素距离，值越小，排除点越多。

投影宽度

即卡尺宽度（*图 13-103* 中黄框的宽度），在 ROI 中顺序排列若干个查找边缘点 ROI，该值描述查找边缘点 ROI 的区域宽度，在一定范围内增大该值可以获取更加稳定的边缘点。

初始拟合

可选局部或全局。

局部

局部最优也就是按照局部的特征点拟合直线，如果局部特征更加准确反映直线所在位置，则采用局部最优，否则采用全局最优。

全局

以查找到的全局特征点进行直线拟合。

拟合方式

拟合方式有最小二乘、huber 和 tukey 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用最小二乘、huber、tukey。

模块结果

*直线查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出直线

记录了查找到的直线起点和终点的坐标信息。

直线起点

point 型，代表输出直线的起点坐标。

直线起点 X

float 型，代表输出直线起点的 X 坐标。

直线起点 Y

float 型，代表输出直线起点的 Y 坐标。

直线终点

point 型，代表输出直线的终点坐标。

直线终点 X

float 型，代表输出直线终点的 X 坐标。

直线终点 Y

float 型，代表输出直线终点的 Y 坐标。

中点

point 型，表示检测到的直线中点。

中点 X

float 型，代表直线中点的 X 坐标。

中点 Y

float 型，代表直线中点的 Y 坐标。

直线角度

float 型，代表输出直线的角度。

拟合误差

float 型，代表实际图形与拟合图形之间的误差。

直线轮廓点

记录了查找到的直线上各个轮廓点的坐标信息。

轮廓点 X

float 型，代表各个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表各个轮廓点的 Y 坐标。

边缘点个数

int 型，代表检出直线上的边缘点个数。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

卡尺框

表示用于定位目标直线所使用的卡尺框，记录了该卡尺框的中心点、宽度、高度和角度信息。

卡尺中心点

point 型，表示卡尺框的中心点，记录了卡尺框中心点的坐标信息。

卡尺中心 X

float 型，代表卡尺框中心点的 X 坐标。

卡尺中心 Y

float 型，代表卡尺框中心点的 Y 坐标。

卡尺宽度

float 型，代表每个卡尺框的宽度，一般相等。

卡尺高度

float 型，代表每个卡尺框的高度，一般相等。

卡尺角度

float 型，代表每个卡尺框与水平方向的夹角，一般相等。

卡尺框检测区

表示可完整包含卡尺框的检测区域，记录了该检测区域的中心点、宽度、高度和角度信息。

卡尺框检测中心点

point 型，表示卡尺框检测区域的中心点，记录了该检测区域的中心点坐标。

卡尺框检测中心 X

float 型，代表卡尺框检测区中心点的 X 坐标。

卡尺框检测中心 Y

float 型，代表卡尺框检测区中心点的 Y 坐标。

卡尺框检测区宽度

float 型，代表每个卡尺框检测区的宽度，一般相等。

卡尺框检测区高度

float 型，代表每个卡尺框检测区的高度，一般相等。

卡尺框检测区角度

float 型，代表每个卡尺框检测区与水平方向的夹角，一般相等。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

13.2.14 直线查找组合

*直线查找组合*模块可检测图像指定区域内的不连续边缘，输出边缘直线起点、终点及角度信息，常用于不连续边缘的检测，也可用于*直线查找*屏蔽区占比过高时的替代方案。所谓不连续边缘，指边缘中间部分存在缺失或是断开。

本节内容包含：

- *使用方法*
  参数配置
  模块结果

使用方法

在流程中：

- *图像源*、*模板匹配*、*位置修正*等模块可作为*直线查找组合*的前序模块，为*直线查找组合*在图像指定区域精确定位矩形。*模板匹配*和*位置修正*做粗定位，*直线查找组合*做精定位。
- *直线查找组合*对后序模块无特殊要求。可接收并处理直线信息的模块均可作为该模块的后续模块。

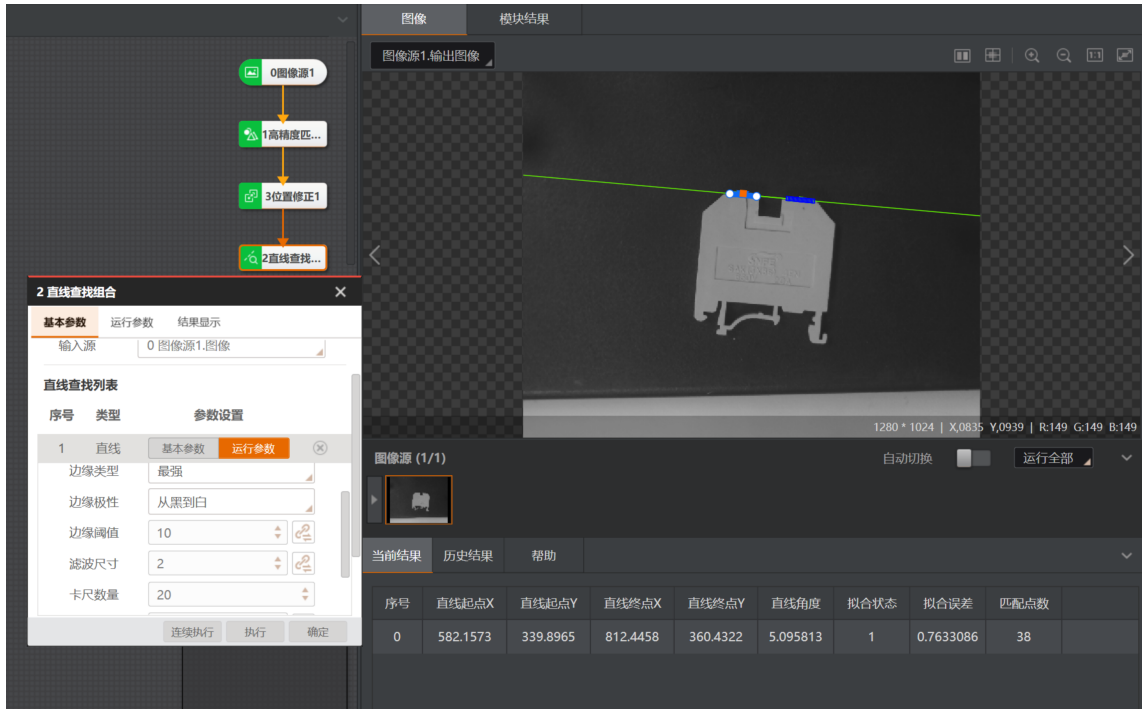


图 13-112 直线查找组合应用示例

参数配置

- 基本参数详情，请参见 *基本参数* 。
- 该模块运行参数定义与*直线查找*模块的同名运行参数相同，详情参见 *直线查找* 。通过配置运行参数，可定义当前模块如何处理输入的数据。
- 结果显示参数详情，请参见 *结果显示* 。

模块结果

*直线查找组合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出直线

记录了查找到的各个直线的起点和终点坐标。

直线起点

point 型，代表输出直线的起点坐标。

直线起点 X

float 型，代表输出直线起点的 X 坐标。

直线起点 Y

float 型，代表输出直线起点的 Y 坐标。

直线终点

point 型，代表输出直线的终点坐标。

直线终点 X

float 型，代表输出直线终点的 X 坐标。

直线终点 Y

float 型，代表输出直线终点的 Y 坐标。

直线角度

float 型，代表输出直线的角度。

输出拟合点

记录了输出直线上拟合点的坐标。

输出拟合点 X

float 型，代表输出直线上各个拟合点的 X 坐标。

输出拟合点 Y

float 型，代表输出直线上各个拟合点的 Y 坐标。

拟合误差

float 型，代表实际图形与拟合图形之间的误差。

拟合状态

int 型，1 代表拟合成功，0 代表拟合失败。

拟合点数

int 型，代表实际拟合点个数。

拟合距离

float 型，代表各个相邻拟合点之间的距离。

匹配点

int 型，代表各个拟合点是否匹配成功，1 代表匹配成功，0 代表匹配失败。

匹配点数

int 型，代表匹配成功的拟合点个数。

最小外接矩形

表示能完整包含目标直线组合的最小矩形，记录了有关该矩形的中心点、宽度、高度和角度信息。

中心点

表示查找到的直线的最小外接矩形的中心点。

中心 X

float 型，代表最小外接矩形中心点的 X 坐标。

中心 Y

float 型，代表最小外接矩形中心点的 Y 坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

卡尺框检测区

box 型，表示可完整包含所有卡尺框的检测区域，记录了该检测区域的中心点、宽度、高度和角度信息。

卡尺框检测中心点

point 型，表示卡尺框检测区域的中心点，记录了该检测区域的中心点坐标。

卡尺框检测中心 X

float 型，代表卡尺框检测区中心点的 X 坐标。

卡尺框检测中心 Y

float 型，代表卡尺框检测区中心点的 Y 坐标。

卡尺框检测区宽度

float 型，代表每个卡尺框检测区的宽度，一般相等。

卡尺框检测区高度

float 型，代表每个卡尺框检测区的高度，一般相等。

卡尺框检测区角度

float 型，代表每个卡尺框检测区与水平方向的夹角。

13.2.15 多直线查找

*多直线查找*模块可同时查找一个检测区域内的多条直线段，主要应用于半导体等行业的测量场景，如硅片间距测量和引脚间距测量。

使用场景

- 需要在同一区域定位多条直线时，如果使用 *直线查找* 模块，需要在流程中放置多个该模块。而如果使用*多直线查找*模块，仅需要放置单个该模块即可。
- 多帧图像的特定区域内，期望边线存在角度变化、位移、伸缩时，对*多直线查找*模块算法的影响较*直线查找*的较小。因为*多直线查找*搜索区域比*直线查找*的搜索区域更大。
- 需要查找多条直线后的二次计算，如计数、特殊要求的间距计算等。

说明

若仅需单纯的间距计算， *间距检测* 模块更适用。

使用方法

在流程中，*多直线查找*的：

- 前序模块通常为 *图像源* 、 *模板匹配* 、 *位置修正* 等模块。*模板匹配*和*位置修正*做直线的粗定位，*多直线查找*做精定位。
- 后序模块可为 *线线测量* 和组合模块等。这些模块的处理*多直线查找*输出的直线数据，实现直线段间距的测量。

以下两张图所示的应用示例中，*多直线查找*输出的多直线信息，依次经组合模块中的 *变量计算* 、 *线线测量* 、 *图形收集* 和 *数据集合* 处理，最终输出多个直线段的间距。

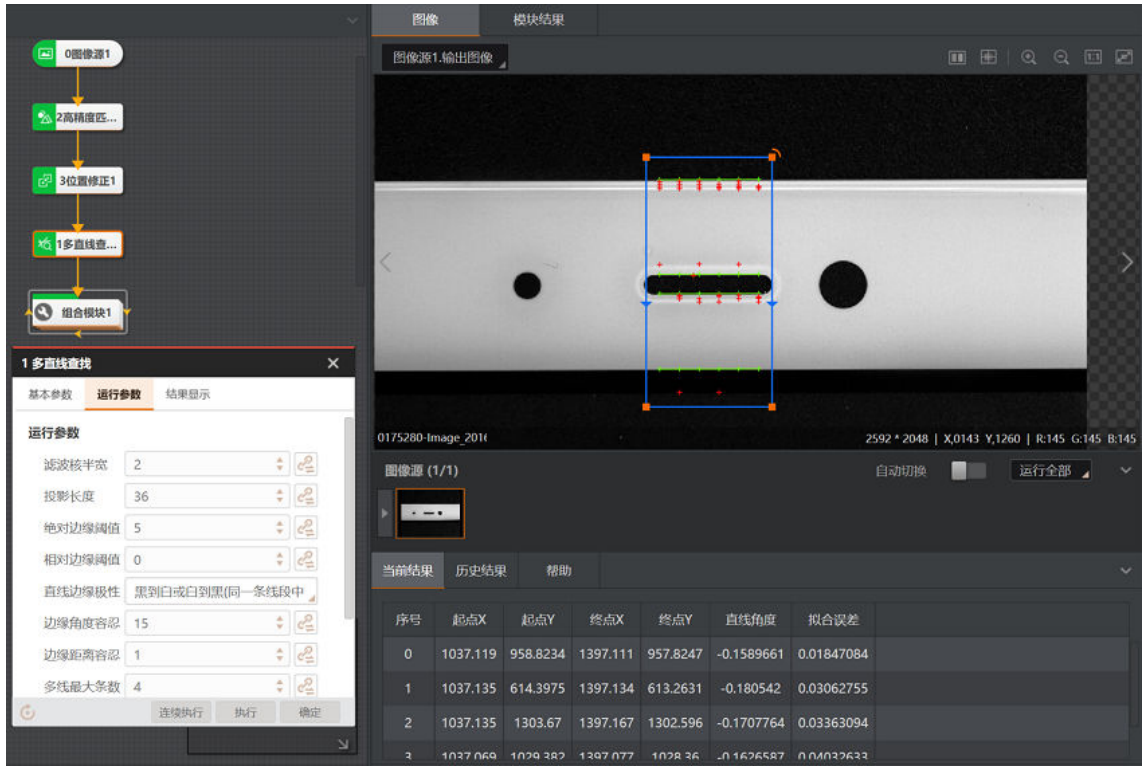


图 13-113 多直线查找输出多直线信息

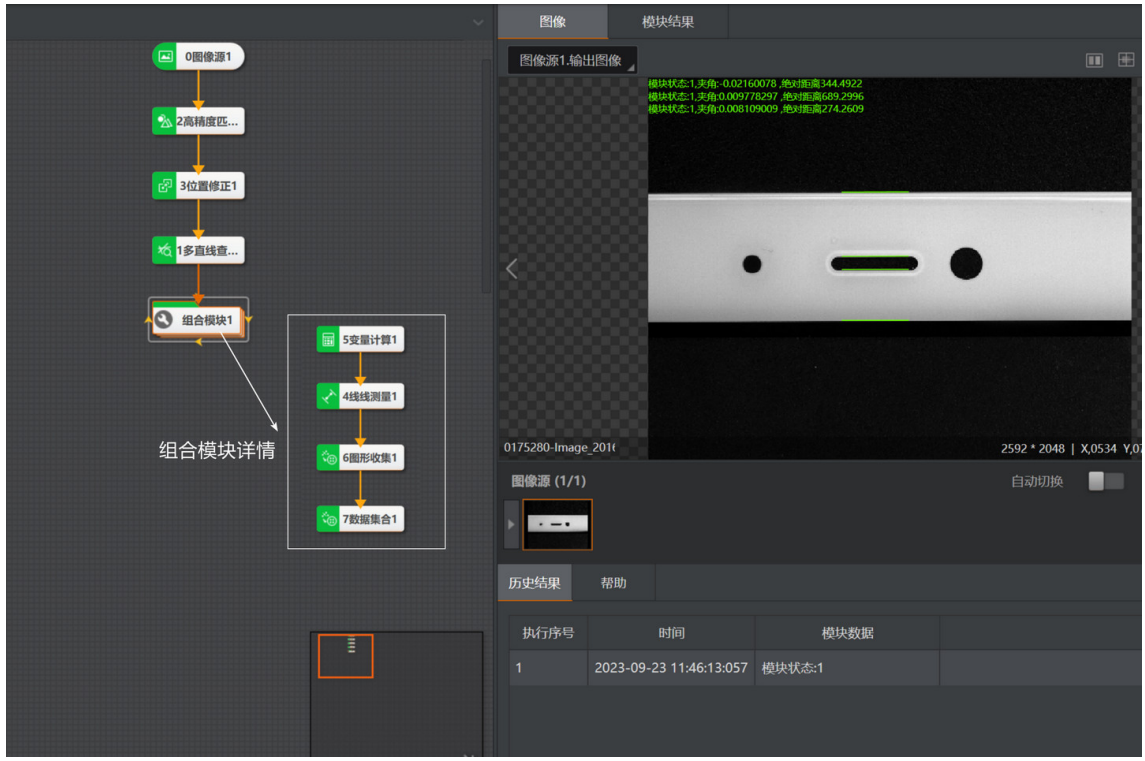


图 13-114 组合模块测量多线段间距

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数*。
- 结果显示参数详情，请参见 *结果显示*。

滤波核半宽

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使检测结果更加稳定，但如果边缘与边缘之间挨得太近时反而会影响边缘位置的精度甚至丢失边缘。

投影长度

决定用于梯度场投影的区域的数量。该值越小，允许工具更细的粒度来分析图像，但可能导致更长的算法耗时。该值越大，算法耗时越小，但可能导致边缘无法被检测出。

建议该参数与**滤波尺寸**取相同值。

绝对/相对边缘阈值

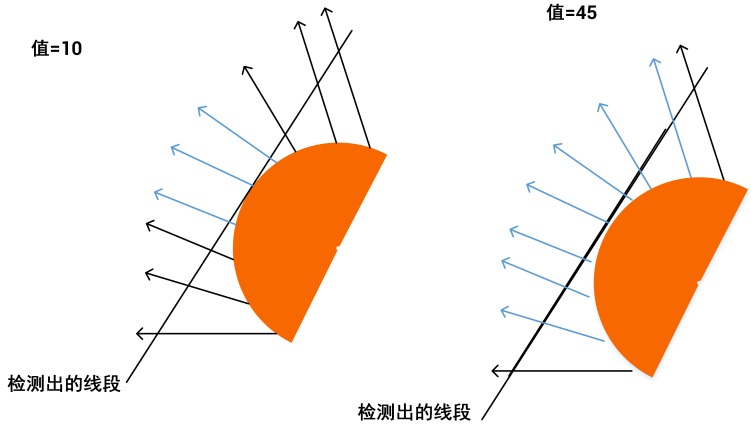
只有边缘梯度阈值大于提取阈值的边缘点才可被检测出。

边缘极性

边缘的灰度变化类型。可选**由黑到白**、**由白到黑**、**任意**，更多详情请参见 ***直线查找*** 的同名运行参数。

边缘角度容忍

边缘点梯度方向与垂直于拟合直线的方向（拟合直线法线方向）的最大允许角度差。增加该值可使算法提取更多的边缘点，从而改变检测出的线段的位置。



边缘角度容忍

边缘点梯度方向与垂直于拟合直线的方向（拟合直线法线方向）的最大允许角度差。增加该值可使算法提取更多的边缘点，从而改变检测出的线段的位置。

边缘距离容忍

边缘点与拟合线的最大允许距离。该值越大，算法提取更多边缘点，进而改变检测出的线段的位置。

多线最大条数

查找目标直线的最大条数。

覆盖率阈值

实际选用的边缘点数量占直线段理想使用的边缘点数量的最小百分比。该值越大，越倾向于输出覆盖率更高的直线段。

说明

覆盖率越高并不说明覆盖的边缘点数越多。需要结合直线段实际长度综合考虑。

旋转角度容忍

所发现的线段的旋转量与定义的梯度搜索方向的最大偏差。该值越低，检测出的线段与梯度搜索方向的平行度越高。

拟合方式

两种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用 huber、tukey。

排序类型

可设置输出直线的顺序，可选得分、ROI 方向排序、ROI 反方向排序。

边缘点数限制最大值

可设置提取特征点数的最大值。推荐使用默认值。

模块结果

*多直线查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

直线个数

int 型，代表定位到的直线的数量。

边缘点个数

int 型，代表定位到的直线上的边缘点个数。

直线段信息

记录了定位到的直线段的起点和终点坐标。

直线起点

起点 X

float 型，代表输出直线起点的 X 坐标。

起点 Y

float 型，代表输出直线起点的 Y 坐标。

直线终点

终点 X

float 型，代表输出直线终点的 X 坐标。

终点 Y

float 型，代表输出直线终点的 Y 坐标。

直线角度

float 型，代表各条输出直线的角度。

拟合误差

float 型，代表真实直线与拟合直线的误差。

直线对比度强度

float 型，代表图像中各条直线边缘的明暗变化程度。该值越大，图像中对应直线的边缘越清晰。

覆盖率分数

float 型，代表拟合直线在图像所有直线中的占比，反映了拟合直线在所有直线中的覆盖率。

在群点数量

int 型，代表图像中聚集在一起的点的数量。

线段索引

int 型，代表从上到下每个线段的索引。

边缘点信息

记录了各条直线上的边缘点坐标、边缘极性、梯度幅值和梯度方向等信息。

边缘点

表示多直线查找模块检测到的边缘点。

轮廓点 X

float 型，代表边缘轮廓点的 X 坐标。

轮廓点 Y

float 型，代表边缘轮廓点的 Y 坐标。

边缘极性

int 型，代表目标对象边缘的亮度变化方向或极性。

梯度幅值

float 型，代表图像上某一点处的梯度的大小或强度。

梯度方向

float 型，代表图像上某一点处的梯度的方向。

边缘距离

float 型，代表图像上某一点到最近边缘的距离。

边缘状态

int 型，代表图像上某个点是否位于边缘。1 代表位于边缘，0 代表不在边缘。

投影区域索引

int 型，代表边缘点所处的投影区域的索引值。

所属线段索引

int 型，代表边缘点所处的线段的索引值。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.16 边缘交点

*边缘交点*模块用于识别图像 ROI 中两条边缘（直线）的交点，并计算交点的坐标信息。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块查找直线的原理和 *直线查找模块的原理* 相同。查找到两条直线后，通过直线方程即可得到两条直线的交点。
以下图为例，两个蓝色框为直线查找的 ROI 区域，绿色或红色的点为直线查找提取到的点，绿色的线为查找到的直线，而这两条直线相交的点（即橙色的点）就是边缘交点。

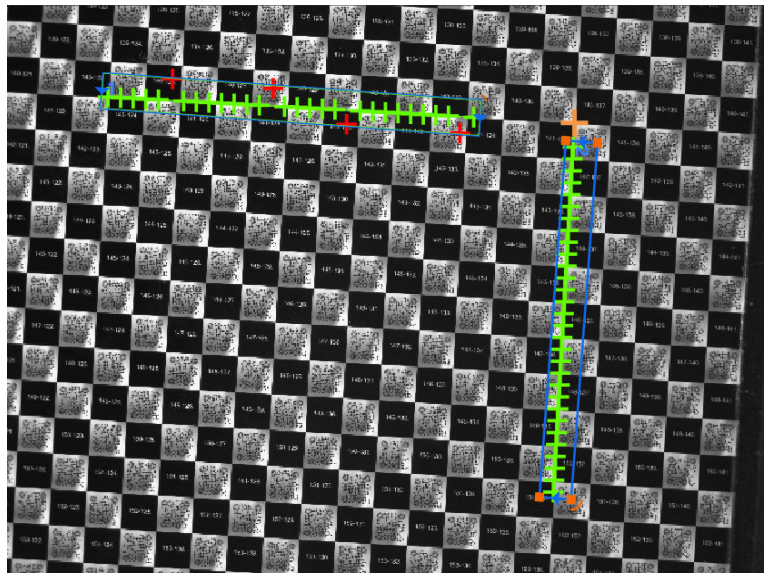


图 13-115 边缘交点示意图

使用方法

在流程中：

- *图像源*、*模板匹配*、*位置修正* 等模块可作为*边缘交点*的前序模块，为*边缘交点*在图像指定区域精确定位边缘交点。*模板匹配*和*位置修正*做粗定位，*边缘交点*做精定位。
- *边缘交点*对后序模块无特殊要求。可接收并处理边缘交点信息的模块均可作为*边缘交点*的后续模块。

下图展示了边缘交点检测的应用示例。

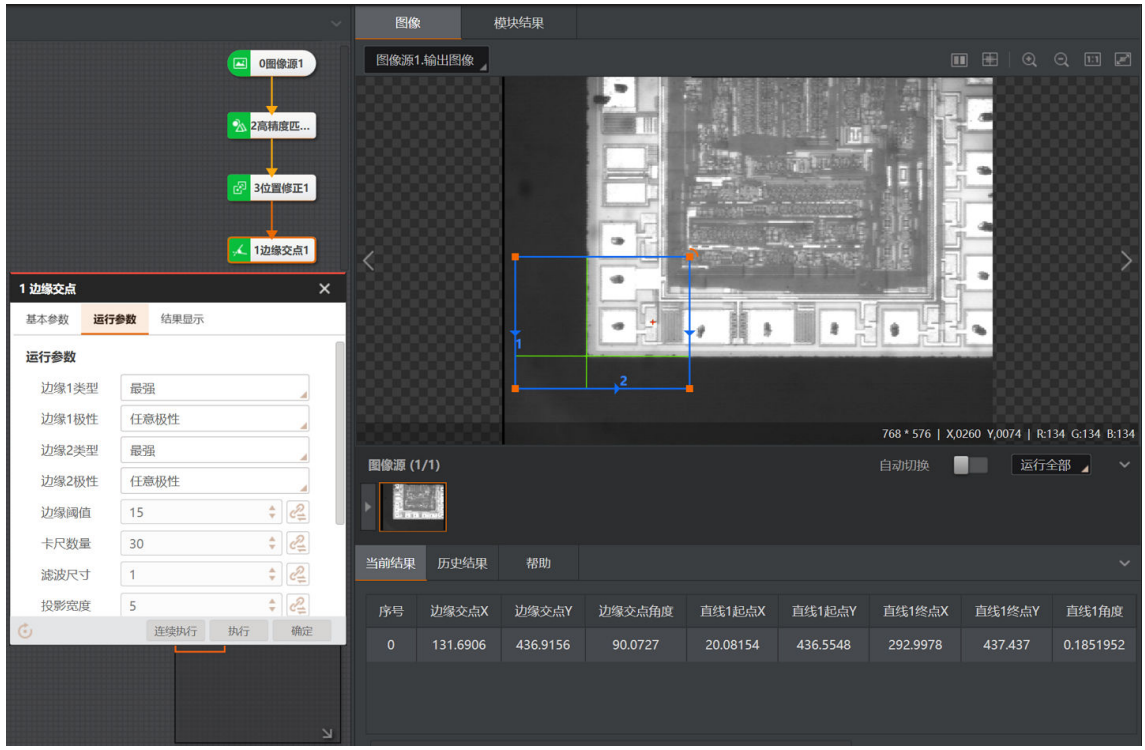


图 13-116 边缘交点检测

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

边缘类型

可选最强、第一条、最后一条。

最强

查找梯度（即灰度值变化）最大的边缘点集合，并将其拟合为直线。

第一条

查找离查找起点最近的边缘点集合，并将其拟合为直线。

最后一条

查找离查找起点最远的边缘点集合，并将其拟合为直线。

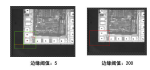
边缘极性

边缘极性有三个选择，即**从黑到白**、**从白到黑**以及**任意**。下图绿色箭头所示为白到黑，红色箭头所示为黑到白。此处“黑”和“白”仅表示灰度值的强弱程度。



边缘阈值

边缘阈值即梯度阈值，范围 0~255，只有边缘梯度阈值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。如下图所示，在阈值设为 5 时是可以识别出边缘并找出边缘交点，而在阈值设为 200 后边缘无法被定位到也就无法找到边缘交点。



滤波尺寸

描述目标边缘的清晰程度，最小值为 1，值越小，表示边缘越清晰且过滤带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间挨得太近（距离小于滤波尺寸）时反而会影响边缘位置的精度甚至丢失边缘。具体原理参考 ***圆查找*** 中的同名运行参数。

卡尺数量

定义卡尺的数量。边缘点由多个卡尺提取出。

剔除点数

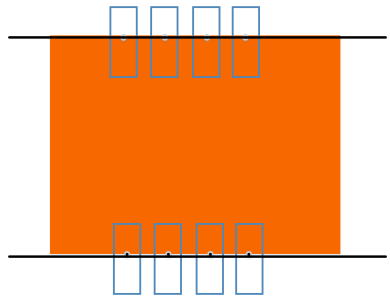
详见 ***圆查找*** 中的同名运行参数。

剔除距离

允许离群点到拟合直线的最大像素距离，值越小，排除点越多，原理以及操作请参考 ***圆查找*** 中的同名运行参数。

投影宽度

即卡尺的宽度(下图中蓝色矩形的宽度)。在一定范围内增大该值可以获取更加稳定的边缘点。



初始拟合

详见 *圆查找* 中的同名运行参数。

拟合方式

包括**最小二乘**、**huber** 和 **tukey** 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用**最小二乘**、**huber** 和 **tukey**。

模块结果

*边缘交点*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

边缘交点

记录了边缘交点的坐标信息。

边缘交点 X

float 型，代表检测到的边缘交点的 X 坐标。

边缘交点 Y

float 型，代表检测到的边缘交点的 Y 坐标。

边缘交点角度

float 型，代表检测到的两条边缘之间的夹角。

边缘直线*（*代表 1-2）

line 型，表示当前模块查找到的所有边缘直线，记录了每条边缘直线的起点和终点坐标。

直线*起点

point 型，代表边缘直线*的起点坐标。

直线*起点 X

float 型，代表边缘直线*起点的 X 坐标。

直线*起点 Y

float 型，代表边缘直线*起点的 Y 坐标。

直线*终点

point 型，代表边缘直线*的终点坐标。

直线*终点 X

float 型，代表边缘直线*终点的 X 坐标。

直线*终点 Y

float 型，代表边缘直线*终点的 Y 坐标。

直线*角度（*代表 1-2）

float 型，代表边缘直线与水平轴之间的夹角。其中，顺时针旋转为正角度，逆时针旋转为负角度。

直线*轮廓点（*代表 1-2）

直线*轮廓点 X

float 型，代表用于检出边缘直线上各个轮廓点的 X 坐标。

直线*轮廓点 Y

float 型，代表用于检出边缘直线上各个轮廓点的 Y 坐标。

直线*轮廓点数（*代表 1-2）

int 型，代表检出边缘直线上轮廓点的个数。

直线*轮廓点状态（*代表 1-2）

int 型，代表检出边缘直线上各个轮廓点的状态，1 代表轮廓点符合要求，0 代表轮廓点不符合要求。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域*（*代表 1-2）

box 型，代表图像中需进行检测的 ROI 区域。

检测区域*中心点

point 型，表示检测区域的中心点。

检测区域*中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域*中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域*宽度

float 型，代表检测区域的宽度。

检测区域*高度

float 型，代表检测区域的高度。

检测区域*角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转则角度为正，逆时针旋转则角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.17 四边形查找

*四边形查找*用于检测图像指定区域内是否存在四边形，输出四边形信息，包括四边形对角线交点、中点线交点等信息。该模块常用于四边形目标物体的定位。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块首先查找图像中的四条直线，再将四条直线组成一个四边形。该模块查找直线的原理和 *直线查找模块的原理* 相同。
以下图为例，四个蓝色框为直线查找的 ROI 区域，绿色或红色的点为直线查找提取到的点，绿色的线为查找到的直线，将这四条线组合起来就是四边形。其中，紫色的点为四边形的四个顶点，黄色的线为四边形的对角线，中间橙色的点为四边形的中心点。

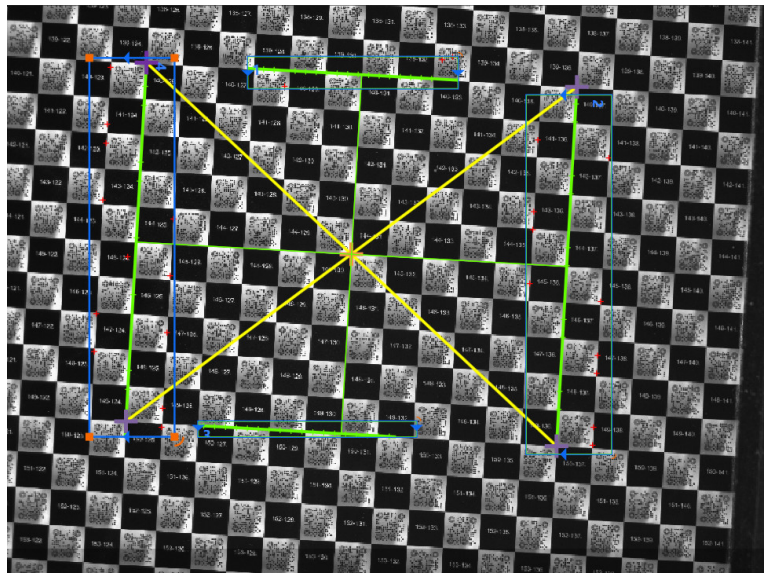


图 13-117 示意图

使用方法

在流程中，*图像源*、*模板匹配*、*位置修正*等模块可作为*四边形查找*的前序模块，为*四边形查找*在图像指定区域精确定位矩形。*模板匹配*和*位置修正*做粗定位，*四边形查找*做精定位。*四边形查找*对后序模块无特殊要求。可接收并处理四边形信息的模块均可作为*四边形查找*的后续模块。

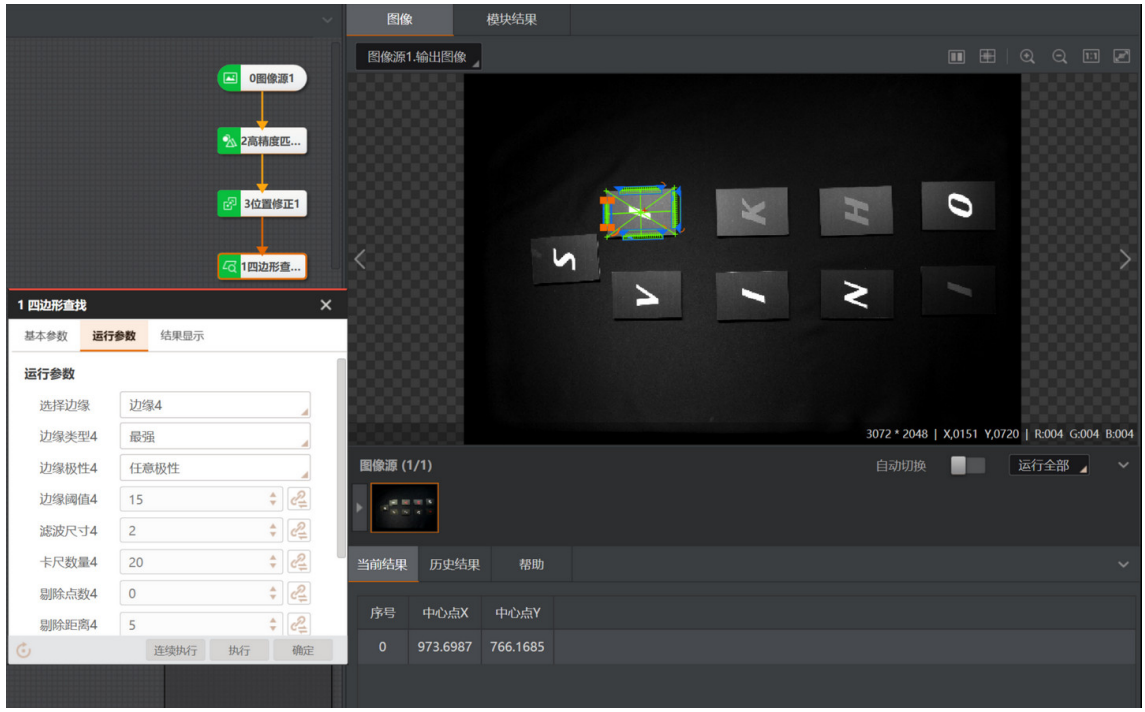


图 13-118 四边形检测应用示例

参数配置

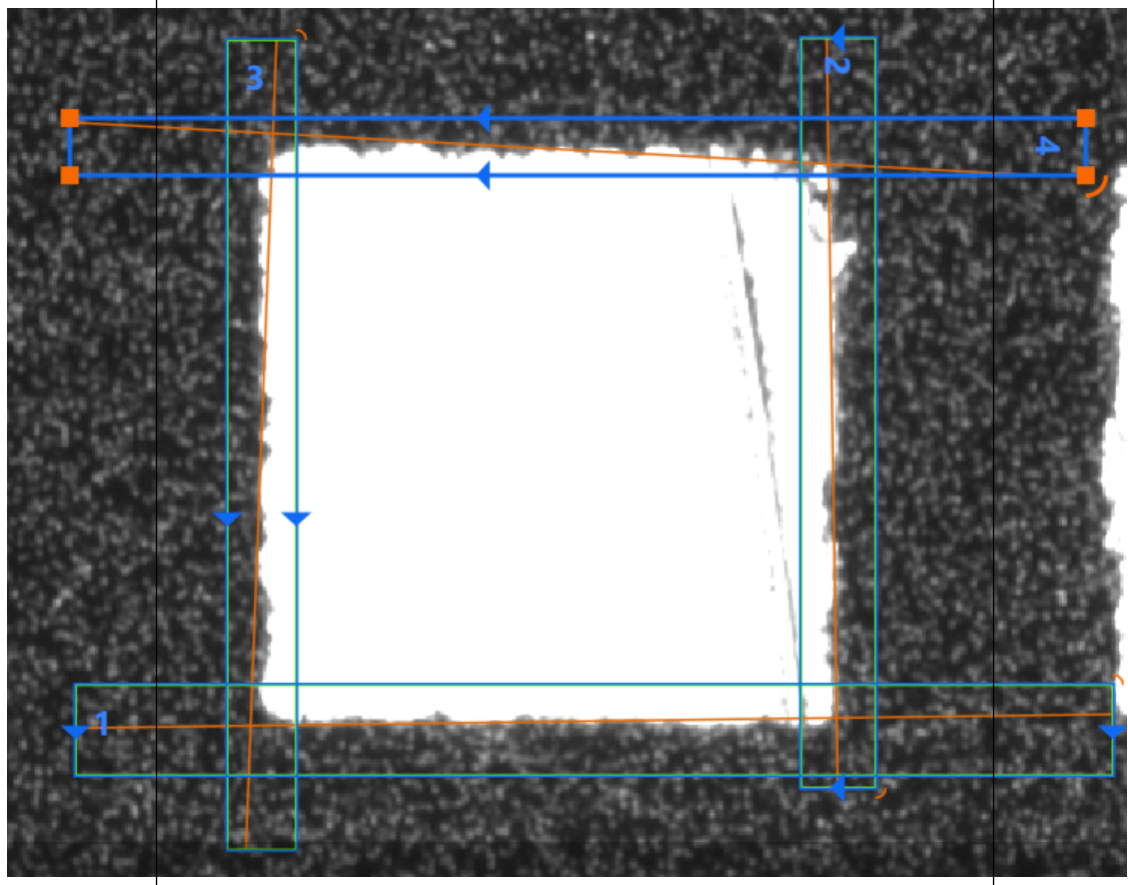
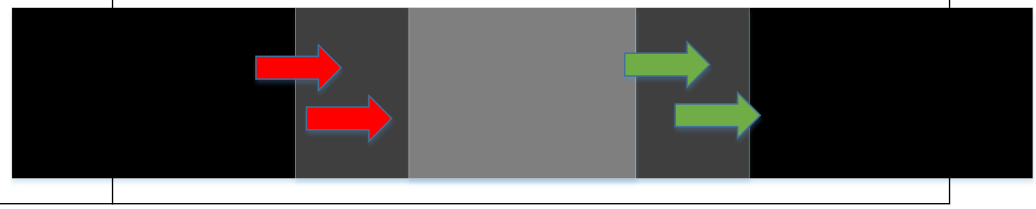
以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

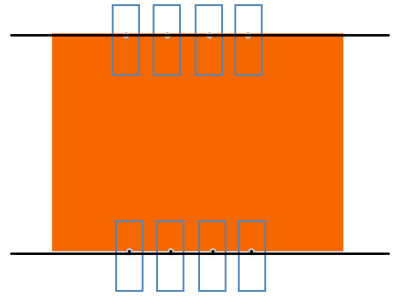
说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

表 13-10 运行参数

参数	描述
选择边缘	从下拉列表中选择四条待查找边缘的任意一条，为其配置本表格中其他运行参数。边缘 1/2/3/4 分别对应四个 ROI 的默认编号，如下图所示。

参数		描述
		 图 13-119 ROI 编号
边缘类型	最强	查找梯度阈值最大的边缘点集合，然后拟合为直线。
	第一条	查找离查找起点最近的边缘点集合，然后拟合为直线。
	最后一条	查找离查找起点最远的边缘点集合，然后拟合为直线。
边缘极性		矩形四条边的边缘极性，包括**从黑到白**、**从白到黑**以及**任意**。下图绿色箭头所示为白到黑，红色箭头所示为黑到白。此处“黑”和“白”仅表示灰度值的强弱程度。 

参数	描述
边缘阈值	边缘阈值即梯度阈值，范围 0~255，只有边缘梯度值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。
滤波尺寸	描述目标边缘的清晰程度，最小值为 1，值越小，表示边缘越清晰且过滤带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间挨得太近（距离小于滤波尺寸）时反而会影响边缘位置的精度甚至丢失边缘。具体原理参考 ***圆查找*** 中的同名运行参数。
卡尺数量	定义卡尺的数量。边缘点由多个卡尺提取出。
剔除点数	详见 ***圆查找*** 中的同名运行参数。
剔除距离	允许离群点到拟合直线的最大像素距离，值越小，排除点越多，原理以及操作请参考 ***圆查找*** 中的同名运行参数。
投影宽度	即卡尺的宽度（下图中蓝色矩形的宽度）。在一定范围内增大该值可以获取更加稳定的边缘点。 
初始拟合	详见 ***圆查找*** 中的同名运行参数。
拟合方式	包括**最小二乘**、**huber** 和 **tukey** 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用**最小二乘**、**huber** 和 **tukey**。

模块结果

*四边形查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

直线*状态（*代表 1-4）

int 型，可用于判断当前模块是否检测到直线，1 代表检测到直线，0 代表未检测到直线。

边缘直线*（*代表 1-4）

记录了当前边缘直线的起点和终点坐标。

直线*起点

表示当前边缘直线的起点坐标。

直线*起点 X

float 型，代表边缘直线*起点的 X 坐标。

直线*起点 Y

float 型，代表边缘直线*起点的 Y 坐标。

直线*终点

表示当前边缘直线的终点坐标。

直线*终点 X

float 型，代表边缘直线*终点的 X 坐标。

直线*终点 Y

float 型，代表边缘直线*终点的 Y 坐标。

直线*角度（*代表 1-4）

float 型，代表边缘直线相对于水平轴的旋转角度，顺时针旋转为正，逆时针旋转为负。

直线*拟合误差（*代表 1-4）

float 型，代表拟合直线与实际直线之间的误差。

直线*轮廓点（*代表 1-4）

记录了检测到的边缘直线上的轮廓点坐标。

边缘*轮廓点 X

float 型，代表边缘直线*轮廓点的 X 坐标。

边缘*轮廓点 Y

float 型，代表边缘直线*轮廓点的 Y 坐标。

边缘*轮廓点数（*代表 1-4）

int 型，代表用于检出边缘直线上轮廓点的个数。

边缘*轮廓点状态（*代表 1-4）

int 型，可用于判断检出边缘直线上各个轮廓点的状态。1 代表轮廓点符合要求，0 代表轮廓点不符合要求。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域*（*代表 1-4）

box 型，代表图像中需进行检测的 ROI 区域。

检测区域*中心点

point 型，表示检测区域的中心点。

检测区域*中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域*中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域*宽度

float 型，代表检测区域的宽度。

检测区域*高度

float 型，代表检测区域的高度。

检测区域*角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转则角度为正，逆时针旋转则角度为负。若检测区域处于正准位姿，则为 0 度。

对角线*（*代表 1-2）

记录了查找到的四边形各对角线的起点和终点坐标。

对角线*起点

表示当前对角线的起点坐标。

对角线*起点 X

float 型，代表四边形对角线起点的 X 坐标。

对角线*起点 Y

float 型，代表四边形对角线起点的 Y 坐标。

对角线*终点

表示当前对角线的终点坐标。

对角线*终点 X

float 型，代表四边形对角线终点的 X 坐标。

对角线*终点 Y

float 型，代表四边形对角线终点的 Y 坐标。

对角线*角度（*代表 1-2）

float 型，代表四边形对角线相对于水平轴的旋转角度，顺时针旋转为正，逆时针旋转为负。

中点线*（*代表 1-2）

中点线*起点

记录了中点线的起点坐标。

中点线*起点 X

float 型，代表四边形中点线起点的 X 坐标。

中点线*起点 Y

float 型，代表四边形中点线起点的 Y 坐标。

中点线*终点

记录了中点线的终点坐标。

中点线*终点 X

float 型，代表四边形中点线终点的 X 坐标。

中点线*终点 Y

float 型，代表四边形中点线终点的 Y 坐标。

中点线*角度（*代表 1-2）

float 型，代表四边形的中点线相对于水平轴的旋转角度，顺时针旋则为正，逆时针转则为负。

临边夹角*（*代表 1-4）

float 型，代表查找到的四边形相邻两边的夹角，顺时针方向为正角度，逆时针方向为负角度。

顶点*（*代表 1-4）

记录了查找到的四边形的顶点坐标。

顶点*X

float 型，代表当前顶点的 X 坐标。

顶点*Y

float 型，代表当前顶点的 Y 坐标。

对角线交点

记录了查找到的四边形对角线的交点坐标。

对角线交点 X

float 型，代表四边形两条对角线交点的 X 坐标。

对角线交点 Y

float 型，代表四边形两条对角线交点的 Y 坐标。

中点线交点

记录了四边形中位线的交点坐标。

中点线交点 X

float 型，代表四边形两条中点线的交点的 X 坐标。

中点线交点 Y

float 型，代表四边形两条中点线的交点的 Y 坐标。

中心点

记录了查找到的四边形的中心点坐标。

中心点 X

float 型，代表四边形中心点的 X 坐标。

中心点 Y

float 型，代表四边形中心点的 Y 坐标。

对边角平分线交点

记录了查找到的四边形的对边角平分线的交点坐标。

对边角平分线交点 X

float 型，代表四边形对边角平分线交点的 X 坐标。

对边角平分线交点 Y

float 型，代表四边形对边角平分线交点的 Y 坐标。

对边角平分线*（*代表 1-2）

记录了查找到的对边角平分线的起点和终点坐标。

对边角平分线*起点

表示查找到的四边形的对边角平分线的起点坐标。

对边角平分线*起点 X

float 型，代表四边形对边角平分线起点的 X 坐标。

对边角平分线*起点 Y

float 型，代表四边形对边角平分线起点的 Y 坐标。

对边角平分线*终点

表示查找到的四边形的对边角平分线的终点坐标。

对边角平分线*终点 X

float 型，代表四边形对边角平分线终点的 X 坐标。

对边角平分线*终点 Y

float 型，代表四边形对边角平分线终点的 Y 坐标。

对边角平分线*角度（*代表 1-2）

float 型，代表四边形的对角平分线相对于水平轴的旋转角度，顺时针旋转则为正，逆时针旋转则为负。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.18 平行线查找

*平行线查找*模块用于检测图像指定区域内的平行线，输出平行线信息，包括平行线起点/终点、平行线中线起点/终点以及平行线间距，常用于部件平行线特征的检测。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块在检测区域内使用卡尺提取特征点。每个卡尺会提取到多个特征点，这些特征点即可拟合成两条平行的直线，从而查找到平行线。

说明

卡尺提取特诊点的过程与*直线查找*模块提取特征点的过程类似。

以下图为例，蓝色框为待检测的 ROI 区域，红色或绿色的点为该区域内提取到的特征点。其中绿色的点拟合成直线，得到两条绿色的直线，即最终查找到的平行线。

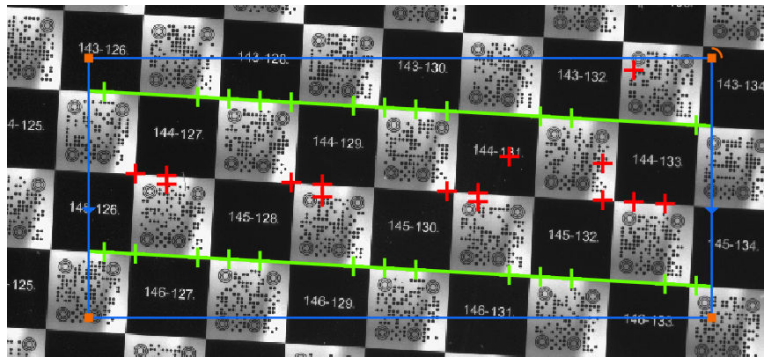


图 13-120 示意图

使用方法

在流程中：

- *图像源*、*模板匹配*、*位置修正*等模块可作为*平行线查找*的前序模块，为*平行线查找*在图像指定区域精确定位平行线。*模板匹配*和*位置修正*做粗定位，*平行线查找*做精定位。
- *平行线查找*对后序模块无特殊要求。可接收并处理平行线信息的模块均可作为*平行线查找*的后续模块。

*平行线查找*的检测结果中，两条绿色粗线为一组平行线，中间细线为中线。

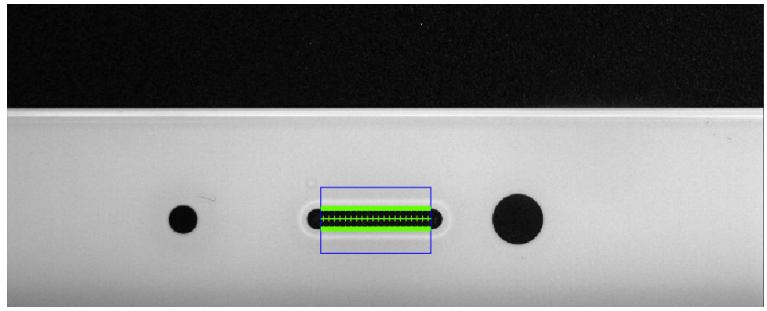


图 13-121 检测结果示例

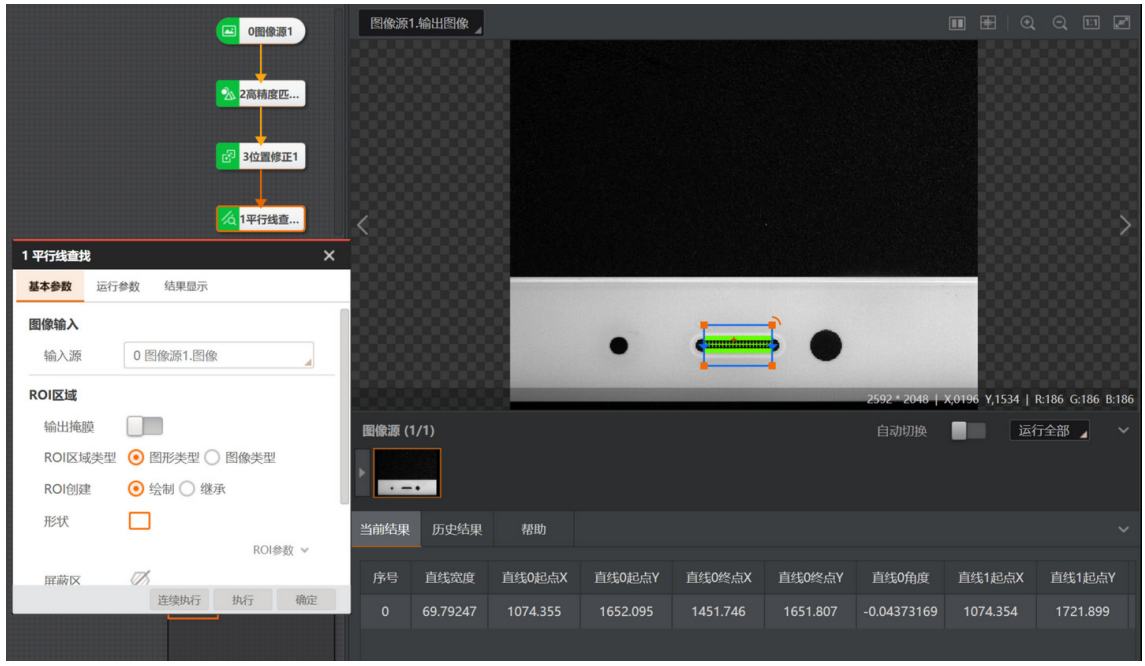


图 13-122 平行线查找应用示例

参数配置

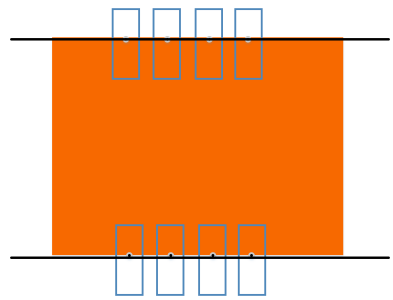
以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数*。
- 结果显示参数详情，请参见 *结果显示*。

表 13-11 运行参数

参数		描述
边缘类型	最宽	检测 ROI 内间距最大的边缘对。
	最窄	检测 ROI 内间距最小的边缘对。
	最强	检测 ROI 内边缘对平均梯度最大的边缘对。
	最弱	检测 ROI 内梯度最小的边缘对。
	第一对	检测 ROI 内边缘对中心与搜索起始点最近的边缘对。
	最后一对	检测 ROI 内边缘对中心与搜索起始点最远的边缘对。
	最接近	检测 ROI 内和理想宽度最接近的边缘对集合。
	最不接近	检测扫 ROI 内和理想宽度最不接近的边缘对集合。
边边缘极性		边缘的灰度值变化类型，包括**从黑到白**、**从白到黑**以及**任意**。下图绿色箭头所示为白到黑，红色箭头所示为黑到白。此处“黑”和“白”仅表示灰度值的强弱程度。 
边缘阈值		边缘阈值即梯度阈值，范围 0~255，只有边缘梯度值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。如下图所示，在阈值设为 10 时是可以识别出边缘并找出平行线，而在阈值设为 200 后，边缘无法被定位到，也就无法找到平行线。

参数	描述
	 边缘阈值：10  边缘阈值：200
滤波尺寸	描述目标边缘的清晰程度，最小值为 1，值越小，表示边缘越清晰且过滤带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间挨得太近（距离小于滤波尺寸）时反而会影响边缘位置的精度甚至丢失边缘。具体原理参考 ***直线查找*** 中的同名参数。
卡尺数量	定义卡尺的数量。边缘点由多个卡尺提取出。
最大角度差	两条目标直线的角度差小于该值时才会被判定为平行线。
剔除点数	详见 ***直线查找*** 中的同名参数。
剔除距离	允许离群点到拟合直线的最大像素距离，值越小，排除点越多，原理以及操作请参考 ***圆查找*** 中的同名运行参数。
滤波尺寸	对噪点起到过滤作用，数值越大抗噪能力越强，得到的边缘数量越少，同时也可能导致目标边缘被筛除。
投影宽度	即卡尺的宽度（下图中蓝色矩形的宽度）。在一定范围内增大该值可以获取更加稳定的边缘点。 

参数	描述
初始拟合	详见 ***直线查找*** 中的同名参数。
拟合方式	包括**最小二乘**、**huber** 和 **tukey** 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用**最小二乘**、**huber** 和 **tukey**。

模块结果

*平行线查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

检测状态

int 型，代表是否检测到平行线，0 代表未检测到平行线，1 代表检测到平行线。

直线宽度

float 型，代表查找到的平行线的宽度。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

边缘直线*（*代表 0-1）

记录了查找到的边缘直线的起点和终点坐标。

直线*起点

表示当前边缘直线的起点坐标。

直线*起点 X

float 型，代表边缘直线*起点的 X 坐标。

直线*起点 Y

float 型，代表边缘直线*起点的 Y 坐标。

直线*终点

表示当前边缘直线的终点坐标。

直线*终点 X

float 型，代表边缘直线*终点的 X 坐标。

直线*终点 Y

float 型，代表边缘直线*终点的 Y 坐标。

直线*角度（*代表 0-1）

float 型，代表边缘直线相对于水平线方向的旋转角度。顺时针旋转，则为正角度；逆时针旋转，则为负角度。

边缘中线

表示两条平行线之间的中线，记录了该中线的起点和终点坐标。

中线起点

表示查找到两条平行线之间的中线。

中线起点 X

float 型，代表中线起点的 X 坐标。

中线起点 Y

float 型，代表中线起点的 Y 坐标。

中线终点

中线终点 X

float 型，代表中线终点的 X 坐标。

中线终点 Y

float 型，代表中线终点的 Y 坐标。

中线角度

float 型，代表中线相对于水平轴的旋转角度，顺时针旋转则为正，逆时针旋转为负。

直线*轮廓点（*代表 0-1）

记录了用于找出平行线的所有轮廓点的坐标。

边缘*轮廓点 X

float 型，代表边缘直线*轮廓点的 X 坐标。

边缘*轮廓点 Y

float 型，代表边缘直线*轮廓点的 Y 坐标。

边缘点个数

int 型，代表边缘直线上的边缘点个数。

边缘*轮廓点状态（*代表 0-1）

int 型，可用于判断查找到的边缘轮廓点的状态，1 表示轮廓点符合要求，0 表示轮廓点不符合要求。

中线轮廓点

记录了用于找出中线的所有轮廓点坐标。

中线轮廓点 X

float 型，代表用于找出中线的各个轮廓点的 X 坐标。

中线轮廓点 Y

float 型，代表用于找出中线的各个轮廓点的 Y 坐标。

中线轮廓点状态

int 型，代表中线轮廓点的状态，1 表示轮廓点符合要求，0 表示轮廓点不符合要求。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.19 平行线计算

*平行线计算*模块可基于指定直线，计算得出该直线的平行线。

本节内容包含：

- *核心概念*
- *使用方法*
- *参数配置*
- *模块结果*

核心概念

通过该模块可按两种方式计算得出平行线——**与直线相距一定距离**和**过直线外一点**。

与直线相距一定距离

最终计算得出指定直线的两条平行线。两条平行线均与指定直线相差指定距离。

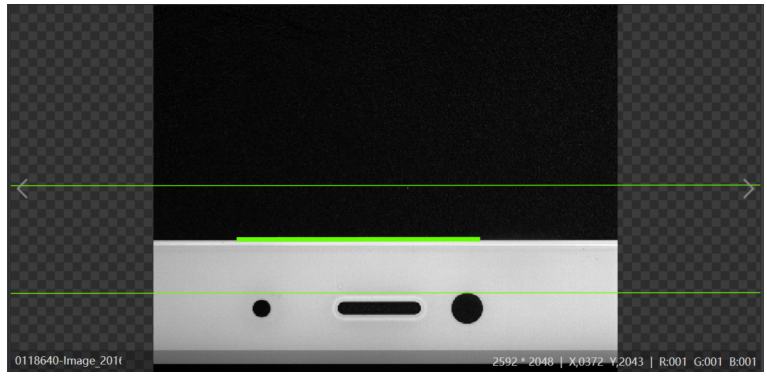


图 13-123 与直线相距一定距离

过直线外一点

最终计算得出指定直线的一条平行线，该平行线经过指定的点。



图 13-124 过直线外一点

使用方法

前后序模块

在流程中，*平行线计算*的

- 前序模块一般为 *直线查找* 。*直线查找*负责为该模块提供直线数据输入。
- 后序模块无特殊要求，可接收并处理平行线数据即可。

主要配置步骤

在流程中调用*平行线计算*模块后，该模块的主要配置步骤如下：

1. 在*输入源*处下拉选择图像数据源。
2. 在*方式选择*下拉列表选择平行线的计算方式（**与直线相距一定距离**或**过直线外一点**。
3. 设置其他输入数据。

- 计算方式设置为**与直线相距一定距离**时，需指定**线输入**，并设置待生成的平行线与该直线的**间距**。
- 计算方式设置为**过直线外一点**时，需指定**点输入**和**线输入**。

4. 切换到模块的*结果显示*页签，对图像显示进行设置。

应用示例

下图中所展示的部件上，较粗的绿线为指定直线，较细的绿线即计算得出的平行线。

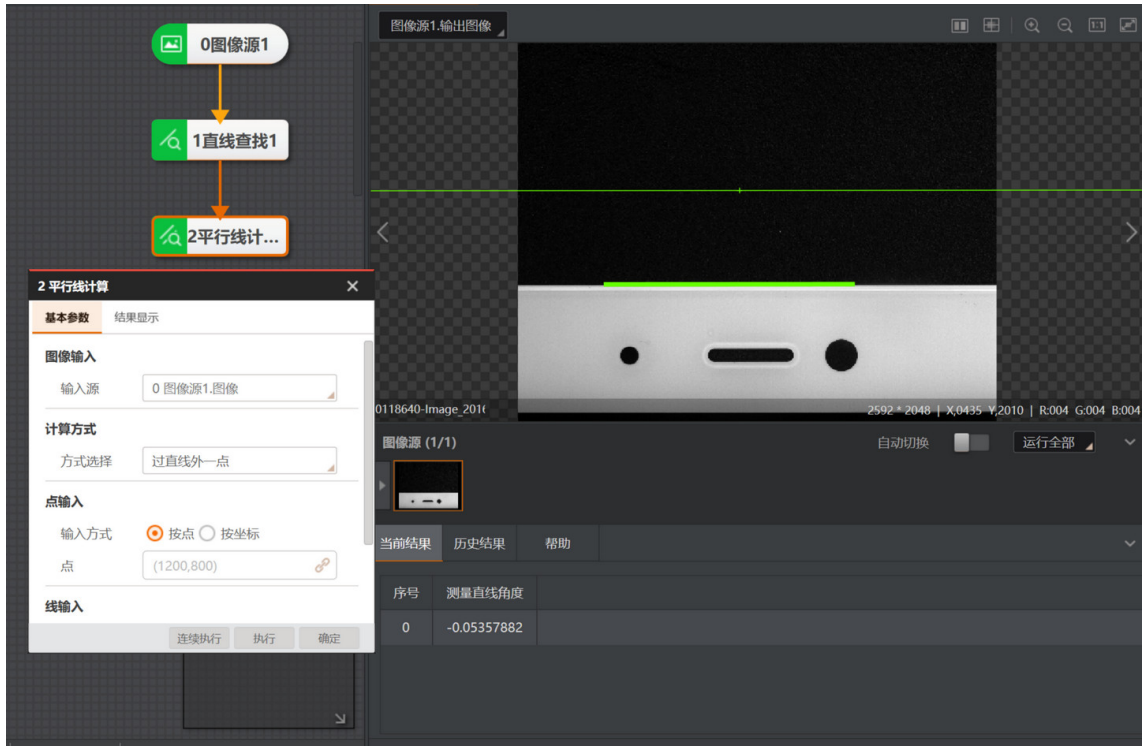


图 13-125 平行线计算应用示例

参数配置

结果显示参数详情，请参见 *结果显示* 。

模块结果

*平行线计算*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量点

point 型，表示输入的测量点坐标。

测量点 X

float 型，代表输入点的 X 坐标。

测量点 Y

float 型，代表输入点的 Y 坐标。

测量直线

line 型，表示输入直线。

测量直线起点

point 型，表示输入直线的起点坐标。

测量直线起点 X

float 型，代表输入直线的起点 X 坐标。

测量直线起点 Y

float 型，代表输入直线的起点 Y 坐标。

测量直线终点

point 型，表示输入直线的终点。

测量直线终点 X

float 型，代表输入直线的终点 X 坐标。

测量直线终点 Y

float 型，代表输入直线的终点 Y 坐标。

测量直线角度

float 型，代表输入直线相对于水平线方向的旋转角度。顺时针旋转为正，逆时针旋转为负。

平行线

平行线起点

平行线起点 X

float 型，当平行线**计算方式**选用**过直线外一点**时，代表检测到的平行线起点 X 坐标。

平行线起点 Y

float 型，当平行线**计算方式**选用**过直线外一点**时，代表检测到的平行线起点 Y 坐标。

平行线终点

平行线终点 X

float 型，当平行线**计算方式**选用**过直线外一点**时，代表检测到的平行线终点 X 坐标。

平行线终点 Y

float 型，当平行线**计算方式**选用**过直线外一点**时，代表检测到的平行线终点 Y 坐标。

平行线*（*代表 1-2）

平行线*起点

表示当前检测到的平行线的起点坐标。

平行线*起点 X

float 型，当平行线**计算方式**选用**与直线相距一定距离**时，代表检测到的平行线起点 X 坐标。

平行线*起点 Y

float 型，当平行线**计算方式**选用**与直线相距一定距离**时，代表检测到的平行线起点 Y 坐标。

平行线*终点

表示当前检测到的平行线的终点坐标。

平行线*终点 X

float 型，当平行线**计算方式**选用**与直线相距一定距离**时，代表检测到的平行线终点 X 坐标。

平行线*终点 Y

float 型，当平行线**计算方式**选用**与直线相距一定距离**时，代表检测到的平行线终点 Y 坐标。

13.2.20 矩形检测

*矩形检测*模块用于检测目标图像中的 ROI 内是否存在矩形。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

在流程中，*图像源*、*模板匹配*、*位置修正*等模块可作为*矩形检测*的前序模块，为*矩形检测*在图像指定区域精确定位矩形。*模板匹配*和*位置修正*做粗定位，*矩形检测*做精定位。
*矩形检测*对后序模块无特殊要求。可接收并处理矩形信息的模块均可作为*矩形检测*的后序模块。

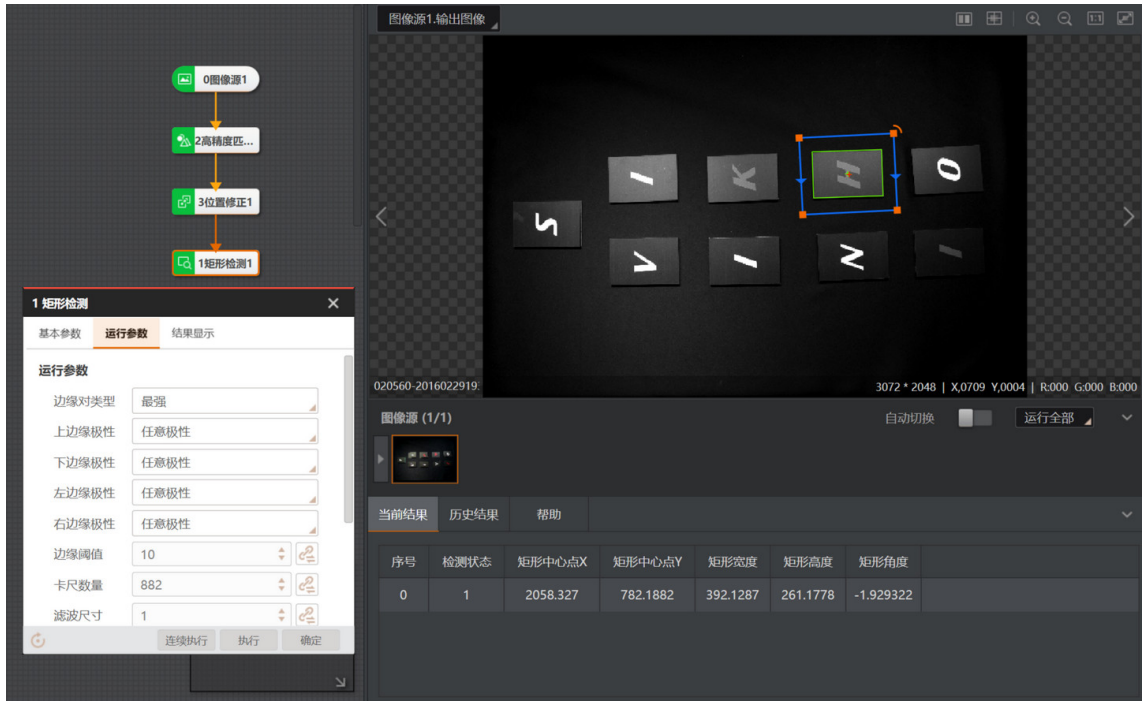


图 13-126 矩形检测应用示例

参数配置

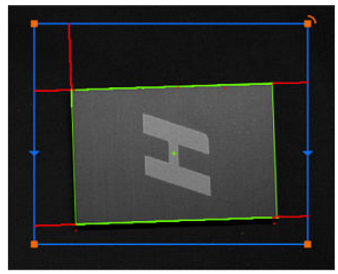
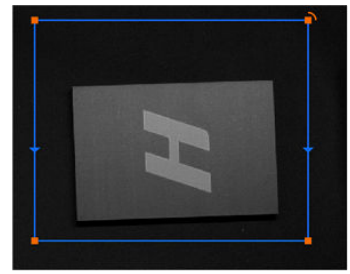
以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

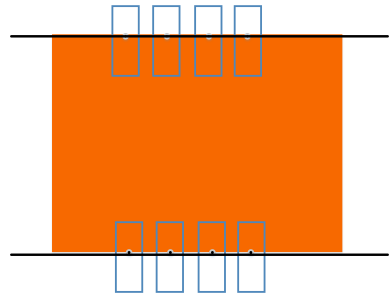
说明

- 基本参数详情，请参见 *基本参数*。
- 结果显示参数详情，请参见 *结果显示*。

表 13-12 运行参数

参数		描述
边缘类型	最宽	检测 ROI 内间距最大的边缘对。
	最窄	检测 ROI 内间距最小的边缘对。
	最强	检测 ROI 内边缘对平均梯度最大的边缘对。
	最弱	检测 ROI 内梯度最小的边缘对。
	第一对	检测 ROI 内边缘对中心与搜索起始点最近的边缘对。

参数		描述
	最后一对	检测 ROI 内边缘对中心与搜索起始点最远的边缘对。
	最接近	检测 ROI 内和理想宽度最接近的边缘对集合。
	最不接近	检测扫 ROI 内和理想宽度最不接近的边缘对集合。
上/下/左/右边边缘极性		矩形四条边的边缘极性，包括**从黑到白**、**从白到黑**以及**任意**。下图绿色箭头所示为白到黑，红色箭头所示为黑到白。此处“黑”和“白”仅表示灰度值的强弱程度。 
边缘阈值		边缘阈值即梯度阈值，范围 0~255，只有边缘梯度值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。如下图所示，在阈值设为 10 时是可以识别出边缘并找出平行线，而在阈值设为 200 后，边缘无法被定位到，也就无法找到平行线。  **边缘阈值：10**  **边缘阈值：200**
滤波尺寸		描述目标边缘的清晰程度，最小值为 1，值越小，表示边缘越清晰且过滤带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间挨得太近（距离小于滤波尺寸）时反而会影响边缘位置的精度甚至丢失边缘。具体原理参考 ***圆查找*** 中的同名运行参数。

参数	描述
卡尺数量	定义卡尺的数量。边缘点由多个卡尺提取出。
剔除点数	详见 *圆查找* 中的同名运行参数。
剔除距离	允许离群点到拟合直线的最大像素距离，值越小，排除点越多，原理以及操作请参考 *圆查找* 中的同名运行参数。
投影宽度	即卡尺的宽度（下图中蓝色矩形的宽度）。在一定范围内增大该值可以获取更加稳定的边缘点。 
初始拟合	详见 *圆查找* 中的同名运行参数。
拟合方式	包括**最小二乘**、**huber** 和 **tukey** 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用**最小二乘**、**huber** 和 **tukey**。

模块结果

*矩形检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

检测状态

int 型，代表是否检测到矩形，1 表示检测到矩形，0 表示未检测到矩形。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

矩形框

表示查找到的矩形，记录了有关该矩形的中心点坐标、宽度、高度和角度。

矩形中心点

表示查找到的矩形的中心点。

矩形中心点 X

float 型，代表矩形中心点的 X 坐标。

矩形中心点 Y

float 型，代表矩形中心点的 Y 坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

矩形角度

float 型，代表找出的矩形基于水平线回正旋转的角度，顺时针为正角度，逆时针为负角度 。

轮廓点

表示用于找出矩形所使用的各个轮廓点。

轮廓点 X

float 型，代表各个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表各个轮廓点的 Y 坐标。

边缘点个数

int 型，表示用于找出矩形的轮廓点个数。

轮廓点状态

int 型，代表用于找出矩形的轮廓点状态。0 代表轮廓点不符合要求，1 代表轮廓点符合要求。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.21 中线查找

*中线查找*模块可用于查找出两条直线的中线。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的算法工作流程包括如下两大主要步骤：

1. 找出用于定位中线的两个点，包括：
   - 直线 1 起点和直线 2 起点之间的中点。
   - 直线 1 终点和直线 2 的终点之间的中点。
2. 基于上述两个中点的位置信息，计算得出直线 1 和直线 2 的中线（如下图橙线所示）。

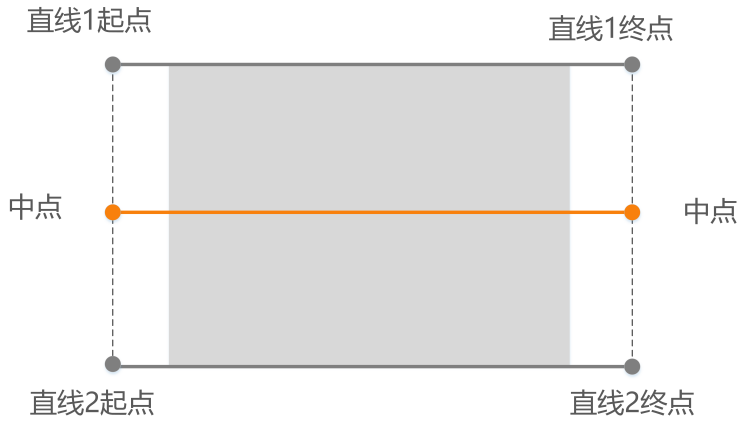


图 13-127 算法工作原理

使用方法

前后序模块

在流程中，*中线查找*的

- 前序模块一般为两个 *直线查找* 。*直线查找*负责为该模块提供直线数据输入。
- 后序模块无特殊要求，可接收并处理直线数据即可。

主要配置步骤

在流程中调用*中线查找*模块后，该模块的主要配置步骤如下：

1. 在*输入源*处下拉选择图像数据源。
2. 分别订阅线 1 和线 2 的输入源。线的输入源有 3 种，分别为**按线**、**按点**和**按坐标**。
   - **按线**：直接从前序模块的模块结果中订阅一条线。
   - **按点**：需从前序模块的模块结果中分别订阅两个点作为线的起点和终点。
   - **按坐标**：需从前序模块的模块结果中分别订阅四个坐标作为起点和终点的 X、Y 坐标。

   说明

   选择一种方式订阅数据源后，切换为其他两种方式时，模块会自动得到其他方式的对应数据源。
3. 切换到模块的*结果显示*页签，对图像显示进行设置。

应用示例

如下动图中所展示的部件上检测出两条细线即为输入的两条直线，中间稍粗的为*中线查找*输出的中线。

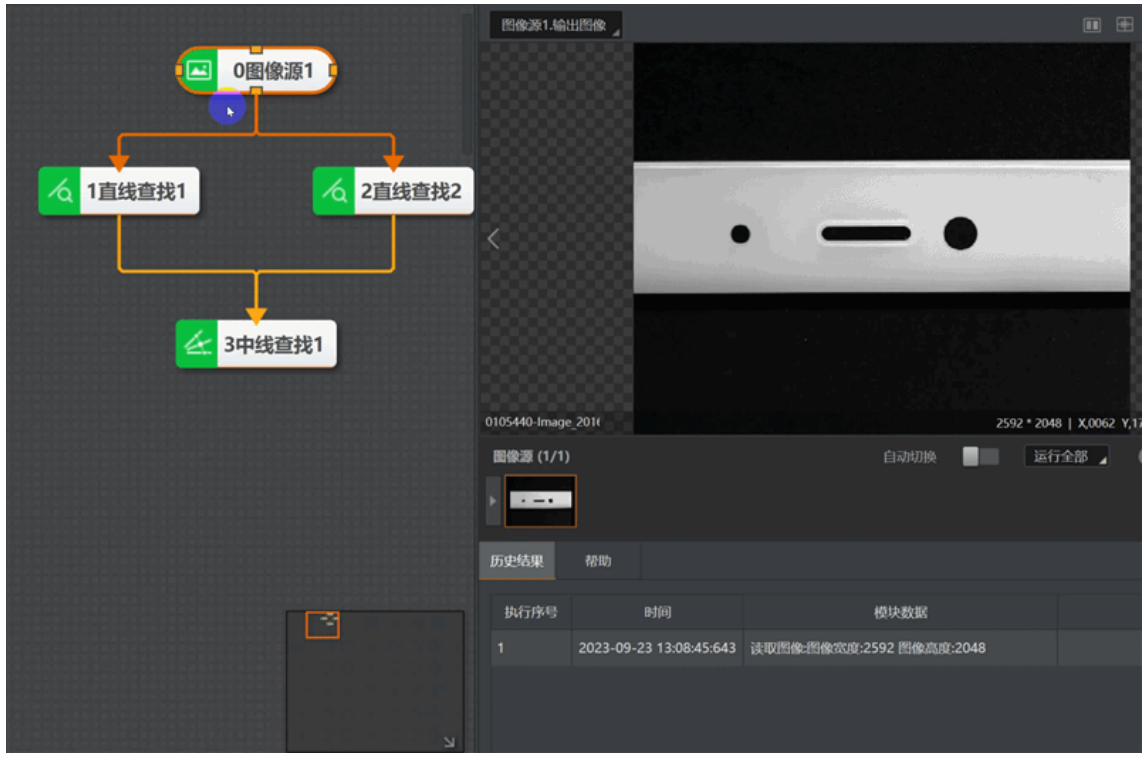


图 13-128 中线查找应用示例

参数配置

结果显示参数详情，请参见 *结果显示* 。

模块结果

*中线查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

中线

表示查找到的中线，记录了有关该中线的起点和终点坐标。

中线起点

表示查找到两条平行线之间的中线。

中线起点 X

float 型，代表中线起点的 X 坐标。

中线起点 Y

float 型，代表中线起点的 Y 坐标。

中线终点

中线终点 X

float 型，代表中线终点的 X 坐标。

中线终点 Y

float 型，代表中线终点的 Y 坐标。

直线角度

float 型，代表输出直线的角度。

测量直线*（*代表 1-2）

line 型，代表用于查找中线的输入直线，记录了有关该直线的起点和终点坐标。

测量直线*起点

point 型，代表输入直线的起点坐标。

测量直线*起点 X

float 型，代表输入直线起点的 X 坐标。

测量直线*起点 Y

float 型，代表输入直线起点的 Y 坐标。

测量直线*终点

point 型，代表输入直线的终点。

测量直线*终点 X

float 型，代表输入直线终点的 X 坐标。

测量直线*终点 Y

float 型，代表输入直线终点的 Y 坐标。

测量直线*角度（*代表 1-2）

float 型，代表输入直线相对于水平轴的旋转角度。顺时针旋转为正，逆时针旋转为负。

13.2.22 垂线查找

*垂线查找*模块可用于指定直线的垂线。

本节包含如下内容:

- *核心概念*
- *使用方法*
- *参数配置*
- *模块结果*

核心概念

该模块可查找两种垂线类型——过点垂线和中垂线。

过点垂线

起始于某个指定的点，且垂直于指定直线的线段。

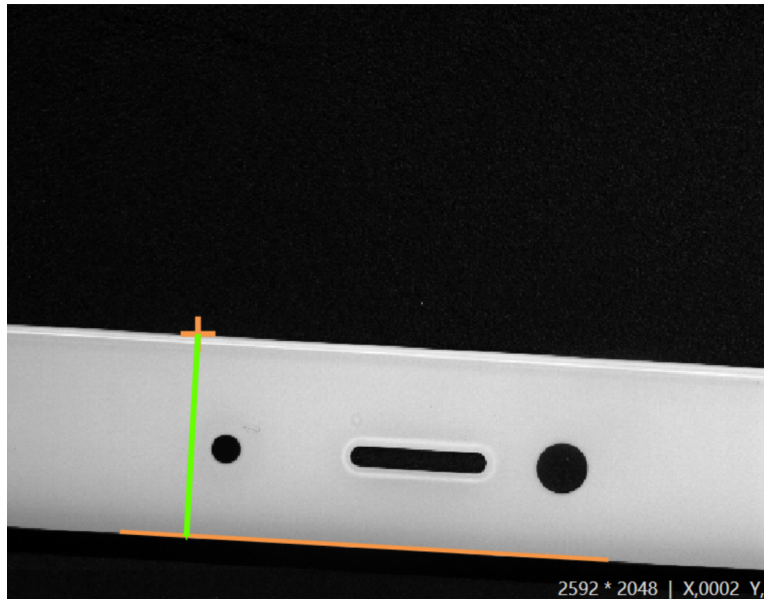


图 13-129 过点垂线

中垂线

指定直线的中垂线。

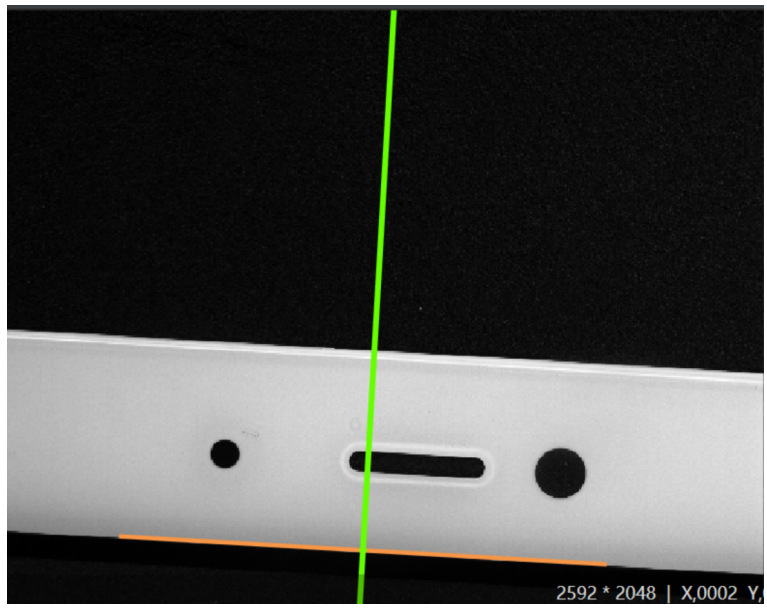


图 13-130 中垂线

使用方法

前后序模块

在流程中，*垂线查找*的

- 前序模块一般为 *直线查找* 。*直线查找*负责为*垂线查找*提供直线数据输入。
- 后序模块无特殊要求，可接收并处理垂线数据即可。

主要配置步骤

在流程中调用*垂线查找*模块后，该模块的主要配置步骤如下：

1. 在*输入源*处下拉选择图像数据源。
2. 选择垂线类型。可选**过点垂线**或**中垂线**。
3. 配置输入数据。
   - 垂线类型设置为**过点垂线**时，需分别指定点和直线的数据源。
   - 垂线类型设置为**中垂线**时，只需指定直线的数据源。
4. 切换到模块的*结果显示*页签，对图像显示进行设置。

应用示例

下图中所展示的部件上，绿色线段即为*垂线查找*输出的过点垂线。



图 13-131 垂线查找应用示例

参数配置

结果显示参数详情，请参见 *结果显示* 。

模块结果

*垂线查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量点

point 型，表示输入的测量点坐标。

测量点 X

float 型，代表输入点的 X 坐标。

测量点 Y

float 型，代表输入点的 Y 坐标。

测量直线

line 型，表示输入直线。

测量直线起点

point 型，表示输入直线的起点坐标。

测量直线起点 X

float 型，代表输入直线的起点 X 坐标。

测量直线起点 Y

float 型，代表输入直线的起点 Y 坐标。

测量直线终点

point 型，表示输入直线的终点。

测量直线终点 X

float 型，代表输入直线的终点 X 坐标。

测量直线终点 Y

float 型，代表输入直线的终点 Y 坐标。

输出直线角度

float 型，代表输出垂线与水平轴之间的夹角，且逆时针旋转为负角度，顺时针旋转为正角度。

垂足点

表示测量点到测量直线的垂足。

垂足点 X

float 型，代表垂足点的 Y 坐标。

垂足点 Y

float 型，代表垂足点的 Y 坐标。

垂线

表示过测量点作测量直线的垂线。

垂足起点

垂线起点 X

float 型，代表垂线起点的 X 坐标。

垂线起点 Y

float 型，代表垂线起点的 Y 坐标。

垂足终点

垂线终点 X

float 型，代表垂线终点的 X 轴坐标。

垂线终点 Y

float 型，代表垂线终点的 Y 轴坐标。

13.2.23 角平分线查找

*角平分线查找*模块可查找两条指定直线的夹角的角平分线。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

前后序模块

在流程中，*角平分线查找*的

- 前序模块一般为 *直线查找* 。*直线查找*负责为*角平分线查找*提供直线数据输入。
- 后序模块无特殊要求，可接收并处理直线数据即可。

主要配置

在流程中调用*角平分线查找*模块后，还需配置该模块的输入数据，包括图像输入的线输入。

应用示例

下图所示的应用示例中，两条绿线即为指定的直线，黄线即为计算得出的角平分线。

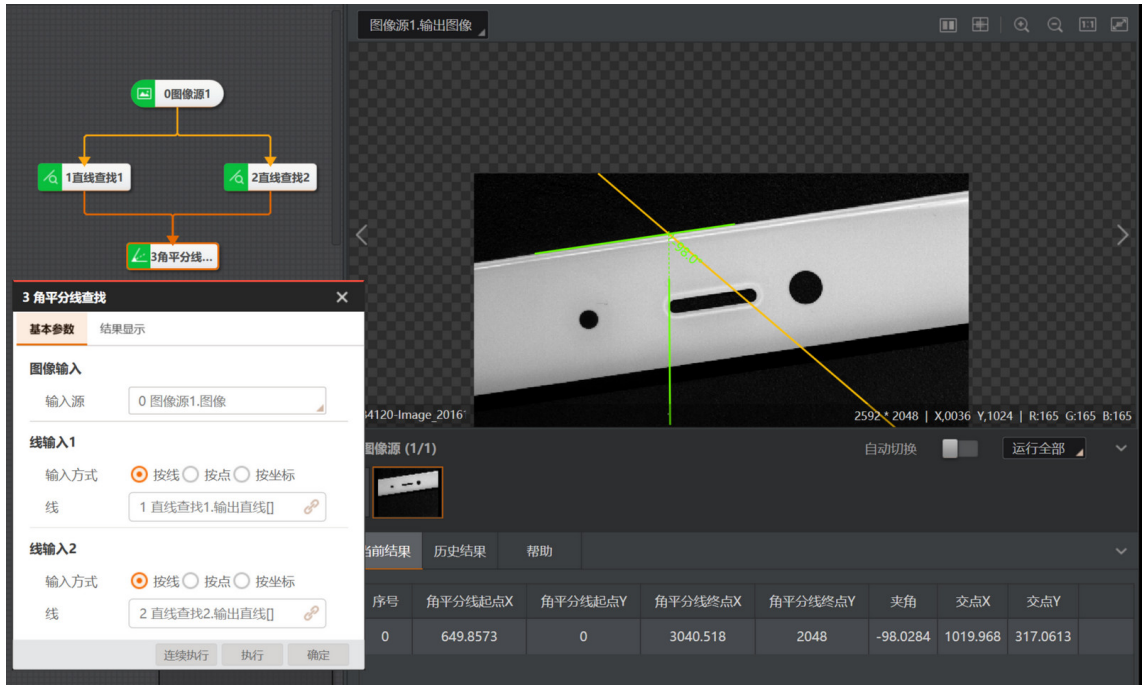


图 13-132 角平分线查找应用示例

参数配置

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模块结果

*角平分线查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量直线*（*代表 1-2）

代表用于查找角平分线的输入直线，记录了有关该直线的起点和终点坐标。

测量直线*起点

point 型，代表输入直线的起点坐标。

测量直线*起点 X

float 型，代表输入直线起点的 X 坐标。

测量直线*起点 Y

float 型，代表输入直线起点的 Y 坐标。

测量直线*终点

point 型，代表输入直线的终点。

测量直线*终点 X

float 型，代表输入直线终点的 X 坐标。

测量直线*终点 Y

float 型，代表输入直线终点的 Y 坐标。

测量直线*角度（*代表 1-2）

float 型，代表输入直线*与水平轴之间的夹角，且逆时针旋转为负角度，顺时针旋转为正角度。

交点

point 型，代表两条测量直线的交点。

交点 X

float 型，代表输入直线 1 与输入直线 2 交点的 X 坐标。

交点 Y

float 型，代表输入直线 1 与输入直线 2 交点的 Y 坐标。

夹角

float 型，代表输入直线 1 与输入直线 2 相交的锐角大小。

角平分线

角平分线起点

角平分线起点 X

float 型，代表输出的角平分线的起点的 X 坐标 。

角平分线起点 Y

float 型，代表输出的角平分线的起点的 Y 坐标 。

角平分线终点

角平分线终点 X

float 型，代表输出的角平分线的终点的 X 坐标 。

角平分线终点 Y

float 型，代表输出的角平分线的终点的 Y 坐标 。

角度标识点*（*代表 1-5）

point 型，代表角度标识点。

角度标识点*X

float 型，点 1：直线向量 1 的起点坐标 X ；点 2：直线向量 1 的终点坐标 X ；点 3：直线向量 1 和直线向量 2 的交点坐标 X ；点 4：直线向量 2 的起点坐标 X ；点 5：直线向量 2 的终点坐标 X。

角度标识点*Y

float 型，代表点 1：直线向量 1 的起点坐标 Y； 点 2：直线向量 1 的终点坐标 Y； 点 3：直线向量 1 和直线向量 2 的交点坐标 Y； 点 4：直线向量 2 的起点坐标 Y； 点 5：直线向量 2 的终点坐标 Y。

13.2.24 卡尺工具

*卡尺工具*主要用于测量物体的宽度、边缘的特征的位置以及图像中边缘对的位置和间距。通过该模块可在图像 ROI 内进行快速且精确的检测和定位。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

工作流程概述

该模块的算法工作流程可概括为：将图像区域内数据沿一个方向投影之后得到一维数据，对该数据进行滤波处理，选择局部极值，并对这些极值按照预设的方式计算得分，最终输出符合得分要求的亚像素极值点。

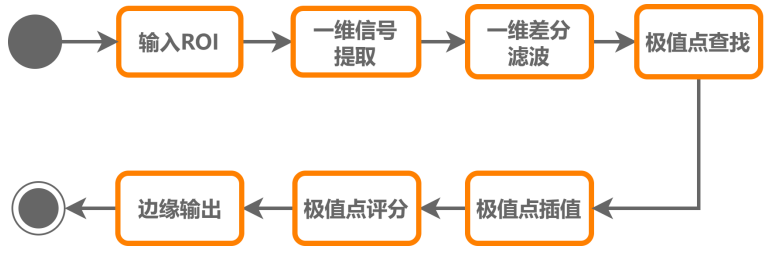


图 13-133 算法工作流程

工作流程详解

上图算法工作流程中各步骤的详情如下。

步骤 1：输入 ROI

从前序模块获取 ROI 输入。

步骤 2：投影（即一维信号提取）

在图像中选取感兴趣区域，即投影区域。该区域的处理相当于沿着某特定方向上的均值滤波处理。一般情况下，有角度的矩形投影区域不与像素网格对齐，卡尺工具通过插值的方法获取采样区域。投影区域的形状目前支持任意角度的矩形。卡尺工具放置的投影区域可能会超出图像的边界。超出图像边界的区域，则填充其他灰度值。

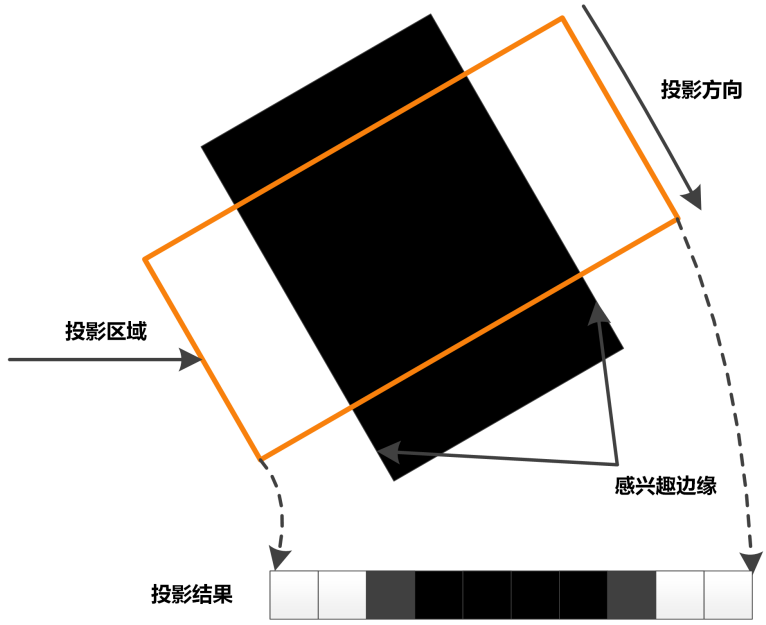


图 13-134 一维信号提取

步骤 3：一维差分滤波

一维投影获取到的 ROI 的一维信息，既包含 ROI 边缘，也包含其他的边缘和噪声。使用差分滤波可以对一维图像起到增强 ROI 边缘和抑制噪声的作用。

如下两张示意图为不同滤波核大小的滤波结果对比。一般情况下，滤波核越大，抑制噪声的能力越强，但极值点位置的精度可能会越差。

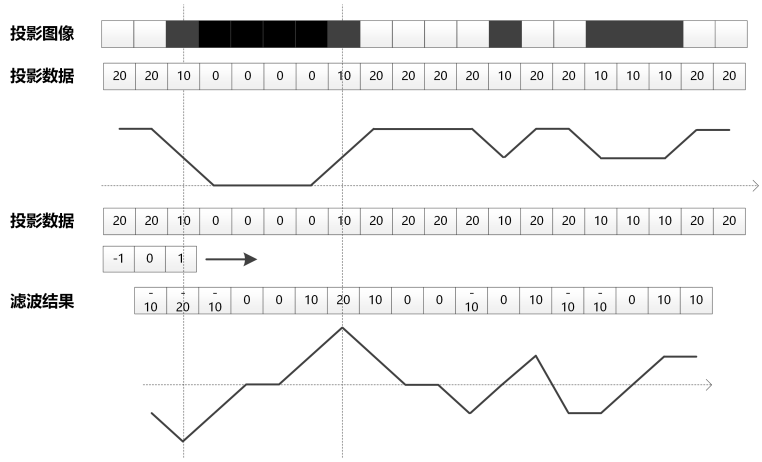


图 13-135 一维差分滤波（滤波核尺寸=3）

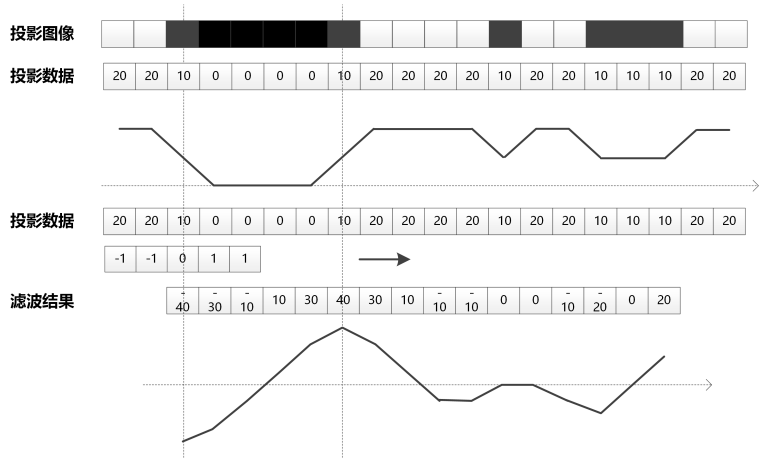


图 13-136 一维差分滤波（滤波核尺寸=5）

步骤 4：极值点查找

算法根据边缘极性类型的不同，选取合适的极值点查找目标，如下表所示。

边缘极性类型	极值点查找目标
白到黑	极小值（谷值）
黑到白	极大值（峰值）
任意极性	极大值（峰值）、极小值（谷值）

如下图所示，绿色箭头所示的边缘极性为“白到黑”，红色箭头所示为“黑到白”。此处的“黑”和“白”是表示灰度值高低的相对概念。经过滤波后，边缘极性将转换成另一种展现形式，即极小值（谷值）和极大值（峰值），分别如上文的示意图*一维差分滤波（滤波核尺寸=3）*和*一维差分滤波（滤波核尺寸=5）*所示。



图 13-137 边缘极性

步骤 5：极值点插值

对提取极值点进行插值，得到亚像素精度的边缘位置。

步骤 6：极值点评分

评分方法类型按评分准则分为**位置评分**、**间距评分**和**对比度评分**三大类。

位置评分

位置评分是根据候选边缘与卡尺工具指定的投影区域中心的相对位置进行评分。若期望 ROI 边缘相对投影区域中心为特定距离，可指定一个绝对位置评分系统，并根据像素绝对距离反馈分数。在边缘对查找模式中，若希望观察候选边缘与投影区域中心间距离相对边缘对宽度大小的变化，可以定义一个相对位置评分系统。在该种评分系统中，原始分数将被归一化，1.0 评分则代表边缘对距离与边缘对宽度大小一致。

- 绝对位置：可通过下文*参数配置*中提及的**位置**配置。按***边缘或边缘对***的中心点相对投影区域中心点的绝对位置差（投影区域中心点左右两边都是值都是正数）评分。
- 相对位置：可通过下文*参数配置*中提及的**相对位置**配置。按***边缘或边缘对***的中心点相对投影区域中心点的位置差（投影区域中心点左右两边一边是负数一边是正数）评分。
- 归一化位置：可通过下文*参数配置*中提及的**归一化位置**配置。按***边缘对***的中心点相对投影区域中心点的绝对归一化距离评分，归一化分母为边线对宽度。
- 归一化相对位置：可通过下文*参数配置*中提及的**归一化相对位置**配置。按**边缘对**的中心点相对投影区域中心点的归一化位置差（可正可负）评分，归一化分母为边缘对宽度。

间距评分

在边缘对查找模式中，边缘对间距评分可以反映边缘对的真实距离（实际测量宽度）与边缘对宽度（外部输入宽度）之间的差异程度。

间距差：可通过下文*参数配置*中提及的**间距差**配置。按“(边缘对距离-边缘对宽度)/边缘对宽度”，单边计分方式，理想情况下该值接近 0。

- 相对间距差：可通过下文*参数配置*中提及的**相对间距差**配置。按(边缘对距离-边缘对宽度)/边缘对宽度，双边计分方式，理想情况下该值接近 0，与间距差不同在于可以计分曲线的不同，而且考虑正负值。
- 间距：可通过下文*参数配置*中提及的**间距**配置。按边对距离/边缘对宽度，理想情况下接近 1。

对比度评分

可对候选边缘进行对比度评分。此处的边缘对比度反映边缘处像素值变化的剧烈程度。最大得分为 1.0，相当于边缘对比度 255（边缘对比度的最大值），如果是边缘对查找模式，则使用两个边缘的平均对比度作为评分因子。

- 对比度：可通过下文*参数配置*中提及的**对比度**配置。可以简单理解为明暗区域边界中明区域与暗区域像素的绝对值差，按边缘对比度或边缘对对比度均值计分。
- 灰度：可通过下文*参数配置*中提及的**灰度**配置。可以简单理解为区域边缘的像素值，但该像素值可能落在亮区域也可能落在暗区域，按边缘灰度或边缘对灰度均值计分。

除了“相对间距差”是**双边计分函数**以外，其他计分方式均为**单边计分函数**。该种函数具有两种形式，如下图所示。



图 13-138 单边计分函数（递增函数）

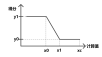


图 13-139 单边计分函数（递减函数）

递增函数指曲线前半段平，后半段上升。递减函数指曲线前半段下降，后半段平，可通过设置水平坐标 x0、x1、xc 来修改函数曲线的走势分布。

双边计分函数可以认为是左右两个单边函数的合并，所以共有四种组合方式，即：

- “递增”函数+“递减”函数
- “递增”函数+“递增”函数
- “递减”函数+“递减”函数
- “递减”函数+“递增”函数

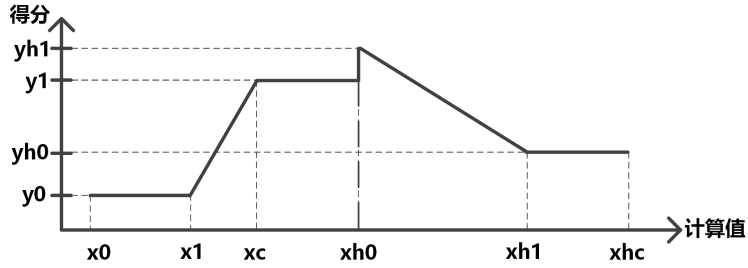


图 13-140 递增函数+递减函数

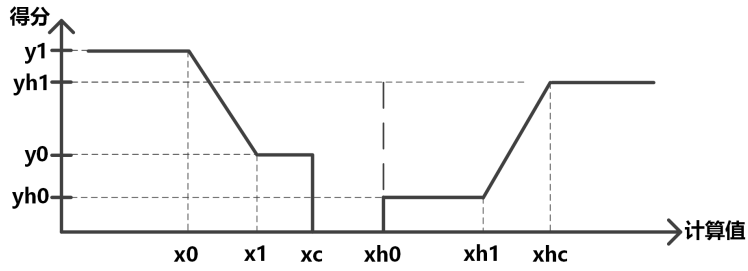


图 13-141 递减函数+递增函数

说明

所有的计分方式既可以单独使用，也可以联合使用。当使用多种计分方式时，需要对各个计分方式进行合并，从而得出总计分结果，总计分的计算公式如下所示：

$$OverallScore = \sqrt[N]{Score_1 \times Score_2 \times \cdots \times Score_N}$$

步骤 7：边缘输出

输出符合要求的边缘轮廓。

使用方法

在流程中，*卡尺工具*主要与 *图像源*、*轮廓匹配*、*位置修正* 组合调用，以便在图像指定区域内精确定位边缘点或边缘点对。通过*模板匹配*和*位置修正*做粗定位，*卡尺工具*做精定位，可实现对指定边缘点或边缘点对的高精度检测。

说明

下图中 ROI（蓝色矩形框）上的箭头表示边缘查找方向。

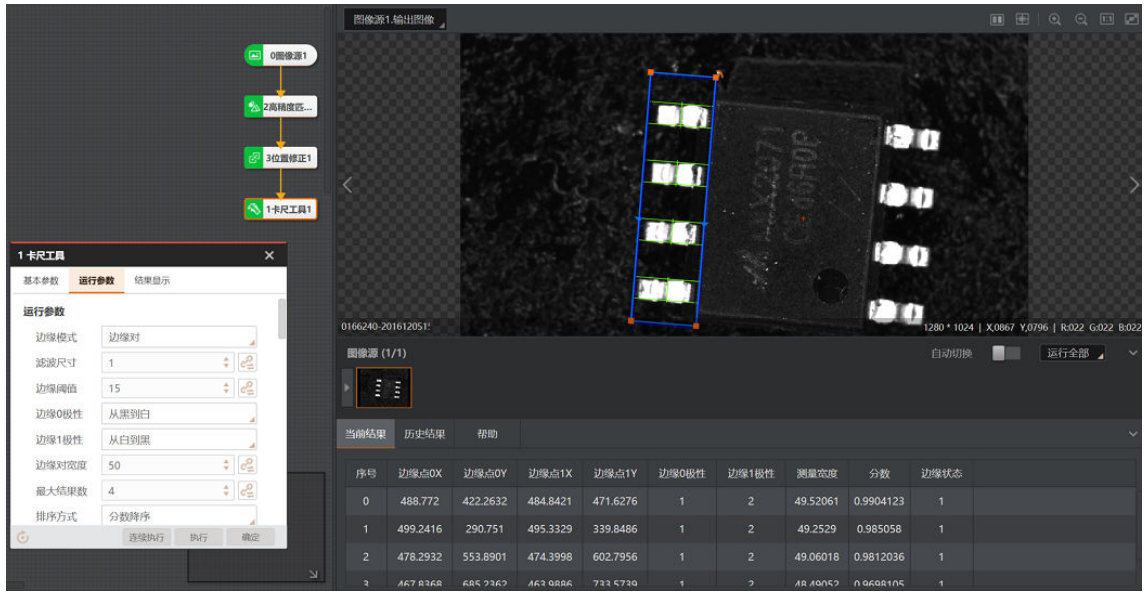


图 13-142 示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

边缘模式

设置检测模式，可选**单边缘**或**边缘对**。

单边缘

检测指定区域内的边缘位置，可用于定位、计数和判断有无等。

边缘对

检测指定区域内的边缘间距 。典型应用为在工件尺寸测量场景中获取特定边缘对间的距离。

滤波尺寸

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使检测结果更加稳定。但如果边缘与边缘之间挨得太近，会影响边缘位置的精度甚至丢失边缘。

边缘阈值

即梯度阈值，取值范围 0~255。设置后，只有边缘梯度阈值大于该值的边缘点才能被检测到。数值越大，抗噪声能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

边缘 0/1 极性

极性表示边缘的灰度值变换趋势，可选**从黑到白**、**从白到黑**或**任意极性**。

说明

边缘模式设置为**边缘对**时，才有**边缘 1 极性**参数。

边缘对宽度

期望输出的边缘对的像素间距尺寸。单独调节该参数无法直接筛选出期望的边缘对，仅当位置归一化计分、相对位置归一化计分、间距计分、间距差计分、相对间距差计分中一个或多个计分方式开启时调节该参数有意义，且作为计分因子的缩放因子使用。

说明

边缘模式设置为**边缘对**时，才有**边缘 1 极性**参数。

最大结果数

期望输出的边缘对最大数量，若实际查找到的对数大于该参数，则按照分数由高到低输出该参数数量的边缘对，否则输出实际边缘对数。

排序方式

设置最终输出的边缘点在*当前结果*页签的显示排序，分为**分数升序**、**分数降序**、**方向正向**、**方向逆向**四种类型。

对比度/边缘对对比度

边缘模式设置为**单边缘**时，该参数为**对比度**；**边缘模式**设置为**边缘对**时，该参数为**边缘对对比度**。

启用后，按“边缘对比度”或“边缘对对比度”均值计分。计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

灰度

启用后，按边缘灰度或边缘对灰度均值计分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

位置

启用后，按边缘或边缘对的中心点相对投影区域中心点的绝对位置之差评分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

相对位置

启用后，按边缘或边缘对的中心点相对投影区域中心点的位置差（可正可负）评分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

归一位置

边缘模式设置为**边缘对**时，才有该参数。启用后，按边缘对的中心点相对投影区域中心点的绝对归一化距离评分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

归一相对位置

边缘模式设置为**边缘对**时，才有该参数。启用后，按边缘对的中心点相对投影区域中心点的归一化位置差（可正可负）评分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

间距

边缘模式设置为**边缘对**时，才有该参数。启用后，按按边对距离/边缘对宽度，单边计分方式评分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

间距差

边缘模式设置为**边缘对**时，才有该参数。启用后，按|(边缘对距离-边缘对宽度)/边缘对宽度|，单边计分方式评分。

计分方式均为上文*模块原理*中提及的单边计分函数。函数形式为递增或递减可通过**曲线类型**设置；**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-138* / *图 13-139* 中的 x0、x1、xc 和 y0 ~ y1。

相对间距差

边缘模式设置为**边缘对**时，才有该参数。启用后，按(边缘对距离-边缘对宽度)/边缘对宽度，双边计分方式评分。

计分方式均为上文*模块原理*中提及的双边计分函数。**左/右曲线**的**起点**、**X 中点**、**终点**和 **Y 计分**分别对应上文 *图 13-140* / *图 13-141* 中的 x0/xh0、x1/xh1、xc/xhc 和 y0 ~ yh0/y1~yh1。

投影方向

可设置卡尺的投影方向，可选如下方向。

- 纵向投影或径向投影：
  针对矩形 ROI，投向矩形宽（纵向投影）；针对扇环 ROI，投向扇环形圆弧（径向投影），边缘点按照圆环方向排列，即边缘点都在一个圆上。
- 横向投影或环向投影：
  针对矩形 ROI，投向矩形高（横向投影）；针对扇环 ROI，投向扇环形半径（环向投影），边缘点按照半径方向排列，即边缘点都在一个半径上。

模糊边缘

开启后，可增强候选边缘点集的提取能力，获取更多候选点集数，从而可在“干扰点较多的图像”或“边缘模糊的图像”场景中更大可能地提取到目标边缘点，但耗时增加明显。

模块结果

*卡尺工具*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

边缘状态

int 型，代表图像上某个点是否位于边缘。1 代表位于边缘，0 代表不在边缘。

边缘个数

int 型，代表查找到的边缘个数。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

边缘*极性（*代表 0-1）

int 型，代表边缘的灰度变化，1 表示从黑到白，2 表示从白到黑，3 表 示任意极性。

边缘*位置（*代表 0-1）

float 型，代表边缘点所在的位置。

边缘点*（*代表 0-1）

边缘点*X

float 型，代表检测到的第（*+1）个边缘点的 X 坐标。

边缘点*Y

float 型，代表检测到的第（*+1）个边缘点的 Y 坐标。

测量宽度

float 型，代表定位到的两个边缘点，在与投影法线方向的宽度 。

直线*起点（*代表 0-1）

直线*起点 X

float 型，代表边缘直线*起点的 X 坐标。

直线*起点 Y

float 型，代表边缘直线*起点的 Y 坐标。

直线*终点（*代表 0-1）

直线*终点 X

float 型，代表边缘直线*终点的 X 坐标。

直线*终点 Y

float 型，代表边缘直线*终点的 Y 坐标。

直线*角度（*代表 0-1）

float 型，代表边缘直线相对于水平线方向的旋转角度。顺时针旋转，则为正角度；逆时针旋转，则为负角度。

圆弧*（*代表 0-1）

圆弧*中心点

圆弧*中心点 X

float 型，代表检测到圆弧中心点的 X 坐标。

圆弧*中心点 Y

float 型，代表检测到圆弧中心点的 Y 坐标。

圆弧*半径

float 型，代表检测到圆弧的半径。

圆弧*起始角度

float 型，代表检测到圆弧的起始角度。

圆弧*角度范围

float 型，代表检测到圆弧的角度范围。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.2.25 边缘查找

*边缘查找*模块可检测指定区域内的边缘点位置，并输出边缘信息，如边缘直线起点和终点坐标。该模块适用于边缘点位置定位、边缘直线间距测量等业务场景。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

边缘为两个灰度值不同的像素区域之间的边界，而边缘像素点则为边缘经过的像素点。

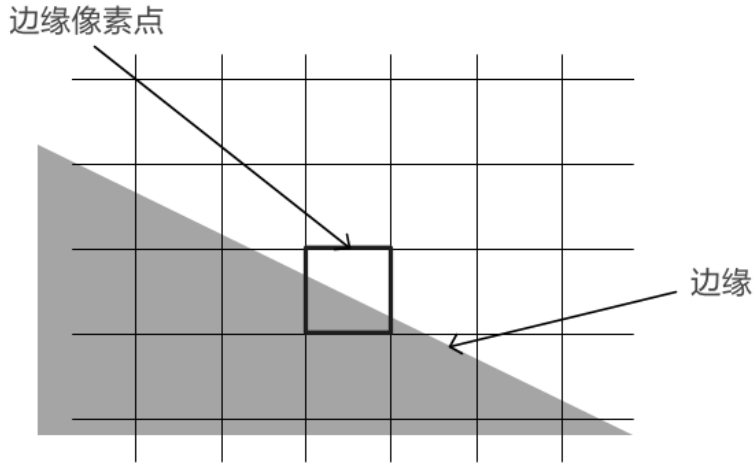


图 13-143 边缘像素点

*边缘查找*的算法工作原理基于*卡尺工具*的单边缘模式，但参数没有*卡尺工具*详尽，且固定了评分方法。因此*边缘查找*在一定程度上减弱了*卡尺工具*的灵活性。
*卡尺工具*算法工作原理详情请参见 *卡尺工具*

使用方法

在流程中：

- *图像源*、*模板匹配*、*位置修正*等模块可作为*边缘查找*的前序模块，为*边缘查找*在图像指定区域精确定位边缘点。*模板匹配*和*位置修正*做粗定位，*边缘查找*做精定位，实现对指定边缘点的高精度检测。
- *几何创建*、*线线测量*和*点线测量*等模块可作为*边缘查找*的后序模块，获取*边缘查找*输出的边缘信息，实现目标测量。

下图展示了检测部件上两线间距的应用示例。

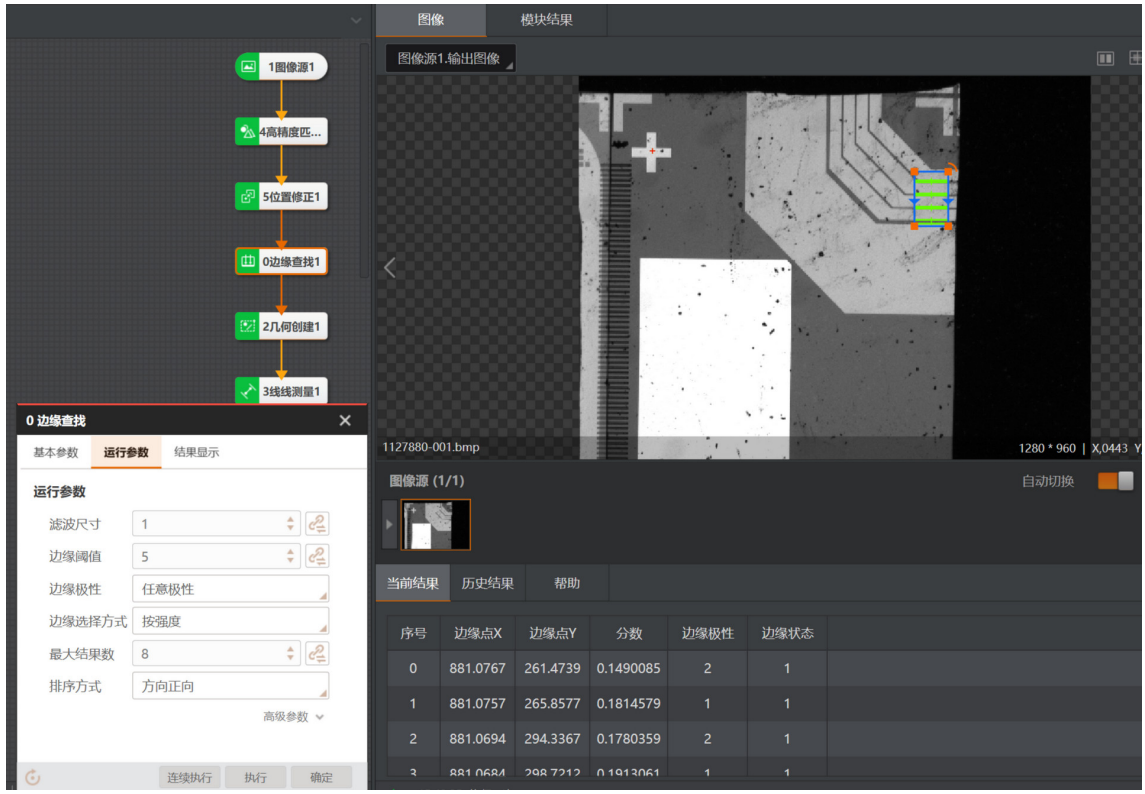


图 13-144 边缘查找模块输出边缘信息

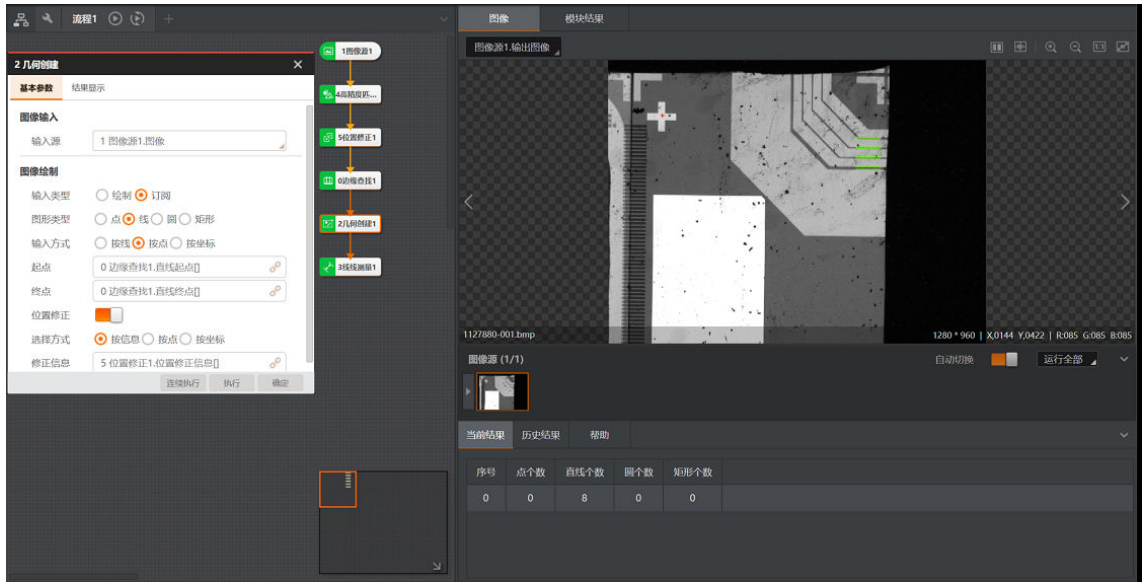


图 13-145 几何创建模块生成直线

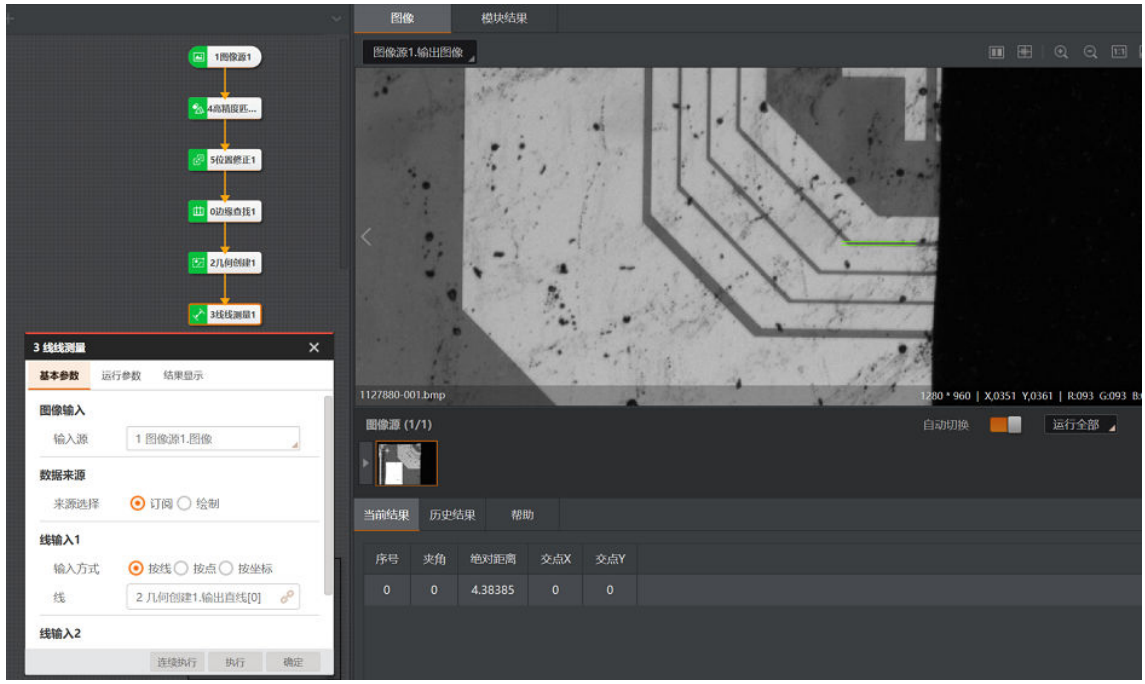


图 13-146 线线测量模块输出两线间距

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

下表仅介绍该模块的部分运行参数详情，其他运行参数与 *卡尺工具* 单边缘模式下的运行参数一致，详情请参见 *卡尺工具* 。

表 13-13 运行参数

参数	描述
排序方式	查找到的边缘在结果显示区的排列顺序，包括**分数降/升序**，即按照边缘的分数降/升序排列；**方向正/逆向**，即按方向（查找方向）正/逆向排列。 **查找方向**：包括**从上到下**和**从左到右**。

参数		描述
		• **从上到下**表示相对于搜索 ROI 按从上到下顺序查找边缘点。若搜索 ROI 旋转 180°则改为从下到上顺序查找边缘点。 • **从左到右**表示相对于搜索 ROI 按从左到右顺序查找边缘点，若搜索 ROI 旋转 180°则改为从右到左顺序查找边缘点。
最大结果数		期望输出边缘最大数量，若实际查找到的个数大于该参数，则按照上述排序方式输出该参数数量的边缘。否则输出实际查找到的所有边缘。该参数仅当边缘类型为所有边缘时生效。
边缘类型	最强边缘	只检测 ROI 内梯度阈值最大的边缘，具体可参见 *圆查找* 模块。
	第一条边缘	只检测 ROI 内与搜索起始点最近的边缘。
	最后一条边缘	只检测 ROI 内与搜索终止点最近的边缘。
	所有边缘	检测 ROI 内所有边缘。

模块结果

*边缘查找*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

边缘状态

int 型，代表图像上某个点是否位于边缘。1 代表位于边缘，0 代表不在边缘。

边缘个数

int 型，代表查找到的边缘个数。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

边缘极性

int 型，代表目标对象边缘的亮度变化方向或极性。

边缘位置

float 型，代表边缘点所在位置。

边缘点

point 型，代表检测到的边缘点。

边缘点 X

float 型，代表检测到边缘点的 X 坐标。

边缘点 Y

float 型，代表检测到边缘点的 Y 坐标。

直线起点

point 型，代表输出直线的起点坐标。

直线起点 X

float 型，代表输出直线起点的 X 坐标。

直线起点 Y

float 型，代表输出直线起点的 Y 坐标。

直线终点

point 型，代表输出直线的终点坐标。

直线终点 X

float 型，代表输出直线终点的 X 坐标。

直线终点 Y

float 型，代表输出直线终点的 Y 坐标。

直线角度

float 型，代表输出直线的角度。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.2.26 顶点检测

*顶点检测*可检测图像指定区域内的顶点，并输出顶点坐标等信息。该模块常用于检测目标物体的顶点。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

*顶点检测*在流程中的前后序模块如下表所示。

表 13-14 前后序模块

前后序模块	描述
前序模块	*图像源*、*模板匹配*、*位置修正*等模块可作为*顶点检测*的前序模块，为*顶点检测*在图像指定区域精确定位边缘点。*模板匹配*和*位置修正*做粗定位，*顶点检测*做精定位，实现对指定顶点的高精度检测。
后序模块	*顶点检测*对后序模块无特殊要求。可接收并处理顶点信息的模块均可作为*顶点检测*的后续模块。

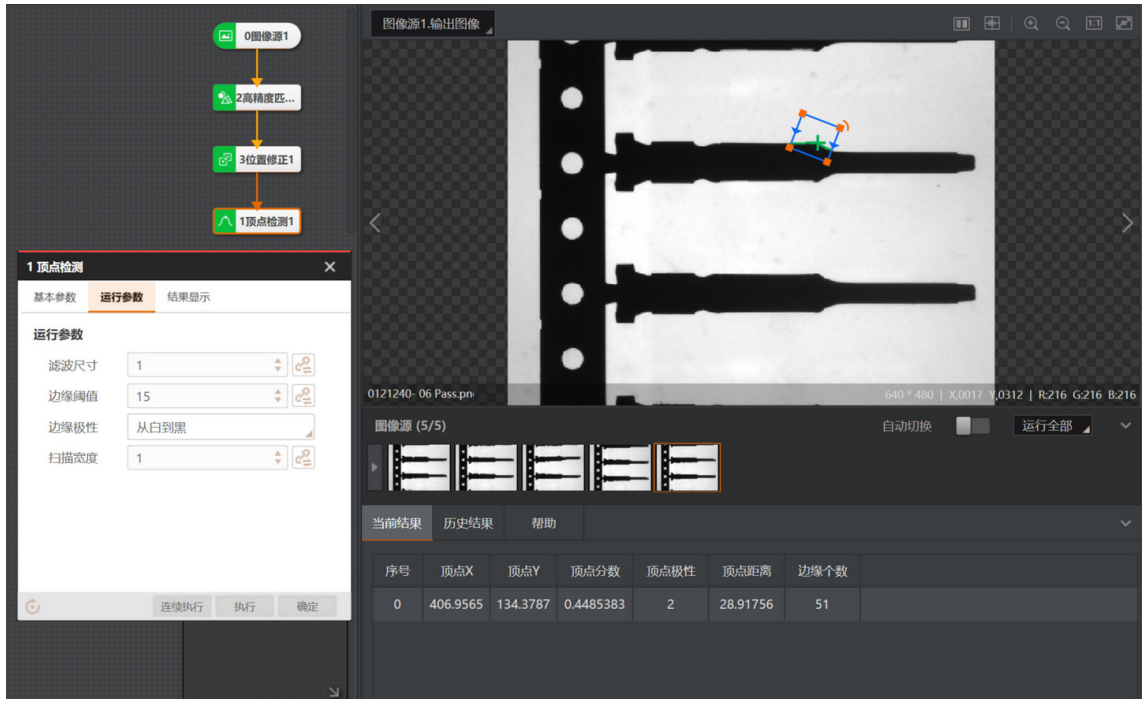


图 13-147 顶点检测应用示例

参数配置

以下仅介绍该模块特有的运行参数（**扫描宽度**）详情，其他运行参数请参见请参见 *卡尺工具* 的同名运行参数。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

扫描宽度

在 ROI 中用于查找边缘点的卡尺的宽度，最小值为 1。在一定范围内增大该值，边缘点数目减少。

模块结果

*顶点检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

顶点分数

float 型，代表顶点检测中边缘点、顶点及其分数均来自内部卡尺计算， 分数可参考卡尺工具中提及的极值点评分。

顶点极性

int 型，代表顶点边缘的灰度变化，1 代表从黑到白，2 代表从白到黑， 3 代表任意极性 。

顶点距离

float 型，代表顶点到 ROI 区域某一边的距离，跟运行参数中的查找方向相对应。

顶点

顶点 X

float 型，代表顶点的 X 坐标。

顶点 Y

float 型，代表顶点的 Y 坐标。

边缘状态

int 型，代表图像上某个点是否位于边缘。1 代表位于边缘，0 代表不在边缘。

边缘个数

int 型，代表查找到的边缘个数。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

边缘极性

int 型，代表目标对象边缘的亮度变化方向或极性。

边缘位置

float 型，代表边缘点所在位置。

边缘点

point 型，代表检测到的边缘点。

边缘点 X

float 型，代表检测到边缘点的 X 坐标。

边缘点 Y

float 型，代表检测到边缘点的 Y 坐标。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.2.27 路径提取

通过*路径提取*模块，可在绘制的路径上等间隔取点或查找边缘点。该模块主要用于点胶行业的胶路检测等场景。

本节内容包含：

- *使用方法*
- *参数配置*
- *模型配置*
- *模块结果*

使用方法

前后序模块

流程中*路径提取*的前后序模块详情如下表所示。

前后序模块	描述
前序模块	一般为 *图像源* 、 *模板匹配* 、 *位置修正* 等模块。前序模块用于路径的粗定位，*路径提取*进行精确的路径点位提取。 **说明** 前序模块包含*位置修正*时，创建位置修正基准点时必须在一张图像上进行。需先运行一次流程，双击*位置修正*模块手动点击创建基准点。
后序模块	后序模块无特殊要求，可接收并处理路径点数据即可。

主要配置步骤

在流程中调用该模块后，该模块的主要配置步骤如下：

1. 配置该模块的基本参数，详情参见 *基本参数* 。
2. 配置该模块的 *运行参数* 。
3. *配置模板* 。

应用示例

下图所示的应用示例中，绿点为与模板匹配的路径点，红点处为缺失的路径点。

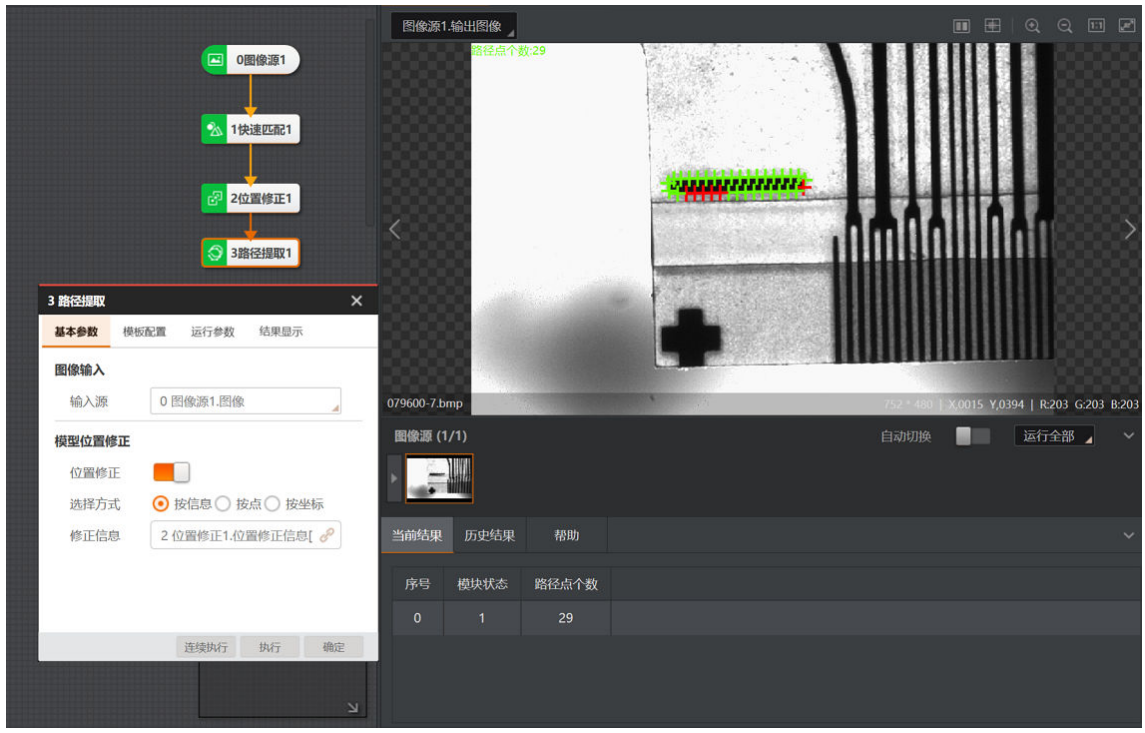


图 13-148 路径提取应用示例

模型配置

请参照以下步骤配置路径模型。

操作步骤

1. 选择*模型配置*页签，并单击*新建模型*。

   如果 *图像源* 模块已正常运行，则该窗口默认显示当前*图像源*模块采集的图像。如需基于其他图像进行模型配置，可单击*选择其他图像*。

2. 绘制路径。

   绘制轨迹路径　单击 ，并在图像上绘制轨迹路径。单击绘制轨迹的节点，拖动光标绘制线段，双击结束绘制。

   绘制扇圆路径　单击 ，并在图像上拖动光标绘制扇圆形路径。

3. **可选操作**: 进行如下可选操作。

   删除路径　右键单击路径，并在弹出的右键菜单上单击*删除*。

插入点	右键单击路径上某个位置，并在弹出的右键菜单上单击*插入点*。
删除点	右键单击路径上某个点，并在弹出的右键菜单上单击*删除点*。
闭合轨迹	右键单击路径，并在弹出的右键菜单上单击*闭合轨迹*，将未闭合的路径轨迹闭合。
调整路径位置	单击选中路径，并拖拽调整其位置。
清空基准	点击*清空基准*可重新设置用于位置修正的基准点。
移动图像	单击 后，将光标移动至图像上并拖拽图像进行移动。
缩放图像	单击 / 缩放图像；或将光标置于图像区域，并滚动鼠标滚轴缩放图像。
清空路径	单击 清空掩膜。
撤销上一步操作	单击 撤销上一步操作。
恢复至撤销前	单击 取消撤销，恢复至撤销前的状态。
图像自适应	单击 将图像大小设置为自适应模式。
全屏显示	单击 将图像全屏显示。

4. 单击 生成模型。

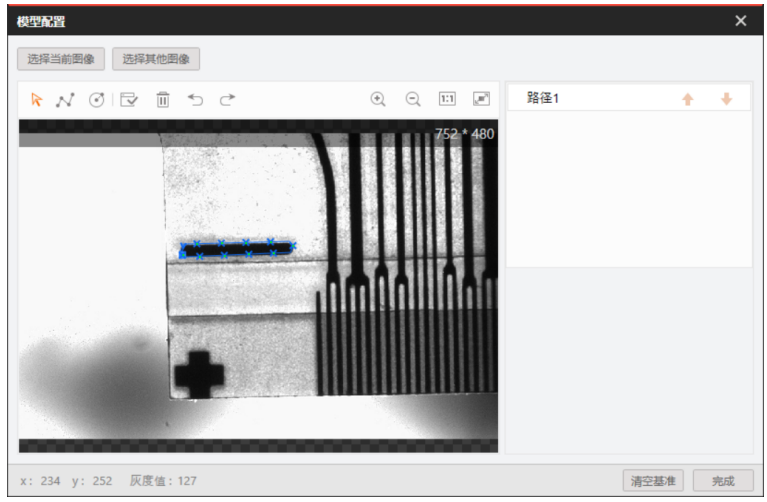


图 13-149 模型效果示例

5. **可选操作**: 在路径列表选择路径，配置训练参数，并重新生成模型，直至模型效果满足需求。

   具体可配置的训练参数取决于 *运行参数* 中**提取方式**的配置。

- **提取方式**配置为**查找边缘点**时，可配置如下训练参数：
  边缘类型

  详情参见 *圆查找* 模块的同名参数。

  边缘极性

  详情参见 *圆查找* 模块的同名参数。

  边缘阈值

  详情参见 *圆查找* 模块的同名参数。

  卡尺宽度/高度

  详情参见 *圆查找* 模块的同名参数。

  滤波尺寸

  详情参见 *圆查找* 模块的同名参数。

  路径点数量

  调整模型中提取路径上的路径点数量，取值范围 2~300。

  位置偏移量

  调整路径点的偏移量，取值范围-1000~1000。
- **提取方式**配置为**等间隔取点**时，可配置如下训练参数：
  路径点数量

  调整模型中提取路径上的路径点数量，取值范围 2~300。

  位置偏移量

  调整路径点的偏移量，取值范围-1000~1000。

6. 单击*模型配置*窗口右下角的*确定*，完成模型配置，并返回*模型配置*页签。
7. **可选操作**: 在*模型配置*页签进行如下可选操作。

   编辑模型　单击*编辑模型*继续编辑模型。

   清空模型　单击 清空模型。

   载入模型　单击 从本地路径载入模型。

   导出模型　单击 将模型导出至本地路径。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

提取方式

设置边缘点的提取方式，包括**等间隔取点**和**查找边缘点**。

等间隔取点

在运行结果中等间距取得模板中对应个数的胶点，检测参数默认不可更改。选择该提取方式时，可启用/关闭**输出圆弧信息**。

输出圆弧信息

开启后，若有圆弧轨迹，该模块会在结果中输出圆弧的圆心和角度。

查找边缘点

运行结果展示为模板附近的边缘点，可自行更改检测参数。

模块结果

*路径提取*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

路径点个数

int 型，代表检测区域内提取到的路径点个数 。

路径*（*代表 1-10）

路径*X

float 型，代表路径点*的 X 坐标。

路径*Y

float 型，代表路径点*的 Y 坐标。

路径*点状态（*代表 1-10）

int 型，代表第*条路径各个路径点的状态，1 代表匹配成功，0 代表匹配失败。

总路径

路径点 X

float 型，代表全部路径的路径点的 X 坐标集。

路径点 Y

float 型，代表全部路径的路径点的 Y 坐标集。

路径点状态

int 型，代表路径各个路径点的状态，1 代表匹配成功，0 代表匹配失败。

路径点类型

int 型，1 代表直线路径点，0 代表圆弧路径点。

13.2.28 目标跟踪

*目标跟踪*模块可用于对目标物体进行跟踪和计数。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块主要利用目标的位置和速度信息，通过数学模型和算法预测目标的运动轨迹，从而实现目标的跟踪和计数。

该模块基于算法对以下五个元素的处理和计算实现。示例场景如下图所示。

- ROI：在相机采集图像中预设的感兴趣区域（Region of Interest）。
- 计数线：一般为 ROI 在跟踪方向（即目标移动方向）上的中线，即上图和下图中的橙色线。
- 有效计数目标：进入 ROI 且越过计数线的目标物体。出现在图像中，但不符合上述条件的物体，不产生计数值累加。
- 跟踪框：算法在图像上生成的可跟踪目标物体的矩形框。
- X/Y 方向速度：跟踪框对于目标物体的跟踪速度。根据目标物体在图像上的移动方向（从左至右或从上至下），跟踪速度分为横向跟踪速度和纵向跟踪速度两种，可分别通过下文*参数配置*中提及的 **X 方向速度**和 **Y 方向速度**配置。

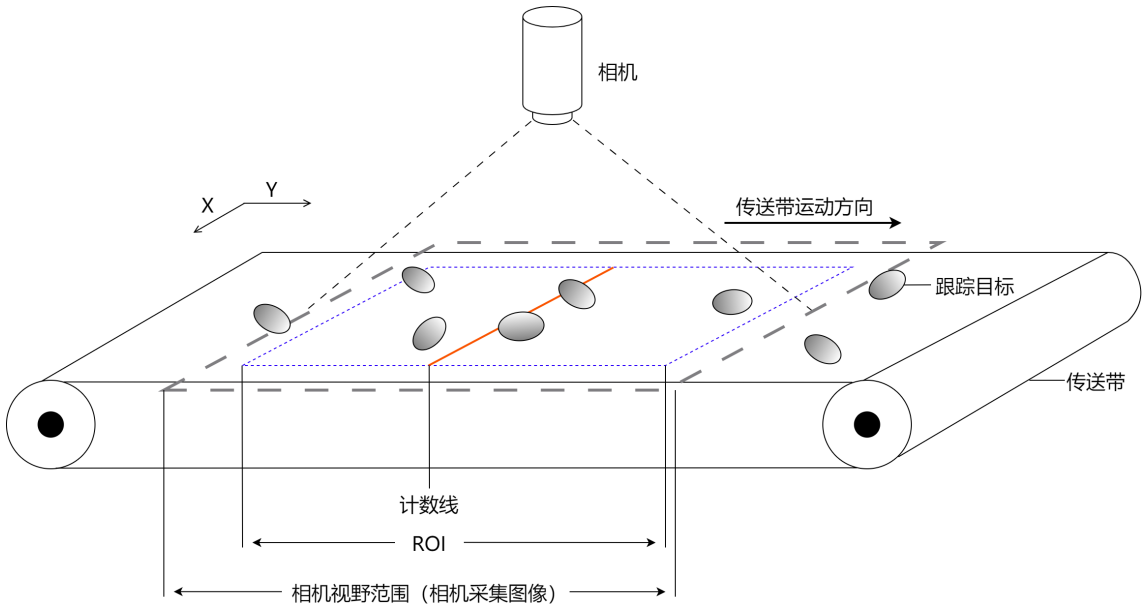


图 13-150 场景示意图

该模块的算法原理为：首先在 ROI 区域发现目标时，获取目标的初始状态信息，包括目标的位置、速度和大小等参数。算法通过获取到的初始状态信息，结合 X/Y 方向速度建立数学模型，预测目标的轨迹，从而实现对目标的跟踪。

说明

- 跟踪过程中，算法会不断更新目标的状态想你想，并根据新的状态信息进行轨迹预测。并将当前的目标检测结果与之前被跟踪的目标关联，确定是否为同一目标。
- 当目标离开视野范围或被其他目标遮挡时，算法将重新寻找目标并获取新的初始状态信息，从而实现目标的连续跟踪和计数。

下图为该模块的使用效果示意。

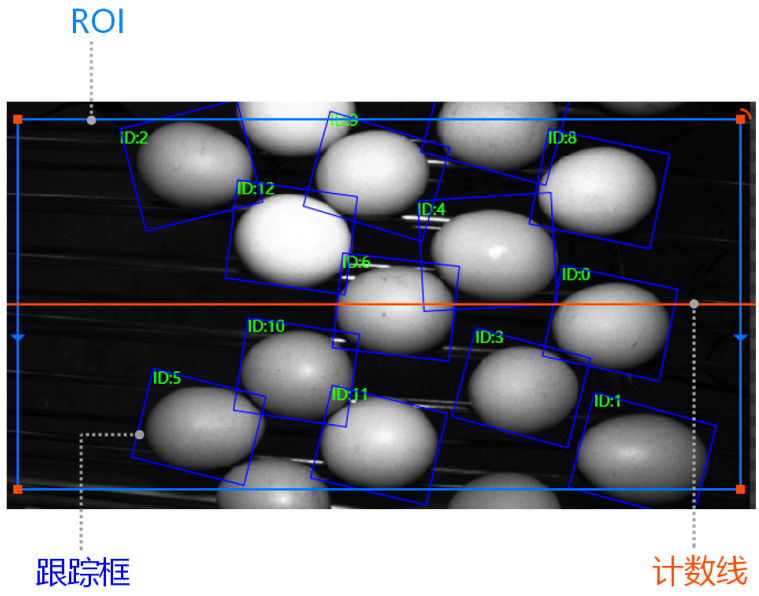


图 13-151 目标跟踪效果图

使用方法

前后序模块

在流程中，*目标跟踪*的：

- 前序模块一般为 *DL 目标检测* 和定位类型的模块（如 *轮廓匹配* 和 *Blob 分析*）。这些模块可为*目标跟踪*提供待跟踪/计数的目标。
- 后序模块无特殊要求。

主要配置步骤

在流程中调用*目标跟踪*模块后，该模块的主要配置步骤如下：

1. 在*基本参数*页签配置图像输入、Box 输入和检测区域。
2. 单击*执行*或*连续执行*查看运行结果。
3. 根据实际需求配置运行参数。例如，跟踪状态不理想时，可调节 **X/Y 方向速度**。
4. （可选）单击*清空计数*清除当前计数，重新从 0 开始计数。

参数配置

以下仅介绍该模块的基本参数和运行参数的详情。通过配置基本参数，可定义数据输入。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

结果显示参数详情，请参见 *结果显示* 。

基本参数详情如下。

图像输入

图像源为采集到的流水线上移动的目标的连续帧图像。图像必须按照采集的时间序列排序。

BOX 输入

为前序模块的输出目标信息矩形，可订阅**目标信息矩形**或**检测区域**。

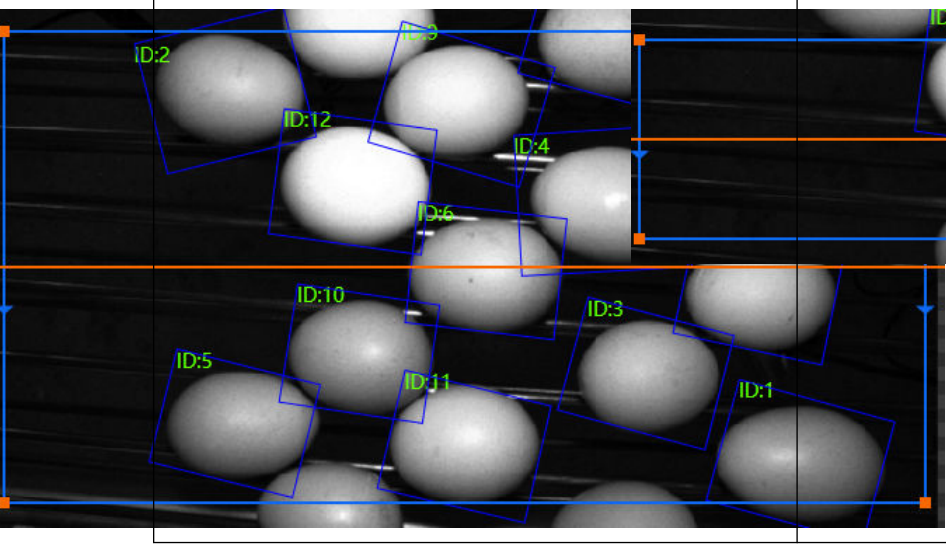
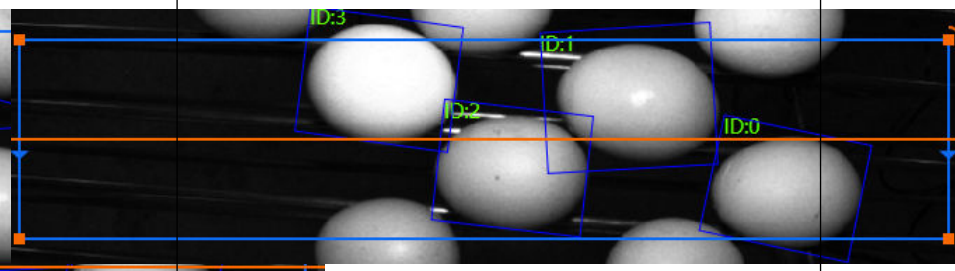
检测区域

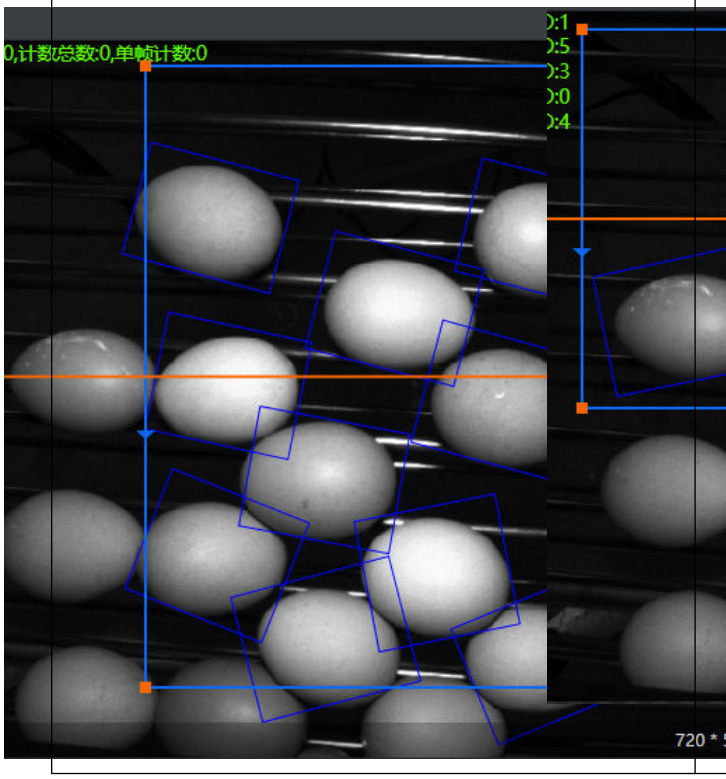
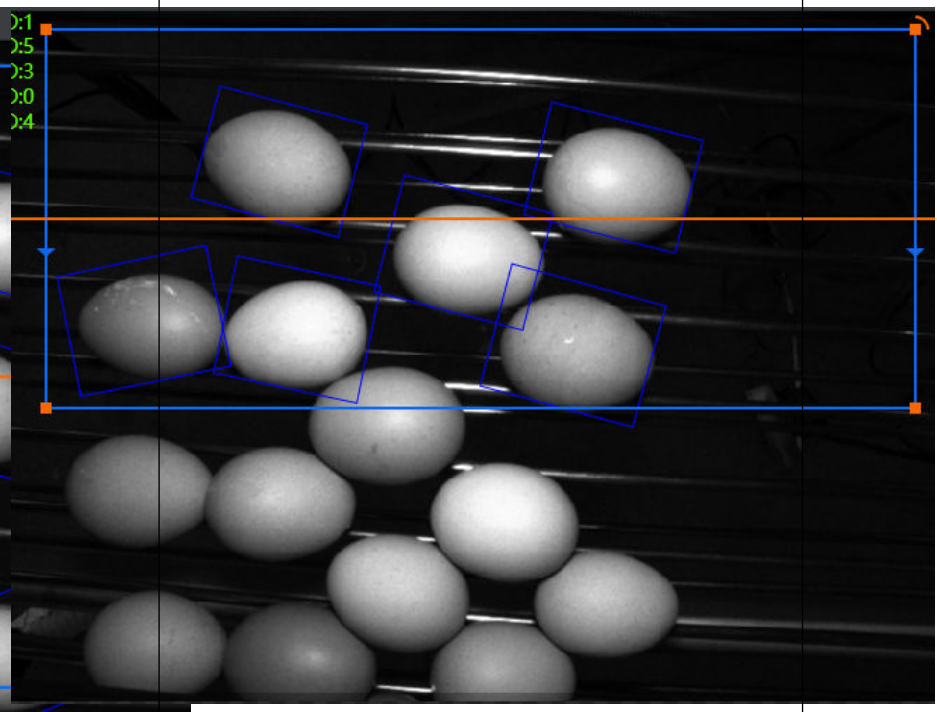
为跟踪和计数的区域。根据目标进入和离开该区域的状态对其进行计数

为了避免计数遗漏，检测区域应尽量位于图像中央，且：

- 区域宽度应尽量与图像宽度相同。
- 区域高度按目标尺寸进行设置，包含 3~5 个目标高度为最佳。

恰当和欠妥的检测区域选取，请参见如下四张示意图。

恰当的检测区域选取	检测区域高度过小
	

检测区域宽度过小	检测区域位置偏上
	

运行参数详情如下。

检测区速度

X 方向速度

跟踪框对在图像上横向移动（从左至右）的物体的跟踪速度，默认值为 1.0。如果跟踪效果不理想，可按需调整取值：

- 跟踪框的移动速度比目标物体移动速度慢时，微调增大取值，例如逐次尝试取值 1.1、1.2、1.3……，直至两者移动速度基本一致。
- 跟踪框的移动速度比目标物体移动速度快时，微调减少取值，例如逐次尝试取值 0.9、0.8、0.7……，直至两者移动速度基本一致。

Y 方向速度

跟踪框对于在图像上纵向移动（从上至下）的物体的跟踪速度，默认值为 1.0。如果跟踪效果不理想，可按需调整取值。调整方式同上。

运行参数

允许缺帧数

若目标检测模块出现未检测到目标的情况，此时目标跟踪模块的**输入 Box** 接收不到数据，定义为丢帧。可通过**允许缺帧数**设置最多丢帧数。

未超过时，不影响目标的计数；超过时，则认为前后是同一个目标，不进行计数。

轨迹重叠率

可根据实际情况对轨迹重叠率参数进行设置，该参数的默认值为 30，范围为 1~100。当两个目标框之间的重叠率超过设置的轨迹重叠率，则会认为是同一个目标。

模块结果

*目标跟踪*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

目标数

int 型，代表当前帧检测区域内跟踪到的目标总数。

计数总数

int 型，代表过去所有帧跨过检测线的所有目标数累计和。

单帧计数

int 型，代表当前帧目标在检测线上的个数 。

多边形轮廓点

轮廓点 X

float 型，代表检测到的目标的轮廓顶点的 X 坐标。

轮廓点 Y

float 型，代表检测到的目标的轮廓顶点的 Y 坐标。

目标 Id 位置

X 坐标

float 型，代表检测到的目标的 id 编号的 X 坐标。

Y 坐标

float 型，代表检测到的目标的 id 编号的 Y 坐标。

边缘点个数

int 型，代表输出的边缘点个数。

目标 Id

int 型，代表检测到的目标 id 编号。

检测区域

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽

float 型，代表 ROI 的宽度。

检测区域高

float 型，代表 ROI 的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

X 方向速度

float 型，代表运行参数中设置的 X 方向速度。

Y 方向速度

float 型，代表运行参数中设置的 Y 方向速度。

检测线

检测线起点

检测线起点 X

float 型，代表检测线起点的 X 坐标。

检测线起点 Y

float 型，代表检测线起点的 Y 坐标。

检测线终点

检测线终点 X

float 型，代表检测线终点的 X 坐标。

检测线终点 Y

float 型，代表检测线终点的 Y 坐标。

直线角度

float 型，代表输出直线的角度。

13.3 测量

测量分类下的模块可用于距离测量、像素统计等，包括线圆测量、圆圆测量、点圆测量等模块。

13.3.1 线圆测量

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

线圆查找模块通过过“圆心点”对“直线”作垂线的方式进行线圆距离的测量，返回的是被测物图像中直线和圆的垂直距离和相交点坐标。该模块多用于工业生产中工件上圆孔与某边之间距离的测量，再根据测量结果确认工件工艺是否合格。
线圆测量模块一般与圆查找、直线查找模块配合使用。圆查找、直线查找作为前序模块，通过圆查找、直线查找定位到图像中需要测量的圆和直线，并输出对应坐标数据给线圆测量模块。线圆测量模块通过订阅接收参数，过“被查找圆的圆心”向“被测量直线”作垂线段，垂线段的长度即为线圆距离。

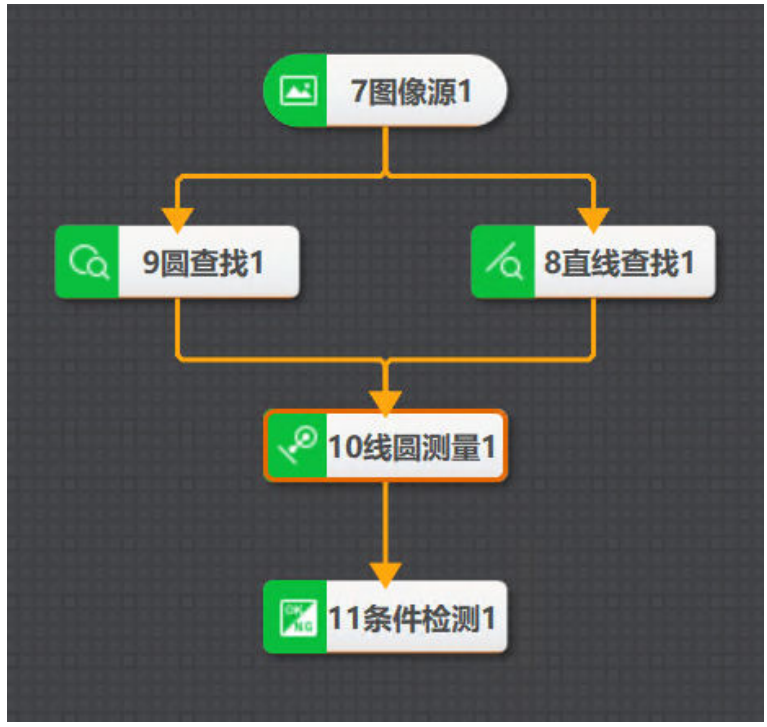


图 13-152 线圆测量示例

该示例通过定位图像中工件的圆孔与上边，将圆孔圆心坐标与直线坐标数据给到线圆测量模块，完成工件线圆距离测量，并通过条件检测模块判断距离是否符合要求的应用。



图 13-153 线圆测量执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  输入源

  可选择前序模块的图像数据。可以是*图像源*模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

说明

来源选择为*订阅*时，该参数可不设置；选择*绘制*时，必须设置。

来源选择

可选订阅或绘制两种方式。

订阅

可直接订阅线和圆的数据。



图 13-154 订阅方式效果

线输入

可选按线、按点、按坐标三种方式订阅直线。

圆输入

可选按圆、按参数两种方式订阅圆。

绘制

绘制方式输入时，相关参数和大部分模块的操作类似，详情参见 *通用模块 ROI 区域* 的介绍。

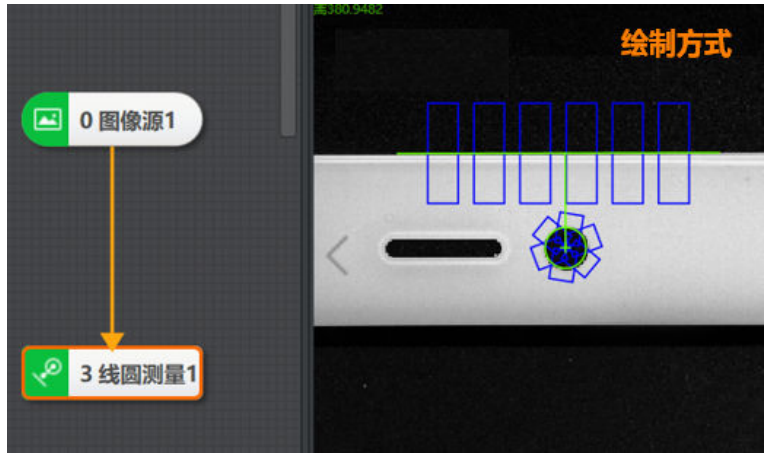


图 13-155 绘制方式效果

- 运行参数：

坐标系选择

可选图像坐标系和特定坐标系。

- 选择图像坐标系时，输出的测量数据为基于图像坐标系的数据；
- 选择特定坐标系时，会基于图像坐标系的数据结合**坐标系矩阵**中选择的标定矩阵文件进行计算，输出实际的物理距离数据。

输出角度范围

设置后可执行输出符合该角度范围内的结果，可选-90°-90°或-180°-180°。

说明

角度是指垂点到圆心的向量与 x 轴正半轴的夹角，如下图所示。下图中，绿色是定位出来的线，黑色是定位出的圆，橙色为垂点到圆心的向量，灰色虚线为 x 轴正半轴，∠α 就是计算的角度。∠α 的正负取决于垂点到圆心的向量方向。若向量指向 y 轴正方向，则为正值；若向量指向 y 轴负方向，则为负值。

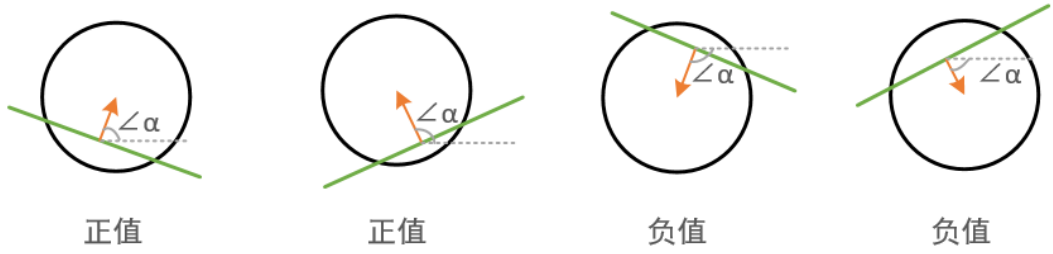


图 13-156 线圆角度示意图

选择类型

设置待测量对象的类型，可选直线 1 或圆 2。

说明

当您选择不同类型时，需要配置的参数也不同。具体请参见下列参数说明。

边缘类型

设置待测量边缘的类型。可选项说明如下：

- 最强：只检测扫描范围内梯度最大的边缘点集合并拟合成直线或圆。
- 第一条：满足条件的第一条直线或圆。
- 最后一条：满足条件的最后一条直线或圆。
- 接近中线：查找最接近区域中线且满足条件的直线。

说明

仅当**选择类型**为**直线 1** 时，您才能选择该选项。

边缘极性

设置待测量边缘的极性，可选项说明如下：

- 从黑到白：从灰度值低的区域过渡到灰度值高的区域的边缘。
- 从白到黑：从灰度值高的区域过渡到灰度值低的区域的边缘 。
- 任意极性：上面两种边缘均被检测。

边缘阈值

边缘阈值也即梯度阈值，取值范围为 0~255。该参数可指定仅当图像中的边缘灰度差大于所设置的边缘阈值时，图像才能被算法检测到。

说明

该参数取值越大，抗噪能力越强，得到的边缘数量越少，甚至可能导致目标边缘点被筛除。请根据实际情况设置该参数。

滤波尺寸

设置滤波器大小，可用于增强目标边缘的清晰程度并抑制噪声。该参数的最小值为 1，数值越小，表示边缘越清晰且过渡带小。

说明

当边缘模糊或有噪声干扰时，增大该值可使检测结果更加稳定。但需注意如果边缘之间的距离小于滤波尺寸，该值设置过大则会影响边缘位置的精度，甚至丢失边缘。建议根据实际边缘过渡带大小设置合适的滤波尺寸。

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，可以获得更佳的查找效果。

初定位

初定位可用于大致判定 ROI 区域内更接近目标圆的区域中心，并将其作为初始圆中心，便于后续对目标圆的精细定位。该功能默认关闭，此时默认 ROI 区域中心为初始圆中心。您也可以开启该功能。开启后，您还需要配置以下参数：

说明

仅当**选择类型**为**圆 2** 时，您才能设置**初定位**。

下采样系数

指定一个下采样系数。该参数取值越大，轮廓点越稀疏，轮廓越不精细。该参数取值的上限为 8。

定位敏感度

指定一个敏感度数值，可用于排除干扰点。该参数取值越大，初定位功能排除噪音干扰的能力越强。

说明

该参数不宜设置过大，否则将容易导致圆初定位失败。请根据实际情况设置该参数。

剔除距离

允许离群点到直线的最大像素距离，超过该距离的点会被排除，值越小，排除点越多。

初始拟合

- 全局：使用查找得到的全局特征点进行直线拟合。
- 局部：按照局部的特征点拟合直线，若局部特征可以更好的体现直线所在位置则采取局部，否则采取全局。

拟合方式

包括最小二乘、huber 和 tukey 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用最小二乘、huber、tukey。

模块结果

*线圆测量*模块的模块结果具体说明如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量直线

line 型，表示输入直线。

测量直线起点

point 型，表示输入直线的起点坐标。

测量直线起点 X

float 型，代表输入直线的起点 X 坐标。

测量直线起点 Y

float 型，代表输入直线的起点 Y 坐标。

测量直线终点

point 型，表示输入直线的终点。

测量直线终点 X

float 型，代表输入直线的终点 X 坐标。

测量直线终点 Y

float 型，代表输入直线的终点 Y 坐标。

测量直线角度

float 型，代表输入直线相对于水平线方向的旋转角度。顺时针旋转为正，逆时针旋转为负。

测量圆环

ANNULUS 型，代表输入或查找到的测量圆环。

圆心点

point 型，代表输入或查找到的圆心点。

圆心点 X

float 型，代表输入或查找到圆的圆心点的 X 坐标。

圆心点 Y

float 型，代表输入或查找到圆的圆心点的 Y 坐标。

圆内径

float 型，代表输入或查找到的测量圆内径。默认为 0 。

圆半径

float 型，代表输入或查找到的圆半径。

圆起始角度

float 型，代表输入或查找到的测量圆起始角度。默认为 0 度。

圆角度范围

float 型，代表输入或查找到的测量圆角度范围。默认为 360 度。

交点*（*代表 1-2）

point 型，代表测量直线与测量圆相交点。

交点*X

float 型，代表两个目标形状交点的 X 坐标。

交点*Y

float 型，代表两个目标形状交点的 Y 坐标。

垂足点

point 型，代表测量圆圆心点垂直于测量直线的交点。

垂足点 X

float 型，代表测量圆圆心点垂直于测量直线的交点坐标 X。

垂足点 Y

float 型，代表测量圆圆心点垂直于测量直线的交点坐标 Y。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

距离

float 型，代表测量圆的圆心垂直于测量直线的距离。

卡尺检测区*（*代表 1-2）

卡尺检测区中心点*

point 型，代表卡尺检测区域的中心点。

卡尺检测区中心 X*

float 型，代表用于查找直线或圆的卡尺框矩形中心的 X 坐标。

卡尺检测区中心 Y*

float 型，代表用于查找直线或圆的卡尺框 矩形中心的 Y 坐标。

卡尺检测区宽度*

float 型，代表用于查找直线或圆的卡尺框宽度。

卡尺检测区高度*

float 型，代表用于查找直线或圆的卡尺框高度 。

卡尺检测区角度*

float 型，代表用于查找直线或圆的卡尺框旋转角度 。

13.3.2 圆圆测量

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

圆圆测量模块通过测量两圆圆心连线长度实现两圆距离的测量，多用于工业生产中工件上圆孔与圆孔间的距离的测量，根据测量结果确认工件工艺是否合格等场景。

圆圆测量模块一般与圆查找模块配合使用。圆查找作为前序模块，通过圆查找定位到图像中需要测量的圆并输出对应坐标数据给圆圆测量模块，圆圆测量模块通过订阅接收参数，过被查找圆的两个圆心作连线线段，连线线段长度即为两个圆之间的距离。

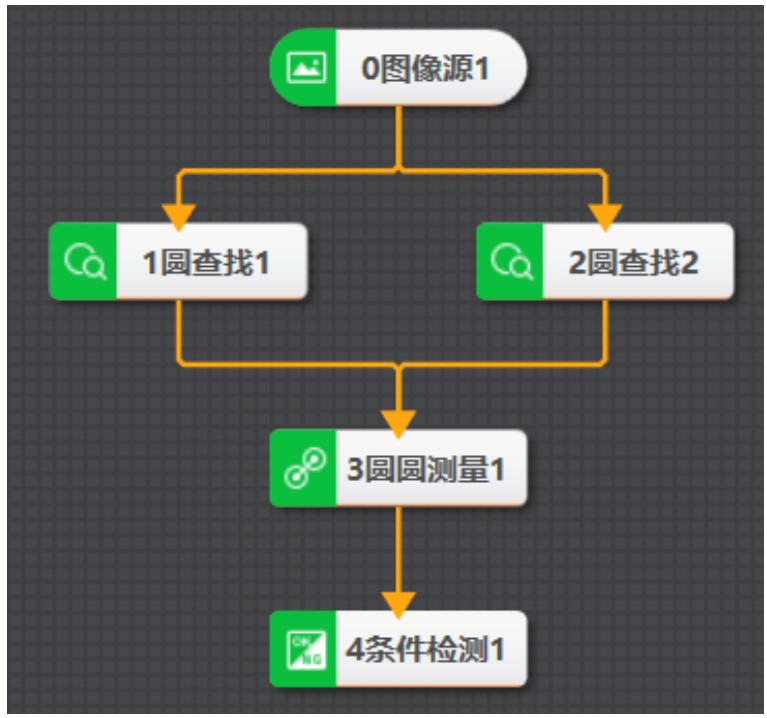


图 13-157 圆圆测量示例

该示例通过定位图像中工件的左右两侧的圆孔，将两圆孔圆心坐标坐标数据给到圆圆测量模块，完成工件圆圆距离测量，并通过条件检测模块判断距离是否符合要求的应用。



图 13-158 圆圆测量执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  输入源

可选择前序模块的图像数据。可以是*图像源*模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

说明

来源选择为*订阅*时，该参数可不设置；选择*绘制*时，必须设置。

来源选择

可选订阅或绘制两种方式。

订阅

可直接订阅 2 个圆的数据。

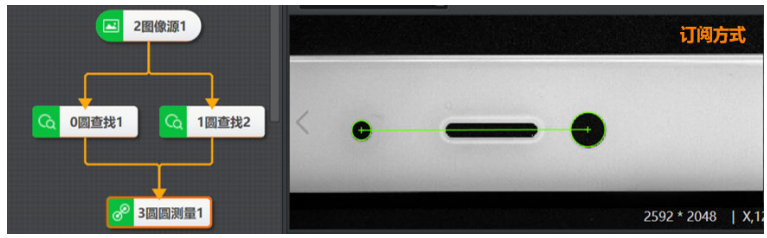


图 13-159 订阅方式效果

圆输入

可选按圆、按参数两种方式订阅圆。

绘制

绘制方式输入时，相关参数和大部分模块的操作类似，详情参见 *通用模块 ROI 区域* 的介绍。

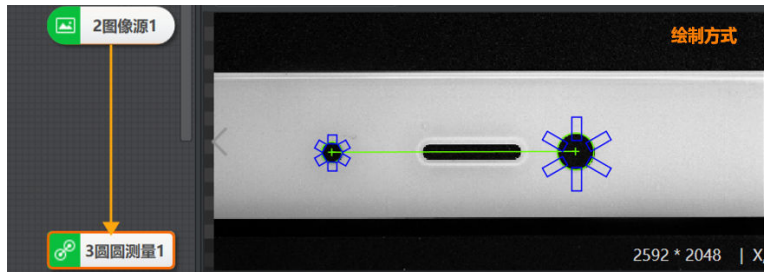


图 13-160 绘制方式效果

- 运行参数：

坐标系选择

可选图像坐标系和特定坐标系。

- 选择图像坐标系时，输出的测量数据为基于图像坐标系的数据；
- 选择特定坐标系时，会基于图像坐标系的数据结合**坐标系矩阵**中选择的标定矩阵文件进行计算，输出实际的物理距离数据。

输出角度范围

设置后可执行输出符合该角度范围内的结果，可选-90°-90°或-180°-180°。

说明

角度是指两个圆心的连线与水平线的夹角。下图∠α 就是计算的角度。默认取输入圆 2 上左侧的角。∠α 的正负取决于圆心连线和水平线的关系。若连线位于水平线下方，则为负值，若在上方，则为正值。左图为负值，右图为正值。

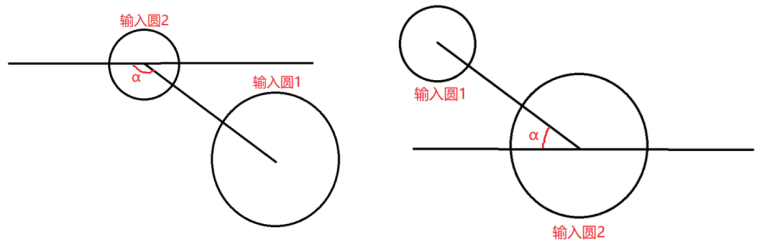


图 13-161 圆圆角度示意图

选择圆

可选圆 1 或圆 2。

边缘类型

设置待测量边缘的类型。可选项说明如下：

- 最强：只检测扫描范围内梯度最大的边缘点集合并拟合成圆。
- 第一条：只检测扫描范围内与圆心距离最小的边缘点集合并拟合成圆 。
- 最后一条：只检测扫描范围内与圆心距离最大的边缘点集合并拟合成圆。

边缘极性

设置待测量边缘的极性，可选项说明如下：

- 从黑到白：从灰度值低的区域过渡到灰度值高的区域的边缘。
- 从白到黑：从灰度值高的区域过渡到灰度值低的区域的边缘 。
- 任意极性：上面两种边缘均被检测。

滤波尺寸

设置滤波器大小，可用于增强目标边缘的清晰程度并抑制噪声。该参数的最小值为 1，数值越小，表示边缘越清晰且过渡带小。

说明

当边缘模糊或有噪声干扰时，增大该值可使检测结果更加稳定。但需注意如果边缘之间的距离小于滤波尺寸，该值设置过大则会影响边缘位置的精度，甚至丢失边缘。建议根据实际边缘过渡带大小设置合适的滤波尺寸。

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，可以获得更佳的查找效果。

初定位

初定位可用于大致判定 ROI 区域内更接近目标圆的区域中心，并将其作为初始圆中心，便于后续对目标圆的精细定位。一般情况下，圆查找前序模块为位置修正，建议关闭初定位。关闭该功能后，默认将 ROI 区域中心作为初始圆中心。若您选择开启该功能，则需要配置以下参数：

下采样系数

指定一个下采样系数。该参数取值越大，轮廓点越稀疏，轮廓越不精细。该参数取值的上限为 8。

定位敏感度

指定一个敏感度数值，可用于排除干扰点。该参数取值越大，初定位功能排除噪音干扰的能力越强。

说明

该参数不宜设置过大，否则将容易导致圆初定位失败。请根据实际情况设置该参数。

剔除距离

允许离群点到圆心的最大像素距离，超过该距离的点会被排除，值越小，排除点越多。

初始拟合

- 全局：使用查找得到的全局特征点进行圆拟合。
- 局部：按照局部的特征点拟合圆，若局部特征可以更好的体现直线所在位置则采取局部，否则采取全局。

拟合方式

包括最小二乘、huber 和 tukey 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用最小二乘、huber、tukey。

模块结果

*圆圆测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量圆环*（*代表 1-2）

测量圆心*

圆*圆心 X

float 型，代表输入或查找到的圆的圆心点的 X 坐标。

圆*圆心 Y

float 型，代表输入或查找到的圆的圆心点的 Y 坐标。

圆*内径

float 型，代表输入或查找到的测量圆内径（默认为 0）。

圆*半径

float 型，代表输入或查找到的圆半径。

圆*起始角度

float 型，代表输入或查找到的测量圆起始角度（默认为 0 度）。

圆*角度范围

float 型，代表输入或查找到的测量圆角度范围（默认为 360 度）。

交点*（*代表 1-2）

point 型，代表测量直线与测量圆相交点。

交点*X

float 型，代表两个目标形状交点的 X 坐标。

交点*Y

float 型，代表两个目标形状交点的 Y 坐标。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

距离

float 型，代表测量圆的圆心垂直于测量直线的距离。

位置关系

string 型，Inside 代表内含，Inscribe 代表内切，Intersect 代表相交，Circumscribe 代表外切，Outside 代表外离。

卡尺检测区*（*代表 1-2）

卡尺检测区中心点*

point 型，代表卡尺检测区域的中心点。

卡尺检测区中心 X*

float 型，代表用于查找直线或圆的卡尺框矩形中心的 X 坐标。

卡尺检测区中心 Y*

float 型，代表用于查找直线或圆的卡尺框 矩形中心的 Y 坐标。

卡尺检测区宽度*

float 型，代表用于查找直线或圆的卡尺框宽度。

卡尺检测区高度*

float 型，代表用于查找直线或圆的卡尺框高度 。

卡尺检测区角度*

float 型，代表用于查找直线或圆的卡尺框旋转角度 。

13.3.3 点圆测量

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

点圆测量模块通过测量点到圆心的连线长度实现点到圆距离的测量，点与圆心的距离为中心距离，点和圆心生成的连线与圆相交的两点为最近距离与最远距离。该模块多用于工业生产中工件上点与圆间距的测量，例如测量齿轮每个齿到轮芯圆的距离、柱状钢材内外径测量等等，根据测量结果确认工件工艺是否合格等场景。

点圆测量模块一般与圆查找、顶点检测、匹配等模块配合使用，圆查找、顶点检测、匹配等模块作为前序模块，通过定位到图像中需要测量的圆和点并输出对应坐标数据给点圆测量模块，点圆测量模块通过订阅接收参数，过“被查找圆的圆心”与“点”作连线，连线线段长度即为中心距离。



图 13-162 点圆测量示例

该示例通过定位图像中齿轮工件的一个齿的顶点以及轮芯的圆环，将坐标数据输入至点圆测量模块，完成齿轮单齿顶点到圆环长度测量的应用。



图 13-163 点圆测量执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

输入源

可选择前序模块的图像数据。

点输入

可选按点、按坐标两种方式订阅点。

圆输入

可选按圆、按参数两种方式订阅圆。

- 运行参数：

坐标系选择

可选图像坐标系和特定坐标系。

- 选择图像坐标系时，输出的测量数据为基于图像坐标系的数据；
- 选择特定坐标系时，会基于图像坐标系的数据结合**坐标系矩阵**中选择的标定矩阵文件进行计算，输出实际的物理距离数据。

输出角度范围

设置后可执行输出符合该角度范围内的结果，可选-90°-90°或-180°-180°。

说明

角度是指点和圆心的连线与水平线的夹角。下图∠α 就是计算的角度。∠α 的正负取决于点和圆心的连线和水平线的关系。若连线位于水平线下方，则为负值，若在上方，则为正值。左图为负值，右图为正值。

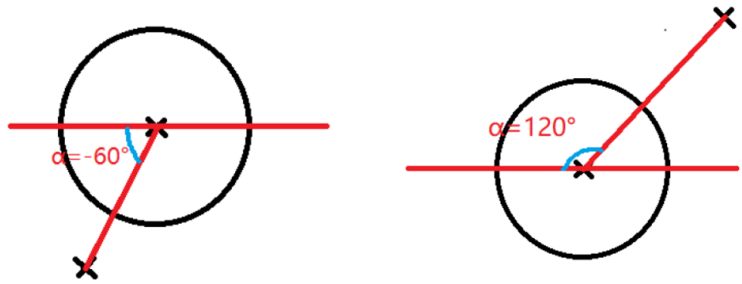


图 13-164 点圆角度示意图

模块结果

*点圆测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

测量点

point 型，表示输入的测量点坐标。

测量点 X

float 型，代表输入点的 X 坐标。

测量点 Y

float 型，代表输入点的 Y 坐标。

测量圆环

ANNULUS 型，代表输入或查找到的测量圆环。

圆心点

point 型，代表输入或查找到的圆心点。

圆心点 X

float 型，代表输入或查找到圆的圆心点的 X 坐标。

圆心点 Y

float 型，代表输入或查找到圆的圆心点的 Y 坐标。

圆内径

float 型，代表输入或查找到的测量圆内径。默认为 0 。

圆半径

float 型，代表输入或查找到的圆半径。

圆起始角度

float 型，代表输入或查找到的测量圆起始角度。默认为 0 度。

圆角度范围

float 型，代表输入或查找到的测量圆角度范围。默认为 360 度。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

中心距离

float 型，代表输入测量点到输入测量圆圆心的距离。

最近距离

float 型，代表输入测量点到输入测量圆上最近的距离。

最远距离

float 型，代表输入测量点到输入测量圆上最远的距离。

13.3.4 点线测量

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

点线测量模块通过测量点到线的垂线段长度实现点到线距离的测量。该模块多用于工业生产中工件上点与直线间距的测量，例如工件角点到单边的距离测量、工件孔点到某边的距离测量等等，根据测量结果确认工件规格型号以及确认工艺是否合格。
点线测量模块一般与直线查找、顶点检测、匹配、边缘交点等模块配合使用，直线查找、顶点检测、匹配、边缘交点等模块作为前序模块，通过定位到图像中需要测量的点和线并输出对应坐标数据至点线测量模块，点线测量模块通过订阅接收参数，过“被查找点”向“被测量直线”作垂线段，垂线线段长度即为距离长度。

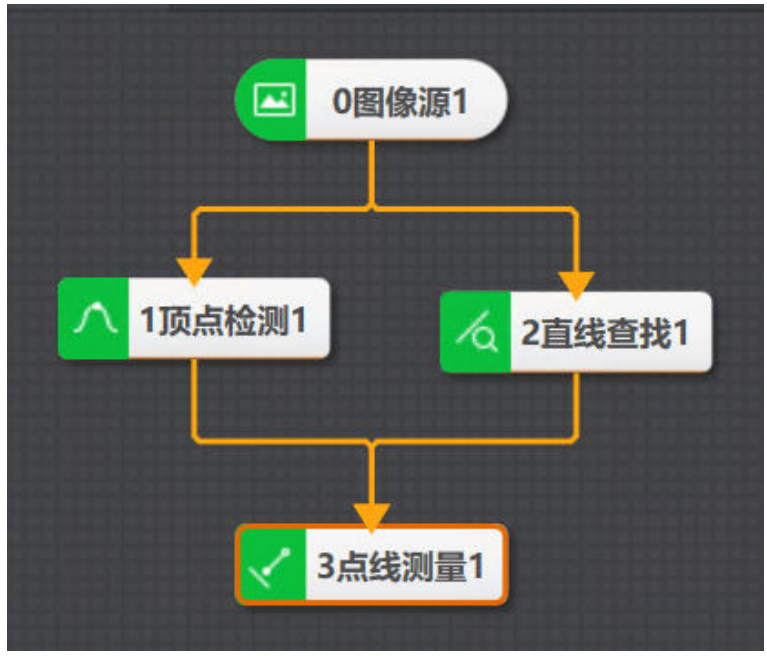


图 13-165 点线测量示例

该示例通过定位图像中工件的一个齿的顶点以及对边，将坐标数据输入至点线测量模块，完成工件单齿最长边的长度测量的应用。

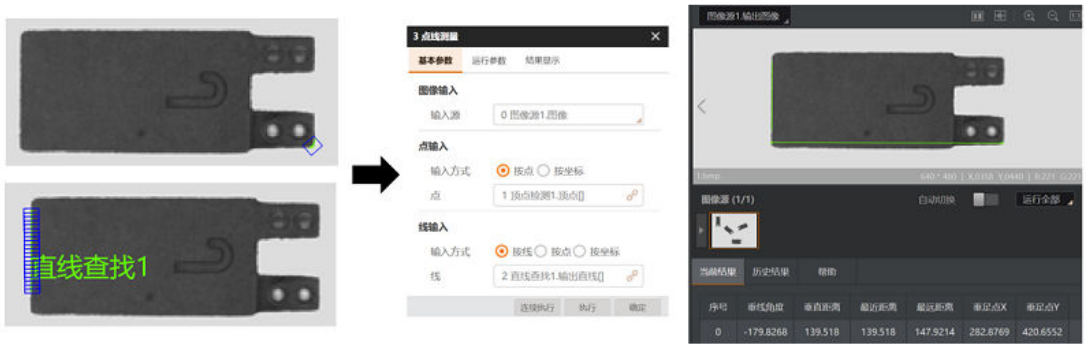


图 13-166 点线测量执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  输入源

  可选择前序模块的图像数据。

  点输入

  可选按点、按坐标两种方式订阅点。

  线输入

  可选按线、按点、按坐标三种方式订阅直线。

- 运行参数：

  坐标系选择

  可选图像坐标系和特定坐标系。

  - 选择图像坐标系时，输出的测量数据为基于图像坐标系的数据；
  - 选择特定坐标系时，会基于图像坐标系的数据结合**坐标系矩阵**中选择的标定矩阵文件进行计算，输出实际的物理距离数据。

  输出角度范围

  设置后可执行输出符合该角度范围内的结果，可选-90°-90°或-180°-180°。

  说明

  角度是指点和垂足的连线与水平线的夹角。下图∠α 就是计算的角度。∠α 的正负取决于连线和水平线的关系。若连线位于水平线下方，则为负值，若在上方，则为正值。左图为负值，右图为正值。

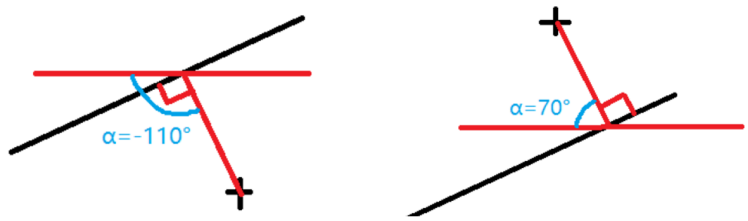


图 13-167 点线角度示意图

模块结果

*点线测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量点

point 型，表示输入的测量点坐标。

测量点 X

float 型，代表输入点的 X 坐标。

测量点 Y

float 型，代表输入点的 Y 坐标。

测量直线

line 型，表示输入直线。

测量直线起点

point 型，表示输入直线的起点坐标。

测量直线起点 X

float 型，代表输入直线的起点 X 坐标。

测量直线起点 Y

float 型，代表输入直线的起点 Y 坐标。

测量直线终点

point 型，表示输入直线的终点。

测量直线终点 X

float 型，代表输入直线的终点 X 坐标。

测量直线终点 Y

float 型，代表输入直线的终点 Y 坐标。

测量直线角度

float 型，代表输入直线相对于水平线方向的旋转角度。顺时针旋转为正，逆时针旋转为负。

垂足点

point 型，代表测量圆圆心点垂直于测量直线的交点。

垂足点 X

float 型，代表测量圆圆心点垂直于测量直线的交点坐标 X。

垂足点 Y

float 型，代表测量圆圆心点垂直于测量直线的交点坐标 Y。

垂线角度

float 型，代表输入测量点垂直于输入测量直线的角。

垂直距离

float 型，代表输入测量点到垂足点的距离。

最近距离

float 型，代表输入测量点到输入测量直线上最近的距离。

最远距离

float 型，代表输入测量点到输入测量直线上最远的距离。

13.3.5 线线测量

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

线线测量模块通过测量直线每个找到的边缘点向对面的线做垂线，计算距离然后求其平均值，平均值即为线线距离。该模块多用于工业生产中工件上两边间距以及夹角的测量，例如测量工件两边距离平均值、判断工件两边夹角是否在合格范围内，根据测量结果确认工件规格型号以及确认工艺是否合格。

线线测量模块一般与直线查找、直线查找组合、多直线查找等可以输出线段数据信息的模块配合使用，直线查找、直线查找组合、多直线查找等模块作为前序模块，通过定位到图像中

需要测量的线并输出对应坐标数据给线线测量模块，线线测量模块通过订阅接收参数，经过计算输出两线段的绝对距离、交点、夹角等参数。

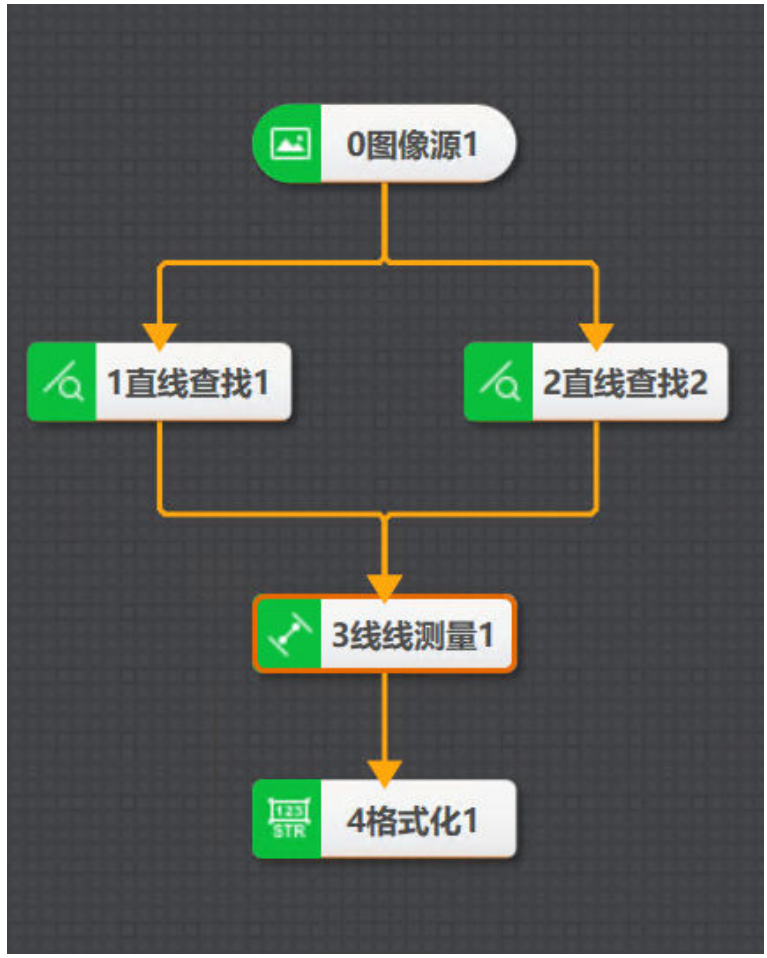


图 13-168 线线测量示例

该示例通过定位图像中工件两条斜边，将坐标数据输入至线线测量模块，完成工件两边夹角角度测量的应用。

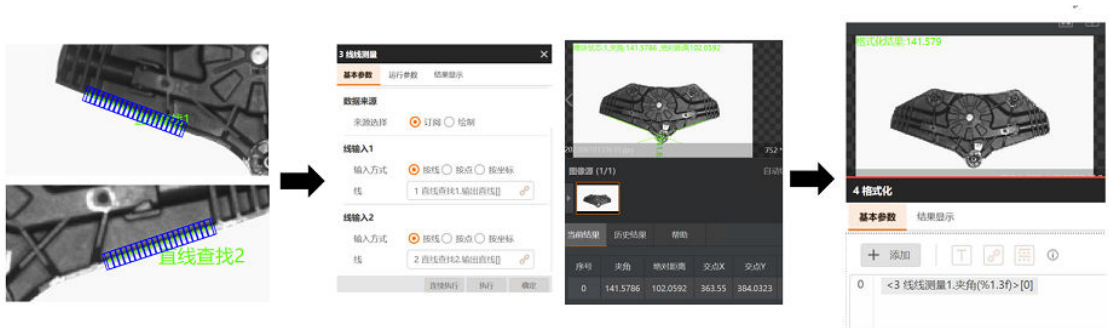


图 13-169 线线测量执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

输入源

可选择前序模块的图像数据。可以是*图像源*模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

说明

来源选择为*订阅*时，该参数可不设置；选择*绘制*时，必须设置。

来源选择

可选订阅或绘制两种方式。

订阅

可直接订阅 2 条线的数据。



图 13-170 订阅方式效果

线输入

可选按线、按点、按坐标三种方式订阅直线。

绘制

绘制方式输入时，您可以选择使用直线卡尺 1 和直线卡尺 2 分别绘制待测量线段，并选择是否开启位置修正功能。更多有关直线卡尺和位置修正的操作说明，请参见 *通用模块 ROI 区域* 。

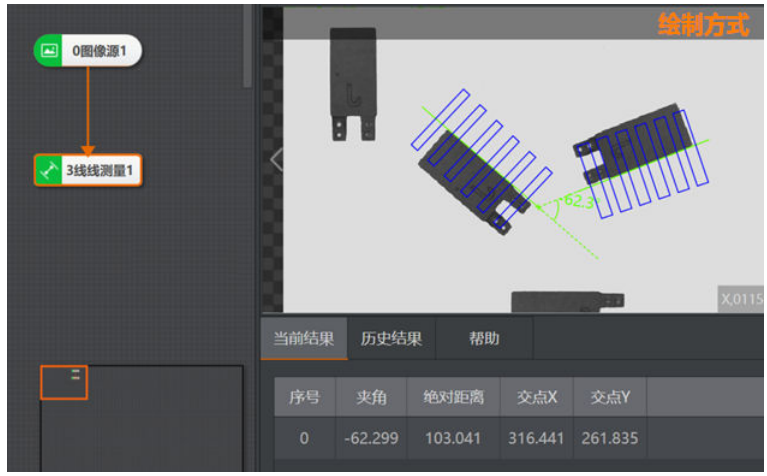


图 13-171 绘制方式效果

- 运行参数：

坐标系选择

可选图像坐标系和特定坐标系。

- 选择图像坐标系时，输出的测量数据为基于图像坐标系的数据；
- 选择特定坐标系时，会基于图像坐标系的数据结合**坐标系矩阵**中选择的标定矩阵文件进行计算，输出实际的物理距离数据。

选择直线

可选直线 1 或直线 2。

边缘类型

- 最强：只检测扫描范围内梯度最大的边缘点集合并拟合成直线。
- 第一条：满足条件的第一条直线。
- 最后一条：满足条件的最后一条直线。
- 接近中线：查找最接近区域中线且满足条件的直线。

边缘极性

- 从黑到白：从灰度值低的区域过渡到灰度值高的区域的边缘。
- 从白到黑：从灰度值高的区域过渡到灰度值低的区域的边缘 。
- 任意极性：上面两种边缘均被检测。

边缘阈值

只有当图像中的边缘灰度差大于所设置的边缘阈值才能被检测出。数值越大，抗噪能力越强，得到的边缘数量越少。

滤波尺寸

描述目标边缘的清晰程度，最小值为 1，值越小，表示边缘越清晰且过渡带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，如果边缘之间距离小于滤波尺寸，会影响边缘位置的精度。

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，可以获得更佳的查找效果。

剔除距离

允许离群点到直线的最大像素距离，超过该距离的点会被排除，值越小，排除点越多。

初始拟合

- 全局：使用查找得到的全局特征点进行直线拟合。
- 局部：按照局部的特征点拟合直线，若局部特征可以更好的体现直线所在位置则采取局部，否则采取全局。

拟合方式

包括最小二乘、huber 和 tukey 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用最小二乘、huber、tukey。

输出角度范围

设置后可执行输出符合该角度范围内的结果，可选-90°-90°或-180°-180°。

说明

角度是指直线交点与两条直线终点方向形成的角，如下图∠α。其输出的正负关系为：从直线 1 到直线 2 方向，顺时针为正，逆时针为负，如下图，左图为负，右图为正。



图 13-172 线线角度示意图

模块结果

*线线测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量直线*（*代表 1-2）

line 型，代表用于查找中线的输入直线，记录了有关该直线的起点和终点坐标。

测量直线*起点

point 型，代表输入直线的起点坐标。

测量直线*起点 X

float 型，代表输入直线起点的 X 坐标。

测量直线*起点 Y

float 型，代表输入直线起点的 Y 坐标。

测量直线*终点

point 型，代表输入直线的终点。

测量直线*终点 X

float 型，代表输入直线终点的 X 坐标。

测量直线*终点 Y

float 型，代表输入直线终点的 Y 坐标。

测量直线*角度（*代表 1-2）

float 型，代表输入直线相对于水平轴的旋转角度。顺时针旋转为正，逆时针旋转为负。

交点

point 型，代表两条测量直线的交点。

交点 X

float 型，代表输入直线 1 与输入直线 2 交点的 X 坐标。

交点 Y

float 型，代表输入直线 1 与输入直线 2 交点的 Y 坐标。

夹角

float 型，代表输入直线 1 与输入直线 2 相交的锐角大小。

绝对距离

float 型，代表两条直线的起点和终点（共 4 个点）到对应另一线直线垂直距离的平均值。

角度标识点*（*代表 1-5）

point 型，代表角度标识点。

角度标识点*X

float 型，点 1：直线向量 1 的起点坐标 X ；点 2：直线向量 1 的终点坐标 X ；点 3：直线向量 1 和直线向量 2 的交点坐标 X ；点 4：直线向量 2 的起点坐标 X ；点 5：直线向量 2 的终点坐标 X。

角度标识点*Y

float 型，代表点 1：直线向量 1 的起点坐标 Y； 点 2：直线向量 1 的终点坐标 Y； 点 3：直线向量 1 和直线向量 2 的交点坐标 Y； 点 4：直线向量 2 的起点坐标 Y； 点 5：直线向量 2 的终点坐标 Y。

卡尺检测区*（*代表 1-2）

卡尺检测区中心点*

point 型，代表卡尺检测区域的中心点。

卡尺检测区中心 X*

float 型，代表用于查找直线或圆的卡尺框矩形中心的 X 坐标。

卡尺检测区中心 Y*

float 型，代表用于查找直线或圆的卡尺框 矩形中心的 Y 坐标。

卡尺检测区宽度*

float 型，代表用于查找直线或圆的卡尺框宽度。

卡尺检测区高度*

float 型，代表用于查找直线或圆的卡尺框高度 。

卡尺检测区角度*

float 型，代表用于查找直线或圆的卡尺框旋转角度 。

13.3.6 点点测量

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

点点测量模块通过测量点到点的连线线段长度实现点到点距离的测量，该模块多用于工业生产中工件上点与点、两圆圆心间距、两直线起点/终点等的测量，例如测量圆形/球型工件圆心点的间距，判断工件间距。以及图像内某两点间距测量等等，根据测量结果确认工件位置、间距以及确认工艺是否合格。

点点测量模块一般与圆查找、顶点检测、直线查找、边缘交点等定位模块配合使用。圆查找、顶点检测、直线查找、边缘交点等模块作为前序模块，通过定位到图像中需要测量的点并输

出对应坐标数据给点点测量模块，点点测量模块通过订阅接收参数，过被查找点作连线线段，连线线段长度即为距离长度。

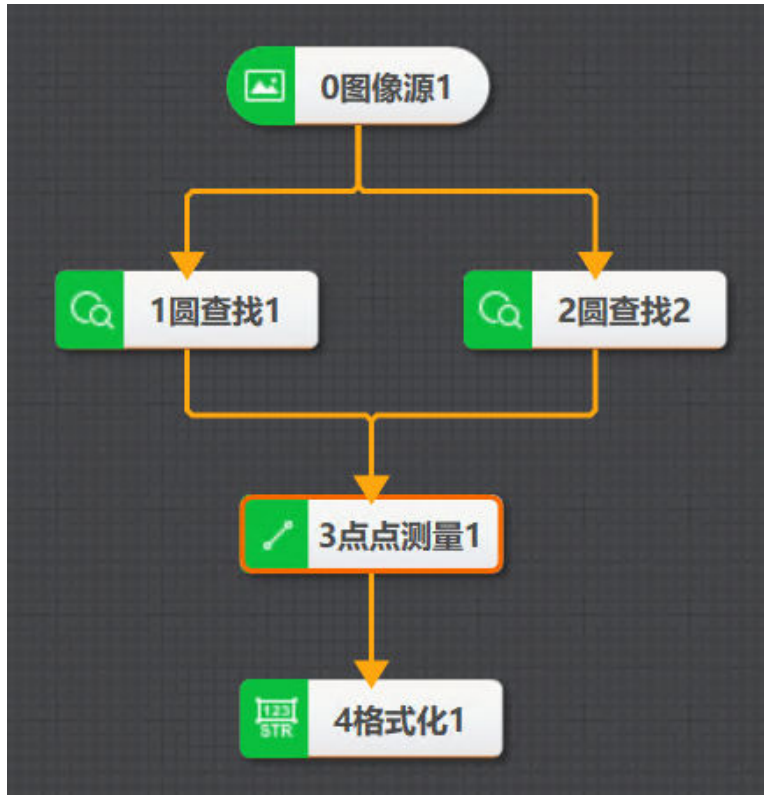


图 13-173 点点测量示例

该示例通过定位图像中圆形工件的圆心点，将圆心坐标数据输入至点点测量模块，完成工件距离测量的应用。



图 13-174 点点测量执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

输入源

可选择前序模块的图像数据。

说明

该参数可不设置。

起点输入
终点输入

可选按点、按坐标两种方式订阅点。

- 运行参数：

坐标系选择

可选图像坐标系和特定坐标系。

- 选择图像坐标系时，输出的测量数据为基于图像坐标系的数据；
- 选择特定坐标系时，会基于图像坐标系的数据结合**坐标系矩阵**中选择的标定矩阵文件进行计算，输出实际的物理距离数据。

输出角度范围

设置后可执行输出符合该角度范围内的结果，可选-90°-90°或-180°-180°。

说明

角度是指点和点的连线与水平线的夹角。下图中，点 1 代表起点，点 2 代表终点，∠α 就是计算的角度。∠α 的正负取决于两点连线和水平线的关系。若连线位于水平线上方，则为正值；若在下方，则为负值。左图为正值，右图为负值。



图 13-175 点点角度示意图

模块结果

*点点测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

测量点*（*代表 1-2）

point 型，代表输入的测量点。

测量点*X

float 型，代表输入测量点的 X 坐标。

测量点*Y

float 型，代表输入测量点的 Y 坐标。

中点

point 型，表示检测到的直线中点。

中点 X

float 型，代表直线中点的 X 坐标。

中点 Y

float 型，代表直线中点的 Y 坐标。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

距离

float 型，代表点与点之间的距离。

13.3.7 亮度测量

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

亮度测量模块针对输入灰度图像的指定 ROI 区域，输出该区域的图像灰度直方图、最小灰度值、最大灰度值、灰度均值、灰度标准差、对比度等信息。

图像灰度**直方图**是对图像中灰度级分布的统计，反映图像中某种灰度出现的频率。灰度直方图是针对数字图像中所有的像素，按照灰度值大小分别统计其出现的频率，即图像中具有某种灰度级的像素个数。直方图的横坐标代表灰度级，纵坐标代表灰度级出现的频率。

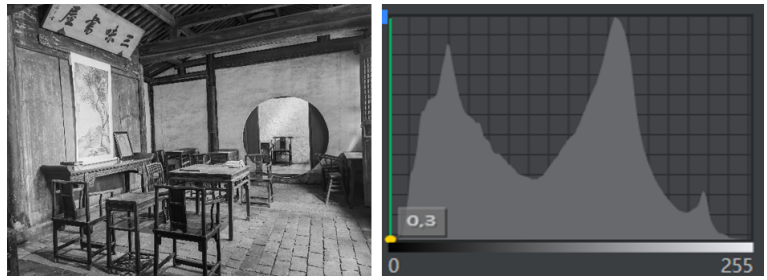


图 13-176 图像灰度直方图

如上图所示，直方图横坐标的范围是 0-255，表示有 255 个灰度级。图的左侧值较小，对应图像中较暗的区域；图的右侧值较大，对应图像中较亮的区域。纵坐标表示对应灰度值出现的次数。

使用方法

亮度测量模块多用于生产中测量生成的图像灰度值是否满足要求、光源或者环境光对生成图像的影响，以及对图像内各区域间的灰度值进行对比测试等。

亮度测量模块一般与采集模块、图像处理等模块配合使用，采集模块作为前序模块，图像处理作为后续模块，采集模块得到的图像通过亮度测量模块进行图像 ROI 内所有像素点的灰度均值和灰度标准差测量，得到需要的图像数据，然后给到后续的图像处理等模块进行后续处理。

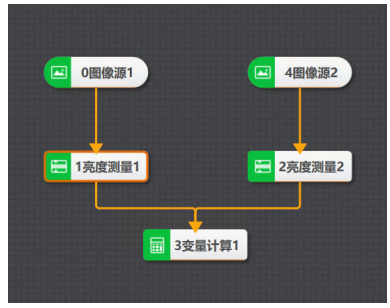


图 13-177 亮度测量方案示例

该示例通过测量两张图所有像素点的对比度值，再通过变量计算完成多图像对比度均值测量的应用。



图 13-178 亮度测量执行结果

参数配置

基本参数如下：

输入源

需要订阅前序模块的图像数据，可以是图像源模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

输出掩膜

可设置是否输出掩膜图像。

ROI 区域类型

- 选择*图形类型*，需设置以下参数：

  ROI 创建

  - *绘制*：自定义选择绘制全图、圆形、扇形、矩形及多边形 ROI 区域。

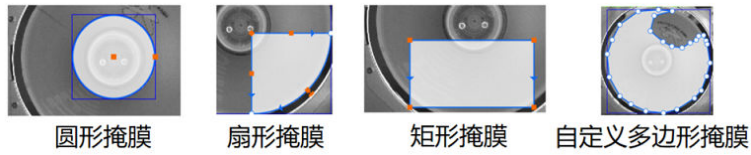


图 13-179 ROI 创建形状

  - *继承*：**按矩形/圆形区域**需要订阅前序模块的输出区域，**按矩形/圆形参数**需要自定义或订阅相关参数。
- 选择*图像类型*，需订阅指定图像作为掩膜图像。

模块结果

*亮度测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

最小值

int 型，代表检测区域图像灰度值的最小值。

最大值

int 型，代表检测区域图像灰度值的最大值。

均值

float 型，代表检测区域图像灰度值的平均值。

标准差

float 型，代表检测区域图像灰度值的标准差。

直方图

int 型，此为长度是 256 的数组，代表在检测区域中每个灰度值所对应的像素个数，

对比度

float 型。使用自动二值化将区域分为较亮和较暗的两个部分，分别计算其灰度均值。两个灰度均值差即为对比度。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.3.8 间距检测

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

间距测量模块用于检测两特征边缘之间的间距，首先查找满足条件的边缘，然后进行距离测量。该模块多用于生产中测量工件宽度、确认工件规格是否满足要求，以及测量两个工件的间距等。

间距检测模块一般与采集模块、匹配定位等模块配合使用，在产线上工件位置不固定，采集模块作为前序模块，采集得到的图像通过定位模块的模型定位到工件具体位置以及位置修正信息，再使用间距检测模块对工件需要测量的位置进行检测。间距检测模块可以订阅位置修正信息进行位置修正，以达到检测不同位置工件的目的。



图 13-180 间距检测示例

该示例通过对工件进行定位，提供位置修正信息给到间距检测模块，实现对不同位置工件宽度进行检测的应用。

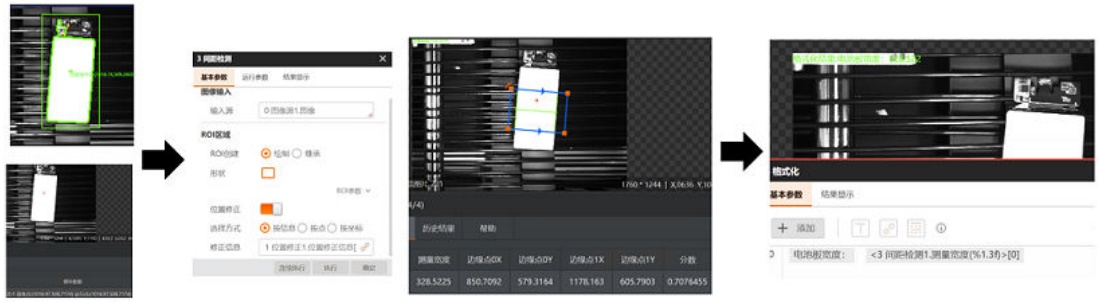


图 13-181 间距检测执行结果

参数配置

基本参数如下：

输入源

需要订阅前序模块的图像数据，可以是图像源模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

ROI 创建

- **绘制**：自定义选择绘制 ROI 区域。
- **继承**：**按区域**需订阅前序模块的矩形输出区域，**按参数**需自定义或订阅 ROI 中心点 X/Y 坐标和宽高度。

运行参数如下：

滤波尺寸

描述目标边缘的清晰程度，最小值为 1，值越小，表示边缘越清晰且过渡带小。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，如果边缘之间距离小于滤波尺寸，会影响边缘位置的精度。

边缘阈值

只有当图像中的边缘灰度差大于所设置的边缘阈值才能被检测出。数值越大，抗噪能力越强，得到的边缘数量越少。

边缘 0/1 极性

- 从黑到白：从灰度值低的区域过渡到灰度值高的区域的边缘。
- 从白到黑：从灰度值高的区域过渡到灰度值低的区域的边缘 。
- 任意极性：上面两种边缘均被检测。

最大结果数

最大的查找的数量。

排序方式

可根据需求按照**分数**或**方向**维度排序。支持按照**分数升序**、**分数降序**、**方向正序**和**方向逆序**排序。

边缘对类型

- 最宽：表示检测范围内间距最大的边缘对。
- 最窄：表示检测范围内间距最小的边缘对。
- 最强：表示检测范围内边缘对平均梯度最大的边缘对。
- 最弱：表示检测范围内梯度最小的边缘对。
- 第一对：表示检测范围内边缘对中心与搜索起始点最近的边缘对。
- 最后一对：表示检测范围内边缘对中心与搜索起始点最远的边缘对。
- 最接近：表示检测扫描范围内和理想宽度最接近的边缘对集合。
- 最不接近：表示检测扫描范围内和理想宽度最不接近的边缘对集合。
- 全部：表示检测扫描范围内所有边缘对。

查找方向

指定检测的查找方向。可选择**四边形从上到下或扇环形由内向外搜索**或**四边形从左到右或扇环形逆时针搜索**。

最小边缘分数

查找到边缘的最小得分，得分低于最小分数的边缘会被过滤。

模块结果

*间距检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

边缘状态

int 型，代表图像上某个点是否位于边缘。1 代表位于边缘，0 代表不在边缘。

边缘个数

int 型，代表查找到的边缘个数。

测量宽度

float 型，代表检测到间距的宽度。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

边缘*极性（*代表 0-1）

int 型，代表边缘的灰度变化。1 表示从黑到白，2 表示从白到黑，3 表 示任意极性。

边缘*位置（*代表 0-1）

float 型，代表边缘点所在的位置。

边缘点*（*代表 0-1）

边缘点*X

float 型，代表检测到的第（*+1）个边缘点的 X 坐标。

边缘点*Y

float 型，代表检测到的第（*+1）个边缘点的 Y 坐标。

直线*起点（*代表 0-1）

直线*起点 X

float 型，代表边缘直线*起点的 X 坐标。

直线*起点 Y

float 型，代表边缘直线*起点的 Y 坐标。

直线*终点（*代表 0-1）

直线*终点 X

float 型，代表边缘直线*终点的 X 坐标。

直线*终点 Y

float 型，代表边缘直线*终点的 Y 坐标。

直线*角度（*代表 0-1）

float 型，代表边缘直线相对于水平线方向的旋转角度。顺时针旋转，则为正角度；逆时针旋转，则为负角度。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.3.9 像素统计

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

像素统计模块用于统计灰度图像中指定 ROI 区域内满足灰度阈值范围的像素数量，并给出该部分像素数量占 ROI 内所有像素数量的比率。图像阈值是指图像中像素灰度值的临界值，图像阈值是图像分割的基础，基于此可完成图像的二值化。
例如：ROI 区域为 2×2 像素格，共计 4 个像素点，灰度值分别为 130、140、150、160。算法参数阈值范围是[10,150]，则像素数量为 3、比率为 0.75。

使用方法

像素统计模块可用于检测图像阈值是否满足项目需求的评估应用。
像素统计模块一般与采集模块、图像处理、逻辑模块等模块配合使用，采集模块作为前序模块，图像处理作为后续模块，采集模块得到的图像通过像素统计模块对 ROI 设定区域内满足高低阈值灰度设置的像素点个数进行统计，再将数据给到后续的逻辑模块进行判断或者图像处理模块进行处理。



图 13-182 像素统计示例

该示例通过将 ROI 指定区域设为覆盖工件，统计图像 ROI 设定区域内满足阈值范围的像素数量以及比率，再通过条件检测完成判断比率是否满足需求的应用。



图 13-183 像素统计执行结果

参数配置

输入源

需要订阅前序模块的图像数据，可以是图像源模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

输出掩膜

可设置是否输出掩膜图像。

ROI 区域类型

- 选择*图形类型*，需设置以下参数：

ROI 创建

- ***绘制***：自定义选择绘制全图、圆形、扇形、矩形 ROI 区域。
- ***继承***：**按矩形/圆形区域**需要订阅前序模块的输出区域，**按矩形/圆形参数**需要自定义或订阅相关参数。

- 选择***图像类型***，需订阅指定图像作为掩膜图像。

阈值范围

需统计区域中的像素灰度值范围，可设置多个阈值范围。

模块结果

*像素统计*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

像素数量

int 型，代表检测区域在指定阈值范围内的像素点数。

比率

float 型，代表检测区域内在指定阈值范围内的像素点数与总的像素点数的比值 。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.3.10 直方图工具

本节内容包含：

- *模块原理*
- *使用方法*

- *参数配置*
- *模块结果*

模块原理

直方图工具针对输入灰度图像的指定 ROI 区域，输出该区域的图像灰度直方图、图像累计直方图、像素个数、最小灰度值、最大灰度值、灰度中值、灰度众数（出现次数最多的灰度）、灰度均值、灰度标准差、对比度等信息。

- 图像灰度**直方图**是对图像中灰度级分布的统计，反映图像中某种灰度出现的频率。灰度直方图是针对数字图像中所有的像素，按照灰度值大小分别统计其出现的频率，即图像中具有某种灰度级的像素个数。直方图的横坐标代表灰度级，纵坐标代表灰度级出现的频率。
- **累积直方图**表示灰度直方图的灰度频数的累计统计，每一个横坐标灰度值代表小于等于此灰度值的频数之和。

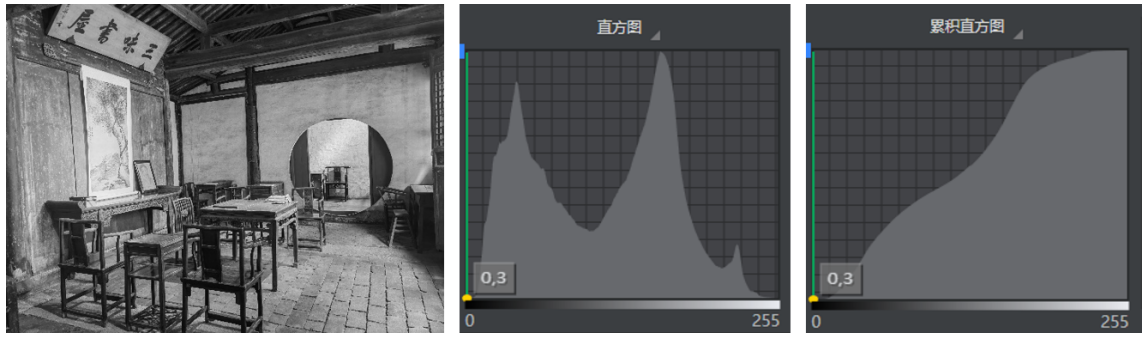


图 13-184 直方图与累积直方图

如上图所示，直方图横坐标的范围是 0-255，表示有 255 个灰度级。图的左侧值较小，对应图像中较暗的区域；图的右侧值较大，对应图像中较亮的区域。纵坐标表示对应灰度值出现的次数。

使用方法

直方图工具模块主要用于对图像进行检测处理的相关应用。

直方图工具一般与采集模块、图像处理、逻辑模块等模块配合使用，采集模块作为前序模块，图像处理作为后续模块，采集模块得到的图像通过直方图工具模块对 ROI 设定区域内的图像进行检测，再将数据给到后续的逻辑模块进行判断或者图像处理模块进行处理。



图 13-185 直方图示例

该示例中先使用直方图工具统计图像 ROI 设定区域内的像素个数、灰度值、对比度等参数，通过条件检测模块判断图像对比度是否满足设定范围，若不满足，则按对比度最大值进行图像增强应用。



图 13-186 直方图工具执行结果

参数配置

输入源

需要订阅前序模块的图像数据，可以是图像源模块输出的图像数据，也可以是图像处理模块处理后的图像数据。

输出掩膜

可设置是否输出掩膜图像。

ROI 区域类型

- 选择*图形类型*，需设置以下参数：

  ROI 创建

  - *绘制*：自定义选择绘制全图、圆形、扇形、矩形及多边形 ROI 区域。

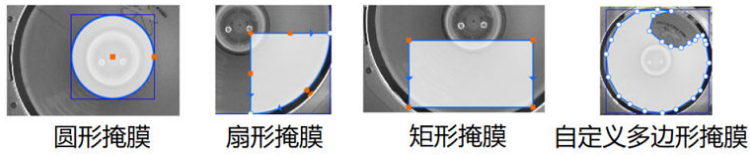


图 13-187 ROI 创建形状

  - *继承*：**按矩形/圆形区域**需要订阅前序模块的输出区域，**按矩形/圆形参数**需要自定义或订阅相关参数。
- 选择*图像类型*，需订阅指定图像作为掩膜图像。

模块结果

*直方图工具*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

像素数量

int 型，代表检测区域在指定阈值范围内的像素点数。

最小值

int 型，代表检测区域图像灰度值的最小值。

最大值

int 型，代表检测区域图像灰度值的最大值。

中值

int 型，代表检测区域内像素灰度值的中值。

峰值

int 型，代表检测区域内出现次数最多的像素值。

均值

float 型，代表检测区域图像灰度值的平均值。

标准差

float 型，代表检测区域图像灰度值的标准差。

对比度

float 型。使用自动二值化将区域分为较亮和较暗的两个部分，分别计算其灰度均值。两个灰度均值差即为对比度。

直方图

int 型，此为长度是 256 的数组，代表在检测区域中每个灰度值所对应的像素个数，

累积直方图

int 型，代表 256 定长数组，检测区域内从 0 开始递增累计的像素点数。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.4 识别

识别分类下的模块可对图像进行处理，包括二维码识别、条码识别、字符识别、DL 字符识别、DL 读码、DL 字符定位、DL 单字符检测和 ML 分类模块。

13.4.1 DL 读码

*DL 读码*基于深度学习模型，帮助您定位和解析场景中一维码和二维码的信息。相较于传统读码模块，该模块的读码准确率较高，且在复杂场景中也可保持较好的识别率。

使用方法

*DL 读码*模块会调用内置的深度学习模型定位和解析码信息，因此您无需自行训练模型。

前后序模块

该模块一般直接与 *图像源* 模块配合使用。后序无指定模块，可处理该模块输出的信息即可。

主要配置方法

该模块支持解析常见的码制，包括 CODE39、QR 码等。使用时，您可根据场景中可能出现的码制，在模块配置的**运行参数**页签中，开启对应的识别选项。

您也可设置相关的其他参数，过滤或调整模块的处理效果。

流程示例

在下图流程中，*DL 读码*模块识别*图像源*图像中的二维码信息。



图 13-188 DL 读码示例效果图

参数配置

本节介绍 *DL 读码*的运行参数。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

条码识别

条码类型

支持 CODE39 码、CODE128 码、库得巴码、EAN8 码、EAN13 码、UPCA 码、UPCE 码、交替 25 码、CODE93 码、MATRIX25 码、MSI 码、CNPOST 码、CODE11 码、IND25 码、ITF14 码。您需根据场景中的条码启用对应的条码类型。

条码个数

期望查找并输出的条码最大数量，若实际查找到的个数小于该参数值，则输出实际数量的条码。

二维码识别

二维码类型

支持 QR 码、DataMatrix 码。您可根据场景中的二维码，启用对应的二维码类型。

二维码个数

期望查找并输出的二维码最大数量，若实际查找到的个数小于该参数值，则输出实际数量的二维码。

堆叠码识别

堆叠码类型

支持 PDF417 码。您可根据场景中的堆叠码，启用对应的堆叠码类型。

堆叠码个数

期望查找并输出的堆叠码最大数量，若实际查找到的个数小于该参数值，则输出实际数量的堆叠码。

运行参数

超时退出时间

算法运行时间超出该参数值，则直接退出，单位 ms。设置为 0 时，超时退出时间则会关闭，以实际算法运行时间为准。默认设置为 1000ms。

高性能模式

开启后，识别效果会更佳，但占用的 CPU 或 GPU 资源会增加。

码行列号

该参数用于辅助判断实际场景中检出的码位置。开启该参数后，需设置**行数量**与**列数量**。您可通过定位的码的行列信息，判断是否存在漏检情况。

在您设置**行数量**和**列数量**后，模块会将按照行列数量，将输入图像平均分割成若干个单元格。运行流程时，模块根据码检测框的外接矩形在相应单元格中所占的比例，计算码的行列信息。

说明

计算行列信息时，输出结果是从 *0* 开始的。换言之，假设某一码的行列信息为 *1，1*，则表示该码在第二行第二列。

假设在一个场景中，**行数量**设置为 *3*、**列数量**设置为 *4*。若码位于第三行第三列，则输出的码行列信息为 *（2，2）*。



图 13-189 码行列号计算方式

码等级

开启**码等级**后，还需设置相关参数。一维码等级仅支持 128 码和 39 码，二维码等级仅支持 QR 码和 DM 码。

二维码等级

ISO 标准

包含 ISO15415、ISO29158 两种标准，可根据需求选择。

评级处理类型

可选处理类型 1、处理类型 2。

处理类型 1

支持 HIK 评级模式。

处理类型 2

支持 ISO 评级模式。

评级模式

可选 ISO 模式、HIK 模式，根据实际需求选择。

极性

包含任意、白底黑码和黑底白码三种形式，可根据所要识别码的极性进行选择。

二维码行数/列数

表示实际二维码包含的行数和列数，一般建议使用默认值，如有特殊情况也可根据实际数值进行填写。组成二维码的最小“方块”或“圆”所在行和列即为一行或一列。

边缘类型

可选连续型、离散型。

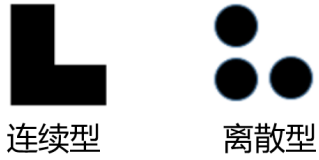


图 13-190 边缘类型

一维码等级

译码评分
边缘确定度
符号对比度
最小反射率
边缘对比度
模块均匀性
可译码性
缺陷度

开启代表该指标的子等级计入总等级。

可译码性 A/B/C/D

可译码性指读取该码的准确性和成功率。A/B/C/D 对应该指标分数的计算等级区间。

缺陷度 A/B/C/D

缺陷度可反映一维码的质量。A/B/C/D 对应该指标分数的计算等级区间。

最小反射率 A

一维码中最暗元素的反射率水平，可反映一维码在不同背景下的可读性和可靠性。

模块均匀性 A/B/C/D

模块均匀性即条形码中的条和空白的宽度是否一致，若宽度不一致则会导致读取失败。A/B/C/D 对应该指标分数的计算等级区间。

对比度 A/B/C/D

对比度即黑色和白色之间的区分程度。A/B/C/D 对应该指标分数的计算等级区间。

模块结果

*DL 读码*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

码个数

int 型，代表识别到码的个数。

码状态

int 型，代表码是否可以识别。1 表示识别成功，不显示表示识别失败，会显示各个码的状态 。

编码信息

string 型，代表各个码蕴含的信息。

码类型

int 型，代表识别到码的类型。

PPM

float 型，一维码中代表成像后最细黑条或白空的像素数，二维码中代表最小模块边长占用的像素数 。

码角点

码角点 X

float 型，代表码四个角点的 X 坐标。

码角点 Y

float 型，代表码四个角点的 Y 坐标。

码角度

float 型，代表码起点与终点连线后与水平方向的夹角，一维码顺时针为正方向，二维码逆时针为正方向。

码镜像标志

int 型，代表码的正反面，0 为反面，1 为正面。

矩形框

box 型，代表识别到码的矩形框。

中心点

point 型，代表识别到码的矩形框中心点。

中心 X

float 型，代表识别到码的矩形框中心点的 X 坐标。

中心 Y

float 型，代表识别到码的矩形框中心点的 Y 坐标。

矩形宽度

float 型，代表识别到码的矩形框宽度。

矩形高度

float 型，代表识别到码的矩形框高度。

矩形角度

float 型，代表识别到码的矩形框角度，由码的点 0 和点 1 所连直线与水平线夹角构成，顺时针为正，逆时针为负。

说明

一般码有 4 个角点，按顺序分别为点 0、点 1、点 2、点 3，不同码制的顶点规则和标准有所差异，具体请以实际为准。

码行号

int 型，代表码在区域的第几行。

码列号

int 型，代表码在区域的第几列。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

码等级

总等级质量

int 型，代表码的总体质量，0(F)、1(D)、2(C)、3(B)、4(A)，数值越 大，代表码质量越好。

译码评分

int 型，代表译码的评分，数值越大代表译码更容易更完整。

符号对比度

int 型，代表评估条码区域深浅是否明显，最小亮度值与最大亮度值之 间的差值，0~4 数值越大表示差值越大。

模块均匀性

int 型，代表一维码（条形码）中条或空白的宽度是否一致。0~4 数 值越大表示均匀性越好。

符号对比度分数

float 型，代表符号对比度的得分。

模块均匀性分数

float 型，代表模块均匀性的得分。

固定程度

int 型，代表衡量空白区、定位图形、校正图形以及其他固有图形的无 损情况是否严重影响参考译码算法对视场终探测和识读能 力，是对固定图案的所有组成部分的总体评级。0~4 数值 越大代表固定程度越好。

轴向不均匀性

int 型，代表评估单元纵向和横向大小的失真度。0~4 数值越大代表不 均匀性越小 。

网格不均匀性

int 型，代表评估网格各单元的最大偏移。0~4 数值越大代表不均匀性 越小。

未使用纠错

int 型，代表评估纠正错误符号所消耗的纠错容量，0~4 数值越大代表消耗纠错容量越少 。

水平打印伸缩

int 型，代表评估印刷单元水平方向的延伸，0~4 数值越大代表延伸越少 。

垂直打印伸缩

int 型，代表评估印刷单元垂直方向的延伸，0~4 数值越大代表延伸越少 。

反射率余量

int 型，代表码中黑色和白色模块反射率之间的差异，数值越大代表反射余量越大。反射率差异越小会导致读码失败。

固定程度分数

float 型，代表固定程度的得分情况。

码轴规整性分数

float 型，代表轴向不均匀性的得分情况。

网格均匀性分数

float 型，代表网格不均匀性的得分情况。

未使用纠错分数

float 型，代表未使用纠错的得分情况 。

水平打印伸缩分数

float 型，代表水平打印伸缩的得分情况。

垂直打印伸缩分数

float 型，代表垂直打印伸缩的得分情况。

反射率余量分数

float 型，代表反射率余量的得分情况。

边缘确定度

int 型，代表码边缘是否清晰明确，0~4 数值越大代表边缘越容易明确 。

最小反射率

int 型，代表最暗的元素的反射率水平，在不同背景下的可读性和可靠 性，0~4 数值越大代表反射水平越高。

最小边缘对比度

int 型，代表相邻条纹的对比度，0~4 数值越大代表对比度越大。

可译码性

int 型，代表读取该码的准确性和成功率，0~4 数值越大代表可译性越高。

缺陷度

int 型，代表码中存在缺陷，用于描述该一维码的质量，0~4 数值越大 代表缺陷越小 。

静区

int 型，代表一维码中有一段特定的区域以此来确定一维码的起始位置 和方向，左右两侧是否有足够的空间，0~4 数值越大代表 静区质量越好。

边缘确定度分数

int 型，代表边缘确定度的得分情况 。

最小反射率分数

float 型，代表最小反射率的得分情况。

最小边缘对比度分数

float 型，代表最小边缘对比度的得分情况。

可译码性分数

float 型，代表可译码性的得分情况。

缺陷分数

float 型，代表缺陷度的得分情况。

静区分数

float 型，代表静区的得分情况。

调试信息

string 型，代表算法调试信息。

算法调试状态字

int 型，代表算法调试信息状态。

算法调试信息大小

int 型，代表算法调试信息大小。

算法调试信息地址

string 型，代表算法调试信息储存地址。

13.4.2 条码识别

*条码识别*模块可定位和识别指定区域内的条码。该模块支持识别 CODE39 码、CODE128 码等多种编码方式，同时支持预处理条码图像，常用于物流读码、商品贴标、零件喷码及条码图像旋转和倾斜的场景。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

一维条形码工作原理

条码通过不同宽度的空白条和黑色条（分别代表 0 和 1），将信息映射成为对应的信息。完整的条码通常包含两侧静区、起始符、数据符、校验位（可选）、终止符。这些元素的排列方式，如下图所示。



图 13-191 条码组成

算法工作步骤

该模块主要通过识别条码的条宽度和寻找匹配码制，解码一维条码信息。具体的算法工作步骤如下：

1. 基于条形码纹理，算法定位候选区，并预处理图像。
2. 逐行扫描条形码，获取条形码黑色条与空白条的边界位置坐标（即下图中绿色点对应坐标）。

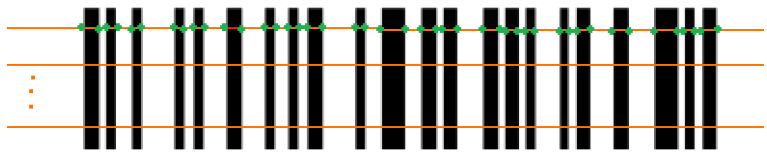


图 13-192 获取坐标

3. 组合匹配黑色条与空白条，并计算黑色条和空白条的宽度。
4. 算法通过匹配起始符和终止符，获得当前检测条码的码制。
5. 基于码制，算法解析数据符内的字符信息。
6. 算法对每一行进行解析，并汇总每行的解析结果，并输出最终的字符信息。

使用方法

您可参照如下内容，在流程中调用该模块。

前序模块

该模块无固定搭配使用模块。在流程中调用该模块时，前序模块可为 *图像源* 、 *仿射变换* 、 *形态学处理* 等模块。图像处理类模块可处理待读码图像，提高读码效率。

后序模块

后序模块无特殊要求，能处理该模块输出信息即可。例如，可以将条码信息和位置输入至 *定位* 、 *通信* 等模块，用作后续处理。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，您可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

CODE39 码/CODE128 码/库得巴码/EAN 码/交替 25 码/CODE93 码

控制模块能否识别对应类型的条码。若您不确定待检测条码的类型，建议开启全部项。

条码个数

设置模块能够查找的最大条码数量。若查找到的条形码数量大于该值，则输出前若干个条形码。

降采样系数

降采样用于降低采样点数，提高图像处理速度。对于一帧分辨率为 N×M 的图像，若降采样系数设置为 k，则算法在该图像的每列每行中，每间隔 k 个点取 1 个点。该例子中，经过降采样后，图像分辨率降低为（N/k）×（M/k）。增大该参数值会降低待处理图像的分辨率，降低轮廓的精细程度，并提高处理效率。但是，当参数值过大时，会丢弃过多图像细节，从而导致无法识别具体信息。

检测窗口大小

条码区域定位窗口大小。默认值 4，当条码中空白间隔比较大时，可以设置得更大，比如 5，但一般要保证条码高度大于窗口大小的 6 倍左右。

静区宽度

静区指条码左右两侧空白区域宽度，默认值 30，稀疏时可尝试设置 50。

去伪过滤尺寸

算法支持识别的最小条码宽度和最大条码宽度。

超时退出时间

设置算法运行的最大允许时间，单位为 ms。该参数设置为 *0* 时，不限制模块运行时间；设置为其他值时，若算法运行时间超出参数设置值，则识别失败，模块不输出字符信息。

模块结果

*条码识别*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

码个数

int 型，代表识别到码的个数。

码状态

int 型，代表码是否可以识别。1 表示识别成功，不显示表示识别失败，会显示各个码的状态 。

编码信息

string 型，代表各个码蕴含的信息。

码类型

int 型，代表识别到码的类型。

码角度

float 型，代表码起点与终点连线后与水平方向的夹角，一维码顺时针为正方向，二维码逆时针为正方向。

PPM

float 型，一维码中代表成像后最细黑条或白空的像素数，二维码中代表最小模块边长占用的像素数 。

矩形框

box 型，代表识别到码的矩形框。

中心点

point 型，代表识别到码的矩形框中心点。

中心 X

float 型，代表识别到码的矩形框中心点的 X 坐标。

中心 Y

float 型，代表识别到码的矩形框中心点的 Y 坐标。

矩形宽度

float 型，代表识别到码的矩形框宽度。

矩形高度

float 型，代表识别到码的矩形框高度。

矩形角度

float 型，代表识别到码的矩形框角度，由码的点 0 和点 1 所连直线与水平线夹角构成，顺时针为正，逆时针为负。

说明

一般码有 4 个角点，按顺序分别为点 0、点 1、点 2、点 3，不同码制的顶点规则和标准有所差异，具体请以实际为准。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

调试信息

string 型，代表算法调试信息。

算法调试状态字

int 型，代表算法调试信息状态。

算法调试信息大小

int 型，代表算法调试信息大小。

算法调试信息地址

string 型，代表算法调试信息储存地址。

13.4.3 二维码识别

*二维码识别*模块用于识别目标图像中的二维码，将读取的二维码信息以字符串的形式输出。该模块支持 QR 码和 DataMatrix 码。该模块对图像要求较高，可高效准确识别多个二维码。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

二维码识别算法用于解析二维码信息。该模块中，算法首先定位二维码，修正二维码姿态并预处理图像，然后进行译码，并保证一定的容错率。

二维码介绍

二维码由固定种类的功能图案（定位图案、校正图案和参考网格）及不同数量的最小模块构成。每个最小模块用于表示 1bit 数据。

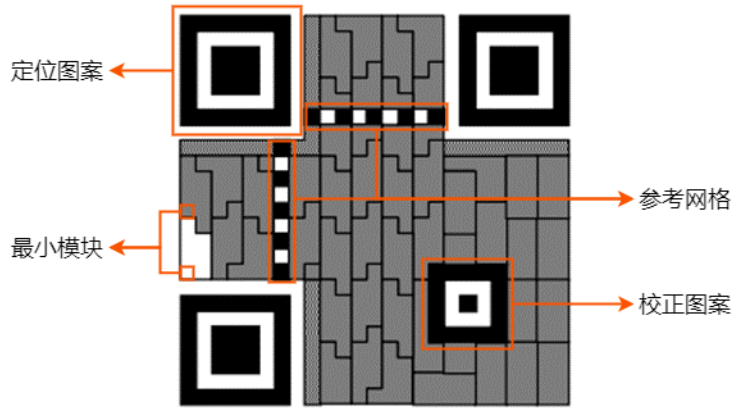


图 13-193 二维码介绍

算法工作步骤

该模块算法工作流如下图所示。



图 13-194 算法工作流程

1. 算法通过二维码纹理信息，定位二维码候选区域。
2. 算法计算二维码梯度，并局部旋正待识别二维码。
3. 算法定位二维码功能图形，并获取最小模块大小、二维码版本号等信息。
4. 算法对二维码建立采样网络。
5. 算法基于采样网络，提取二维码中每个最小模块的数据。
6. 算法依据二维码编码原理，获取二维码所含信息。

使用方法

前序模块

该模块无固定搭配使用模块。在流程中调用该模块时，前序模块可为 *图像源* 、 *仿射变换* 、 *形态学处理* 等模块。图像处理类模块可处理待读码图像，提高读码效率。

后序模块

后序模块无特殊要求，能处理该模块输出信息即可。例如，可以将二维码字符和位置信息输入至 *定位* 、 *通信* 等模块，用作后续处理。

参数配置

以下仅介绍该模块的运行参数的详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

QR 码/DataMatrix 码

控制模块能否识别对应类型的二维码。若您不确定待检测二维码的类型，建议开启全部项。

二维码个数

期望查找并输出的二维码最大数量，若实际查找到的个数小于该参数，则输出实际数量的二维码。

说明

有些场景中二维码个数不确定，若要识别所有出现的二维码，则该参数可设置为场景中二维码个数最大值。在部分应用中，背景纹理较复杂，当前参数值可以适当大于需要识别的二维码个数。

极性

包含任意、白底黑码和黑底白码三种形式，可根据实际情况进行选择。

边缘类型

包含连续型、离散型和兼容模式三种类型，如下图所示。左边表示连续型，右边表示离散型，兼容模式可兼容其他两种类型。



图 13-195 边缘类型

降采样倍数

降采样用于降低采样点数，提高图像处理速度。对于一帧分辨率为 N×M 的图像，若降采样系数设置为 k，则算法在该图像的每列每行中，每间隔 k 个点取 1 个点。该例子中，经过降采样后，图像分辨率降低为（N/k）×（M/k）。增大该参数值会降低待处理图像的分辨率，降低轮廓的精细程度，并提高处理效率。但是，当参数值过大时，会导致无法识别具体信息。

码宽范围

二维码所占的像素宽度，码宽范围包含最大最小码的像素宽度。

镜像模式

控制算法是否镜像采集图像，可选**任意**、**镜像**和**非镜像**三种模式。当采集图像为从镜像时（即与实际图像颠倒），可选择**镜像**。

QR 畸变

包括畸变和非畸变两种。当要识别的二维码打印在瓶体上或者类似物流的软包上有褶皱时，可选择**畸变**。

超时退出时间

设置算法运行的最大允许时间，单位为 ms。该参数设置为 *0* 时，不限制模块运行时间；设置为其他值时，若算法运行时间超出参数设置值，则识别失败，模块不输出字符信息。

应用模式

选择对应的应用场景。

普通模式

适用于一般场景。

专家模式

适用于二维码较难识别的场景。

极速模式

识别效率最高，但识别能力较弱，适用于单一类型的读码场景。选择该模式时，**极性**、**边缘类型**和 **DM 码类型**等参数需设置为场景中对应类型；若设置为**兼容模式**，则极速模式不生效。

DM 码类型

模块能够识别的 DM 码类型。可选**正方形**、**长方形**和**兼容模式**。选择**兼容模式**时，模块能够识别正方形和长方形的 DM 码。

模块结果

*二维码识别*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

码个数

int 型，代表识别到码的个数。

码状态

int 型，代表码是否可以识别。1 表示识别成功，不显示表示识别失败，会显示各个码的状态 。

编码信息

string 型，代表各个码蕴含的信息。

码角度

float 型，代表码起点与终点连线后与水平方向的夹角，一维码顺时针为正方向，二维码逆时针为正方向。

矩形框

box 型，代表识别到码的矩形框。

中心点

point 型，代表识别到码的矩形框中心点。

中心 X

float 型，代表识别到码的矩形框中心点的 X 坐标。

中心 Y

float 型，代表识别到码的矩形框中心点的 Y 坐标。

矩形宽度

float 型，代表识别到码的矩形框宽度。

矩形高度

float 型，代表识别到码的矩形框高度。

矩形角度

float 型，代表识别到码的矩形框角度，由码的点 0 和点 1 所连直线与水平线夹角构成，顺时针为正，逆时针为负。

说明

一般码有 4 个角点，按顺序分别为点 0、点 1、点 2、点 3，不同码制的顶点规则和标准有所差异，具体请以实际为准。

PPM

float 型，一维码中代表成像后最细黑条或白空的像素数，二维码中代表最小模块边长占用的像素数 。

评估版本

int 型，101 代表 DM 码，102 代表 QR 码。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

调试信息

string 型，代表算法调试信息。

算法调试状态字

int 型，代表算法调试信息状态。

算法调试信息大小

int 型，代表算法调试信息大小。

算法调试信息地址

string 型，代表算法调试信息储存地址。

13.4.4 DL 字符定位

*DL 字符定位*模块帮助您定位图像中的字符区域。该模块通过神经网络提取输入图像中的文本位置信息，可在文本背景复杂或者位置不固定场景中帮助您较为准确地定位字符位置。

使用场景

*DL 字符定位*通过深度学习模型提取输入图像中字符形态、轮廓和语义等信息。相较于传统的定位算法（例如 *轮廓匹配*、*Blob 分析* 等定位模块），基于深度学习模型的 *DL 字符定位*对场景要求更低、目标形态宽容度更高。同时，该模块还可用于定位旋转字符以及同时定位多行文本。

在获得字符定位信息后，该模块会对每一个定位结果（定位框）评分。这个评分即为置信度，用于表示定位框是字符框的可能性大小。置信度得分越高，则该定位框是文本区域的可能性越大。

使用方法

您可参照以下介绍，了解如何获取该模块所需的模型、可选前后序模块及如何在流程中使用该模块。

模型获取

该模块中已提供简易模型，可用于定位背景干扰少、字符对比度高等简单场景中的字符。

若该模块中自带的模型无法满足业务需求，您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您可通过该模块的**模型文件路径**加载模型，并开启**方案存模型**，将模型文件保存至方案中。

前后序模块

该模块的前后序模块详情如下。

前序模块

使用时，前序模块可为 *图像源* 。

该模块对字符尺寸占比有要求。您需确保字符高度和输入图像的长边的比值大于 2.5%。假设输入了一张图像分辨率为 800×600 的图像，字符高度需至少为 20 像素。

说明

若输入图像的字符占比不满足需求，您可通过 *轮廓匹配* 、 *Blob 分析* 、 *位置修正* 、 *仿射变换* 等模块，裁剪输入图像。

后序模块

后序常搭配 *DL 字符识别* 。后者订阅该模块输出的字符位置信息，并基于位置信息识别字符内容。

流程示例

如下图所示，*图像源*模块将图像信息输入至 *DL 字符定位*模块。该模块定位字符信息，并将定位信息输入至 *DL 字符识别*模块，用于识别字符信息。

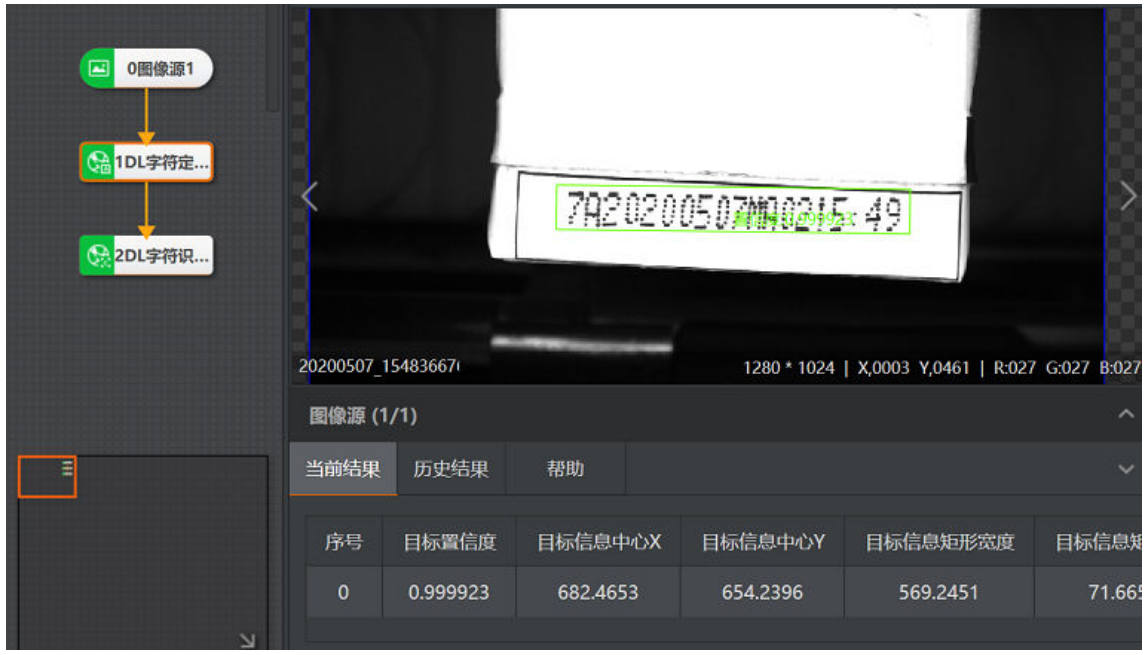


图 13-196 使用示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

设置该模块运行时使用的模型文件（.bin）。此处默认加载模块内部的对应的模型文件，您也可以加载自行训练的模型文件。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

获取模型 ROI

开启后，模块自动获取模型文件中的 ROI 信息。此时，您无需配置*基本参数*中的 **ROI 区域**。

模型 ROI 可在 VisionTrain 或 AI 训练平台训练模型时设置。设置后，训练的模型包含 ROI 信息。

按 ROI 裁图

开启时，模块仅预测 ROI 内图像，忽略 ROI 外信息。

关闭时，模块识别整张输入图像，但根据其他设置，如 **ROI 区域**，筛选过滤结果。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小分数

设置预测结果的最小可信度。若某预测框的可信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

高级参数

目标排序

设置模块输出预测结果的顺序。在需识别多目标的场景中，您可设置该参数，以确保输出的一致性。

- 按中心点 X/Y 坐标升序：按照预测框中心点的 X/Y 坐标值，从小到大排序预测结果。
- 按最小分数降序：按照预测结果的确信程度，从大到小排序预测结果。

缩放系数

设置预测框的大小。有效值范围为 0.01~2.00，默认值为 1.00。增大该参数值，会增大预测框的尺寸。

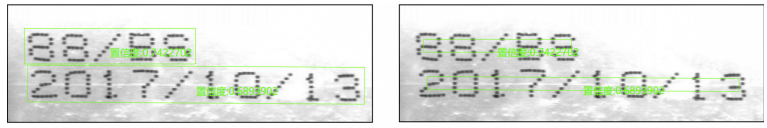


图 13-197 结果对比

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

过滤参数

文本对比

用于筛选特定文本。开启时，模块基于当前定位结果，再次通过文本对比进行过滤。适用于定位到的文本行差异较大时，筛选特定文本行。

开启后，可导入已有模型文件（.gall）或手动注册。您可参照以下步骤手动进行注册。

1. 单击*注册文本对比*，进入**注册图像**。

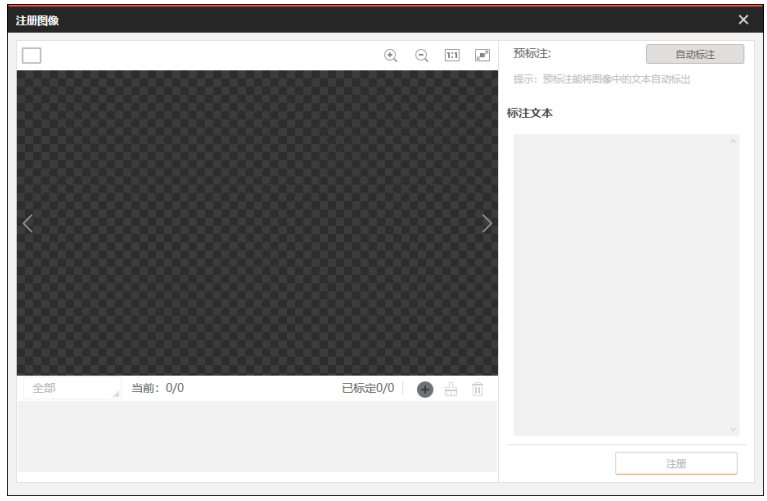


图 13-198 注册图像

2. 单击 ⊕ 添加图像。
3. 通过*自动标注*或 □，标注图像中想要识别的目标。
4. 单击*注册*。
5. 调整**结果文本数**和**最小相似度**。

结果文本数

设置过滤后的最大目标数量。

最小相似度

设置筛选结果和模型文件的最小相似程度。若相似程度小于该参数值，则模块过滤掉该结果。

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

字符角度使能

用于筛选预测结果。开启时，需设置**角度范围**。若预测结果框相对于水平方向的旋转超出设置的**角度范围**，则模块舍弃该预测结果。此处的旋转指：预测结果框相对于水平线进行顺时针旋转时，角度大于 0；逆时针旋转时，角度小于 0。

字符宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

字符高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

模块结果

*DL 字符定位*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

目标个数

int 型，代表检测区域内检测到的定位框个数。

目标置信度

float 型，代表每个定位框的置信度。

目标状态

int 型，代表每个定位框的状态。1 代表定位框正常，0 代表定位框异常。

目标信息矩形

目标信息中心

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.4.5 DL 字符识别

*DL 字符识别*模块依赖于深度学习模型，识别物体表面的字符信息。该模块可用于食药品包装、3C 电子、汽车零配件生产、半导体、物流等行业，实现生产日期、批号、产品编号等信息的自动识别。

模块原理

基于深度学习模型的字符识别可利用神经网络识别图像中的文本信息。该模块基于前序模块输出的文本定位信息，使用深度学习模型识别图像中的文字并输出识别结果。
模块的工作流程大致如下：

1. 提取图像语义特征。
   算子通过卷积操作提取图像不同层级的语义信息。
2. 提取文字序列特征。
   算子借助循环神经网络中的 LSTM 层，将上一步中提取到的语义特征转化为文字序列特征。
3. 输出结果。
   算子将上一步输出的文字序列转化为字符信息。

使用场景

*DL 字符识别*模块应用深度学习模型识别图像中的文本信息，可用于识别物体的表面字符。

说明

若需识别单字符，建议您调用 *DL 单字符检测*。

传统的 OCR 识别基于人工选择的特征，存在鲁棒性较弱，应用范围局限的问题。在字体形态存在变化、存在背景干扰或无固定特征的场景中，其识别效果将会变弱。
相较于 OCR 识别，基于深度学习模型的 *DL 字符识别*模块兼容性更好。在字符形态不一、字符与背景对比度低、图像背景存在干扰、或字符轻微黏连畸变但仍肉眼可辨等复杂场景中，该模块仍可保持良好的识别效果。

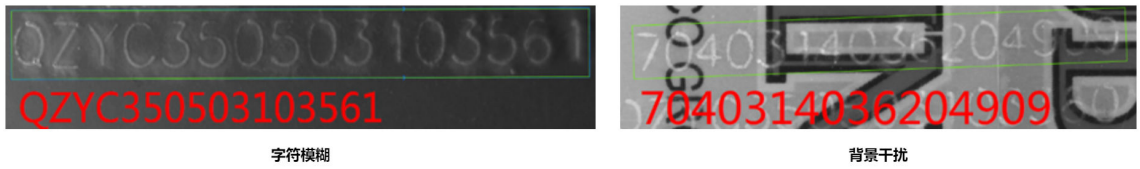


图 13-199 应用场景示例

针对弯曲字符场景，例如文字围绕环形呈现时，您可通过 *圆环展开* 等图像处理模块展开字符后，调用该模块识别字符。

说明

若无法展开字符，建议您调用 *DL 单字符检测*，分别识别每一个字符。

使用方法

您可参照以下介绍，了解如何获取该模块所需的模型、可选前后序模块及如何在流程中使用该模块。

说明

该类型算法模型较为轻量，一般场景中推荐使用 *DL 字符识别 C*。

模型获取

该模块中已提供简易模型，可用于定位背景干扰少、字符对比度高等简单场景中的字符。

若该模块中自带的模型无法满足业务需求，您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您可通过该模块的**模型文件路径**加载模型，并开启**方案存模型**，将模型文件保存至方案中。

说明

训练该模块使用的模型前，建议您先检查采集图像中文字是否容易分辨。若图像中的字符无法分辨，则您需调整成像质量，并确保实际使用的图像中文字较易分辨。
在运行流程时，也应确保输入至该模块的图像中文字较易分辨，否则，识别效果可能会下降。

前后序模块

该模块的前后序模块详情如下。

前序模块

前序模块需提供字符定位信息，辅助该模块识别字符信息。可选的模块分为两类：

- 传统方式：使用 *轮廓匹配*、*Blob 分析* 等模块匹配字符或图像中的固定特征。目标无序摆放场景中，您还需要通过 *位置修正* 模块修正字符定位框。

  说明

  使用该方式时，您需确保匹配的特征较为稳定，且该特征与需识别的字符的位置关系固定。

- 深度学习方式：使用 *DL 字符定位* 。

  说明

  使用 *DL 字符定位*时，需确保字符的高度与输入图像长边的比值大于 2.5%，以保证良好的定位效果。

后序模块

后序模块可为 *字符比较* 等模块，对比字符信息与标准字符信息。

流程示例

该示例演示了识别牛奶盒上的字符信息。

*图像源*模块将采集图像输入至 *DL 字符定位*。经过定位后，*DL 字符识别*识别定位区域文本信息，并将文本传递给*格式化*。经过格式化后，*字符比较*模块对比标准字符信息。



图 13-200 流程示例

模型性能调优

实际使用中，模型性能受到输入图像质量、模块设置等因素影响。此处列举了几种模型性能不佳的情况以及应对方法。您可以参照以下调整方式，优化模块内模型和您训练的模型的表现性能。

内置模型识别结果不佳

识别结果不佳通常表现为含误识别字符。

- 检查前序模块 *DL 字符定位* 输出结果中，预测结果（文本框）是否正常。预测文本框的大小影响信息识别，若过小，则 *DL 字符识别*模块识别的字符不完整，进而导致识别结果不佳。
- 若待识别文本包含易混淆字符，例如数字 0 和字母 O、数字 1 和大写字母 I、大写字母 V 和小写字母 v 等，您可通过 *参数配置* 中的 *字符校验*，针对字符串规则设置校验规则。
- 检查输入图像质量。待识别图像中，字符应清晰、完整。若不满足，可通过 *图像处理* 中的模块调整图像成像质量，或调整相机。
- 检查待识别图像中的字符字体。当前，内置模型可支持点阵等各类常规字体。部分字体，例如花体，内置模型可能不支持，此时，您应自行训练对应的模型，并将训练的模型导入到模块内使用。

自行训练的模型识别结果不佳

自行训练的模型性能取决于模型训练集质量和训练参数。您可以检查以下几个方面：

- 检查用于训练的图像数量。应至少使用 10~20 张图像。
- 检查用于训练的图像是否包含实际检测场景中可能出现的字符。假设训练模型时，使用了 15 张图像，并且实际检测场景中，会出现大小写英文字母及 0~9 数字。那么，该 15 张图像中，应包含上述所有字符。
- 检查模型训练集图像标注。若标注存在问题，则训练的模型会识别效果差。此时，您应修改对应标注并重新训练模型。
- 检查训练模型时，设置的训练参数是否合理。具体的参数释义，请参见 VisionTrain 或 AI 训练平台的用户手册。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 ***基本参数*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

模型参数

模型文件路径

设置该模块运行时使用的模型文件（.bin）。此处默认加载模块内部的对应的模型文件，您也可以加载自行训练的模型文件。

说明

该模块提供 4 个模型文件。如需更换加载的模型，您可将光标悬浮在 ，并在菜单中单击对应模型的 。您也可以进入.\Applications\Module(sp)\x64\Recognition 路径，获取对应模型文件。

- textrec_fast_en.bin：适用于识别英文字符，模型识别速度较快，精度略低。
- textrec_fast_ch.bin：适用于中文字符，模型识别速度较快，精度略低。
- textrec_accuracy_en.bin：适用于英文字符，模型识别精度高，速度略慢。
- textrec_accuracy_ch.bin：适用于中文字符，模型识别精度高，速度略慢。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

算法参数

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

高级参数

最小翻转分数

设置模块对检测区域倾斜度的容忍程度。当待检测区域的评分大于该值，模块在识别该区域字符信息时，首先进行翻转。

说明

当该参数值设置为 1 时，模块直接识别字符信息；设置为 0 时，模块在识别时默认翻转字符信息。

最大文本行数

设置单个 ROI 内，模块识别的可能结果数量最大值。

当单个 ROI 文本行包含易混淆的字符，例如大写字母 V 和小写字母 v 时，模块输出多个该文本行识别可能结果，且不同结果具有不同的置信度。此时，若可能结果的数量大于该参数值，模块按照结果的置信度大小，优先输出置信度高的若干个结果，并舍弃其余识别结果。

过滤参数

字符校验

用于设置不同 ROI 的校验规则。单击*字符校验*进入配置窗口，可设置该模块字符校验的规则。

单一校验

开启后，对所有 ROI 区域，模块均按照 **ROI 列表**中的第一个校验规则，过滤字符。

ROI 列表

对应模块**基本参数**中设置的 ROI 区域。您可对各个 ROI 区域，分别设置校验规则。

启用字符校验

用于设置单个 ROI 的激活状态。开启后，则模块开始应用该 ROI 的校验规则。

识别字符个数

设置模块能够识别的最大字符数量。若实际识别的字符数量大于该参数值，模块舍弃超出参数值的对应字符，仅输出参数值数量的目标。

设置后，需要对每个字符，设置对应的字符类型。

设置字符类型

设置每一个字符所属的字符类型，并按照设置的字符类型过滤输出结果。字符类型包括全部、数字（0~9）、大写字母（A~Z）、小写字母（a~z）、特殊字符、空格和自定义。

说明

- 特殊字符：其覆盖范围为自定义字符类型中的特殊字符。
- 自定义：可根据需求对已有字符库中的字符选择组合形式，完成设置后将鼠标放置在自定义设置框上可以显示已设定的组合形式。

模块结果

*DL 字符识别*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

检测区域状态

int 型，代表每个检测区域的状态，1 代表示模块功能正常，0 代表模块功能异常。

检测序号

string 型，代表每个检测区域的序号和文本框序号。例如 2-1，代表第二个检测区域的第一个文本框。

字符个数

int 型，代表每个文本框中检测到的字符串的字符个数。

字符串置信度

float 型，代表每个文本框中检测到的字符串的置信度。

字符信息

string 型，代表每个文本框中检测到的字符串的内容。

字符置信度

float 型，代表每个文本框中检测到的字符串中单个字符的置信度。

最佳字符串信息

string 型，代表置信度最高的字符串的内容。

最优字符个数

int 型，代表置信度最高的字符串的长度。

最优字符串置信度

float 型，代表置信度最高的字符串的置信度。

检测区域个数

float 型，代表检测区域个数。

模型标识

int 型，代表模型标识。

调试信息

string 型，代表算法调试信息。

算法调试状态字

int 型，代表算法调试信息状态。

算法调试信息大小

int 型，代表算法调试信息大小。

算法调试信息地址

string 型，代表算法调试信息储存地址。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.4.6 DL 单字符检测

*DL 单字符检测*模块可用于定位和识别输入图像中的单个字符，适用于图像中存在无法展开的弯曲字符、单个字符或不规则文本行场景。

使用场景

该模块与 *DL 字符识别* 的用途基本一致。但不同于 *DL 字符识别*，*DL 单字符检测*整合定位和识别功能，适用于环形文本、文本行不规整字符的定位和识别。

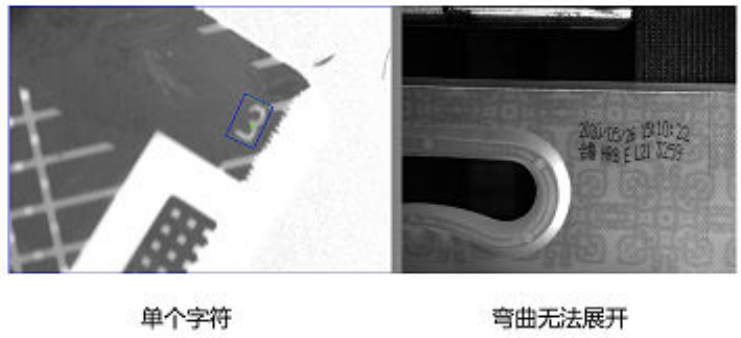


图 13-201 DL 单字符检测应用场景

在识别多个单字符后，您可将识别的单字符信息组合成文本行，从而识别完整的字符串。

使用方法

您可以参照如下内容，使用该模块实现单字符识别。

模型获取

该模块中已提供简易模型，可用于定位背景干扰少、字符对比度高等简单场景中的字符。

若该模块中自带的模型无法满足业务需求，您可使用或 AI 训练平台自行训练模型。训练完成并获取模型后，您可通过该模块的**模型文件路径**加载模型，并开启**方案存模型**，将模型文件保存至方案中。

说明

训练该模块使用的模型前，建议您先检查采集图像中文字是否容易分辨。若图像中的字符无法分辨，则您需调整成像质量，并确保实际使用的图像中文字较易分辨。
在运行流程时，也应确保输入至该模块的图像中文字较易分辨，否则，识别效果可能会下降。

前后序模块

该模块通常用于溯源信息和校验打印字符场景。根据应用场景需求的不同，常用搭配的前后序模块有所差异。

- 溯源信息：该场景中通常需要识别和保存字符。此时，您可搭配调用 ***图像源*** 模块作为前序模块。
- 校验打印字符：该场景需在识别字符后，比较识别到的字符和标准字符信息，以校验打印字符。此时，您可调用 ***字符比较*** 接收识别的文本信息，并比较标准字符内容。

通常来说，字符在图像中的占比越大，该模块的识别效果越好。因此，在***图像源***模块后，您还可以调用 ***Blob 分析*** 、 ***仿射变换*** 等模块，预处理输入图像。

说明

若您需裁剪输入图像和使用自行训练的模型，建议您基于裁剪后的图像进行标注和模型训练。

流程示例

下图流程演示了校验字符打印的流程。该流程中，***DL 单字符检测***模块识别***图像源***模块输入图像中的文本信息，并将文本输入至***字符比较***模块，实现字符校验。



图 13-202 DL 单字符检测示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

设置该模块运行时使用的模型文件（.bin）。此处默认加载模块内部的对应的模型文件，您也可以加载自行训练的模型文件。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

高级参数

目标排序

设置模块输出预测结果的顺序。

- 按中心点 X/Y 坐标升序：按照预测框中心点的 X/Y 坐标值，从小到大排序预测结果。
- 按置信度降序：按照预测结果的确信程度，从大到小排序预测结果。

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

过滤参数

字符过滤

用于过滤字符规则，筛选 ROI 内字符识别结果。

启用字符校验

开启时，需配置**识别字符个数**及**设置字符类型**。

识别字符个数

设置模块能够识别的最大字符数量。若实际识别的字符数量大于该参数值，模块舍弃超出参数值的对应字符，仅输出参数值数量的目标。

设置后，需要对每个字符，设置对应的字符类型。

设置字符类型

设置每一个字符所属的字符类型，并按照设置的字符类型过滤输出结果。字符类型包括全部、数字（0~9）、大写字母（A~Z）、小写字母（a~z）、特殊字符、空格和自定义。

说明

- 特殊字符：其覆盖范围为自定义字符类型中的特殊字符。
- 自定义：可根据需求对已有字符库中的字符选择组合形式，完成设置后将鼠标放置在自定义设置框上可以显示已设定的组合形式。

文本宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

文本高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

模块结果

*DL 单字符检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

文本行个数

int 型，检测区域内检测到的文本行个数，不超过最大查找个数。

文本行状态

int 型，1 代表检测到文本行，0 代表未检测到文本行，该参数输出个数与文本行个数保持一致 1 代表检测到文本行，0 代表未检测到文本行，该参数输出个数与文本行个数保持一致。

字符置信度

float 型，代表每个文本框中检测到的字符串中单个字符的置信度。

字符状态

int 型，代表每个文本行中单个字符的状态。1 代表检测到字符，0 代表未检测到字符。

字符信息

string 型，代表每个文本框中检测到的字符串的内容。

字符串信息

string 型，代表检出的单个字符组合成一个字符串。

字符个数

int 型，代表每个文本框中检测到的字符串的字符个数。

字符串置信度

float 型，代表每个文本框中检测到的字符串的置信度。

检测序号

string 型，代表每个检测区域的序号和文本框序号。例如 2-1，代表第二个检测区域的第一个文本框。

字符信息矩形

字符信息中心

字符信息中心 X

float 型，代表单字符框位置中心点的 X 坐标。

字符信息中心 Y

float 型，代表单字符框位置中心点的 Y 坐标。

字符信息矩形宽度

float 型，代表单字符框的宽度。

字符信息矩形高度

float 型，代表单字符框的高度。

字符信息矩形角度

float 型，代表单字符框的角度。

文本框信息

文本框中心点

文本框中心 X

float 型，代表检测到的文本行中心点 X 坐标。

文本框中心 Y

float 型，代表检测到的文本行中心点 Y 坐标。

文本框宽度

float 型，代表检测到的文本行宽度。

文本框高度

float 型，代表检测到的文本行高度。

文本框角度

float 型，代表检测到的文本行角度。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.4.7 字符识别

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

字符识别分为训练和识别两部分：

- 训练阶段：在设定的训练区域内，利用自适应分割先获取单个字符，并针对单个字符提取字符特征信息，同时根据输入的字符信息利用分类器进行训练，生成字符库。
- 识别阶段：直接设定检测区域，工具会自动进行分割和特征提取，然后将单个字符特征传入到训练库中计算距离度量，通过分类器输出识别结果。

使用方法

字符识别模块是通过字库训练达到识别图片中字符的作用，无需深度学习训练。该模块常用于字符种类较少、字符位置较稳定且成像清晰的识别场景。
该模块无固定搭配使用模块，前序模块可搭配几何变换、形态学处理等图像处理模块对字符效果优化，将字符清晰的图片输入至字符识别模块，字符识别模块会输出第一可能识别结果和候选识别结果，默认读取第一可能结果，成功提取字符后输出至后序模块。

参数配置

字库训练

单击可弹出字库训练界面，操作步骤如下：

1. 框选目标字符区域。
2. 单击***提取字符***，会出现已被红色框分割的字符，如下图所示。



图 13-203 字库训练

3. 单击*训练字符*，此处需输入对应的字符真值，并单击*添加至字符库*即可完成训练。若识别不准确可重复训练，如下图所示。



图 13-204 训练字符

字符过滤

单击可弹出字符过滤界面，操作步骤如下：

1. *启用字符过滤*可提高识别的准确性，此时在相应的位置只识别所设置的字符类型，如下图所示。



图 13-205 字符过滤

2. 自定义设置**识别字符个数**以及设置每个字符的类型，包括全部、数字、大写字母、小写字母、特殊字符、空格、自定义等几种类型，可以通过**自定义**来定义容易误读的字符，但前提是自定义字符应存在于字符库中。

字符极性

包含白底黑字和黑底白字两种类型。

字符宽度范围

设置字符的最小宽度和最大宽度。

宽度类型

包含可变类型和等宽类型两种类型。当字符宽度一致时，建议选择**等宽类型**；当字符宽度有差异，建议选择**可变类型**。

字符宽度范围

设置字符的最小高度和最大高度。

二值化系数

二值化阈值参数，范围为[0, 100]。

片段面积范围

单个字符片段的面积范围。

合格阈值

能够被识别字符的最小得分。

距离阈值

字符片段到文本基线的距离，大于该值则删除。

忽略边框

选择**是**则忽略与 ROI 区域粘连的字符，选择**否**则不忽略。

主方向范围

允许文本行倾斜的最大角度范围。

倾斜角范围

允许字符倾斜的最大角度范围。

字符最小间隙

两个字符间的最小横向间距。

行间最小间隙

多行字符间的最小间隙 。

最大宽高比

单个字符外接矩形的最大宽高比。

分类方法

该参数配合相似度类型使用，有距离最近、权重最高和频率最高三种方式。

字宽滤波使能

是否开启字符间字符宽度的滤波使能。

笔画宽度范围

单个笔画的宽度范围，在打开**宽度滤波使能**后才能生效。

相似度类型

可选择欧式距离和余弦距离，不同类型会影响其识别率。

模块结果

*字符识别*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

识别结果数量

int 型，代表识别的字符集合的个数。

字符个数

int 型，代表每个文本框中检测到的字符串的字符个数。

字符信息

string 型，代表每个文本框中检测到的字符串的内容。

候选字符

string 型，代表识别的得分次高的字符结果。

字符分数

float 型，代表识别的得分最高的字符的得分，数组，其长度等于字符个数 。

字符框

字符中心点

字符中心 X

float 型，代表识别的字符的中心 X 坐标 。

字符中心 Y

float 型，代表识别的字符的中心 Y 坐标 。

字符矩形宽度

float 型，代表识别的字符的宽度，数组，其长度等于字符个数。

字符矩形高度

float 型，代表识别的字符的高度，数组，其长度等于字符个数。

字符矩形角度

float 型，代表识别的字符的角度，数组，其长度等于字符个数。

字符矩形倾斜度

float 型，代表识别的字符的矩形倾斜角度，数组，其长度等于字符个数。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

第一字符

string 型，代表识别的得分最高的字符，数组，其长度等于字符个数，与字符信息一致、

第一字符得分

float 型，代表识别的得分最高的字符的得分，数组，其长度等于字符个数。

第二字符

string 型，代表识别的得分次高的字符，数组，其长度等于字符个数，与候选字符一致。

第二字符得分

float 型，代表识别的得分次高的字符的得分，数组，其长度等于字符个数。

置信度

float 型，代表识别的字符的置信度。数组，其长度等于字符个数。

13.4.8 ML 分类

*ML 分类*可帮助您预测输入图像的类别。该模块基于传统机器学习方式，通过提取的图像特征和人为标注信息训练分类模型，并使用该模型推理输入图像类别，从而实现类似人类视觉判读的效果。该模块广泛应用于物体识别和分拣。

模块原理

*ML 分类*模块基于输入图像的图像特征和内置的标签训练分类器模型进行训练，获取对应的推理模型。该模型能够推理图像类别，从而实现图像分类。

该模块的算法工作流程分为训练和推理阶段。

- 训练阶段：您需从待分类数据样本中选择部分有代表性的样本，并手动标记其类别。算子使用选择的特征提取器提取图像特征，并基于类别标签训练分类模型。模型包含特征类型及分类模型。
- 推理阶段：算子使用训练阶段中的模型，提取新数据的特征、预测其分类并最终输出新数据的类别信息。

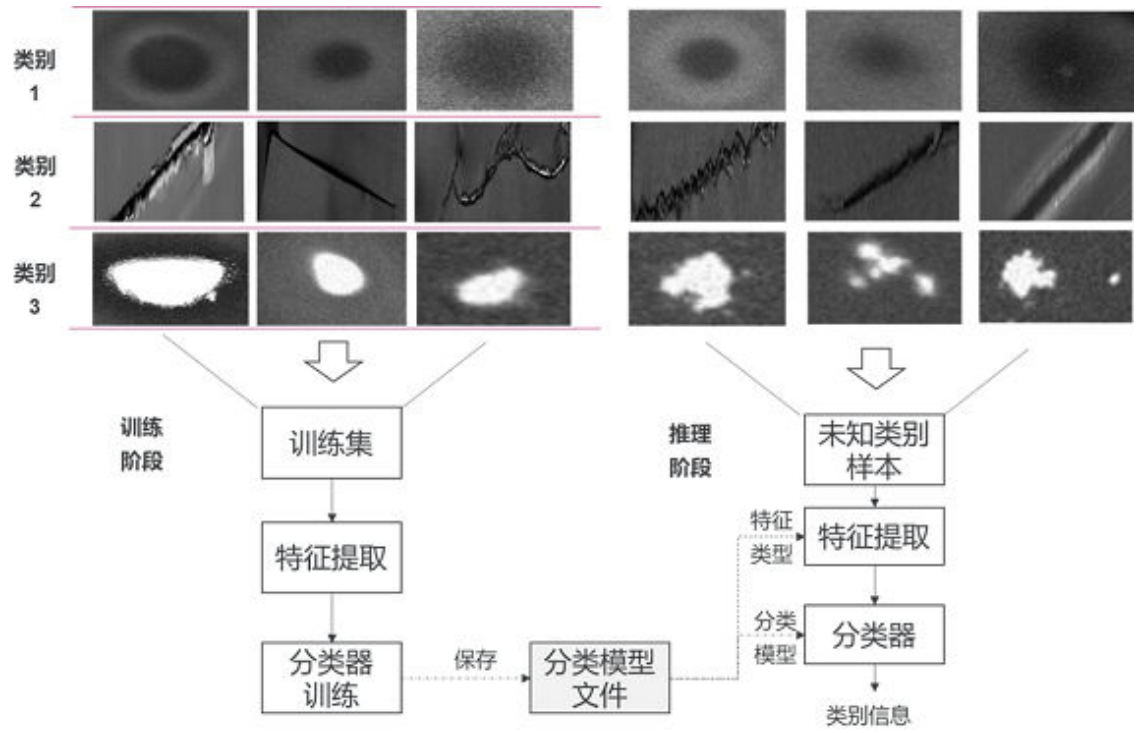


图 13-206 ML 分类算法流程

该模块功能与 *DL 分类* 模块类似，两者的差别在于 ML 分类基于机器学习实现，DL 分类基于深度学习实现。

使用方法

您可以参照如下内容，在流程中使用该模块实现图像分类。在流程中添加该模块后，您需首先手动分类部分图像，详情请参见 *分类训练* 。

前后序模块

该模块的常见前后序模块如下所示。

前序模块

该模块通常搭配搭配 *图像源* 模块。*图像源*模块向 *ML 分类*模块传输需分类的图像。

后序模块

该模块无指定后序模块。后序模块可接收和处理该模块输出的图像或分类信息即可。

主要配置步骤

1. 在流程中添加*图像源*和 *ML 分类*模块，并将 *ML 分类*模块连接至*图像源*模块。
2. 双击 *ML 分类*模块，配置相应参数。

3. 在 *分类训练* 页签，完成模型分类训练。
4. 执行流程进行分类。

示例方案

在**分类训练**中训练分类模型后，您可通过该模块分类场景中的目标。

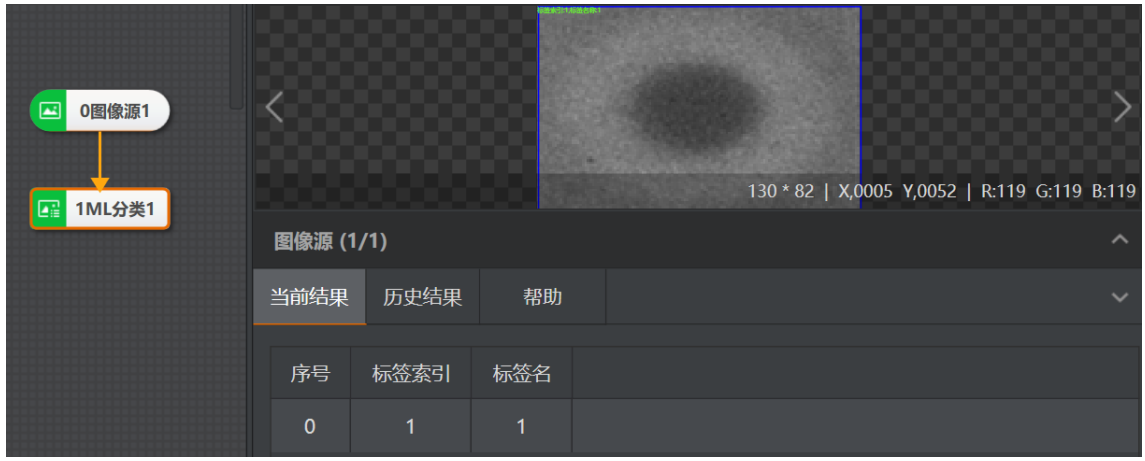


图 13-207 ML 分类执行结果

分类训练

分类训练指算子根据已分类的样本图，学习这些样本图的图像特征及对应类别，并建立分类模型的过程。在归类样本图过程中，您需准确地对具有代表性的样本图进行分类，以确保训练出的分类模型具备良好的分类性能。

若无可加载的模型文件，需先通过*分类训练*生成模型。

操作步骤

1. 在流程中双击 *ML 分类*模块。
2. 在 *ML 分类*对话框的*分类训练*页签，单击*训练*。
3. 添加图像类别。
   1) 在*分类训练*对话框的右上方，单击*添加*。
   2) 在*添加新类别*对话框，单击 ⊞ 。

      可通过多次单击 ⊞ ，一次性添加多个类别；也可分多次添加类别。
   3) 在*类别名称*下方的输入框内，输入图像类别的名称。
   4) 单击*确定*。

      添加成功后，会出现在*分类训练*对话框右上角的类别列表中。选中类别后，单击 🗑 图标，可进行删除操作。
4. 在*分类训练*对话框下方，单击 ⊞ 图标。添加需标记的图像。

可通过如下多种方式添加图像。

当前图像	添加该模块基本参数中输入源处订阅的图像。
缓存图像	可加载*缓存图像*模块缓存的图像。
导入本地图像	从本机选择图像添加，每次只能导入一张图像。
导入文件夹	从本机选择文件夹，将文件夹中的图像均添加。
按类别导入文件夹	可将本机文件夹中的图像按照某种类别直接导入。 **说明** 使用该功能时，应先选中图像需标记的类别。

5. 选中需标记类别的图像，再选中对应的类别，即可完成单张图像的类别标记。
   重复该步骤，完成多张图像的标记。标记的图像越多越好，也提高模块检测的正确性。

说明

标记图像至少应大于 4 张，类别应大于 2 种，每种类别的图像至少有 2 张图像，方可进行训练。

6. **可选操作**: 单击 可清空图像已标记的图像类别；单击 可清空已添加的图像。
7. 配置训练参数。

最大迭代次数

拟合算法的最大执行次数。

最小迭代误差

当训练器迭代的误差小于该数值时，则停止训练。

分类器类型

可选 SVM 分类器和 RF 分类器。

SVM 分类

即支持向量机分类器。其目标是确定一个分类超平面，从而将不同类别的数据分隔开。适用于训练样本较少，类内差异较小，类间差距较大的场景。

RF 分类器

即随机森林分类器。此为集成多个决策树进行分类的一种算法。其中决策树通过递归地划分训练数据集，将数据集划分成若干个子集，每个子集对应一个类别。适用于训练样本较多，类别较多的场景。相较于 SVM 分类器，可应用类内差异稍大一些的场景。

- 纹理特征参数：

纹理特征

开启该参数后，可设置图像的纹理特征。通过设置以下参数实现。

说明

纹理特征和**灰度特征**需至少开启其中一种。

HOG

HOG 特征是基于图像局部区域的梯度方向直方图统计信息。它对图像的边缘和纹理等特征敏感，可有效描述和提取图像中目标的形状和结构信息。适用于需要分类的每一类图像具有固定的几何形状或轮廓、类间形状差异较明显、且姿态固定的样本。

LBP

LBP 特征是一种用于描述图像局部纹理特征的算子。其具有灰度不变性和旋转不变性。适用于类内具有稳定的纹理分布、且类间的纹理具有明显差异的场景。

GLCM

即灰度共生矩阵，是一个描述具有某种空间位置关系的两个像素灰度的联合分布的矩阵。适用于类内具有固定的或相近的纹理分布、且类内纹理不考虑姿态的场景。

边缘密度

边缘密度特征是一种基于图像边缘像素点数量及其分布情况的图像特征。其可以描述图像中的边缘信息和结构特征。适用于类内的纹理分布密度较稳定、且类间存在明显差异的场景。

极坐标投影

将图像转换到极坐标下，并对转换后的图像在水平和垂直方向上进行投影，将投影得到的一维向量作为特征。适用于类内样本在水平和竖直方向上纹理分布具有一致性或相近、且类间分布有显著差异的场景。

- 灰度特征参数：

灰度特征

开启该参数后，可设置图像的灰度特征。通过设置以下参数实现。

说明

纹理特征和**灰度特征**需至少开启其中一种。

灰度范围

即图像像素灰度值的范围。适用于对具有固定的最大灰度值差异或者最小灰度值界限的不同类别样本。

灰度均值

即图像像素灰度值的平均值。适用于对具有固定的灰度平均值界限的不同类别样本。

灰度方差

即图像像素灰度值的方差。适用于对具有固定的灰度方差界限的不同类别样本。

灰度能量

即指图像像素灰度级别的平方和。描述的是图像中纹理信息的整体强度和均匀程度，能够反映图像的能量分布情况。适用类内纹理分布的强度和均匀程度相同或相近、且类间纹理分布有显著差异的场景，对纹理分布的姿态不敏感。

各向异性

是一种描述图像灰度分布的特征，反映图像在不同方向上的灰度变化程度。适用类内具有相同的灰度分布方向、且类间在相同方向上、灰度分布具有明显差异的场景

水平投影

表示将二维图像在水平方向上灰度值的求和，最终得到一个表示水平方向上灰度值的投影，可反映图像在水平方向上的能量分布情况。适用类内水平方向上灰度空间分布相近、且类间存在明显差异的场景。

垂直投影

表示将二维图像在垂直方向上进行灰度值的求和，最终得到一个表示垂直方向上灰度值的投影，可反映图像在垂直方向上的能量分布情况。适用类内垂直方向上灰度空间分布相近、且类间存在明显差异的场景

灰度直方图

反映一幅图像中各灰度级像素出现的频率与灰度级的关系。以灰度级为横坐标，频率为纵坐标，绘制的关系图像就是一幅灰度图像的直方图。适用于类内样本整体具有相同或相近灰度分布、且类间具有显著灰度差异的场景。

8. 单击*训练*进行分类训练，生成模型。

完成训练后，*ML 分类*对话框的*分类训练*页签显示已注册的图像类型和信息。通过右上角的 可将生成的模型以 bin 文件形式导出。同时，也支持导入已训练的模型。

后续处理

模块使用生成的模型结合设置的基本参数和运行参数完成检测。

参数配置

此处仅对该模块的运行参数进行介绍，基本参数结果显示各模块大同小异，此处不再赘述，具体参见 *基本参数* 和 *结果显示* 。

模型文件路径

选择之前训练生成的模型文件。

说明

- 若该模块已通过*分类训练*进行注册，则直接使用生成的模型。此处可不加载模型文件。
- 若既完成*分类训练*，也设置了模型文件路径，则优先使用加载的模型文件。

方案存模型

开启后，将模型数据保存到方案文件或流程文件中，跨机加载方案时不需要再输入模型文件路径。

模块结果

*ML 分类*模块的模块结果具体如下：

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

预测信息个数

int 型，代表获取的分类个数。

标签索引

int 型，代表获取的分类的索引，数组，其长度等于预测信息个数。

标签名

string 型，代表获取的分类的名称，数组，其长度等于预测信息个数。

13.5 缺陷检测

缺陷检测相关模块主要用于识别图像中呈现的零部件特征缺陷。

13.5.1 表面缺陷滤波

*表面缺陷滤波*模块适用于简单、无纹理背景的表面缺陷检测场景，可检测工件表面点状、线状、块状等不同形状的缺陷（划伤、破损、异物等），且不受材料表面亮度变化影响。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

基础工作原理

在对材料表面的缺陷进行检测时，缺陷区域往往呈现出与背景不同的灰度特征，导致表面灰度分布出现突变。表面缺陷滤波算法将该图像特征抽象为“异常灰度像素值在一定空间范围内构成的区域”，同时结合缺陷的空间与灰度分布特征，生成图像空域滤波器，对图像中的像素以及其邻域像素的分布进行分析。

下图示例为缺陷区域的一维信号波形。该图中，缺陷区域灰度值出现异于背景灰度分布的突变，且波形近似于高斯函数。

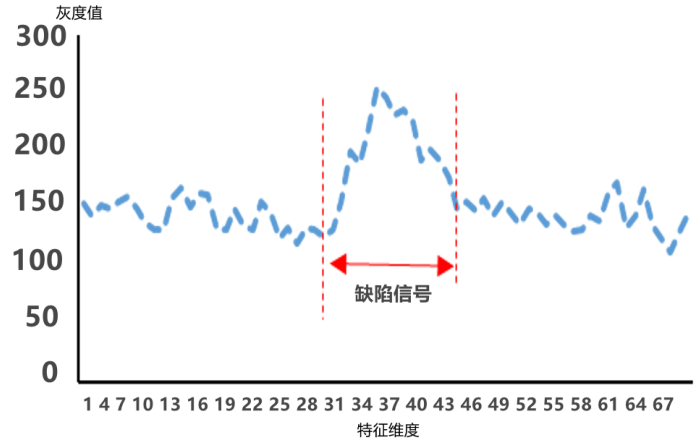


图 13-208 缺陷区域一维信号波

算法根据上图缺陷信号波形，构造一个与缺陷信号波形相同的高斯函数，并基于该函数在大于缺陷区域的滤波窗口内生成卷积核。

$$kernel = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

为了对不同方向的缺陷进行检测，将生成的卷积核进行旋转，得到不同方向的卷积核（示例如下）。



使用上述卷积核对缺陷图像进行滤波，并综合各个方向的滤波结果，得到最终的滤波响应结果。

$$F = \sum_{i=1}^{i=n} \omega_i f\left(\theta_i\right)$$

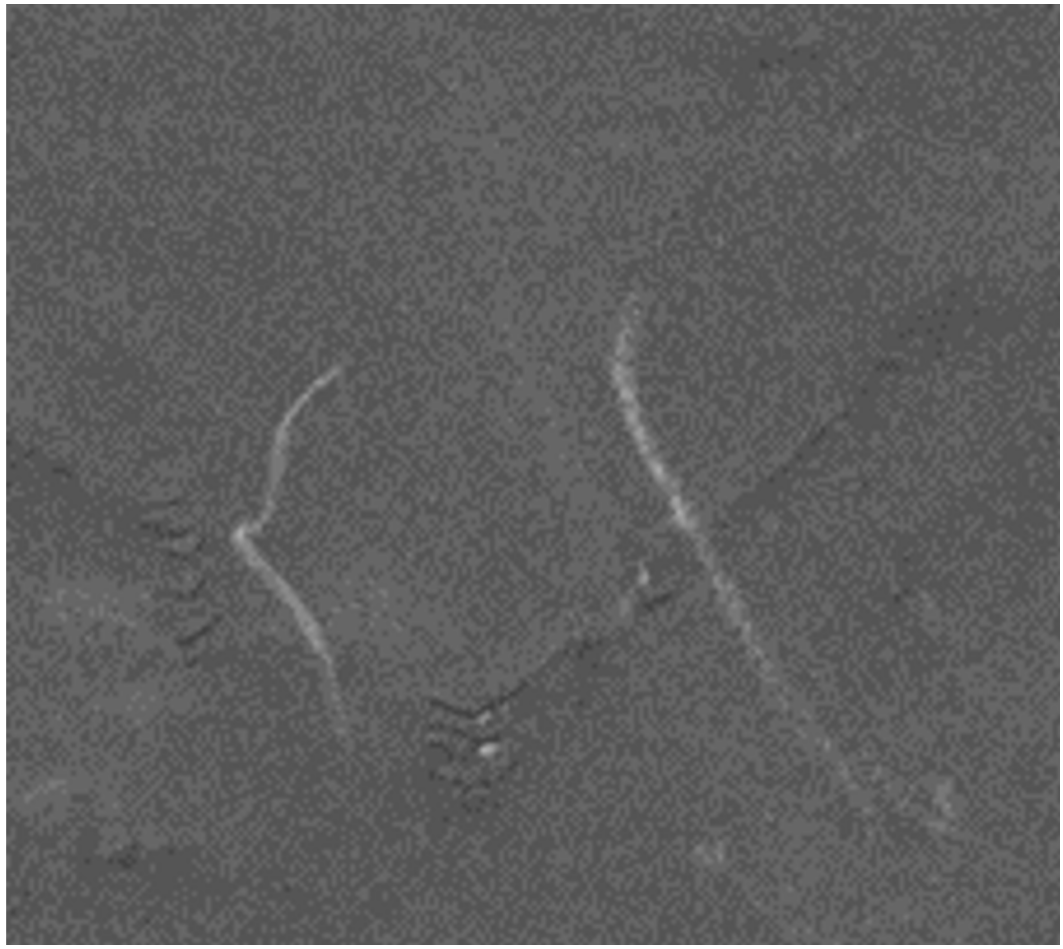


图 13-209 缺陷原图

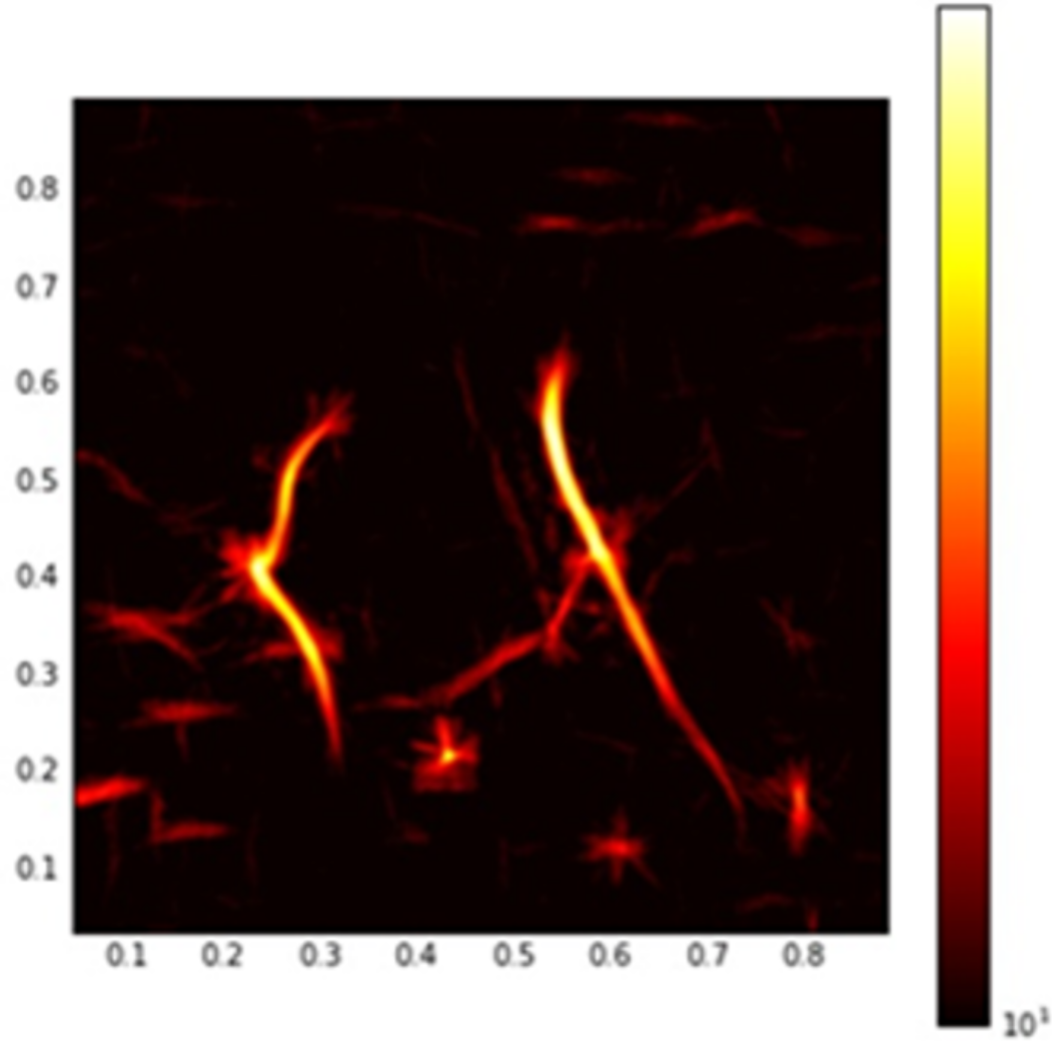


图 13-210 滤波响应结果

算法工作流程

基于上述基础工作原理，可将算法主要工作流程归纳如下：

1. 生成不同方向的卷积核。
2. 分别使用生成的各卷积核对图像进行滤波处理。
3. 综合各方向的滤波结果生成最终的缺陷滤波响应图。
4. 根据滤波响应值判断缺陷是否存在。如果存在，同时输出缺陷的严重程度。

使用方法

在流程中，*表面缺陷滤波*的前后序模块详情如下。

表 13-15 前后序模块

前后序模块	描述
前序模块	前序模块为 *图像源*，可为*表面缺陷滤波*提供图像输入
后序模块	后序模块为 *Blob 分析*，可处理*表面缺陷滤波*生成的滤波响应图，并筛选出符合条件的缺陷 blob

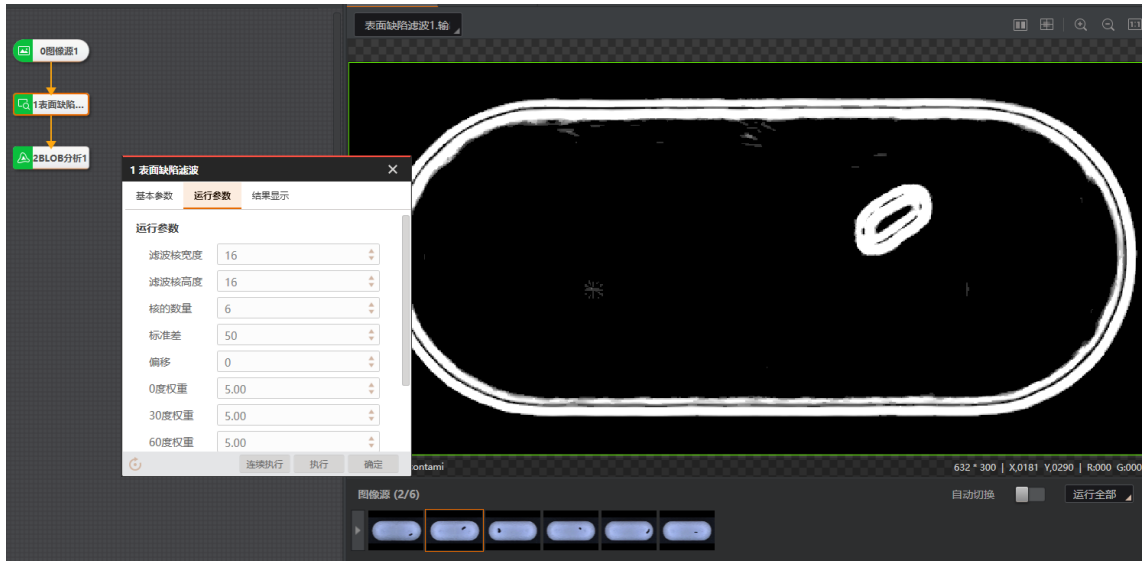


图 13-211 示例

参数配置

以下仅介绍该模块的运行参数详情。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

滤波模式

可选快速和高精度两种模式。

- 快速模式：算法耗时较低，但对指定方向缺陷的滤波响应抑制效果不精确 。
- 高精度模式：算法耗时较高，可通过设置权重精确抑制或增强指定方向缺陷的滤波响应。

卷积核宽度/高度

即卷积核尺寸，决定卷积核的视野大小。在缺陷检测时，模块对卷积核所覆盖的图像区域进行检测，因此需确保卷积核尺寸大于所要检测最大缺陷的宽度或高度的最小值。

设置原则：当设置的卷积核尺寸大于图像中缺陷尺寸的最小值时，针对相同大小的缺陷，卷积核尺寸越大，滤波效果越好，但算法耗时会显著增加。此时建议对需检测图像进行尺寸缩放以降低图像大小，但需确保需检测的缺陷信息无明显损失。

核的数量

即卷积核分析的角度颗粒度。软件根据设置的数值，在 0 到 180 度之间均匀分布卷积核方向。卷积核个数越多，算法检测的角度方向越多，能对更多方向的缺陷产生显著的滤波响应。

随着卷积核个数的增大，算法滤波效果提升并不明显，但算法耗时会相应增加。请根据实际情况设置，一般设置为 6 即可满足大部分需求。对于缺陷较为明显的样本，甚至可设置更少数量的卷积核个数。

标准差

即卷积核函数中的高斯标准差。标准差值越大，则波形的能量越发散，能对越大范围的信号产生响应，能检测更多缺陷特征较为微弱的区域；此外，对噪声信号的响应也会越明显。

设置该参数时，需根据缺陷信号的强度和表面粗糙度综合确定。

- 检测的缺陷越微弱，则该参数需设置越大的取值，使卷积核能对更大范围的信号产生滤波响应。
- 检测的表面越粗糙，则该参数需设置越小的取值，避免噪声核背景的干扰。检测表面粗糙度与可检测出的缺陷微弱程度相关，表面越平滑则能检测缺陷信号的微弱程度越低。

偏移

在整体滤波响应的结果中减掉该参数值。其作用为在调整波长和标准差等参数后，整体调整滤波响应的范围。

- 当整体滤波响应强度偏大，使很多噪声和背景产生干扰时，通过增大该参数可使整体响应值减小，直到满足将噪声的偏移降低到 0 为止。
- 当整体滤波响应偏小而又不存在过多的噪声，则适当减小该参数，可使滤波响应值朝着增大的方向偏移。

0/30/60/90/120/150 度权重

滤波方向权重参数，包含 0 度、30 度、60 度、90 度、120 度和 150 度这 6 个方向的滤波响应权重。

当卷积核的个数不等于 6 时，其他方向的滤波响应权重由这 6 个方向的权重自适应计算得到。通过改变不同方向的权重，可控制某个方向的滤波结果对整体滤波响应值的作用。取值范围为-10 ~ +10，值越大，则该方向的滤波响应在进行结果融合时，所起到的作用越强。

模块结果

*表面缺陷滤波*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.5.2 字符缺陷检测

可通过*字符缺陷检测*模块准确识别字符缺陷，包括字符脏污、缺失、差异等缺陷。该模块适用于检测各类部件的字符缺陷，如半导体芯片上雕刻字符缺陷和包装行业印刷字符缺陷。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *字符模型*
- *模块结果*

模块原理

*字符缺陷检测*可将目标图像与标准图像进行比对，从而进行字符的缺陷检测，分为建模和检测两个流程。

建模

建模只需要注册 OK 样本，并保存 OK 样本中有效字符的特征信息。
整体流程：

1. 输入待检测图像。
2. 设定定位基准，用于对待检测图像粗定位。
3. 进行字符分割，获得文本行/单词/字符区域。
4. 对分割的字符区域进行建模，获得字符的特征信息。
5. 进行模型训练，将所有 OK 样本的字符特征信息保存到模型中。

检测

整体流程：

1. 输入待检测图像以及建模得到的模型。
2. 对输入的待检测图像进行模板匹配。
3. 结合模板匹配定位结果、建模时的基准位置及检测区域，进行位置修正，获得当前检测图像的检测区域。
4. 进行字符匹配，获得与建模时对应的文本行/单词/字符。
5. 将上一步匹配到的文本行/单词/字符与模型中的文本行/单词/字符进行特征比对，计算精确的缺陷位置。

使用方法

前序后模块

在流程中，*字符缺陷检测*模块的前序模块为 *图像源* 和 *高精度匹配* 。*图像源*为*字符缺陷检测*提供图像输入， *高精度匹配* 为*字符缺陷检测*提供可继承的特征模板。



图 13-212 应用示例

主要配置步骤

在流程中调用*字符缺陷检测*模块后，该模块的主要配置步骤如下。

1. 执行一次流程，使*字符缺陷检测*获取*图像源*输入的图像。
2. 双击流程中的*字符缺陷检测*，打开该模块的配置窗口，依次配置 *基本参数* 、 *字符模型* 、 *运行参数* 和 *结果显示* 。
3. 单击*执行*，查看该模块的运行结果。

字符模型

*字符缺陷检测*模块需将目标图像与基准图像比对，判断目标图像的字符是否存在缺陷。创建字符模型即创建上述基准图像的过程。

前提条件

已执行一次流程，使*字符缺陷检测*模块获取到*图像源*输入的图像。

请参照如下步骤创建并配置字符模型。

操作步骤

1. 在*字符缺陷检测*模块的配置窗口，选择*字符模型*页签。
2. 单击*模型训练*打开*字符训练模型*窗口。
3. 设置基准，创建定位模型。
   - 选项 1：在该窗口左上角选择**直接创建**（默认为该选项），并从工具栏选择任意掩膜绘制工具，在当前图像上绘制待检测字符轮廓。单击 生成粗定位模型。
   - 选项 2：单击*直接创建*处，选择**继承**，并在*模型选择*下拉列表选择 *高精度匹配* 模块生成的模型，如下图所示。



图 13-213 模型继承

4. 完成相关参数的设置。

模型参数

选择**直接创建**时，需设置该参数。

尺度模式

设置**速度尺度**的配置模式，可选**自动**或**手动**模式。如果**自动**模式能满足特征提取需求则无需调节；如果不能满足要求，可切换为**手动**模式并手动调整**速度尺度**。

速度尺度

设置特征匹配速度。该值越大表示图像金字塔顶层的图像越小，相应提取的边缘点就越稀疏，进而加快特征匹配速度，但是过大的取值可能会导致漏检。取值范围为 1~20。

特征尺度

表示图像金字塔底层的大小，调整该值可调节匹配算法的抗形变能力。“抗形变能力”指算法在匹配目标存在一定形变的情况下也能匹配到目标的能力。该值越大表示图像金字塔底层的搜索图像越小，提取的特征点越稀疏，但抗形变能力越强，匹配速度越快。取值过大可能导致误检，定位精度也会有一定程度下降。

说明

该参数取值不能大于**速度尺度**取值。

阈值模式

可选自动以及手动两种模式 。

自动：根据目标字符自动决定阈值参数，自动适应。

手动：以定的阈值作为查找的阈值参数。

对比度阈值

阈值模式设置为**手动**时生效，查找的最低对比度数值。

搜索参数

选择**直接创建**和**继承**时，均需设置该参数。

绘制 ROI

默认为全屏，此时对整个图像进行粗定位检测。启用后可在图像上绘制 ROI 区域，此时仅对 ROI 区域内进行粗定位检测。

最小匹配分数

匹配分数指特征模型与搜索图像中目标的相似程度，即相似度阈值。目标与模型的相似度达到该阈值时才会被搜索到。最大取值为 1，表示完全契合，默认为 0.5。

角度范围

匹配目标相对于已创建模型的角度变化范围。如需搜索有旋转变化的目标，则需相应设置，默认范围-180°~ 180°。

匹配极性

极性表示特征图形到背景的颜色过渡情况（由黑到白或由白到黑）。查找目标的边缘极性和特征模型的极性不一致但仍要保证目标被查找到时，则需将该参数设置成不考虑极性；如不需要，可设置成考虑极性，缩短查找时间。

5. 单击 *下一步*，设定检测目标。
   1) 单击 ，并在图像上拖动光标绘制检测区域。
   2) 根据实际需求设置字符分割相关参数，定义该模块如何提取字符。

最小字符面积

设置面积阈值，查找面积大于设置数值的目标字符。

最小字符宽度

设置字符宽度的阈值，查找宽度大于设置数值的目标字符。

二值化尺度

设置的数值越低，可提取的二值化点越少，但也越可靠，请根据实际需求设置。

二值化窗口大小

设置二值化点的领域大小。

二值化阈值

- **字符极性**选择黑底白字时，若字符的灰度值高于设置的数值，则二值图中字符显示为白色，其余均为黑色。
- **字符极性**选择白底黑字时，若字符的灰度值低于设置的数值，则二值图中字符显示为黑字，其余均为白色。

显示二值图

开启以后会进行图像二值化，背景图像和字符灰度差更明显。

字符极性

可选黑底白字或白底黑字。

字符分割方式

可选文本行分割、单词分割和字符分割。

6. 单击 *下一步*，设置精定位。
   1) 若上一步中提取的字符效果不理想，可重新绘制效果不理想的区域并设置精定位相关参数。

缩放模式

可选取手动和自动两种模式，推荐使用自动模式

精度金字塔尺度

表示提取特征颗粒的精细程度，当精细尺度取值为 1 时精细程度最大，取边缘点数量最多，精度最高

速度金字塔尺度

数值越大抽取边缘点就越稀疏，但会加快匹配速度，取值范围 1~20。

阈值模式

可选自动以及手动两种模式 。

自动：根据目标字符自动决定阈值参数，自动适应。

手动：以定的阈值作为查找的阈值参数。

模型低阈值

阈值模式设置为**手动**时生效，查找的最低阈值参数。

2) 根据实际需求设置训练参数并单击*训练模型*。

最小匹配分数

匹配分数指特征模型与搜索图像中目标的相似程度，即相似度阈值。目标与模型相似度达到该阈值时才会被搜索到。最大取值为 1，表示完全契合，默认为 0.5。

角度范围

匹配目标相对于已创建模型的角度变化范围。如需搜索有旋转变化的目标，则需相应设置，默认范围-180°~ 180°。

阈值类型

自动阈值：根据目标图像自适应阈值参数。

模型阈值：以模型的对比度阈值作为匹配阶段的阈值参数。

手动阈值：以用户设定的阈值作为查找的阈值参数。

边缘阈值

阈值类型选择**手动阈值**时，需要设置。

位置纠正

开启后，如果检测的字符位置有所调整，只要调整范围在宽度和高度方向容忍的尺度范围内，仍可成功提取字符。

宽度方向容忍
高度方向容忍

宽度/高度方向允许偏离模型的尺度，开启*位置纠正*时需设置。

7. 单击*下一步*，设置检测区域掩膜（即无需识别的区域）。
   1) 单击 ，启用掩膜绘制功能。
   2) 在检测区域内拖动光标，绘制无需识别字符的区域。

8. 单击 *下一步*，批量对图像进行模型训练。
   1) 加载多张建模图像进行统计训练。
   2) 选中每张图像时会进行模型训练，根据提示信息对每张图像设置是否统计。合格图像标记为*统计当前图像*，不合格图像标记为*不统计当前图像*。



图 13-214 统计训练

9. 单击*完成*。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 字符模板相关参数详情，请参见 *字符模型*
- 结果显示参数详情，请参见 *结果显示* 。

- 字符检测参数详情如下。

  缺陷类型

共有三种类型，可选亮缺陷、暗缺陷、亮暗缺陷。

亮/暗缺陷阈值

亮/暗缺陷的最小灰度值。

亮/暗缺陷尺度

亮/暗缺陷的最小尺度。

边缘容忍度

取值越大对缺陷容忍程度越高。

面积大小阈值

检测图像与高低阈值图像比较得到差异二值图，二值图中大于面积阈值图的 blob 认定为缺陷。

- 字符精定位参数详情如下。

最小匹配分数

匹配分数指特征模板与搜索图像中目标的相似程度，即相似度阈值。目标与模板的相似度达到该阈值时才会被搜索到。最大取值为 1，表示完全契合，默认为 0.5。

角度范围

表示待匹配目标相对于已创建模板的角度变化范围。如需搜索存在旋转变化的目标，则需设置该参数。

X/Y 尺度范围

表示待匹配目标相对于已创建模板的缩放尺度变化范围。如需搜索在图像中存在大小缩放的目标，则需配置该参数。

位置纠正

开启后，如果检测的字符位置有所调整，只要调整幅度在宽度和高度方向容忍的尺度范围内，仍可成功提取字符。

宽度/高度方向容忍

宽度/高度方向允许偏离模板的尺度，开启**位置纠正**时需设置。

阈值类型

选择边缘对比度阈值的类型。该模块仅输出边缘对比度高于该阈值的轮廓。

自动阈值：根据目标图像自适应边缘对比度阈值。

模板阈值：以模板的边缘对比度阈值作为匹配阶段的阈值

手动阈值：以手动设置的**边缘阈值**（即边缘对比度阈值）作为匹配阶段的阈值。

- 字符粗定位参数性情如下。

最小匹配分数

同上文**最小匹配分数**。

匹配极性

极性表示特征图形到背景的颜色过渡情况(由黑到白或由白到黑)。查找目标的边缘极性和特征模板的极性不一致但仍要保证目标被查找到时，则需将该参数设置成**不考虑极性**；如不需要，可设置成**考虑极性**，缩短查找时间。

角度范围

同上文**角度范围**。

X/Y 尺度范围

同上文 **X/Y 尺度范围**。

阈值类型

同上文**阈值类型**。

模块结果

*字符缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

字符状态

int 型，代表每个文本行中单个字符的状态。1 代表检测到字符，0 代表未检测到字符。

字符框个数

int 型，代表待检测的字符框个数。

字符框

字符框中心点

字符框中心 X

float 型，代表字符框中心点的 X 轴坐标。

字符框中心 Y

float 型，代表字符框中心点的 Y 轴坐标。

字符框宽度

float 型，代表字符框的宽度。

字符框高度

float 型，代表字符框的高度。

字符框角度

float 型，代表字符框较长边相对于 X 轴正方向的旋转角度，顺时针为正，逆时针为负。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

缺陷个数

int 型，代表检测到的缺陷个数。

缺陷框

box 型，代表缺陷框

缺陷框中心点

point 型，代表缺陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷图像

缺陷图像数据

image 型，代表完成缺陷检测后以二值图输出的缺陷图像。

缺陷图像宽度

int 型，代表输出图像的宽度，一般单位为像素。

缺陷图像高度

int 型，代表输出图像的高度，一般单位为像素。

缺陷图像像素格式

int 型，代表输出图像的像素格式。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.5.3 异常检测

*异常检测*模块可通过边缘特征或灰度特征检测物料异常情况，并输出异常缺陷的位置信息。

本节内容包含：

- *模块原理*
- *使用方法*
- *缺陷模型*
  - *直接创建缺陷模型*
  - *继承缺陷模型*
- *参数配置*
- *模块结果*

模块原理

图像异常检测算子的工作流程分为注册阶段和检测阶段两大阶段。注册阶段，该算子基于合格样本提取特征，并构造特征模型。在之后的检测阶段，该算子将目标图像与合格样本图像进行比较：

- 如果目标图像存在与合格样本图像不同的像素区域，该算子将根据匹配程度给出匹配分数，并将匹配分数与预设的分数阈值进行比较。
- 如果目标图像的匹配分数低于阈值，则判定图像为存在缺陷。

该算子的整体工作流程见下图。

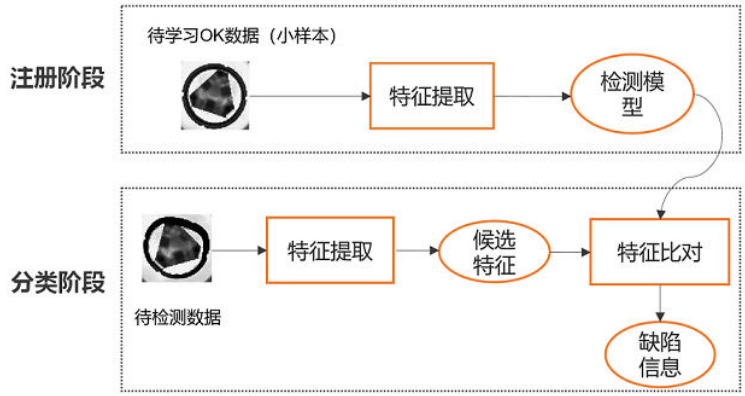


图 13-215 图像异常检测算子原理

说明

上图中，矩形元素表示算子的执行动作，椭圆形元素表示动作执行后算子生成的结果对象。

注册阶段

算子在注册阶段基于您注册的合格样本构造检测模型。该阶段算子的主要工作流程如下：

1. 从合格样本中提取图像特征，获取特性向量。
2. 基于特性向量构造检测模型，描述 OK 图像的特征分布。

分类阶段

算子在分类阶段，基于“待检测数据特征与模型特征的相似度”，将检测数据标为 NG 或 OK。该阶段算子的主要工作流程如下：

1. 对待检测数据按照与注册时相同的方式提取候选特征。
2. 将候选特征与模型的特征进行比对。

3. 根据两者的相似度，输出相似度得分。
4. - 若待检测数据中存在相似度得分低于指定阈值的区域，则将该区域判定为缺陷区域，同时将该检测数据标为 NG 图并输出。
   - 若待检测数据中不存在相似度得分低于指定阈值的区域，则将该检测数据标为 OK 图并输出。

使用方法

前后序模块

在流程中，该模块的前序模块一般为*图像源*。由于该模块的算法工作流程中存在模板匹配过程，所以无需进行粗定位。

主要配置步骤

在流程中调用该模块后，该模块的主要配置步骤如下：

1. 配置该模块的基本参数，详情参见 *基本参数* 。
2. 配置 *直接创建缺陷模型* 。
3. 配置该模块的 *运行参数* 。
4. 配置该模块的 *结果显示* 。
5. 执行或连续执行该模块，查看该模块的输出结果。

应用示例

下图为通过*异常检测*模块检测部件表面结构异常的示例。

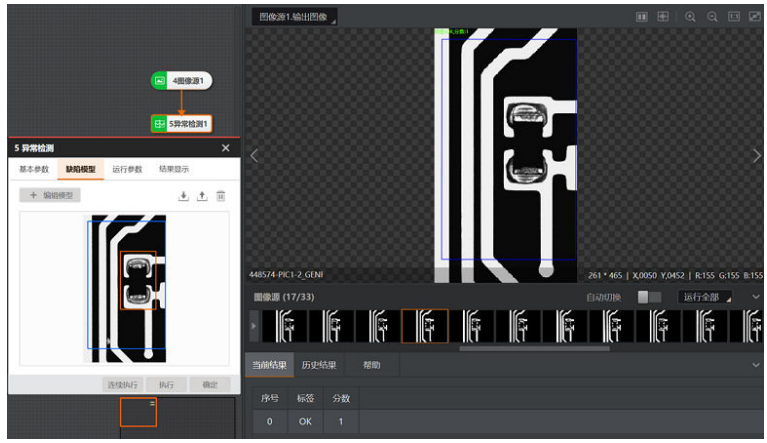
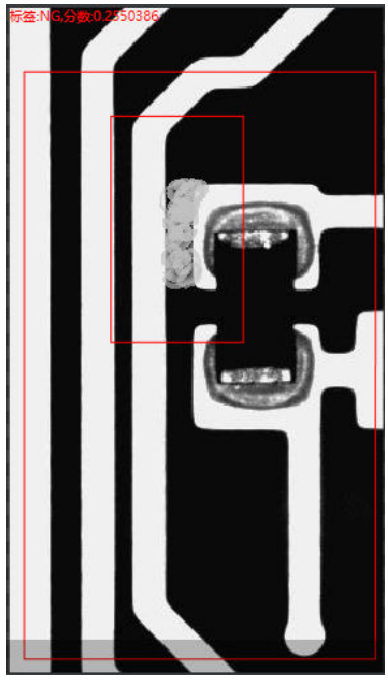
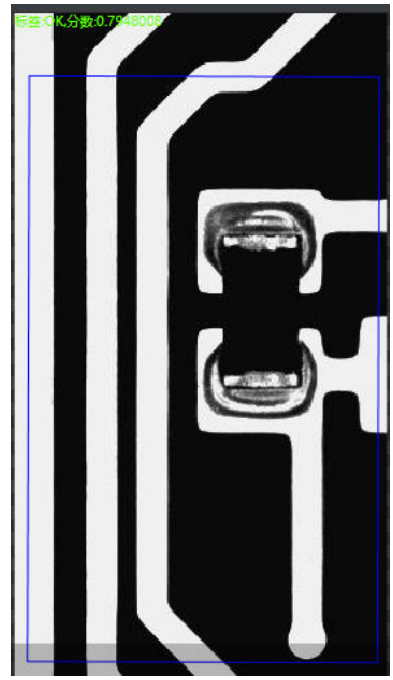


图 13-216 应用示例

如下左图和右图分别为异常检测结果异常和正常的示例。

NG	OK
	

缺陷模型

*异常检测*模块需将目标图像与合格样本作比对，判断目标图像是否存在缺陷。创建缺陷模型的过程，即“注册上述合格样本，并基于合格样本进行算法训练”的过程。

缺陷模型分为 *直接创建* 和 *继承* 两种方式。

直接创建缺陷模型

选择**直接创建**时，需先创建轮廓匹配模型，再修正检测区域，最后进行训练。

前提条件

已执行一次流程，使*异常检测*模块获取到*图像源*输入的图像。

请参照如下步骤创建并配置缺陷模型。

操作步骤

1. 在*异常检测*模块的配置窗口，选择*缺陷模型*页签。
2. 选择**直接创建**并单击*+ 新建模型*，打开*异常检测*窗口。

该窗口默认显示当前*图像源*模块采集的图像。如需基于其他图像进行模板配置，可单击*选择其他图像*。

3. **可选操作**: 设计基准，建立定位模型。

说明

建立定位模型可避免因为位置偏差导致的误检。

1) 在图像上特征明显的区域绘制掩膜。算法将在掩膜区域内提取图像特征。

说明

具体如何绘制掩膜区域参见 *模块中模型配置的掩膜区域* 章节。

2) 进行如下可选操作。

可选操作	描述
删除掩膜	右击掩膜，并在弹出的右键菜单上单击*删除*。
复制掩膜	右击掩膜，并在弹出的右键菜单上单击*复制*。
调整掩膜位置	单击选中掩膜，并拖拽调整其位置。
选择匹配中心	单击 ，并在图像上的合适位置单击，将其设置为模板的匹配中心。
移动图像	单击 后，将光标移动至图像上并拖拽图像进行移动。
缩放图像	单击 / 缩放图像；或将光标置于图像区域，并滚动鼠标滚轴缩放图像。
清空掩膜	单击 清空掩膜。
撤销上一步操作	单击 撤销上一步操作。
恢复至撤销前	单击 取消撤销，恢复至撤销前的状态。
图像自适应	单击 将图像大小设置为自适应模式。
全屏显示	单击 将图像全屏显示。

3) 单击 生成模板。

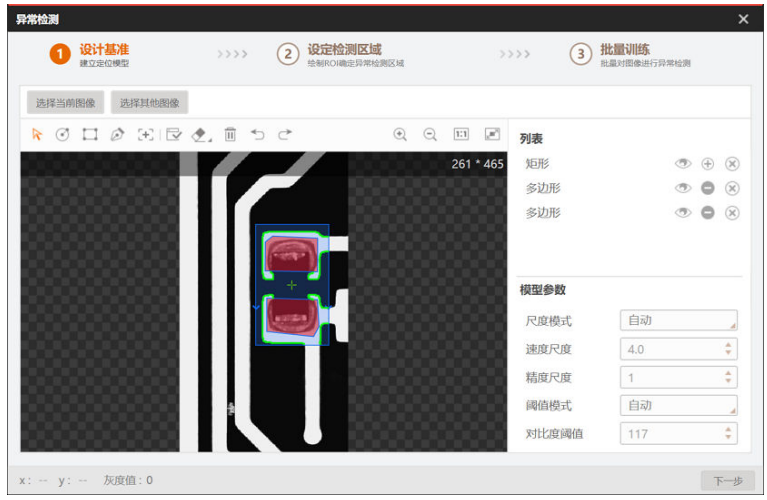


图 13-217 设定基准

4) **可选操作**: 如果模型中提取了不必要的轮廓点，单击 ，并将光标悬浮至这些轮廓点将其擦除。

可单击 右下角的三角形图标，并在弹出的浮窗滑动滑块调整“橡皮擦”的大小。

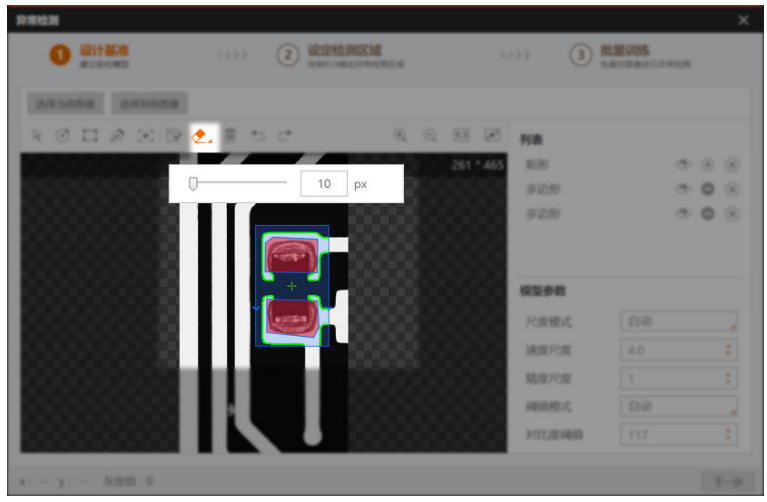


图 13-218 调整“橡皮擦”大小

5) **可选操作**: 在掩膜列表进行如下可选操作。

可选操作	描述
隐藏/显示掩膜	单击 / 隐藏或显示掩膜。
屏蔽掩膜	单击 ，再单击 重新生成模板，可屏蔽该掩膜中的特征点提取。
删除掩膜	单击 删除对应的掩膜。

6) **可选操作**: 如果生成的定位模型无法满足需求，调整模型参数取值并重新生成模型，直至模型满足需求。

尺度模式、**速度尺度**、**特征尺度**、**阈值模式**和**边缘阈值**这些模型参数详情请参见*轮廓匹配*模块的 *特征模板* 。其中**特征尺度**和*轮廓匹配*模型参数中的**特征尺度**含义相同。

4. 单击 *下一步*，设定检测区域，可以是整张图像，也可以是图像中的部分区域。

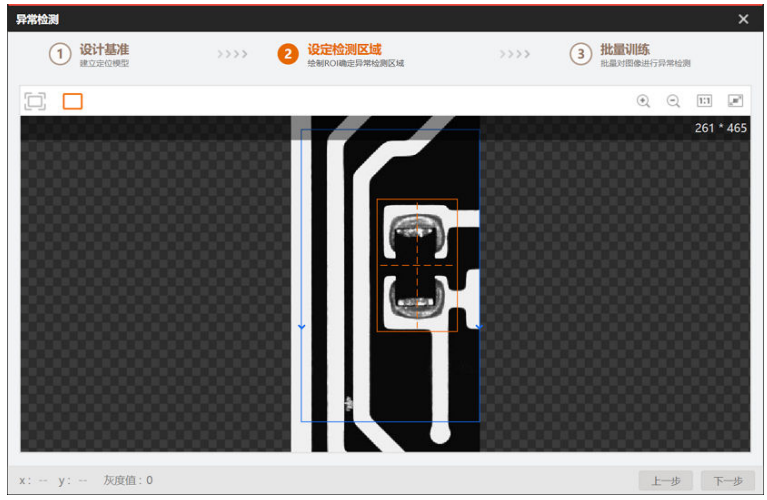


图 13-219 设定检测区域

5. 单击 *下一步*，对图像进行批量训练。

1) 单击 并在弹出菜单上按需选择操作项，加载多张建模图像（包括 OK 图像和 NG 图像）进行统计训练。

说明

至少需要一张 OK 图像进行建模，一般选择 3~5 张建模图像。

操作项	描述
当前图像	导入当前的建模图像。
缓存图像	导入*缓存图像*模块缓存的图像。

操作项	描述
本地图像	从本地路径导入一张或多张图像。
导入文件夹	选择本地路径下的某个文件夹，将该文件夹下的所有图像批量导入。
批量导入 OK 图像	选择本地路径下的某个文件夹，将该文件夹下的所有图像批量导入为 OK 图像。
批量导入 NG 图像	选择本地路径下的某个文件夹，将该文件夹下的所有图像批量导入为 NG 图像。

2) **可选操作**: 设置**尺度模式**和**特征类型**。

尺寸模式

定义缺陷检测精度的配置模式，分为**自动**和**手动**两种。若**自动**模式满足针对缺陷的特征提取需求，则无需调节。

- **自动**：算法根据输入图像自动设置适配该图像的**最小缺陷尺寸**（具体定义见下文）。
- **手动**：选用该模式时，需手动设置**最小缺陷尺寸**取值。

最小缺陷尺寸

需检测出的缺陷的最小宽度（单位：pixel）。取值越小，缺陷检测精度越高，耗时越长。

建议根据实际检测需求合理设置。例如，假设图像中存在 A 缺陷（尺寸较大）和 B 缺陷（尺寸较小），但仅需检测出 A 缺陷，那么可将该参数取值设置为略大于 B 缺陷尺寸的取值。

背景噪声抑制类型

可选自动或手动。设置为手动时，根据输入的**背景噪声抑制阈值**对背景进行抑制；设置为自动时，需要在样本图像上绘制**背景噪声抑制区域**，算法将根据该区域自动计算阈值进行抑制。

背景噪声抑制阈值

抑制背景扰动导致的异常。该数值越高，对背景波动导致异常的抑制作用就越强，可削弱背景不一致对检测的影响，降低误检的发生。

背景噪声抑制区域

需自行绘制进行抑制的背景区域，算法将根据该区域自动计算背景噪声抑制阈值，避免对区域内的背景波动产生误检。

下采样率

如果该参数值为 k，则即是在原图中每行每列每隔 k 个点取一个点组成一幅图像。因此该参数值越大，轮廓点越稀疏，轮廓越不精细。该参数值不宜设置过大。

特征类型

设置算法在模型训练阶段从图像中提取的特征类型，包括**全部**、**边缘特征**和**灰度特征**。

- **全部**：提取轮廓特征和灰度特征。
- **边缘特征**：仅提取边缘特征。
- **灰度特征**：仅提取灰度特征。

3) 选择 OK 图像，并在“模型注册”处单击 *OK*，将该图像标记为 OK。
4) 选择 NG 图像，并在“模型注册”处单击 *NG*，将该图像标记为 NG。
5) 单击 *训练*。
6) **可选操作**: 在“模型验证”处，滑动滑块调整 OK 标准。

此处的标准数值表示验证图像与模型的相似度。数值越大，相似度越高。

说明

验证参数中的**匹配极性**、**角度范围**和**阈值类型**的详情，请参见 *轮廓匹配* 中的同名参数。**边缘阈值**详情请参见下文的 *参数配置* 。

7) 导入验证图像，并单击 *模型验证*，获取统计结果。
8) 单击 *完成*，返回 *缺陷模型* 页签。



图 13-220 批量训练

6. 在 *缺陷模型* 页签进行如下可选操作。

编辑模板　单击 *编辑模板* 继续编辑模板。

清空模板　单击 清空模板。

载入模板　单击 从本地路径载入模板。

导出模板　单击 将模板导出至本地路径。

继承缺陷模型

可继承前序模块的矩形 ROI 或者掩膜图像进行缺陷模型训练。

前提条件

已执行一次流程，使*异常检测*模块获取到图像源输入的图像。

请参照如下步骤创建并配置缺陷模型。

操作步骤

1. 在*异常检测*模块的配置窗口，选择*缺陷模型*页签。
2. 选择*继承*并单击*+ 新建模型*，打开*异常检测*窗口。
3. 设置*继承方式*。可选*按区域*或*按图像*。
   - 若选择*按区域*，则需从前序模块订阅区域。
   - 若选择*按图像*，则需从前序模块选择掩膜图像。



图 13-221 按区域继承

4. 单击 并在弹出菜单上按需选择操作项，加载多张建模图像（包括 OK 图像和 NG 图像）进行统计训练。

注意

至少需要加载一张 OK 图像进行建模，一般选择 3 至 5 张建模图像。

操作项	描述
当前图像	导入当前的建模图像。
缓存图像	导入*缓存图像*模块缓存的图像。
本地图像	从本地路径导入一张或多张图像。
导入文件夹	选择本地路径下的某个文件夹，将该文件夹下的所有图像批量导入。
批量导入 OK 图像	选择本地路径下的某个文件夹，将该文件夹下的所有图像批量导入为 OK 图像。
批量导入 NG 图像	选择本地路径下的某个文件夹，将该文件夹下的所有图像批量导入为 NG 图像。

5. **可选操作**: 设置**尺寸模式**和**特征类型**，参数详情分别参见 *直接创建缺陷模型* 中已提及的 *尺寸模式* 和 *特征类型* 。
6. 选择 OK 图像，并在“模型注册”处单击 *OK*，将该图像标记为 OK。
7. 选择 NG 图像，并在“模型注册”处单击 *NG*，将该图像标记为 NG。
8. 单击“批量训练”处的*训练*，完成模型训练。



图 13-222 模型训练完成

9. **可选操作**: 在“模型验证”处，滑动滑块调整 OK 标准。

说明

此处的标准数值表示验证图像与模型的相似度。数值越大，相似度越高。

10. 单击*完成*，返回*缺陷模型*页签。
11. **可选操作**: 在*缺陷模型页*签进行如下可选操作。

编辑模板　单击*编辑模板*继续编辑模板。

清空模板　单击 清空模板。

载入模板　单击 从本地路径载入模板。

导出模板　单击 将模板导出至本地路径。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

分数阈值

低于评分阈值的目标将被判定为 NG。

说明

缺陷模板选择继承时，运行参数只有该参数，无以下参数。

最小匹配分数

缺陷模型与图像中目标的相似程度，即相似度阈值。相似度达到该阈值时，目标才会被搜索到，最大取值为 1，表示目标与模型完全一致。

匹配极性

极性表示特征图形到背景的颜色过渡情况。如果查找目标的边缘极性和缺陷模型的极性不一致时，仍要保证目标被查找到，则匹配极性需设置成**不考虑极性**。如不需要则可设置成**考虑极性**，可缩短查找时间。

角度范围

待匹配目标相对于已创建模型的角度变化范围，若要搜索有旋转变化的目标则需设置该参数，有效值范围-180~ 180。

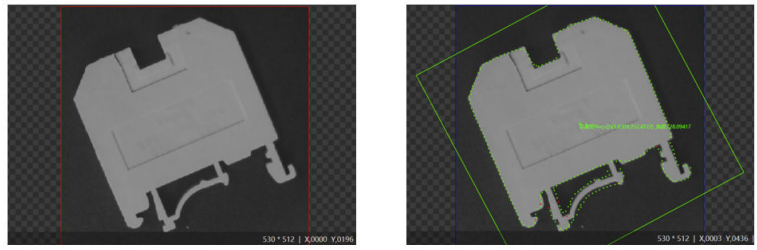


图 13-223 效果示例

尺度范围类型

设置待匹配目标相对于已创建模板的缩放尺度范围类型。搜索存在尺度缩放的目标时，需设置该参数，可从整体或分别从 X、Y 轴的缩放尺度范围进行匹配，可选**尺度范围**和**尺度范围 XY** 两种类型。各类型具体说明如下：

尺度范围

从整体设置待匹配目标相对于已创建模板的缩放尺度范围。当您选择该类型后，还需要在**尺度范围**字段继续设置尺度的变化范围，取值范围是 0.1~ 1。

尺度范围 XY

可分别从 X、Y 轴设置待匹配目标相对于已创建模板的缩放尺度范围。当您选择该类型后，需要在 **X 尺度范围**和 **Y 尺度范围**字段分别设置待匹配目标相对于已创建模板在 X 轴和 Y 轴上的变化范围，取值范围是 0.1~ 10。

阈值类型

设置边缘阈值的类型。边缘阈值指边缘对比度阈值，即轮廓点和背景的灰度值的差距阈值。仅大于该阈值的轮廓点可被提取。

自动阈值

根据目标图像自适应边缘阈值。

模板阈值

以模板的边缘阈值经过内部转换后作为匹配阶段的边缘阈值。

手动阈值

以手动设定的***边缘阈值***作为边缘阈值。

延拓阈值

延拓阈值为特征在图像边缘显示不全时，特征缺失的部分相对于完整的特征的比例。当被查找的目标被图像边缘截断时，设置延拓阈值可保证目标被找到。

下图中只要设置延拓阈值大于 40 时就可以保证图像上方被截断的目标被查找到。

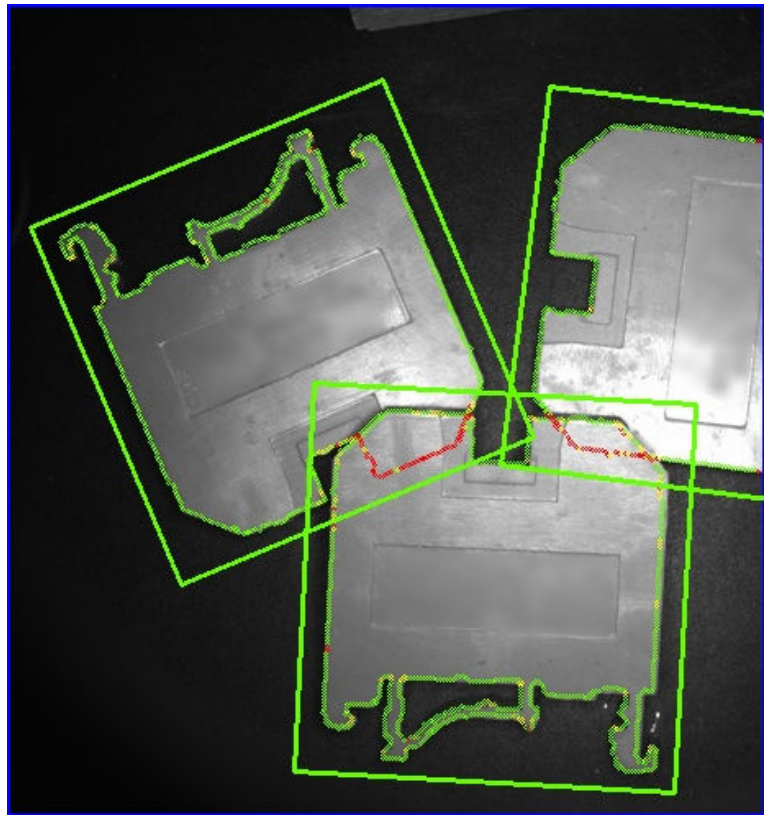


图 13-224 效果示例

模块结果

*异常检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

对比状态

int 型，代表将算法分数与设定阈值进行比较，大于该阈值输出 1，表示无缺陷，小于该阈值输出 0，表示有缺陷 。

标签

string 型，代表将算法分数与设定阈值进行比较，大于该阈值输出 OK， 表示无缺陷，小于该阈值输出 NG，表示有缺陷 。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

检测区域个数

float 型，代表检测区域个数。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

匹配状态

int 型，1 代表匹配成功，0 代表匹配失败。

匹配框

box 型，代表将匹配对象框选起来的最小矩形框。

匹配框中心

point 型，代表将匹配对象框选起来的最小矩形框的中心点。

匹配框中心 X

float 型，代表匹配框中心点的 X 坐标。

匹配框中心 Y

float 型，代表匹配框中心点的 Y 坐标。

匹配框宽度

float 型，代表匹配框的宽度。

匹配框高度

float 型，代表匹配框的高度。

角度

float 型，代表匹配框较长边旋转至水平轴的角度，顺时针旋转为正，逆时针旋转为负。

13.5.4 边缘模型缺陷检测

通过*边缘模型缺陷检测*模块，可将目标物体边缘与基准模型进行对比，输出边缘缺陷信息，如边缘偏移和断裂。

本节内容包含：

- *模块原理*
- *使用方法*
- *边缘模型*
- *参数配置*
- *模块结果*

模块原理

该模块的边缘缺陷检测算法包括位置偏移缺陷检测、断裂缺陷检测和阶梯缺陷检测。

说明

该模块的边缘检测基于 *卡尺工具* 。

位置偏移缺陷检测

边缘位置缺陷检测模块通过检测边缘与模型边缘的距离进行缺陷判定。如果当前点的距离超过预设的阈值（即下文 *参数配置* 中的**位置偏移阈值**），则判定为位置偏移缺陷点。如果存在连续位置偏移缺陷点且长度大于预设的阈值（即下文 *参数配置* 中的**缺陷长度阈值**），则构成位置偏移缺陷区域。

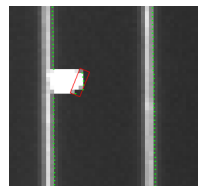


图 13-225 位置偏移缺陷示例

断裂缺陷检测

如果查找边缘点失败，则当前卡尺检测区域被认定为断裂点。
此时将卡尺中心赋值为边缘点，如果连续相连的断裂点长度满足预设的阈值（即下文 *参数配置* 的**缺陷长度阈值**），则判定为是断裂缺陷区域。

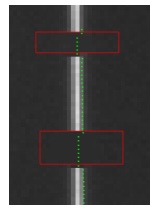


图 13-226 断裂缺陷

阶梯缺陷检测

将检测出的边缘与基准模型中的边缘进行对比。如果相邻点在垂直方向上的偏离距离大于预设的阈值（即下文 *参数配置* 的**阶梯偏离高度**），则判定为阶梯点。如果相邻阶梯点的尺寸大于预设的阈值（即下文 *参数配置* 的**最小阶梯长度**，则认为是阶梯缺陷区域。

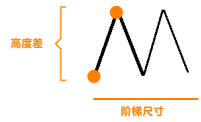


图 13-227 阶梯缺陷

使用方法

前后序模块

如果待检测目标在视野内存在位姿变化，*边缘模型缺陷检测*的前序模块一般包括 *图像源*、*位置修正* 和 *模板匹配* 等模块。前序模块用于对边缘做粗定位，*边缘模型缺陷检测*进行精定位。

如果待检测目标在视野内无位姿变化，*边缘模型缺陷检测*可仅与*图像源*搭配使用。

说明

前序模块包含*位置修正*时，创建位置修正基准点时必须在一张图像上进行。需先运行一次流程，双击*位置修正*模块手动点击创建基准点。

*边缘模型缺陷检测*对后序模块无特殊要求。

主要配置步骤

在流程中调用该模块后，该模块的主要配置步骤如下：

1. 配置该模块的基本参数，详情参见 *基本参数* 。
2. 配置 *边缘模型* 。
3. 配置该模块的 *运行参数* 。
4. 配置该模块的 *结果显示* 。
5. 执行或连续执行该模块，查看该模块的输出结果。

应用示例

下图所示的应用示例中，与模型不匹配的边缘，在结果图像中标红显示。

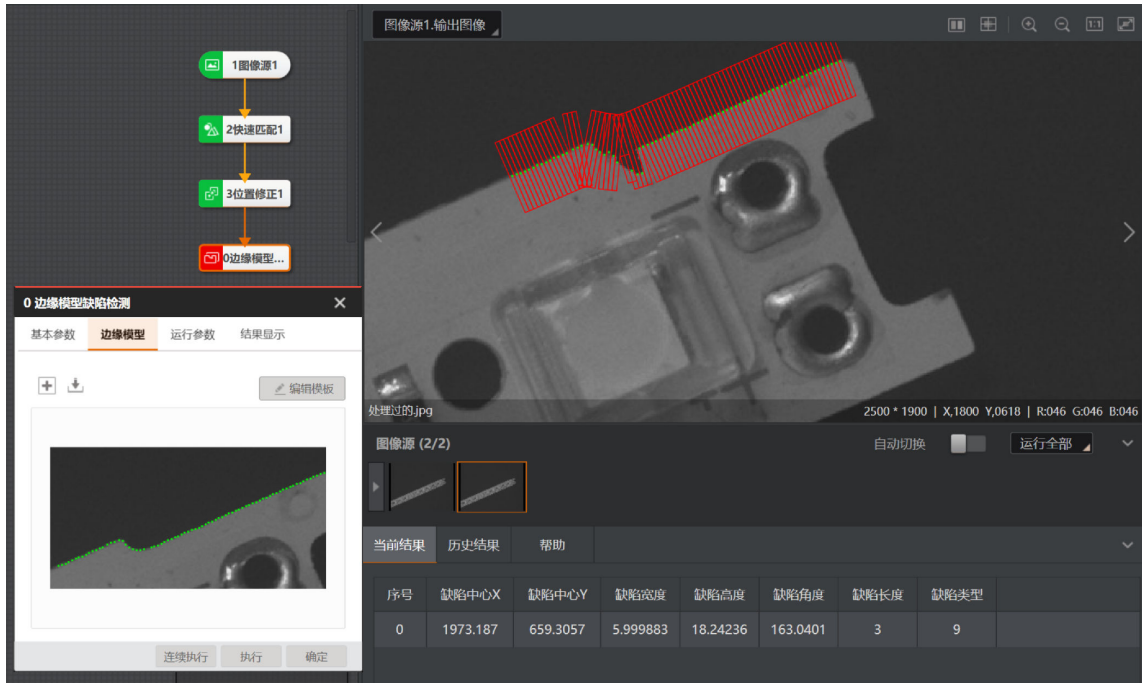


图 13-228 应用示例

边缘模型

*边缘模型缺陷检测*模块需将目标图像的轮廓与基准图像比对，判断目标图像的轮廓是否存在缺陷。创建边缘模型即创建上述基准图像的过程。

前提条件

已执行一次流程，使*边缘模型缺陷检测*模块获取到*图像源*输入的图像。

请参照如下步骤创建并配置边缘模型。

操作步骤

1. 在*边缘模型缺陷检测*模块的配置窗口，选择*边缘模型*页签。

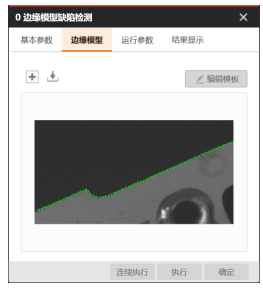


图 13-229 边缘模型页签

2. 单击 ⊞ 打开*模板配置*窗口。

   该窗口默认显示当前*图像源*模块采集的图像。如需基于其他图像进行模板配置，可单击*选择其他图像*。

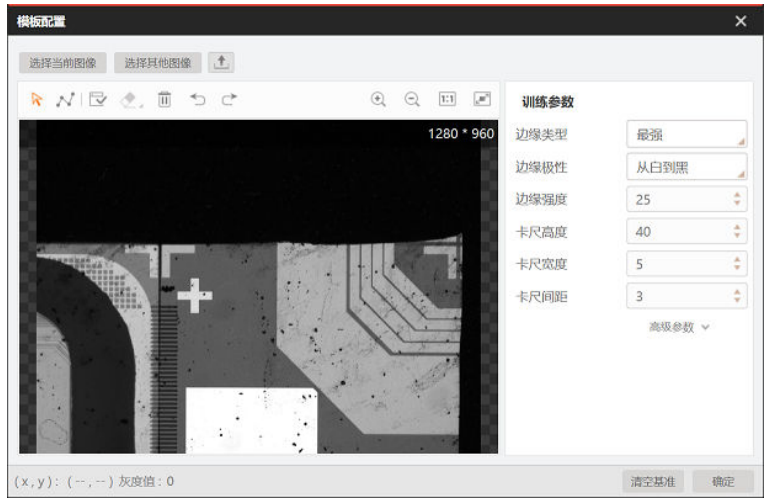


图 13-230 模板配置窗口

3. 单击 ，并在图像上绘制待提取为模型的轮廓轨迹。

   单击可绘制轨迹的节点，拖动光标可绘制线段，双击可结束绘制。

4. 可选操作: 进行如下可选操作。

移动图像	单击 后，将光标移动至图像上并拖拽图像进行移动。
缩放图像	单击 / 缩放图像；或将光标置于图像区域，并滚动鼠标滚轴缩放图像。
清空轨迹	单击 清空轨迹。
撤销上一步操作	单击 撤销上一步操作。
恢复至撤销前	单击 取消撤销，恢复至撤销前的状态。
图像自适应	单击 将图像大小设置为自适应模式。
全屏显示	单击 将图像全屏显示。
删除轨迹节点	将光标悬浮至轨迹正上方，直至显示轨迹节点（见下图）。此时右键单，并在弹出的右键菜单上单击*删除*。
插入轨迹节点	将光标悬浮至轨迹正上方，直至显示轨迹节点（见下图）。此时右键单，并在弹出的右键菜单上单击*插入点*。
删除轨迹节点	将光标悬浮至轨迹正上方，直至显示轨迹节点（见下图）。此时右键单，并在弹出的右键菜单上单击*删除点*。
闭合轨迹	将光标悬浮至轨迹正上方，直至显示轨迹节点（见下图）。此时右键单，并在弹出的右键菜单上单击*闭合轨迹*，将未闭合的路径轨迹闭合。
调整轨迹节点位置	将光标悬浮至轨迹正上方，直至显示轨迹节点（见下图）。此时拖拽节点调整其位置。
清空基准	如果模板中用于位置修正的基准点不符合要求，单击*清空基准*可重新设置用于位置修正的基准点。

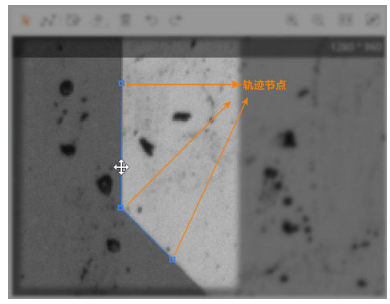


图 13-231 轨迹节点

5. 单击 生成模型。

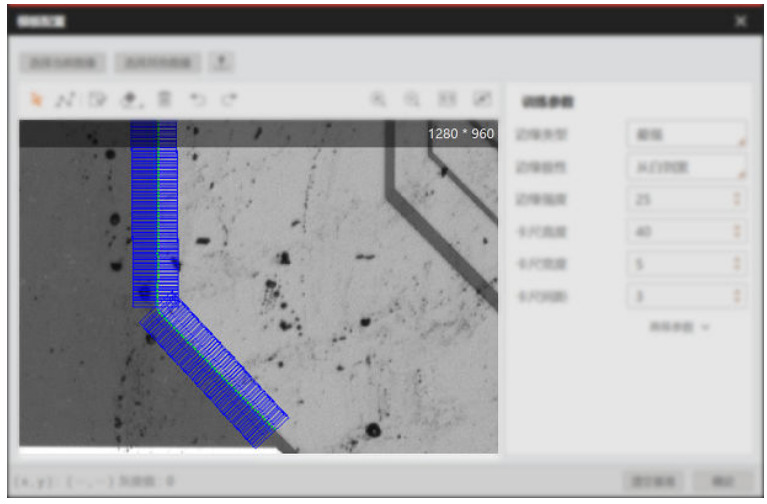


图 13-232 模型效果示例

6. **可选操作**: 如果模型中提取了不必要的轮廓点，单击 ，并将光标悬浮至这些轮廓点将其擦除。

   可单击 右下角的三角形图标，并在弹出的浮窗滑动滑块调整“橡皮擦”的大小。

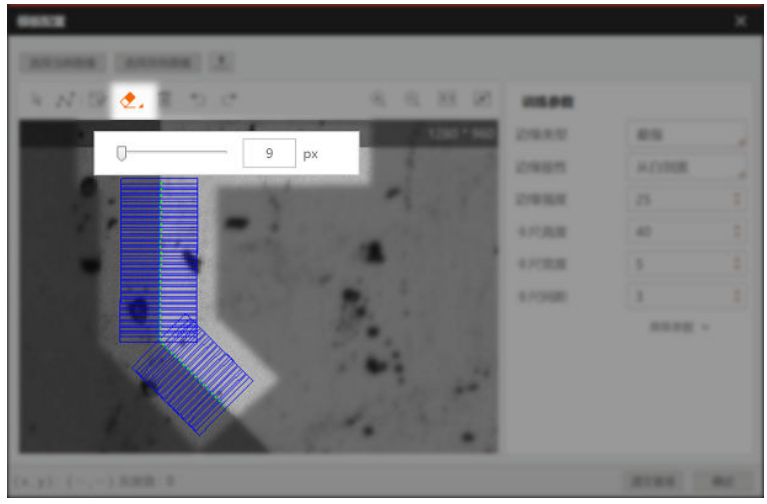


图 13-233 调整“橡皮擦”大小

7. **可选操作**: 如果基于默认取值的训练参数生成的模型无法满足需求，调整参数取值并重新生成模型，直至模型满足需求。

   边缘类型

   定义待查找的边缘类型。可设置为**最强**、**第一条**或**最后一条**。

   最强

仅以下边缘类型可被检测出：查找方向上梯度大于下文提及的**边缘阈值**，且变化幅度最大的边缘。

第一条

仅以下边缘类型可被检测出：查找方向上梯度大于**边缘阈值**的第一条边缘。

最后一条

仅以下边缘类型可被检测出：查找方向上梯度大于**边缘阈值**的最后一条边缘。

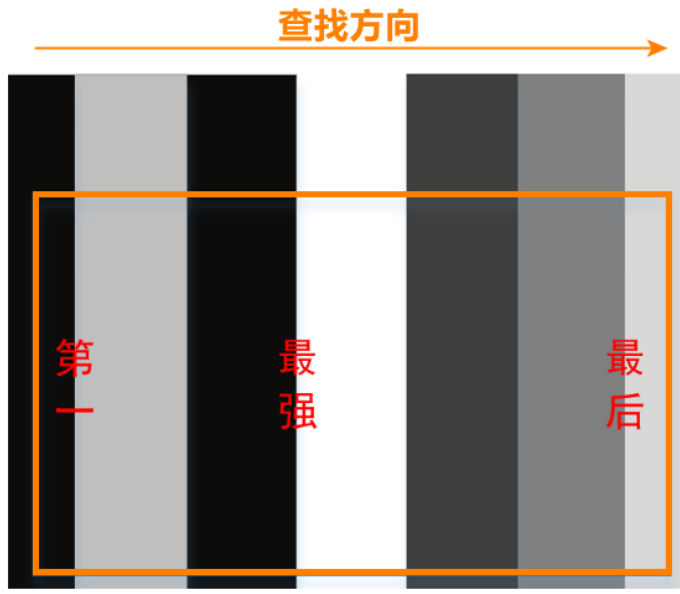


图 13-234 边缘类型

边缘极性

定义待查找的边缘的极性。极性指灰度变化的类型，可设置为*任意极性*、**从黑到白**或**从白到黑**。此处的“黑”和“白”分别用来描述灰度值的高和低。

边缘强度

边缘强度即梯度阈值，取值范围 0~255。仅梯度大于该值的边缘点可被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

卡尺高度

该值描述 *卡尺*（可理解为“用于检测边缘点的 ROI”）的高度。当边缘查找不准确时可适当增大该值。

卡尺宽度

该值描述 *卡尺* 的宽度。在一定范围内增大该值可以获取更加稳定的边缘点。噪点较多时建议取较大值，但相应的耗时将增大。

卡尺间距

该值描述相邻 ***卡尺*** 的间距。间距越小，获取的边缘点越多，但相应的耗时将增大。

边缘精细检测

如果不开启，则基于绘制轨迹放置卡尺并提取边缘点。

如果开启，算法首先将基于绘制轨迹放置卡尺并提取边缘点；然后使用边缘点形成新的轨迹，并再次放置卡尺提取边缘点。开启的作用是，通过实际查找的边缘点修正手动绘制的轨迹，降低手动绘制的轨迹不准确可能引入的风险，使边缘更加精确。

滤波尺寸

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定。但如果边缘与边缘之间距离小于滤波尺寸，边缘位置的精度将受影响，甚至可能丢失边缘。

8. **可选操作**: 单击 将模板保存至本地路径。
9. 单击*模板配置*窗口右下角的*确定*，完成模板配置，并返回*边缘模型*页签。
10. 在*边缘模型*页签进行如下可选操作。

编辑模板	单击*编辑模板*继续编辑模板。
载入模板	单击 从本地路径载入模板。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 ***基本参数*** 。
- 边缘模型相关训练参数详情，请参见 ***边缘模型*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

运行参数中的检测参数与 ***边缘模型*** 中的训练参数相同，其余运行参数详情如下。

缺陷长度阈值

如果存在连续位置偏移缺陷点，且由这些缺陷点构成的轮廓的长度大于该参数值，则判定这些缺陷点所在区域存在位置偏移缺陷。

位置缺陷使能

开启并配置如下参数后，该模块可检测位置偏移缺陷。

位置偏移阈值

如果目标图像边缘点与建模图像的标准边缘点的距离大于该阈值，则判定为位置偏移缺陷待选点。

断裂缺陷使能

开启后，该模块可检测断裂缺陷。

阶梯缺陷使能

开启并按需配置如下子参数后，该模块可检测锯齿类缺陷。该类缺陷尺寸和面积较小，且偏离幅度较低。

阶梯偏离高度

检测图像中，如果相邻的两个边缘点高度偏差若大于该阈值，则判定为阶梯缺陷候选点。

最小阶梯长度

阶梯缺陷候选点相邻的个数超该值，则判定为阶梯缺陷。

灰度辅助检测

开启后，算法基于建模时的理想边缘灰度信息提取边缘，具体机制如下：

根据模型边缘灰度信息，对检测卡尺输出的边缘进行评分，其中平均灰度和灰度邻域评分各占 0.5 的权重，卡尺输出的后续边缘所占权重依次递减 20%。最后统计评分信息，将得分最高的边缘输出。

滤波尺寸

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定。但如果边缘与边缘之间距离小于滤波尺寸，边缘位置的精度将受影响，甚至可能丢失边缘。

追踪容忍

距离阈值，用于边缘点修正，去除干扰点。若为 0，则不作边缘点修正，若大于 0，则在该距离阈值内寻找更符合边缘趋势的边缘点。

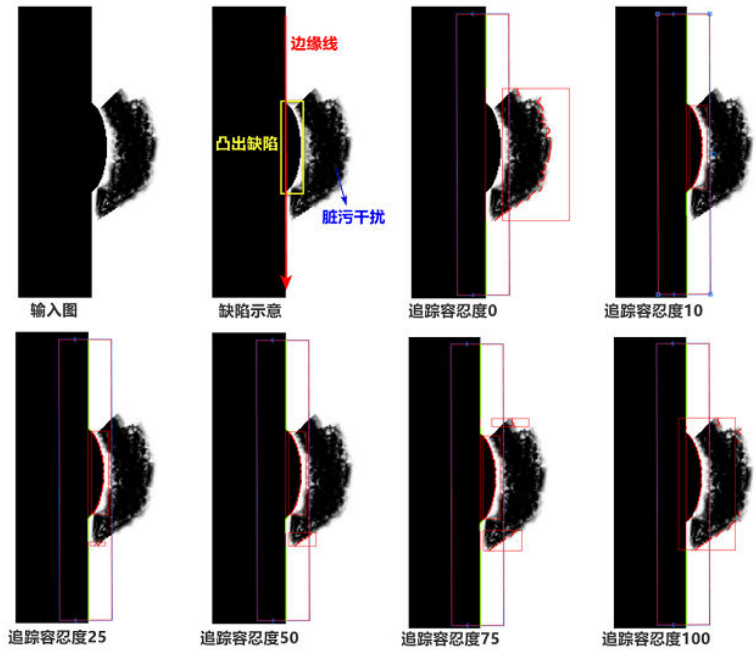


图 13-235 追踪容忍效果示意

模块结果

*边缘模型缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

缺陷框

box 型，代表缺陷框

缺陷框中心点

point 型，代表缺陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

卡尺框

box 型，表示用于定位目标对象所使用的卡尺框，记录了该卡尺框的中心点、宽度、高度和角度信息。

卡尺中心点

point 型，表示卡尺框的中心点，记录了卡尺框中心点的坐标信息。

卡尺中心 X

float 型，代表卡尺框中心点的 X 坐标。

卡尺中心 Y

float 型，代表卡尺框中心点的 Y 坐标。

卡尺宽度

float 型，代表每个卡尺框的宽度，一般相等。

卡尺高度

float 型，代表每个卡尺框的高度，一般相等。

卡尺角度

float 型, 代表每个卡尺框与水平方向的夹角。一般对应位置角度相差的大小为检测圆弧的角度范围。

缺陷卡尺起始索引

int 型，代表检测出的缺陷在所有卡尺中的起始位置索引。

缺陷卡尺终止索引

int 型，代表检测出的缺陷在所有卡尺中的终止位置索引。

缺陷类型

int 型，代表边缘缺陷的类型。

- 6 代表边缘位置偏移。
- 7 代表边缘凸起或凹坑缺陷。
- 8 代表边缘断裂缺陷 。
- 9 代表边缘阶梯差缺陷。

缺陷长度

float 型，代表沿着轨迹方向的缺陷长度，即所占卡尺数量。

边缘点

point 型，代表检测到的边缘点。

边缘点 X

float 型，代表检测到边缘点的 X 坐标。

边缘点 Y

float 型，代表检测到边缘点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

理想卡尺点

理想卡尺点 X

float 型，代表理想中检测到边缘点的 X 坐标。

理想卡尺点 Y

float 型，代表理想中检测到边缘点的 Y 坐标。

13.5.5 边缘对模型缺陷检测

*边缘对模型缺陷检测*模块可将实际提取到的边缘对与预先建立的边缘对模型作比较，分析边缘对的存在性、位置、宽度等信息，判断是否存在边缘断裂等边缘对缺陷。该模块常用于待检边缘区域呈现不规则分布的情况，例如呈曲线分布。

模块原理

该模块算法包括宽度缺陷检测、位置缺陷检测、断裂缺陷检测、阶梯缺陷检测和气泡缺陷检测。

说明

该模块的边缘检测基于 *卡尺工具* 。

宽度缺陷检测

将查找到的边缘对与模型中的边缘对进行比较，判断边缘对是否存在宽度缺陷，包括宽度过窄和过宽，如下图所示存在宽度过宽缺陷。



图 13-236 宽度缺陷

位置缺陷检测

将模型中的边缘对信息与检测出的边缘对作比较，如果超过一定数量的连续多个卡尺内理想边缘对中心位置与实际边缘对中心位置的差值大于预设的阈值（（即下文 ***参数配置*** 中的**位置偏移阈值**），则当前卡尺位置存在边缘对位置缺陷。

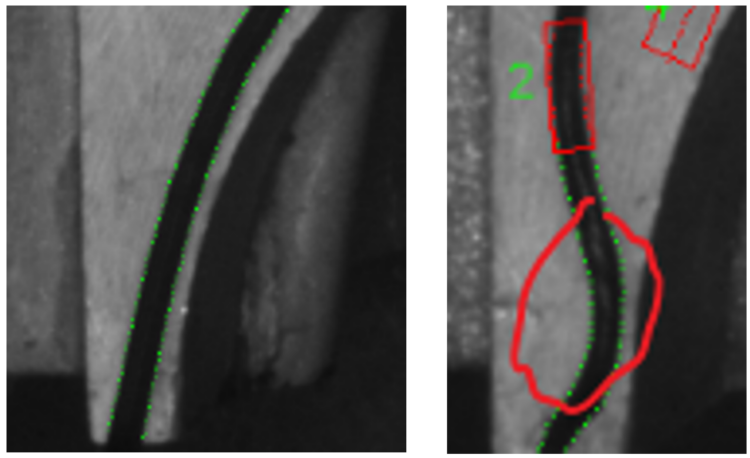


图 13-237 位置缺陷

断裂缺陷检测

如果实际查找边缘点对失败，则判定当前卡尺区域为候选断裂点，并将卡尺中心赋值为边缘点对；如果相邻的断裂点长度满足缺陷长度阈值，则认为存在断裂缺陷。

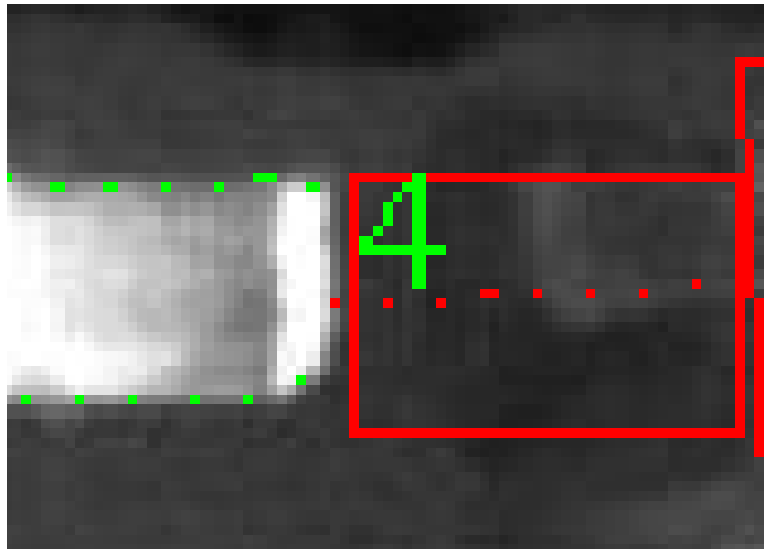


图 13-238 断裂缺陷

阶梯缺陷

将检测出的边缘对与基准模型中的边缘对进行对比。如果相邻点在垂直方向上的偏离距离大于预设的阈值（即下文 *参数配置* 的**阶梯偏离高度**），则判定为阶梯点。如果相邻阶梯点的尺寸大于预设的阈值（即下文 *参数配置* 的**最小阶梯长度**，则认为是阶梯缺陷区域。

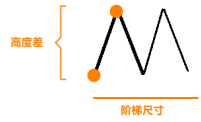


图 13-239 阶梯缺陷

气泡缺陷检测

将模型边缘对平均灰度与查找到的边缘对的平均灰度进行比较，如果两者差值大于预设的阈值（即下文 *参数配置* 中的**灰度合格阈值**），则判定为缺陷候选点；如果相邻缺陷候选点的个数超过预设的阈值（即下文中的**气泡缺陷长度**，则判定相邻缺陷候选点所在位置存在气泡缺陷。

说明

此处的“平均灰度”指“边缘点对”连线上所有点的灰度均值（双线性插值）。

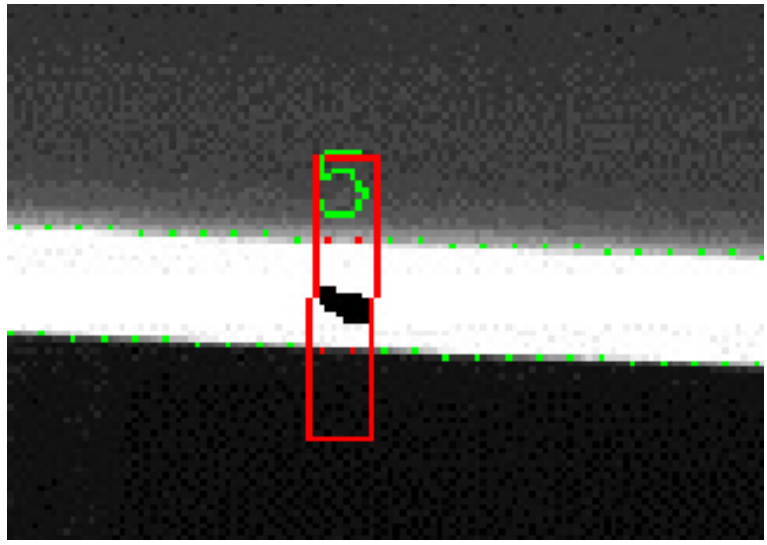


图 13-240 气泡缺陷

使用方法

前后序模块

在流程中，*边缘模型缺陷检测*的：

- 前序模块一般为 *图像源*。如果待检测物体在图像中存在位姿变化，也可搭配 *模板匹配* 和 *位置修正* 模块使用。

  说明

  前序模块包含 *位置修正* 时，创建位置修正基准点时必须在一张图像上进行。需先运行一次流程，双击 *位置修正* 模块手动点击创建基准点。
- 后序模块无特定要求。例如可搭配 *脚本* 和 *发送数据* 模块使用，将数据按指定格式发出。

主要配置步骤

在流程中调用该模块后，该模块的主要配置步骤如下：

1. 配置该模块的基本参数，详情参见 *基本参数*。
2. 配置 *边缘模型*。
3. 配置该模块的运行参数，详情参见下文的 *参数配置*。
4. 配置该模块的 *结果显示*。
5. 执行或连续执行该模块，查看该模块的输出结果。

边缘模型

该模块的边缘模型的创建与配置步骤，与 *边缘模型缺陷检测* 的 *边缘模型* 基本一致。两者区别在于该模块的边缘模型除了可通过手动绘制轨迹生成，也可通过设置轨迹的起始范围自动生成，具体过程如下：

1. 单击**起始范围**处的圆形图标，并在模型图像上绘制轨迹的起始范围。
2. 单击***生成轨迹***自动生成边缘对模型。

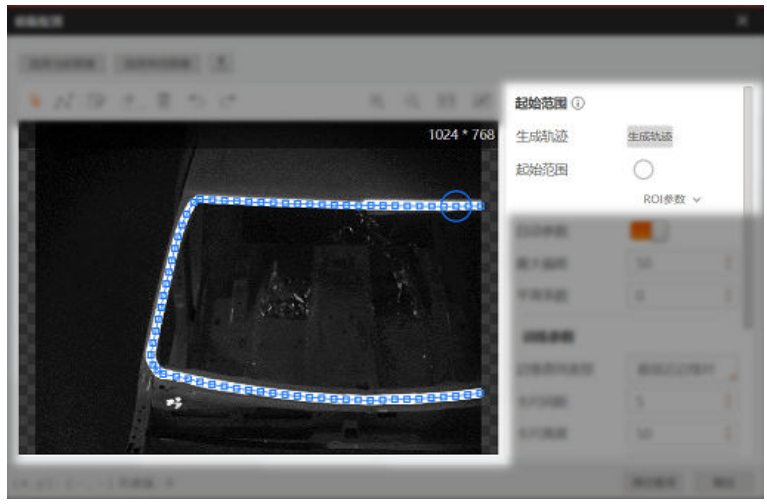


图 13-241 自动生成轨迹

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

边缘查找类型

设置需查找的边缘对类型。

最宽边缘对

查找间距最大的边缘对。

最接近边缘对

查找与下文**理想宽度**取值最接近的边缘对。

最强最接近边缘对

查找边缘梯度最强且与下文**理想宽度**取值最接近的边缘对。

卡尺高度

该值描述 *卡尺* （可理解为“用于检测边缘点的 ROI”）的高度。当边缘查找不准确时可适当增大该值。

卡尺宽度

该值描述 *卡尺* 的宽度。在一定范围内增大该值可以获取更加稳定的边缘点。噪点较多时建议取较大值，但相应的耗时将增大。

理想宽度

设置边缘对的理想宽度。

边缘阈值

边缘阈值即梯度阈值，取值范围 0~255。仅梯度大于该值的边缘点可被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

边缘 0/1 极性

定义待查找的两条边缘的极性。极性指灰度变化的类型，可设置为**从黑到白**、**从白到黑**或**任意极性**。此处的“黑”和“白”分别用来描述灰度值的低和高。

中心点使能

开启后，该模块将输出的边缘 0 和边缘 1 之间中点的 X/Y 轴坐标以及中点状态（1：正常，0：异常）。

缺陷长度阈值

如果存在连续位置偏移缺陷点，且由这些缺陷点构成的轮廓的长度大于该参数值，则判定这些缺陷点所在区域存在位置偏移缺陷。

宽度缺陷使能

开启后将检测宽度缺陷是否在合格比例内。

宽度合格比例

设置两条边缘之间的宽度的合格比例范围。以“将宽度合格比例设置为[50,150]”为例，该范围表示实际查找到的边缘对的宽度是模型边缘对宽度的 50%至 150%时宽度合格，否则将被作为宽度缺陷的候选边缘点对。

位置缺陷使能

开启并配置如下参数后，该模块可检测位置偏移缺陷。

位置偏移阈值

如果目标图像边缘点与建模图像的标准边缘点的距离大于该阈值，则判定为位置偏移缺陷待选点。

断裂缺陷使能

开启后将检测边缘断裂缺陷。

阶梯缺陷使能

开启并按需配置如下子参数后，可检测锯齿类缺陷。该类缺陷尺寸和面积较小，且偏离幅度较低。

阶梯偏离高度

检测图像中，如果相邻的两个边缘点高度偏差若大于该阈值，则判定为阶梯缺陷候选点。

最小阶梯长度

阶梯缺陷候选点相邻的个数超该值，则判定为阶梯缺陷。

气泡缺陷使能

灰度合格阈值

以默认的[75,150]为例，该取值范围表示：如果实际查找到的边缘对的灰度均值为模型中边缘对平均灰度的 75%到 150%时，则为 OK，否则作为气泡缺陷的候选边缘点对；

灰度突变阈值

如果相邻像素点的灰度差异值大于该阈值，则判定该像素点为一个灰度突变缺陷。

气泡缺陷长度

如果被判定为气泡缺陷的候选边缘点对的个数超过该值，则这些边缘点对所在位置存在气泡缺陷。

最大突变次数

针对灰度突变来说，如果灰度突变个数大于该值，则判定当前卡尺存在气泡缺陷。

灰度辅助检测

开启后，算法基于建模时的理想边缘灰度信息提取边缘，具体机制如下：

根据模型边缘灰度信息，对检测卡尺输出的边缘进行评分，其中平均灰度和灰度邻域评分各占 0.5 的权重，卡尺输出的后续边缘所占权重依次递减 20%。最后统计评分信息，将得分最高的边缘输出。

滤波尺寸

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定。但如果边缘与边缘之间距离小于滤波尺寸，边缘位置的精度将受影响，甚至可能丢失边缘。

追踪容忍

距离阈值，用于边缘点修正，去除干扰点。若为 0，则不作边缘点修正，若大于 0，则在该距离阈值内寻找更符合边缘趋势的边缘点。

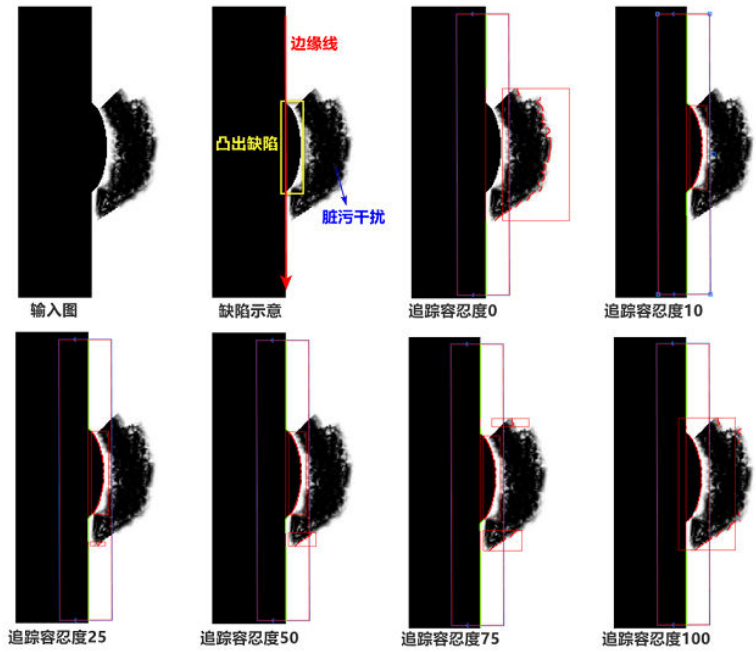


图 13-242 追踪容忍效果示意

模块结果

*边缘对模型缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

边缘对个数

int 型，代表生成的边缘对数量总和。

最大宽度

float 型，代表生成的边缘对之间的最大宽度。

最小宽度

float 型，代表生成的边缘对之间的最小宽度。

平均宽度

float 型，代表生成的边缘对之间的平均宽度。

边缘对宽度

float 型，代表边缘对的宽度。

卡尺框

box 型，表示用于定位目标对象所使用的卡尺框，记录了该卡尺框的中心点、宽度、高度和角度信息。

卡尺中心点

point 型，表示卡尺框的中心点，记录了卡尺框中心点的坐标信息。

卡尺中心 X

float 型，代表卡尺框中心点的 X 坐标。

卡尺中心 Y

float 型，代表卡尺框中心点的 Y 坐标。

卡尺宽度

float 型，代表每个卡尺框的宽度，一般相等。

卡尺高度

float 型，代表每个卡尺框的高度，一般相等。

卡尺角度

float 型, 代表每个卡尺框与水平方向的夹角。一般对应位置角度相差的大小为检测圆弧的角度范围。

缺陷框

box 型，代表缺陷框

缺陷框中心点

point 型，代表缺陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷卡尺起始索引

int 型，代表检测出的缺陷在所有卡尺中的起始位置索引。

缺陷卡尺终止索引

int 型，代表检测出的缺陷在所有卡尺中的终止位置索引。

缺陷类型

int 型，代表边缘缺陷的类型。

- 1 代表边缘对宽度缺陷。
- 2 代表边缘对位置偏移。
- 3 代表边缘对阶梯缺陷。
- 4 代表边缘对断裂缺陷。
- 5 代表边缘对气泡缺陷。

缺陷长度

float 型，代表沿着轨迹方向的缺陷长度，即所占卡尺数量。

边缘*轮廓点（*代表 0-1）

边缘*轮廓点 X

float 型，代表边缘直线*轮廓点的 X 坐标。

边缘*轮廓点 Y

float 型，代表边缘直线*轮廓点的 Y 坐标。

边缘中点

point 型，代表实际检测到的边缘 0 或边缘 1 的中点。

边缘中点 X

float 型，代表实际检测到的边缘 0 或边缘 1 的中点的 X 坐标。

边缘中点 Y

float 型，代表实际检测到的边缘 0 或边缘 1 的中点的 Y 坐标。

边缘*轮廓点状态（*代表 0-1）

int 型，代表用于检测出边缘直线的轮廓点的状态。1 代表轮廓点符合要求，0 代表轮廓点不符合要求。

边缘中点状态

int 型，1 代表正常，其他代表异常。

边缘点个数

int 型，代表输出的边缘点个数。

理想边缘*边缘点（*代表 0-1）

理想边缘*边缘点 X

float 型，代表理想中检测到的边缘 0 的边缘点的 X 轴坐标。

理想边缘*边缘点 Y

float 型，代表理想中检测到的边缘 0 的边缘点的 Y 轴坐标。

优化轨迹点

优化轨迹点 X

float 型，代表优化轨迹点的 X 轴坐标。

优化轨迹点 Y

float 型，代表优化轨迹点的 Y 轴坐标。

理想轨迹点

理想轨迹点 X

float 型，代表理想轨迹点的 X 轴坐标。

理想轨迹点 Y

float 型，代表理想轨迹点的 Y 轴坐标。

理想轨迹卡尺

理想轨迹卡尺中心

理想轨迹卡尺中心 X

float 型，代表卡尺框中心点 X 轴坐标。

理想轨迹卡尺中心 Y

float 型，代表卡尺框中心点 Y 轴坐标。

理想轨迹卡尺宽度

float 型，代表理想卡尺框的宽度，单位一般为像素。

理想轨迹卡尺高度

float 型，代表理想卡尺框的高度，单位一般为像素。

理想轨迹卡尺角度

float 型，代表理想卡尺框与水平方向的夹角。

优化轨迹状态

int 型，1 代表轨迹已成功优化，0 代表优化失败。

13.5.6 直线边缘缺陷检测

*直线边缘缺陷检测*模块通过分析图像中的直线边缘特征，能够精确定位并标记缺陷，并输出缺陷的位置和大小等信息。该模块常用于待检边缘区域呈现明显的直线状分布的场景。

模块原理

该模块将实际提取到的边缘与基准边缘直线比较，通过分析边缘的存在性和位置等信息，明确是否存在边缘缺陷。常见缺陷类型包括断裂和位置偏移等。

- 断裂：根据断裂与基准边缘直线的位置关系，细分为左断裂和右断裂。
- 位置偏移：根据缺陷与基准边缘直线的位置关系，细分为左缺陷、右缺陷。

该模的算法工作流程概览如下图所示。

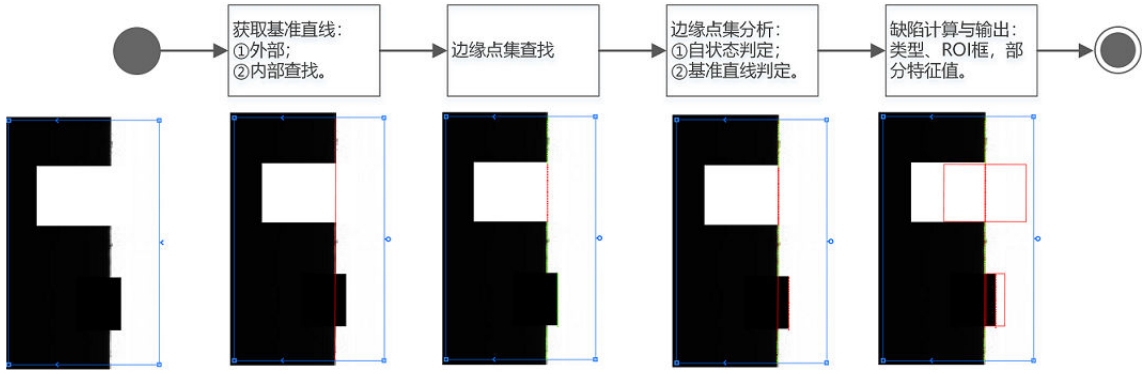


图 13-243 算法工作流程概览

上图中各步骤解析如下。

1. 选定检测区域，获取基准边缘直线。
   获取基准边缘直线的方式分如下两种：
   - 方式 1：外部生成基准直线边缘并直接传入该模块。
     该方式对应如下三种场景：
     - 前序模块已经获取该边缘直线。
     - 已有图集的待检边缘区域无法有效提取稳定边缘直线。
     - 基准边缘直线已知。
   - 方式 2：该模块内部通过 ***直线查找*** 自行查找基准直线边缘。
2. 提取边缘点集。
   在基准边缘直线与检测 ROI 框的相交直线段上均等分布 N 个 ***卡尺工具***，每个卡尺提取一个边缘点，组成含有 N 个边缘点的边缘点集。换言之，即通过边缘采样的方式模拟待检边缘区域的边缘信息。

   说明

   如果卡尺查找边缘点失败，则用卡尺中心点作为该边缘点。

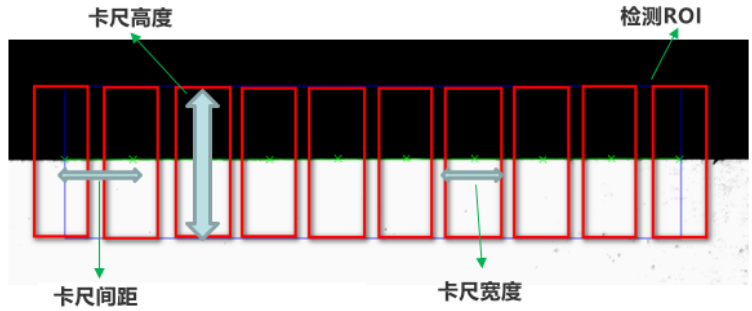


图 13-244 卡尺工具的分布方式

3. 分析边缘点集。
   分如下两种分析方式：
   - 方式 1：判定边缘点自身状态。如果查找失败，则为断裂边缘点，否则为 OK 边缘点。
   - 方式 2：与基准直线比较。如果当前边缘点与基准边缘直线距离偏大，则认为是偏移边缘点，否则为 OK 边缘点。
4. 计算缺陷并输出缺陷信息。
   a. 将连续的断裂边缘点聚类，得到断裂区域；将连续的缺陷边缘点聚类，得到缺陷区域。
   b. 分别计算每个区域的面积和尺寸特征，并输出满足大于一定面积或尺寸要求的缺陷区域。

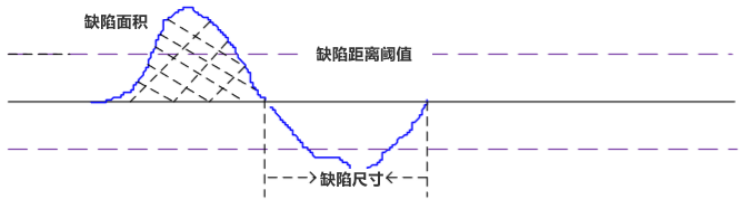


图 13-245 缺陷面积与尺寸定义

上图中，水平实线为基准边缘直线，蓝色点集为实际检测的边缘点集。
某些场景下仅需要检测基准直线的同侧缺陷，其他场景基于基准直线明确边缘缺陷的位置。如下图所示，如果缺陷在基准直线轨迹方向的左侧，则为左断裂或左缺陷；如果缺陷在轨迹方向右侧，则为右断裂或右缺陷。

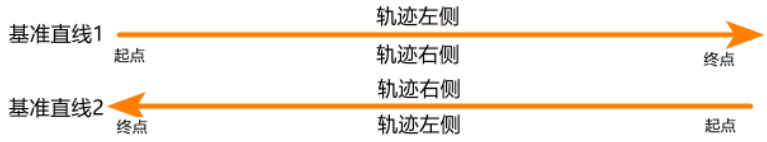


图 13-246 缺陷极性

使用方法

在流程中，*直线边缘缺陷检测*的前后序模块详情如下。

表 13-16 前后序模块

前后序模块	描述
前序模块	一般将 *图像源* 作为*直线边缘缺陷检测*的前序模块，提供图像输入。前序模块还可包括 *直线查找*，以实现更精确的目标定位
后续模块	无特定要求

一般场景下，在流程中调用*直线边缘缺陷检测*后，完成该模块的 ROI 绘制即可对目标图像进行直线边缘缺陷检测。如果检测结果不理想，可调整下文 *参数配置* 中提及的运行参数，直至检测结果符合预期。

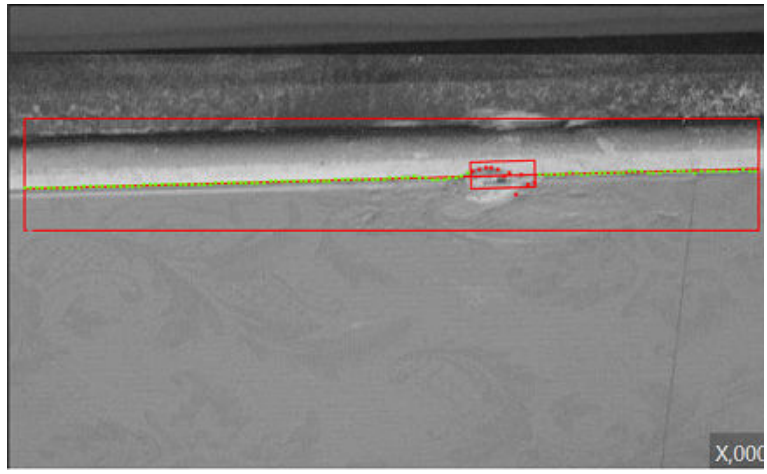


图 13-247 检测结果示例

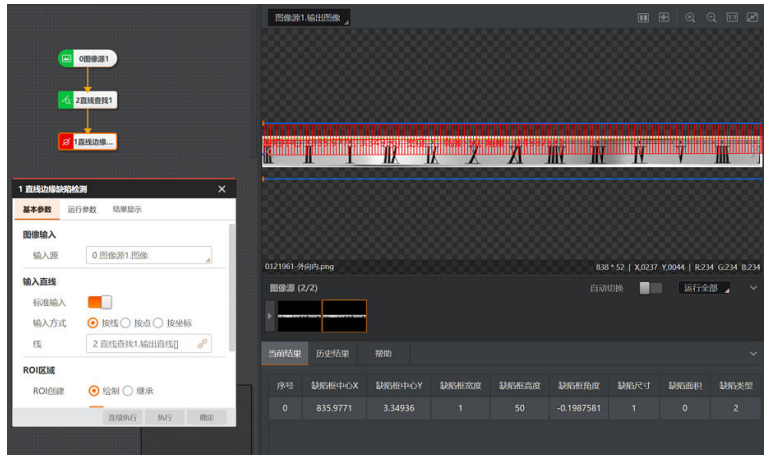


图 13-248 应用示例

参数配置

以下仅介绍该模块的运行参数详情和部分基本参数详情。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

该模块的前序模块包含*直线查找*时，基本参数中需配置**输入直线类型**。

输入直线类型

设置直线的输入方式，包括**按线**、**按点**和**按坐标**。

按线

选择*直线查找*模块输出的直线作为输入直线。

按点

将**起点**和**终点**分别设置为*直线查找*模块输出直线的起点和终点。

按坐标

将**起点 X 坐标**、**起点 Y 坐标**、**终点 X 坐标**、**终点 Y 坐标**分别设置为*直线查找*模块输出直线的起点 X 坐标、起点 Y 坐标、终点 X 坐标和终点 Y 坐标。

可通过调整如下运行参数，定义当前模块如何处理输入的数据。

边缘类型

定义待查找的边缘类型。可设置为**最强**、**第一条**或**最后一条**。

最强

仅以下边缘类型可被检测出：查找方向上梯度大于下文提及的**边缘阈值**，且变化幅度最大的边缘。

第一条

仅以下边缘类型可被检测出：查找方向上梯度大于**边缘阈值**的第一条边缘。

最后一条

仅以下边缘类型可被检测出：查找方向上梯度大于**边缘阈值**的最后一条边缘。

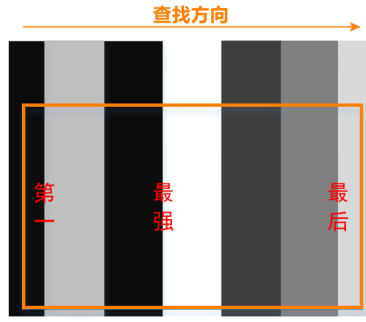


图 13-249 边缘类型

边缘极性

定义待查找的边缘的极性。极性指灰度变化的类型，可设置为**从黑到白**、**从白到黑**或**任意极性**。此处的“黑”和“白”分别用来描述灰度值的高和低。

滤波尺寸

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定。但如果边缘与边缘之间距离小于滤波尺寸，边缘位置的精度将受影响，甚至可能丢失边缘。

边缘阈值

边缘阈值即梯度阈值，取值范围 0~255。仅梯度大于该值的边缘点可被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

卡尺高度

该值描述 *卡尺*（可理解为“用于检测边缘点的 ROI”）的高度。当边缘查找不准确时可适当增大该值。

卡尺宽度

该值描述 *卡尺* 的宽度。在一定范围内增大该值可以获取更加稳定的边缘点。噪点较多时建议取较大值，但相应的耗时将增大。

卡尺间距

该值描述相邻 *卡尺* 的间距。间距越小，获取的边缘点越多，但相应的耗时将增大。

缺陷极性

有**轨迹左侧**、**轨迹右侧**和**轨迹两侧缺陷**三种极性。详情见上文*模块原理*中的 *图 13-246*。

缺陷距离阈值

如果查找到的边缘点与基准边缘直线的距离大于该阈值，则判定为待筛选缺陷点。

缺陷尺寸/面积使能

如果开启**缺陷尺寸使能**和（或）**缺陷面积使能**，则根据对应阈值做进一步筛选。相关原理请参见上文*模块原理*中的 *图 13-245*。

模块结果

*直线边缘缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

ROI 缺陷个数

int 型，数组形式，代表每个 ROI 检测到的缺陷个数。

缺陷框

box 型，代表缺陷框

缺陷框中心点

point 型，代表缺陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷框索引

int 型，代表缺陷框的索引、序号，通常为顺序递增的数组，数组长度和缺陷框数据的长度相同。

缺陷尺寸

float 型，代表沿着轨迹方向的缺陷长度。

缺陷面积

float 型，代表缺陷框的面积。

卡尺起始值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的起始序号点。

卡尺终止值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的终止序号点。

缺陷类型

int 型，代表边缘缺陷的类型。

- 1 代表缺陷。
- 2 代表断裂。
- 3 代表左缺陷。
- 4 代表右缺陷。
- 5 代表左断裂。
- 6 代表右断裂。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

边缘点

表示多直线查找模块检测到的边缘点。

轮廓点 X

float 型，代表边缘轮廓点的 X 坐标。

轮廓点 Y

float 型，代表边缘轮廓点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

ROI 边缘点个数

int 型，数组形式，代表每个 ROI 检测到的边缘点个数。

卡尺框

box 型，表示用于定位目标对象所使用的卡尺框，记录了该卡尺框的中心点、宽度、高度和角度信息。

卡尺中心点

point 型，表示卡尺框的中心点，记录了卡尺框中心点的坐标信息。

卡尺中心 X

float 型，代表卡尺框中心点的 X 坐标。

卡尺中心 Y

float 型，代表卡尺框中心点的 Y 坐标。

卡尺宽度

float 型，代表每个卡尺框的宽度，一般相等。

卡尺高度

float 型，代表每个卡尺框的高度，一般相等。

卡尺角度

float 型, 代表每个卡尺框与水平方向的夹角。一般对应位置角度相差的大小为检测圆弧的角度范围。

理想卡尺点状态

int 型，代表理想中直线边缘点的创建状态。1 表示创建成功，0 表示创建失败。

理想卡尺点数量

int 型，代表理想中直线边缘点的数量。

ROI 理想卡尺点数量

int 型，代表每个 ROI 理想检测到的直线边缘点数量。

标准直线

line 型，代表标准直线。

标准直线起点

point 型，代表标准直线起点。

标准直线起点 X

float 型，代表标准直线起点的 X 坐标 。

标准直线起点 Y

float 型，代表标准直线起点的 Y 坐标 。

标准直线终点

point 型，代表标准直线终点。

标准直线终点 X

float 型，代表标准直线终点的 X 坐标 。

标准直线终点 Y

float 型，代表标准直线终点的 Y 坐标 。

标准直线角度

float 型，代表标准直线相对于 X 轴正方向顺时针偏移的角度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.5.7 直线对缺陷检测

*直线对缺陷检测*模块适用于检测发生形变或者断裂的一对直线之间的缺陷。该模块适用于检测矩形工件边缘的形变和缺损、判断工件边缘的规整程度、查找毛刺和污垢。

使用方法

该模块可作为*图像源*模块的后序模块，直接接收*图像源*的图像输入。
如果检测物在视野内存在姿态变化，该模块的前序模块中可加入 *轮廓匹配* 等定位模块进行粗定位；再使用 *位置修正* 和 *直线查找* 模块完成直线精定位；最后该模块将*直线查找*输出的直线作为基准边缘，用于后续的缺陷检测。
完成该模块的 ROI 绘制和运行参数配置后，即可对目标图像进行直线对缺陷检测。

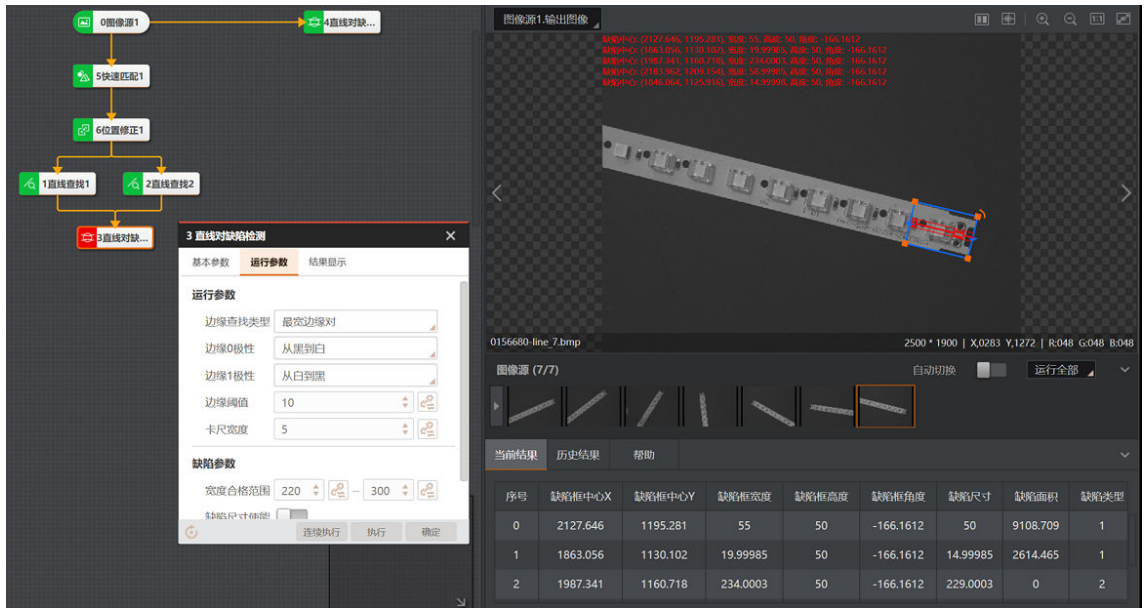


图 13-250 应用示例

参数配置

- 基本参数详情，请参见 ***基本参数*** 。
- 该模块运行参数与 ***圆弧对缺陷检测*** 的类似，具体请参见该模块的同名参数。
- 结果显示参数详情，请参见 ***结果显示*** 。

模块结果

*直线对缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

ROI 缺陷个数

int 型，数组形式，代表每个 ROI 检测到的缺陷个数。

缺陷框

box 型，代表缺陷框

缺陷框中心点

point 型，代表缺陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷框索引

int 型，代表缺陷框的索引、序号，通常为顺序递增的数组，数组长度和缺陷框数据的长度相同。

缺陷尺寸

float 型，代表沿着轨迹方向的缺陷长度。

缺陷面积

float 型，代表缺陷框的面积。

卡尺起始值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的起始序号点。

卡尺终止值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的终止序号点。

缺陷类型

int 型，代表缺陷的类型。

- 1 代表缺陷。
- 2 代表断裂。
- 11 代表位置偏移缺陷。
- 13 代表宽度缺陷 。
- 14 代表亮度缺陷。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

边缘*轮廓点（*代表 0-1）

边缘*轮廓点 X

float 型，代表边缘直线*轮廓点的 X 坐标。

边缘*轮廓点 Y

float 型，代表边缘直线*轮廓点的 Y 坐标。

边缘*轮廓点状态（*代表 0-1）

int 型，代表用于检测出边缘直线的轮廓点的状态。1 代表轮廓点符合要求，0 代表轮廓点不符合要求。

卡尺点

point 型，代表理想中检测到边缘点。

理想卡尺点 X

float 型，代表理想中检测到边缘点的 X 坐标。

理想卡尺点 Y

float 型，代表理想中检测到边缘点的 Y 坐标。

理想卡尺点状态

int 型，代表理想中直线边缘点的创建状态。1 表示创建成功，0 表示创建失败。

理想卡尺点数量

int 型，代表理想中直线边缘点的数量。

ROI 理想卡尺点数量

int 型，代表每个 ROI 理想检测到的直线边缘点数量。

标准直线

line 型，代表标准直线。

标准直线起点

point 型，代表标准直线起点。

标准直线起点 X

float 型，代表标准直线起点的 X 坐标 。

标准直线起点 Y

float 型，代表标准直线起点的 Y 坐标 。

标准直线终点

point 型，代表标准直线终点。

标准直线终点 X

float 型，代表标准直线终点的 X 坐标 。

标准直线终点 Y

float 型，代表标准直线终点的 Y 坐标 。

标准直线角度

float 型，代表标准直线对于水平线顺时针方向偏移的角度。

缺陷综合信息

box 型，代表缺陷综合信息。

最小宽度

float 型，代表模块生成的直线对或圆弧对之间的最小宽度。

最大宽度

float 型，代表模块生成的直线对或圆弧对之间的最大宽度。

平均宽度

float 型，代表模块生成的直线对或圆弧对之间的平均宽度。

近似面积

float 型，代表模块生成的直线对或圆弧对之间的近似面积总和。

最小边缘点*（*代表 0-1）

point 型，代表直线对之间生成的最小宽度间距中下直线边缘点。

最小边缘点 X

float 型，代表直线对之间生成的最小宽度间距中下直线边缘点的 X 坐标。

最小边缘点 Y

float 型，代表直线对之间生成的最小宽度间距中下直线边缘点的 Y 坐标。

最小边缘点*极性（*代表 0-1）

int 型，1 代表从黑到白，2 代表从白到黑。

最小边缘分数

float 型，代表根据算法计算出的最小边缘对分数。

最小边缘距离

float 型，代表最小边缘对之间的距离。

最小边缘状态

int 型，1 代表找到最小边缘对，0 代表没找到。

最大边缘点*（*代表 0-1）

point 型，代表直线对之间生成的最大宽度间距中下直线边缘点。

最大边缘点 X

float 型，代表直线对之间生成的最大宽度间距中下直线边缘点的 X 坐标。

最大边缘点 Y

float 型，代表直线对之间生成的最大宽度间距中下直线边缘点的 Y 坐标。

最大边缘点*极性（*代表 0-1）

int 型，1 代表从黑到白，2 代表从白到黑。

最大边缘分数

float 型，代表根据算法计算出的最大边缘对分数。

最大边缘距离

float 型，代表最大边缘对之间的距离。

最大边缘状态

int 型，1 代表找到最大边缘对，0 代表没找到。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.5.8 圆弧对缺陷检测

*圆弧对缺陷检测*模块可检测圆弧的凹凸部分和断裂部分，并输出相关信息。该模块适用于检测“呈现圆环状分布，且具有一定径向宽度”的边缘区域。

模块原理

*圆弧对缺陷检测*模块一般将实际提取到的边缘点对与基准（同心）圆弧对比较，通过分析边缘对的存在性、宽度等信息，明确是否存在边缘对缺陷。常见缺陷类型有断裂、凸起或凹陷等。

该模块基本算法原理与 *直线边缘缺陷检测* 基本相同，但在如下四项差异：

表 13-17 原理差异

差异项	描述
待检测区域特征	待检区域需具有“呈现圆环状分布，且具有一定径向宽度”的特征，且较为明显。
查找对象	该模块查找的是“边缘点对”，而非“边缘点”。
缺陷极性判定	该模块不判定缺陷极性，仅检测目标线条是否存在断裂或宽度异常（凸出或凹陷）缺陷。
内部逻辑	该模块获取基准圆弧对的方式也分为“从外部获取”和“内部自行查找”两种，但“内部自行查找”的实现逻辑不同——该模块基于 *圆查找* 分别查找圆环外径圆和内径圆，并组成同心圆弧对。

使用方法

该模块的前序模块一般为 *图像源* 和 *圆查找* 。

该模块主要配置步骤如下：

1. 针对待检测的圆弧对，绘制 ROI。
2. 基于前序的两个*圆查找*模块输出的圆弧半径数据（两者相减）和实际检测需求，判定圆弧宽度合格范围。
3. 在*运行参数*页签设置 *圆弧宽度合格范围* 参数。
4. 执行一次该模块，输出检测结果图像。
5. 如果检测效果不理想，调整下文 *参数配置* 中提及的运行参数，并再度执行该模块，直至检测结果符合预期。

下图示例中，输出图像中标为红色的圆弧对，即为宽度不符合预期的缺陷圆弧对。

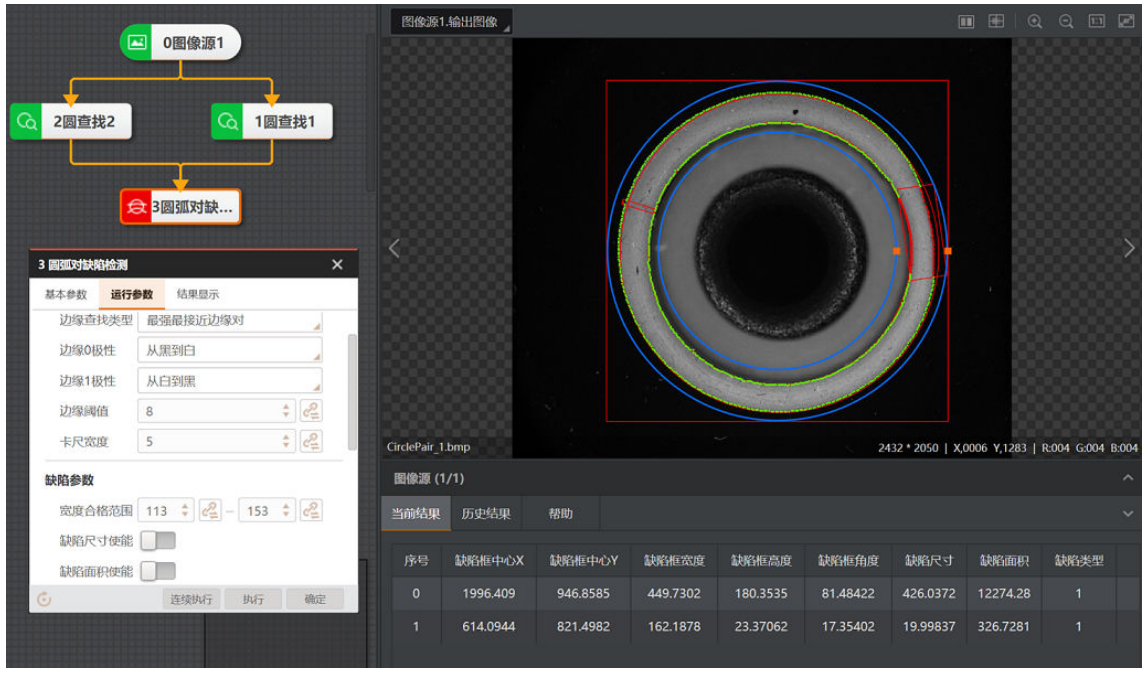


图 13-251 应用示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

边缘查找类型

设置需查找的边缘对类型。

最宽边缘对

查找间距最大的边缘对。

最接近边缘对

查找与下文**理想宽度**取值最接近的边缘对。

最强最接近边缘对

查找边缘梯度最强且与下文**理想宽度**取值最接近的边缘对。

边缘 0/1 极性

定义待查找的两条边缘的极性。极性指灰度变化的类型，可设置为**从黑到白**、**从白到黑**或**任意极性**。此处的“黑”和“白”分别用来描述灰度值的低和高。

边缘阈值

边缘阈值即梯度阈值，取值范围 0~255。仅梯度大于该值的边缘点可被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

卡尺宽度

该值描述 *卡尺* 的宽度。在一定范围内增大该值可以获取更加稳定的边缘点。噪点较多时建议取较大值，但相应的耗时将增大。

宽度合格范围

该参数为圆弧对缺陷检测的主要调节参数,只有在该参数范围内的圆弧对宽度才算作合格,超出范围的视作缺陷，在设置该参数前可使用卡尺工具测量出大致的圆弧对宽度。



图 13-252 圆弧对宽度范围示意图

缺陷尺寸使能

多个缺陷点投影到基准（同心）圆弧，组成的像素尺寸大于**缺陷尺寸阈值**，则判定为缺陷尺寸生效。

缺陷面积使能

缺陷轮廓与基准（同心）圆弧围成的面积是缺陷面积，开启后，仅面积大于**缺陷面积阈值**的缺陷能被查找到。

理想宽度

内外两圆弧的理想间距。

卡尺高度

详情参见*直线边缘缺陷检测*模块的 *卡尺高度* 。

滤波尺寸

详情参见*直线边缘缺陷检测*模块的 *滤波尺寸* 。

卡尺间距

详情参见*直线边缘缺陷检测*模块的 *卡尺间距* 。

卡尺数量

详情参见*圆查找*模块的 *卡尺数量* 。

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，为获取更佳查找效果，建议配合**剔除阈值**设置。

剔除阈值

允许离群点到拟合圆的最大像素距离，值越小，排除点越多。

容忍角度

可接受的目标角度差。

追踪容忍角度

边缘追踪所允许偏移的最大像素。

模块结果

*圆弧对缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

ROI 缺陷个数

int 型，数组形式，代表每个 ROI 检测到的缺陷个数。

检测圆弧

ANNULUS 型，代表检测圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点。

检测圆弧中心 X

float 型，代表圆弧 ROI 中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表圆弧 ROI 中心点的 Y 坐标。

检测圆弧内径

float 型，代表圆弧 ROI 的内径。

检测圆弧外径

float 型，代表圆弧 ROI 的外径。

检测圆弧起始角度

float 型，代表圆弧 ROI 起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

检测圆弧角度范围

float 型，代表圆弧 ROI 从起始边开始至终止边所跨过的角度范围。

缺陷框

box 型，代表缺陷框。

缺陷框中心点

point 型，代表陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷框索引

int 型，代表缺陷框的索引、序号，通常为顺序递增的数组，数组长度和缺陷框数据的长度相同。

缺陷尺寸

float 型，代表沿着轨迹方向的缺陷长度。

缺陷面积

float 型，代表缺陷框的面积。

卡尺起始值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的起始序号点。

卡尺终止值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的终止序号点。

缺陷类型

int 型，代表缺陷的类型。

- 1 代表缺陷。
- 2 代表断裂。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

边缘*轮廓点（*代表 0-1）

边缘*轮廓点 X

float 型，代表边缘直线*轮廓点的 X 坐标。

边缘*轮廓点 Y

float 型，代表边缘直线*轮廓点的 Y 坐标。

边缘*轮廓点状态（*代表 0-1）

int 型，代表用于检测出边缘直线的轮廓点的状态。1 代表轮廓点符合要求，0 代表轮廓点不符合要求。

边缘*轮廓点数（*代表 0-1）

int 型，代表检测到的直线*的边缘点个数。

卡尺点

point 型，代表理想中检测到边缘点。

理想卡尺点 X

float 型，代表理想中检测到边缘点的 X 坐标。

理想卡尺点 Y

float 型，代表理想中检测到边缘点的 Y 坐标。

理想卡尺点状态

int 型，代表理想中直线边缘点的创建状态。1 表示创建成功，0 表示创建失败。

理想卡尺点数量

int 型，代表理想中直线边缘点的数量。

ROI 理想卡尺点数量

int 型，代表每个 ROI 理想检测到的直线边缘点数量。

标准圆环

ANNULUS 型，代表标准圆环。

标准圆环圆心

point 型，代表标准圆环中线点。

标准圆环圆心 X

float 型，代表标准圆环中心点的 X 坐标。

标准圆环圆心 Y

float 型，代表标准圆环中心点的 Y 坐标。

标准圆环内径

float 型，代表标准圆环内侧圆环的圆半径。

标准圆环外径

float 型，代表标准圆环外侧圆环的圆半径。

标准圆环起始角度

float 型，代表标准圆环起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

标准圆环角度范围

float 型，代表标准圆环从起始边开始至终止边所跨过的角度范围。

缺陷综合信息

box 型，代表缺陷综合信息。

最小宽度

float 型，代表模块生成的直线对或圆弧对之间的最小宽度。

最大宽度

float 型，代表模块生成的直线对或圆弧对之间的最大宽度。

平均宽度

float 型，代表模块生成的直线对或圆弧对之间的平均宽度。

近似面积

float 型，代表模块生成的直线对或圆弧对之间的近似面积总和。

最小边缘点*（*代表 0-1）

point 型，代表直线对之间生成的最小宽度间距中下直线边缘点。

最小边缘点 X

float 型，代表直线对之间生成的最小宽度间距中下直线边缘点的 X 坐标。

最小边缘点 Y

float 型，代表直线对之间生成的最小宽度间距中下直线边缘点的 Y 坐标。

最小边缘点*极性（*代表 0-1）

int 型，1 代表从黑到白，2 代表从白到黑。

最小边缘分数

float 型，代表根据算法计算出的最小边缘对分数。

最小边缘距离

float 型，代表最小边缘对之间的距离。

最小边缘状态

int 型，1 代表找到最小边缘对，0 代表没找到。

最大边缘点*（*代表 0-1）

point 型，代表直线对之间生成的最大宽度间距中下直线边缘点。

最大边缘点 X

float 型，代表直线对之间生成的最大宽度间距中下直线边缘点的 X 坐标。

最大边缘点 Y

float 型，代表直线对之间生成的最大宽度间距中下直线边缘点的 Y 坐标。

最大边缘点*极性（*代表 0-1）

int 型，1 代表从黑到白，2 代表从白到黑。

最大边缘分数

float 型，代表根据算法计算出的最大边缘对分数。

最大边缘距离

float 型，代表最大边缘对之间的距离。

最大边缘状态

int 型，1 代表找到最大边缘对，0 代表没找到。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.5.9 圆弧边缘缺陷检测

*圆弧边缘缺陷检测*模块可检测工件或产品上的弧边缘缺陷，如断裂、凹陷、凸点、磨损等问题。

模块原理

该模块的主要检测途径是在圆弧上自动创建一定数量的 *卡尺*，比对相邻卡尺中部分圆弧中心点 x 轴、y 轴和角度偏移，并根据预设的阈值大小判断缺陷是否成立。
如下图所示，检测结果中红框标注部分即圆弧缺陷，绿点即正常的圆弧边缘。



图 13-253 检测结果示例

使用方法

在流程中，*圆弧边缘缺陷检测*模块的前后序模块详情如下：

- 前序模块：通常为*图像源*，为*圆弧边缘缺陷检测*提供图像输入。前序模块还可包括 *圆查找*，以实现更精确的目标定位。
- 后序模块：无特定要求。

*圆弧边缘缺陷检测*具体的配置方法如下图所示，当圆轮廓比较模糊时建议开启**标准输入**。

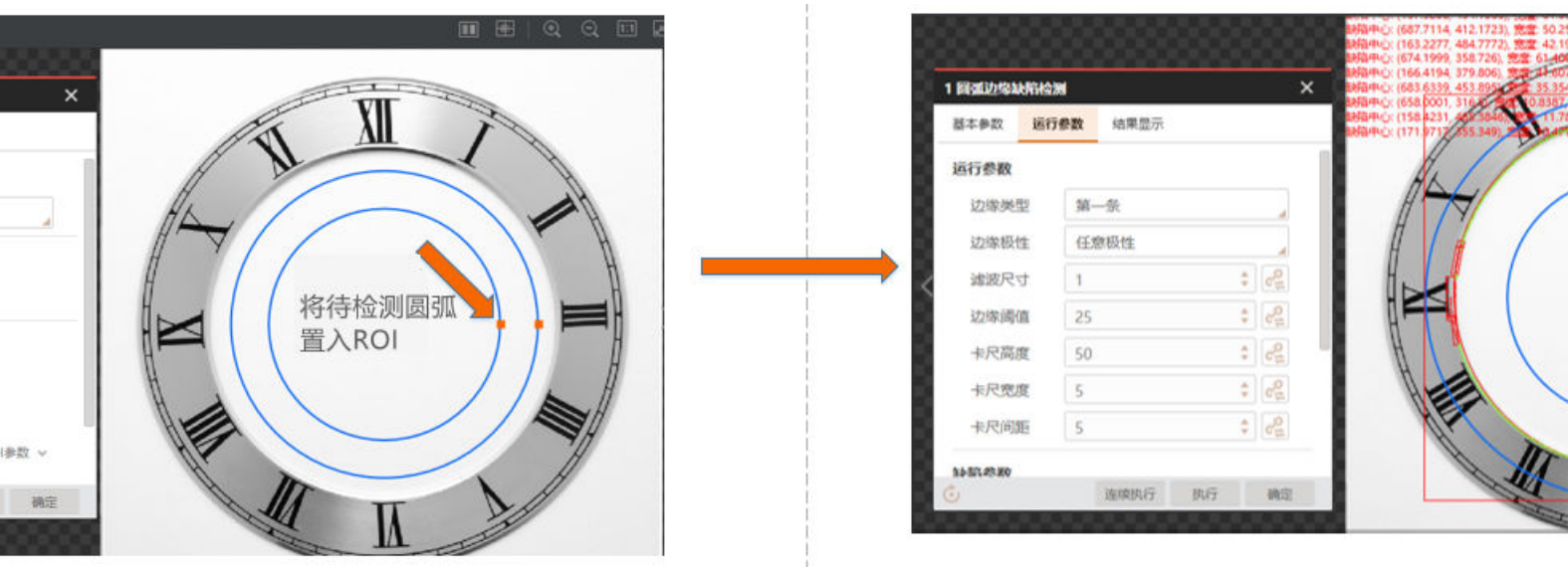


图 13-254 配置方法简易图示

参数配置

圆输入

可以选择“按圆”输入，直接链接前面圆查找的输出圆。也可以选择“按参数”自定义圆的圆心坐标、半径。

ROI 区域

可以继承也可以绘制 ROI 区域。

边缘类型

有最强、第一条和最后一条三种类型。

边缘极性

有从白到黑、从黑到白和任意极性三种极性。

滤波尺寸

用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间距离小于滤波尺寸时反而会影响边缘位置的精度甚至丢失边缘，该值须要根据实际情况设置。

边缘阈值

边缘阈值即梯度阈值，范围 0~255，只有边缘梯度阈值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。

卡尺高度

在 ROI 中环形分布若干个边缘点查找 ROI，该值描述扫描边缘点查找 ROI 的区域高度。当边缘查找不准确时可适当增大该值。

卡尺宽度

在一定范围内增大该值可以获取更加稳定的边缘点。

卡尺间距

在 ROI 中环形分布若干个边缘点查找 ROI，每个 ROI 之间的像素间距。

缺陷极性

有轨迹左侧、右侧和轨迹两侧等三种极性，沿着检测框 BOX 的方向看，检测边缘的左侧为轨迹左侧，其他的依次对应。

缺陷距离阈值

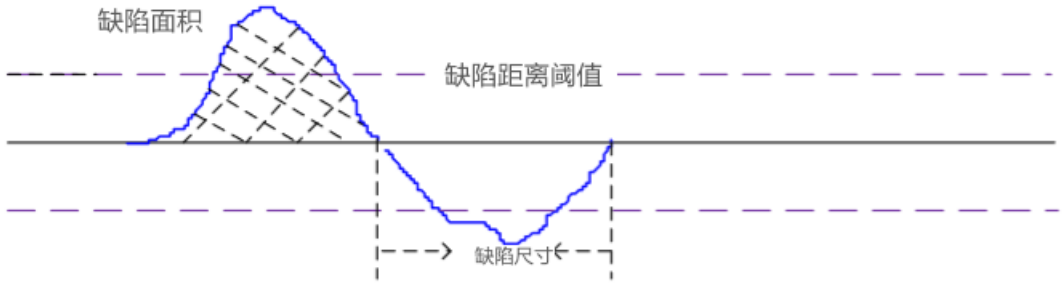
边缘点距离拟合直线的距离，若距离大于阈值，则判定为待筛选缺陷点，若尺寸或面积使能打开，则需要进一步根据对应阈值进行筛选。

缺陷尺寸使能

多个缺陷点投影到拟合直线，组成的像素尺寸大于阈值，则判定为缺陷尺寸生效。

缺陷面积使能

缺陷轮廓与标准直线围成的面积是缺陷面积，缺陷面积在使能设置范围内的缺陷才可能被查找到，如下图所示。



卡尺数量

用于扫描边缘点的 ROI 区域数量。

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，为获取更佳查找效果，建议与剔除距离结合使用。

剔除阈值

允许离群点到拟合圆的最大像素距离，值越小，排除点越多。

追踪容忍度

边缘追踪所允许偏移的最大像素。

模块结果

*圆弧边缘缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

ROI 缺陷个数

int 型，数组形式，代表每个 ROI 检测到的缺陷个数。

检测圆弧

ANNULUS 型，代表检测圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点。

检测圆弧中心 X

float 型，代表圆弧 ROI 中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表圆弧 ROI 中心点的 Y 坐标。

检测圆弧内径

float 型，代表圆弧 ROI 的内径。

检测圆弧外径

float 型，代表圆弧 ROI 的外径。

检测圆弧起始角度

float 型，代表圆弧 ROI 起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

检测圆弧角度范围

float 型，代表圆弧 ROI 从起始边开始至终止边所跨过的角度范围。

缺陷框

box 型，代表缺陷框。

缺陷框中心点

point 型，代表陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷框索引

int 型，代表缺陷框的索引、序号，通常为顺序递增的数组，数组长度和缺陷框数据的长度相同。

缺陷尺寸

float 型，代表沿着轨迹方向的缺陷长度。

缺陷面积

float 型，代表缺陷框的面积。

卡尺起始值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的起始序号点。

卡尺终止值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的终止序号点。

缺陷类型

int 型，代表缺陷的类型。

- 1 代表缺陷。
- 2 代表断裂。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

边缘点

表示多直线查找模块检测到的边缘点。

轮廓点 X

float 型，代表边缘轮廓点的 X 坐标。

轮廓点 Y

float 型，代表边缘轮廓点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

ROI 边缘点个数

int 型，数组形式，代表每个 ROI 检测到的边缘点个数。

卡尺点

point 型，代表理想中检测到边缘点。

理想卡尺点 X

float 型，代表理想中检测到边缘点的 X 坐标。

理想卡尺点 Y

float 型，代表理想中检测到边缘点的 Y 坐标。

理想卡尺点状态

int 型，代表理想中直线边缘点的创建状态。1 表示创建成功，0 表示创建失败。

理想卡尺点数量

int 型，代表理想中直线边缘点的数量。

ROI 理想卡尺点数量

int 型，代表每个 ROI 理想检测到的直线边缘点数量。

标准圆环

ANNULUS 型，代表标准圆环。

标准圆环圆心

point 型，代表标准圆环中线点。

标准圆环圆心 X

float 型，代表标准圆环中心点的 X 坐标。

标准圆环圆心 Y

float 型，代表标准圆环中心点的 Y 坐标。

标准圆环内径

float 型，代表标准圆环内侧圆环的圆半径。

标准圆环外径

float 型，代表标准圆环外侧圆环的圆半径。

标准圆环起始角度

float 型，代表标准圆环起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

标准圆环角度范围

float 型，代表标准圆环从起始边开始至终止边所跨过的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.5.10 边缘组合缺陷检测

*边缘组合缺陷检测*模块可组合*直线边缘缺陷检测*和*圆弧边缘缺陷检测*进行边缘缺陷检测，并输出缺陷个数、缺陷位置和缺陷大小等信息。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

在流程中，该模块的：

- 前序模块一般为 *图像源* 、*模板匹配* 和 *位置修正* 。这些模块搭配使用可对存在位姿变化的物料进行特征定位。。
- 后序模块可使用 *脚本* 和 *发送数据* ，将数据按指定格式发出。



图 13-255 应用示例

参数配置

可在该模块配置窗口的*运行参数*页签配置*直线边缘缺陷检测*和*圆弧边缘缺陷检测*模块的基本参数和运行参数。

- 基本参数详情，请参见 *基本参数* 。
- 运行参数详情，请参见 *直线边缘缺陷检测* 和 *圆弧边缘缺陷检测* 。
- 结果显示参数详情，请参见 *结果显示* 。



图 13-256 运行参数页签

模块结果

*边缘组合缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

缺陷框

box 型，代表缺陷框。

缺陷框中心点

point 型，代表陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷尺寸

float 型，代表沿着轨迹方向的缺陷长度。

缺陷面积

float 型，代表缺陷框的面积。

卡尺起始值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的起始序号点。

卡尺终止值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的终止序号点。

缺陷类型

int 型，代表边缘缺陷的类型。

- 6 代表边缘位置偏移。
- 8 代表边缘断裂缺陷 。
- 9 代表边缘阶梯差缺陷。

边缘点

表示多直线查找模块检测到的边缘点。

轮廓点 X

float 型，代表边缘轮廓点的 X 坐标。

轮廓点 Y

float 型，代表边缘轮廓点的 Y 坐标。

轮廓点状态

int 型，代表每个轮廓点是否被检测出。0 代表未检测到轮廓点，1 代表检测到轮廓点。

边缘点个数

int 型，代表输出的边缘点个数。

卡尺点

point 型，代表理想中检测到边缘点。

理想卡尺点 X

float 型，代表理想中检测到边缘点的 X 坐标。

理想卡尺点 Y

float 型，代表理想中检测到边缘点的 Y 坐标。

理想卡尺点状态

int 型，代表理想中直线边缘点的创建状态。1 表示创建成功，0 表示创建失败。

理想卡尺点数量

int 型，代表理想中直线边缘点的数量。

标准圆环

ANNULUS 型，代表标准圆环。

标准圆环圆心

point 型，代表标准圆环中线点。

标准圆环圆心 X

float 型，代表标准圆环中心点的 X 坐标。

标准圆环圆心 Y

float 型，代表标准圆环中心点的 Y 坐标。

标准圆环内径

float 型，代表标准圆环内侧圆环的圆半径。

标准圆环外径

float 型，代表标准圆环外侧圆环的圆半径。

标准圆环起始角度

float 型，代表标准圆环起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

标准圆环角度范围

float 型，代表标准圆环从起始边开始至终止边所跨过的角度范围。

标准直线

line 型，代表标准直线。

标准直线起点

point 型，代表标准直线起点。

标准直线起点 X

float 型，代表标准直线起点的 X 坐标 。

标准直线起点 Y

float 型，代表标准直线起点的 Y 坐标 。

标准直线终点

point 型，代表标准直线终点。

标准直线终点 X

float 型，代表标准直线终点的 X 坐标 。

标准直线终点 Y

float 型，代表标准直线终点的 Y 坐标 。

标准直线角度

float 型，代表标准直线相对于 X 轴正方向顺时针偏移的角度。

13.5.11 边缘对组合缺陷检测

*边缘对组合缺陷检测*可组合*直线对缺陷检测*和*圆弧对缺陷检测*进行边缘对缺陷检测，并输出缺陷的数量、位置和面积等信息。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

在流程中，该模块的：

- 前序模块可为 *图像源* 、*模板匹配* 和 *位置修正* 。这些模块搭配使用可对存在位姿变化的物料进行特征定位。还可在前序模块中搭配 *圆查找* 和 *直线查找* ，获取边缘直线或圆弧作为该模块的基准边缘，提高缺陷查找效率。
- 后序模块可使用 *脚本* 和 *发送数据* ，将数据按指定格式发出。

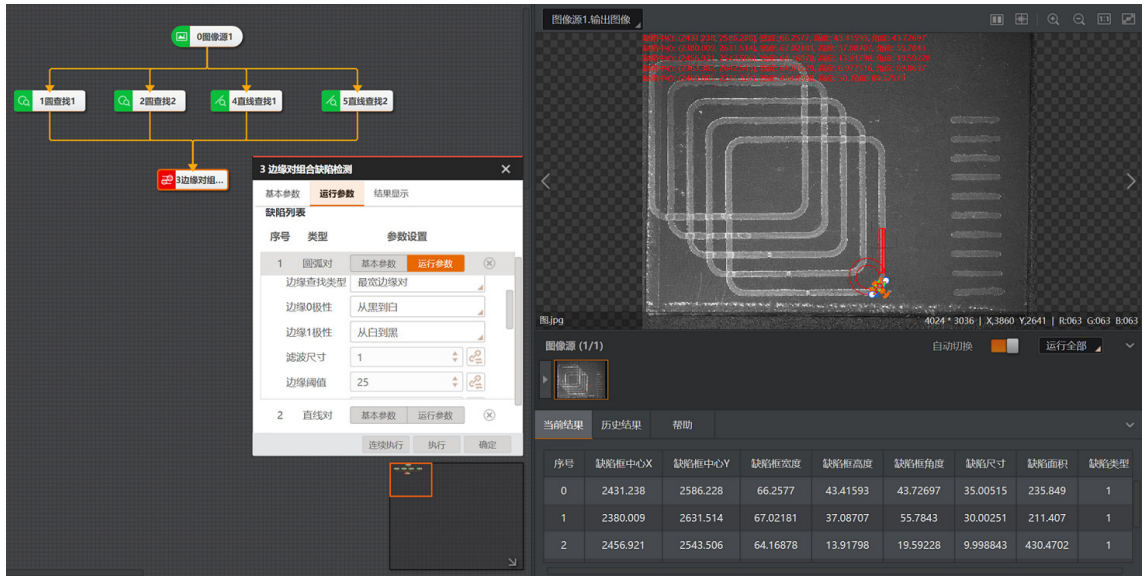


图 13-257 应用示例

参数配置

可在该模块配置窗口的*运行参数*页签配置这两个模块的基本参数和运行参数。

- 基本参数详情，请参见 *基本参数* 。
- 运行参数包含**中心点使能**和**缺陷列表**，**缺陷列表**参见 *直线对缺陷检测* 和 *圆弧对缺陷检测* 。

  中心点使能

  开启后，该模块将输出边缘 0 和边缘 1 之间中点的 X/Y 轴坐标以及中点状态（1：正常，0：异常） 。
- 结果显示参数详情，请参见 *结果显示* 。

模块结果

*边缘对组合缺陷检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

缺陷个数

int 型，代表检测到的缺陷个数。

缺陷框

box 型，代表缺陷框。

缺陷框中心点

point 型，代表陷框中心点。

缺陷框中心 X

float 型，代表缺陷框中心点的 X 坐标。

缺陷框中心 Y

float 型，代表缺陷框中心点的 Y 坐标。

缺陷框宽度

float 型，代表缺陷框的宽度。

缺陷框高度

float 型，代表缺陷框的高度。

缺陷框角度

float 型，代表缺陷框以初始点开始顺时针旋转的角度。

缺陷尺寸

float 型，代表沿着轨迹方向的缺陷长度。

缺陷面积

float 型，代表缺陷框的面积。

卡尺起始值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的起始序号点。

卡尺终止值

int 型，代表根据卡尺宽度和卡尺间距从直线起始点开始顺序放置序号点的终止序号点。

缺陷类型

int 型，代表边缘缺陷的类型。

- 6 代表边缘位置偏移。
- 8 代表边缘断裂缺陷 。
- 9 代表边缘阶梯差缺陷。

边缘*轮廓点（*代表 0-1）

边缘*轮廓点 X

float 型，代表边缘直线*轮廓点的 X 坐标。

边缘*轮廓点 Y

float 型，代表边缘直线*轮廓点的 Y 坐标。

边缘*轮廓点状态（*代表 0-1）

int 型，代表用于检测出边缘直线的轮廓点的状态。1 代表轮廓点符合要求，0 代表轮廓点不符合要求。

边缘*轮廓点数（*代表 0-1）

int 型，代表检测到的直线*的边缘点个数。

边缘中点

point 型，代表实际检测到的边缘 0 或边缘 1 的中点。

边缘中点 X

float 型，代表实际检测到的边缘 0 或边缘 1 的中点的 X 坐标。

边缘中点 Y

float 型，代表实际检测到的边缘 0 或边缘 1 的中点的 Y 坐标。

边缘中点状态

int 型，1 代表正常，其他代表异常。

边缘中点数

int 型，代表“直线 0 与直线 1”或“圆弧 0 与圆弧 1”对应轮廓点连线中点的个数。

卡尺点

point 型，代表理想中检测到边缘点。

理想卡尺点 X

float 型，代表理想中检测到边缘点的 X 坐标。

理想卡尺点 Y

float 型，代表理想中检测到边缘点的 Y 坐标。

理想卡尺点状态

int 型，代表理想中直线边缘点的创建状态。1 表示创建成功，0 表示创建失败。

理想卡尺点数量

int 型，代表理想中直线边缘点的数量。

标准圆环

ANNULUS 型，代表标准圆环。

标准圆环圆心

point 型，代表标准圆环中线点。

标准圆环圆心 X

float 型，代表标准圆环中心点的 X 坐标。

标准圆环圆心 Y

float 型，代表标准圆环中心点的 Y 坐标。

标准圆环内径

float 型，代表标准圆环内侧圆环的圆半径。

标准圆环外径

float 型，代表标准圆环外侧圆环的圆半径。

标准圆环起始角度

float 型，代表标准圆环起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

标准圆环角度范围

float 型，代表标准圆环从起始边开始至终止边所跨过的角度范围。

标准直线

line 型，代表标准直线。

标准直线起点

point 型，代表标准直线起点。

标准直线起点 X

float 型，代表标准直线起点的 X 坐标 。

标准直线起点 Y

float 型，代表标准直线起点的 Y 坐标 。

标准直线终点

point 型，代表标准直线终点。

标准直线终点 X

float 型，代表标准直线终点的 X 坐标 。

标准直线终点 Y

float 型，代表标准直线终点的 Y 坐标 。

标准直线角度

float 型，代表标准直线相对于 X 轴正方向顺时针偏移的角度。

缺陷综合信息

最小宽度

float 型，代表模块生成的直线对或圆弧对之间的最小宽度。

最大宽度

float 型，代表模块生成的直线对或圆弧对之间的最大宽度。

平均宽度

float 型，代表模块生成的直线对或圆弧对之间的平均宽度。

近似面积

float 型，代表模块生成的直线对或圆弧对之间的近似面积总和。

13.5.12 边缘位置趋势分析

*边缘位置趋势分析*模块可通过边缘点的距离信息分析出物体边缘的走势，输出图像中“边缘点坐标”和“边缘点与 ROI 边缘距离”等位置信息。通过这些信息可进一步获取到期望的图像边缘点。

本节包含如下内容：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

边缘类趋势分析包括边缘趋势分析和边缘对趋势分析，是边缘缺陷分析库的重要组成部分，均属于卡尺系工具。该模块通过边缘点到查找到的 ROI 某边的距离信息分析物体边缘的整体走势。

算法处理流程如下：

1. 根据预设的运行参数在 ROI 区域内获取符合要求的卡尺区域集合。
2. 在每个卡尺区域获取一个边缘点，并获取所有边缘点到 ROI 某边的距离信息（包括最大值、最小值、平均值等）。
3. 根据设置的距离阈值提取满足要求的点集，并得到该点集距离中的最大距离、最小距离及平均距离，同时分别输出一个最大距离和最小距离对应的点。

使用方法

该模块前序后模块详情见下表。

表 13-18 前后序模块

前后序模块	描述
前序模块	如果待检测目标在视野内存在位姿变化，该模块的前序模块一般可包括*图像源*、*模板匹配*、*Blob 分析*和*位置修正*等。其中*模板匹配*或*BLOB 分析*用于进行粗定位，*位置修正*用于实时修正检测区域。

前后序模块	描述
	如果待检测目标在视野内无位姿变化，该模块可仅与*图像源*搭配使用。
后序模块	该模块后序模块可接 *点集*，以获取图像边缘点和最大/最小距离点。

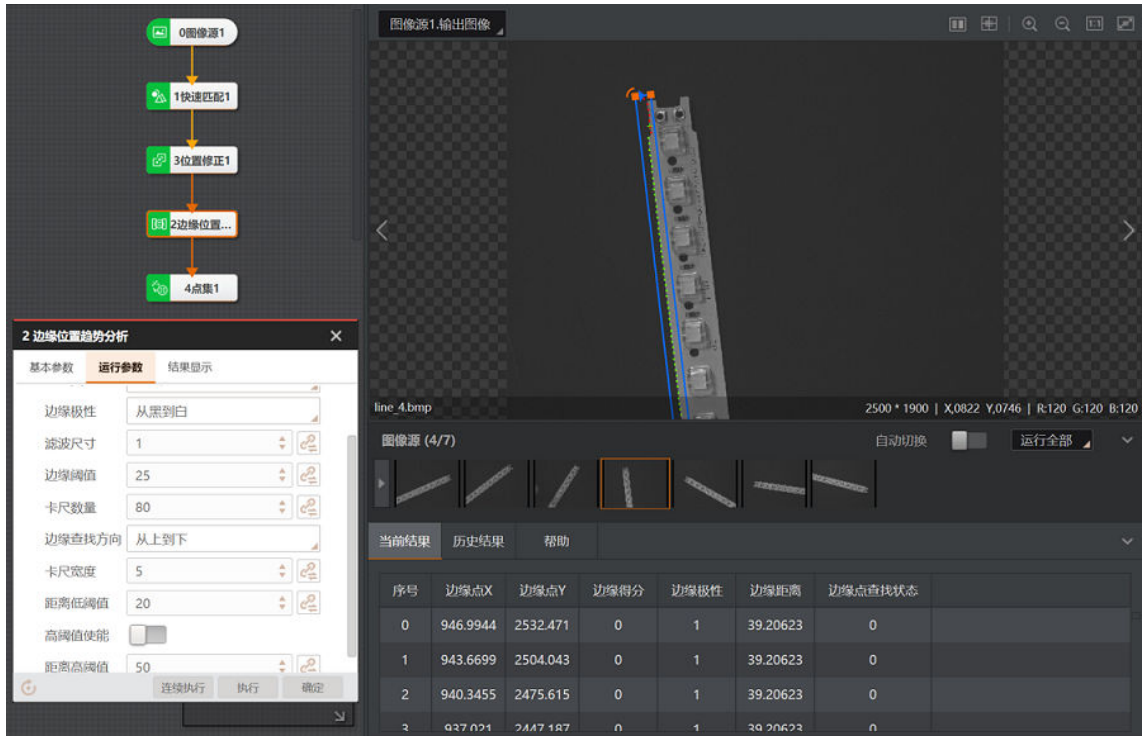


图 13-258 应用示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数*。
- 结果显示参数详情，请参见 *结果显示*。

表 13-19 运行参数

参数	描述
边缘类型	可设置边缘检测的类型，可选最强、第一条边缘和最后一条边缘。

参数	描述
	• 最强：只检测扫描范围内梯度阈值最大的边缘点集合。 • 第一条边缘：只检测扫描范围内与搜索起始点最近的边缘点集合。 • 最后一条边缘：只检测扫描范围内与搜索终止点最近的边缘点集合。
边缘极性	可设置边缘检测的极性，可选任意极性、从黑到白、从白到黑。 • 从黑到白：从灰度值低的区域过渡到灰度值高的区域的边缘。 • 从白到黑：从灰度值高的区域过渡到灰度值低的区域的边缘。 • 任意极性：以上两种边缘均可检测。
滤波尺寸	用于增强边缘和抑制噪声，最小值为 1。需根据实际需求设置。 当边缘模糊或有噪声干扰时，增大该值有利于使检测结果更稳定。但若边缘与边缘之间距离小于滤波尺寸，反而会影响边缘位置的精度甚至丢失边缘。
边缘阈值	边缘阈值即梯度阈值，范围 0~255。 当边缘梯度阈值大于该值的边缘点，方可被检测到。 数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。
卡尺数量	查找 ROI 区域内，用于扫描边缘点的卡尺 ROI 数量。
边缘查找方向	可设置查找边缘的方向，可选从上到下、从左到右。 • 从上到下：ROI 角度为 0°时，表示相对于设置的 ROI 区域，按从上往下的顺序查找边缘点。若调整 ROI 角度，则查找方向需同步调整角度。 • 从左到右：ROI 角度为 0°时，表示相对于设置的 ROI 区域，按从左往右的顺序查找边缘点。若调整 ROI 角度，则查找方向需同步调整角度。
卡尺宽度	即查找边缘点 ROI 的区域宽度，在一定范围内增大该值可获取更稳定的边缘点。 同时边缘点 ROI 的区域高度由查找 ROI 的高度决定。
距离低阈值	对应距离最小值。需配合高阈值一起使用，高阈值相关介绍请见高阈值使能和距离高阈值参数。

参数	描述
	• 当设置的距离低阈值＜距离高阈值时，则边缘距离满足“≥低阈值且≤高阈值”的边缘点集组成目标点集。 • 当设置的距离低阈值≥距离高阈值时，则边缘距离满足“＞低阈值或＜高阈值”的边缘点组成目标点集。
高阈值使能	开启该功能时，根据所有边缘点集的边缘距离分布自适应计算距离高阈值；关闭该功能时，则自定义设置距离高阈值。
距离高阈值	仅在关闭高阈值使能时，有该参数。对应距离的最大值。

模块结果

*边缘位置趋势分析*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

边缘点个数

int 型，代表输出的边缘点个数。

边缘点提取总数

int 型，代表边缘趋势上的边缘点个数。

边缘点最小距离

float 型，代表边缘点之间的最小距离 。

最小距离索引点

点 X

float 型，代表最小距离的点的 X 轴坐标。

点 Y

float 型，代表最小距离的点的 Y 轴坐标。

边缘点最大距离

float 型，代表边缘点之间的最大距离。

最大距离索引点

点 X

float 型，代表最大距离的点的 X 轴坐标。

点 Y

float 型，代表最大距离的点的 Y 轴坐标。

边缘点平均距离

float 型，代表边缘点之间的平均距离。

目标边缘点总数

int 型，代表边缘趋势上的边缘点个数。

目标边缘点最小距离

float 型，代表边缘趋势上的边缘点之间的最小距离。

目标边缘点最大距离

float 型，代表边缘趋势上的边缘点之间的最大距离。

目标边缘点平均距离

float 型，代表边缘趋势上的边缘点之间的平均距离。

卡尺数量

int 型，对应运行参数中的卡尺数量。

边缘点

point 型，代表检测到的边缘点。

边缘点 X

float 型，代表检测到边缘点的 X 坐标。

边缘点 Y

float 型，代表检测到边缘点的 Y 坐标。

边缘得分

float 型，代表边缘对的得分。

边缘极性

int 型，代表边缘点的极性。

边缘距离

float 型，代表边缘点之间的距离。

边缘点查找状态

int 型，代表边缘点是否在边缘上。1 代表正常，其他代表异常。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.5.13 边缘对位置趋势分析

*边缘对位置趋势分析*模块可输出图像中边缘对之间的信息（如边缘对的距离）。通过这些信息可筛选点位。

模块原理

边缘类趋势分析包括边缘趋势分析和边缘对趋势分析，是边缘缺陷分析库的重要组成部分，均属于卡尺系工具。该模块通过边缘点到查找到的点对距离信息分析物体边缘的整体走势。算法处理流程如下：

1. 根据预设的运行参数在 ROI 区域内获取符合要求的卡尺区域集合。
2. 在每个卡尺区域获取一个边缘点对（即两个边缘点），并获取所有边缘点对各自的距离信息（包括最大值、最小值、平均值等）。
3. 最终根据设置的距离阈值提取满足要求的点对集，并得到该点对集距离中的最大距离、最小距离及平均距离，同时分别输出一个最大距离和最小距离对应的点。

使用方法

该模块前序后模块详情见下表。

表 13-20 前后序模块

前后序模块	描述
前序模块	如果待检测目标在视野内存在位姿变化，该模块的前序模块一般可包括*图像源*、*模板匹配*、*Blob 分析*和*位置修正*等。其中*模板匹配*或*BLOB 分析*用于进行粗定位，*位置修正*用于实时修正检测区域。 如果待检测目标在视野内无位姿变化，该模块可仅与*图像源*搭配使用。
后序模块	该模块后序模块可接*点集*模块，以获取图像边缘点和最大/最小距离点。还可接*脚本*和*发送数据*，以将数据按指定格式发出。

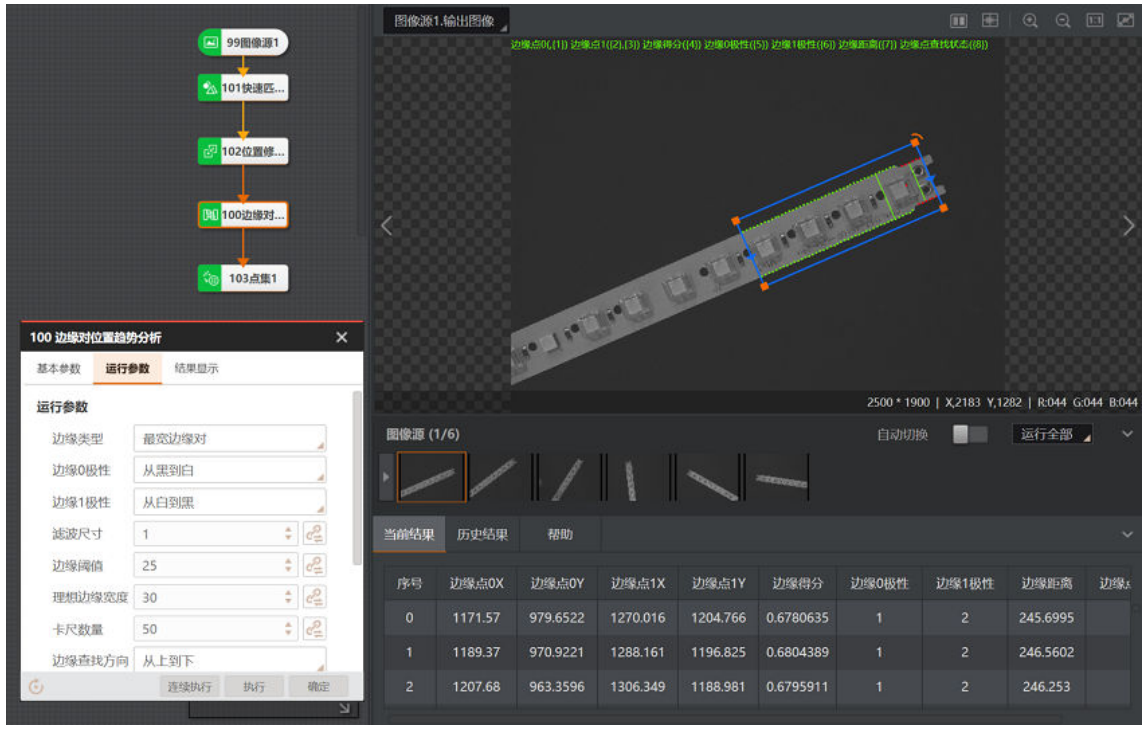


图 13-259 应用示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见*基本参数*。
- 结果显示参数详情，请参见*结果显示*。

表 13-21 运行参数

参数	描述
边缘类型	设置边缘检测的类型，可选最宽边缘对、最窄边缘对、最强边缘对、最弱边缘对、第一个边缘对、最后一个边缘对、最接近边缘对、最不接近边缘对。 • 最宽边缘对：检测扫描范围内间距最大的边缘对集合。 • 最窄边缘对：检测扫描范围内间距最小的边缘对集合。 • 最强边缘对：检测扫描范围内边缘对平均梯度值最大的边缘对集合。 • 最弱边缘对：检测扫描范围内边缘对平均梯度值最小的边缘对集合。 • 第一个边缘对 : 检测扫描范围内边缘对中心与搜索起始点最近的边缘对集合。 • 最后一个边缘对 : 检测扫描范围内边缘对中心与搜索终止点最近的边缘对集合。 • 最接近边缘对 : 检测扫描范围内边缘对距离与**理想边缘宽度**参数差异最小的边缘对集合。 • 最不接近边缘对 : 检测扫描范围内边缘对距离与**理想边缘宽度**参数异最大的边缘对集合。
边缘 0/1 极性	可设置边缘 0/1 检测的极性，可选任意极性、从黑到白、从白到黑。 • 从黑到白：从灰度值低的区域过渡到灰度值高的区域的边缘。 • 从白到黑：从灰度值高的区域过渡到灰度值低的区域的边缘。 • 任意极性：以上两种边缘均可检测。 边缘 0 为边缘对中的首边缘，边缘 1 为边缘对中的尾边缘。
滤波尺寸	用于增强边缘和抑制噪声，最小值为 1。当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间距离小于滤波尺寸时反而会影响边缘位置的精度甚至丢失边缘，该值须要根据实际情况设置。
边缘阈值	边缘阈值即梯度阈值，范围 0~255，只有边缘梯度阈值大于该值的边缘点才被检测到。数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。
理想边缘宽度	期望找到的边缘对集合中每对边缘的绝对距离。 当**边缘类型**选择最接近边缘对或最不接近边缘对时生效，选择其他边缘类型时该参数无效。
卡尺数量	查找 ROI 区域内，用于扫描边缘点的卡尺 ROI 数量。

参数	描述
边缘查找方向	可设置查找边缘的方向，可选从上到下、从左到右。 • 从上到下：ROI 角度为 0°时，表示相对于设置的 ROI 区域，按从上往下的顺序查找边缘点。若调整 ROI 角度，则查找方向需同步调整角度。 • 从左到右：ROI 角度为 0°时，表示相对于设置的 ROI 区域，按从左往右的顺序查找边缘点。若调整 ROI 角度，则查找方向需同步调整角度。
卡尺宽度	即查找边缘点 ROI 的区域宽度，在一定范围内增大该值可获取更稳定的边缘点。 同时边缘点 ROI 的区域高度由查找 ROI 的高度决定。
距离低阈值	对应距离最小值。需配合高阈值一起使用，高阈值相关介绍请见高阈值使能和距离高阈值参数。 • 当设置的距离低阈值＜距离高阈值时，则边缘距离满足“≥低阈值且≤高阈值”的边缘点对集组成目标点对集合。 • 当设置的距离低阈值≥距离高阈值时，则边缘距离满足“＞低阈值或＜高阈值”的边缘点对集组成目标点对集合。
高阈值使能	开启该功能时，根据所有边缘点对集的边缘距离分布自适应计算距离高阈值；关闭该功能时，则自定义设置距离高阈值。
距离高阈值	仅在关闭高阈值使能时，有该参数。对应距离的最大值。
中心点使能	开启后，该模块将输出的边缘 0 和边缘 1 之间中点的 X/Y 轴坐标以及中点状态（1：正常，0：异常）。

模块结果

*边缘对位置趋势分析*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

边缘对个数

int 型，代表共查找的边缘对个数。

边缘对提取总数

int 型，代表查找的边缘对有效个数。

边缘对最小距离

float 型，代表边缘对中的最小距离。

最小距离拟合直线

拟合直线点*（*代表 0-1）

拟合点*X

float 型，代表边缘对中最小距离的边缘*的点的 X 轴坐标。

拟合点*Y

float 型，代表边缘对中最小距离的边缘*的点的 Y 轴坐标。

边缘对最大距离

float 型，代表边缘对中的最大距离。

最大距离拟合直线

拟合直线点*（*代表 0-1）

拟合点*X

float 型，代表边缘对中最大距离的边缘*的点的 X 轴坐标。

拟合点*Y

float 型，代表边缘对中最大距离的边缘*的点的 Y 轴坐标。

边缘对平均距离

float 型，代表边缘对的平均距离。

目标边缘对总数

int 型，代表有效边缘对的总个数。

目标边缘对最小距离

float 型，代表有效边缘对间的最小距离。

目标边缘对最大距离

float 型，代表有效边缘对间的最大距离。

目标边缘对平均距离

float 型，代表有效边缘对间的平均距离。

卡尺数量

int 型，对应运行参数中的卡尺数量。

边缘点*（*代表 0-1）

边缘点*X

float 型，代表边缘对中的边缘*的 X 轴坐标。

边缘点*Y

float 型，代表边缘对中的边缘*的 Y 轴坐标。

边缘中点

point 型，代表实际检测到的边缘 0 或边缘 1 的中点。

边缘中点 X

float 型，代表实际检测到的边缘 0 或边缘 1 的中点的 X 坐标。

边缘中点 Y

float 型，代表实际检测到的边缘 0 或边缘 1 的中点的 Y 坐标。

边缘中点状态

int 型，1 代表正常，其他代表异常。

边缘得分

float 型，代表边缘对的得分。

边缘*极性（*代表 0-1）

int 型，代表边缘对中边缘*的极性。

边缘距离

float 型，代表边缘点之间的距离。

边缘点查找状态

int 型，代表边缘点是否在边缘上。1 代表正常，其他代表异常。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6 边缘学习

边缘学习属于深度学习的一种，相关介绍参见 *深度学习* 。

基于运行环境差异分类

与深度学习类模块一样，边缘学习类模块也分为 C 和 G 两种类型，这两种模块的原理与参数基本一致，仅在于运行环境不同。

表 13-22 边缘学习类模块分类（基于运行环境差异）

类型	运行环境	要求
模块名称后缀为“C”	基于 CPU 运行	推荐使用 i7-8 代及以上的 CPU
模块名称后缀为“G”	基于显卡 GPU 运行	确保使用显存 6 GB 及以上的独立显卡。显卡相关推荐参见 *推荐硬件* 。

基于原理差异分类

按工作原理分类，边缘学习类模块可细分为“分类”、“检测”和“分割”三个类型。

表 13-23 边缘学习分类（基于原理差异）

类型	模块	检测目标特点	使用场景	功能优势
分类	*注册分类*	• 多类别（最多支持 100 个类别）、多目标 • 不同类样本差异明显、同类样本一致性好的物体	适用于每个类别样本量较少，且在使用过程中有新增类别的场景	/
	异常分类	• OK 和 NG 两种类别 • 两类检测目标类间的关键特征存在可辨识差异 • 同类检测目标间存在轻微差异	适用于每个类别样本量较少且在使用过程中明确不存在第三种类别的场景	• 相较于 *注册分类* 具有更高的准确率和分类效率 • 通过学习 10 张以内的样本数据即可实现有效分类

类型	模块	检测目标特点	使用场景	功能优势
	多类别分类	• 多类别（至少 2 类，最多支持 32 类） • 不同类检测目标类间差异较明显 • 同类检测目标间存在轻微差异	适用于每个类别样本量较少，且在使用过程中有新增类别的场景	• 相较于 *注册分类* 具有更高的准确率和分类效率 • 通过学习 10 张以内的样本数据即可实现有效分类
检测	*注册检测*	• 单类别、多目标 • 形态变化小、无重叠和背景干扰的物体	适用于产线换型频繁、要求模型迭代速度快的定位检测场景	/
	全监督检测	• 多类别、多目标 • 形态变化小、无严重重叠的物体或区域 • 检测目标和背景存在明显的可辨识差异	适用于简单的定位分类场景	• 相较于 *注册检测* 模块，检出率和准确率更高 • 通过学习少量数据即可满足简单场景的检测需求
	有无检测	• 单类别、单目标 • 形态存在变化的柔性物体 • 检测目标目标和背景存在可辨识差异	适用于简单的目标存在性检测场景	通过学习 10 张以内的样本数据即可满足简单场景的检测需求
	学习计数	• 单类别、多目标 • 形态存在变化的柔性物体 • 检测目标间无重叠，且目标和背景存在可辨识差异	适用于简单的目标数量统计场景	通过学习 10 张以内的样本数据即可满足简单场景的计数需求
分割	*注册分割*	• 单类别、多目标 • 形态简单、相互无遮挡的物体或区域 • 检测目标和背景存在明显的可辨识差异	适用于通过注册模板分割简单场景中的对应类别目标	/

13.6.1 学习计数

学习计数模块可对单一类别的目标进行计数和定位，适用于形态存在变化的柔性物体，例如挤压变形的零食袋、褶皱的快递标签等。该模块所需训练数据量较少，可通过学习少量注册图像实现检测计数，其中注册图像应标注全部待计数目标。

模块原理

学习计数模块通过从注册图像中提取目标的特征数据，并在检测图像中搜索并定位具有同类特征的目标，从而实现目标数量统计功能。该模块仅支持对图像中单个类别的目标进行检测与计数。

算法分为注册和检测两个步骤，并支持对效果不佳的异常样本追加学习，如下图所示。

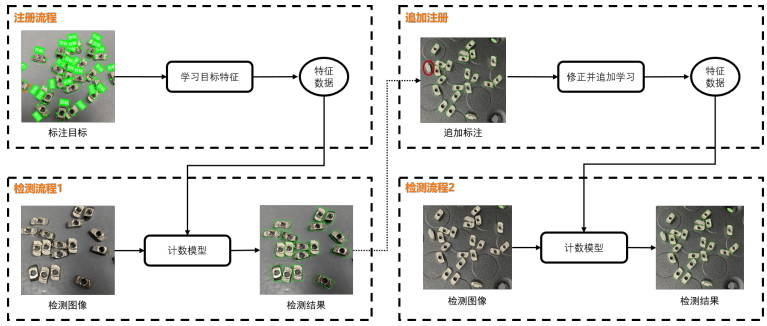


图 13-260 算法流程

1. 注册步骤：算法获取到待注册图像，这些图像包含目标标注。基于这些标注，算法通过深度学习提取目标的特征数据。
2. 检测步骤：算法将注册步骤中的特征数据与待检测图像输入至计数模型，在检测图像中搜索并定位具有同类特征的目标。
3. 追加注册与检测步骤：当出现新样本或异常样本时，可修正检测结果并作为追加注册数据。算法获取到追加注册图像和修正结果后，学习这些新样本的特征并更新到检测流程中，从而能够适应新样本并提升检测效果。

使用方法

您可以参照如下内容，使用该模块检测简单场景中单一类别的目标，并对其进行计数和定位。

在流程中添加该模块后，需完成注册类别，详情请参见 *注册类别* 。

前后序模块

使用该模块时，需确保前序模块有 *图像源* 模块，采集需进行检测的图像。

配置步骤

1. 在流程中添加*图像源*和*学习计数*，并将*学习计数*模块连接至*图像源*模块。
2. 双击*学习计数*配置相应参数。
3. 在**注册类别**页签，完成注册。
4. 执行流程进行检测。
5. 当出现新样本或未检出目标的异常样本时，在**注册类别**页签单击*追加注册*，将新特征数据或修正后的特征数据更新至检测流程中。
6. 执行流程再次进行检测。

示例方案

如下图所示，注册少量含多个方形螺母的样本图像后，您可通过该模块实现对方形螺母的计数和定位，从而完成视野内单一类别目标数量的统计任务。

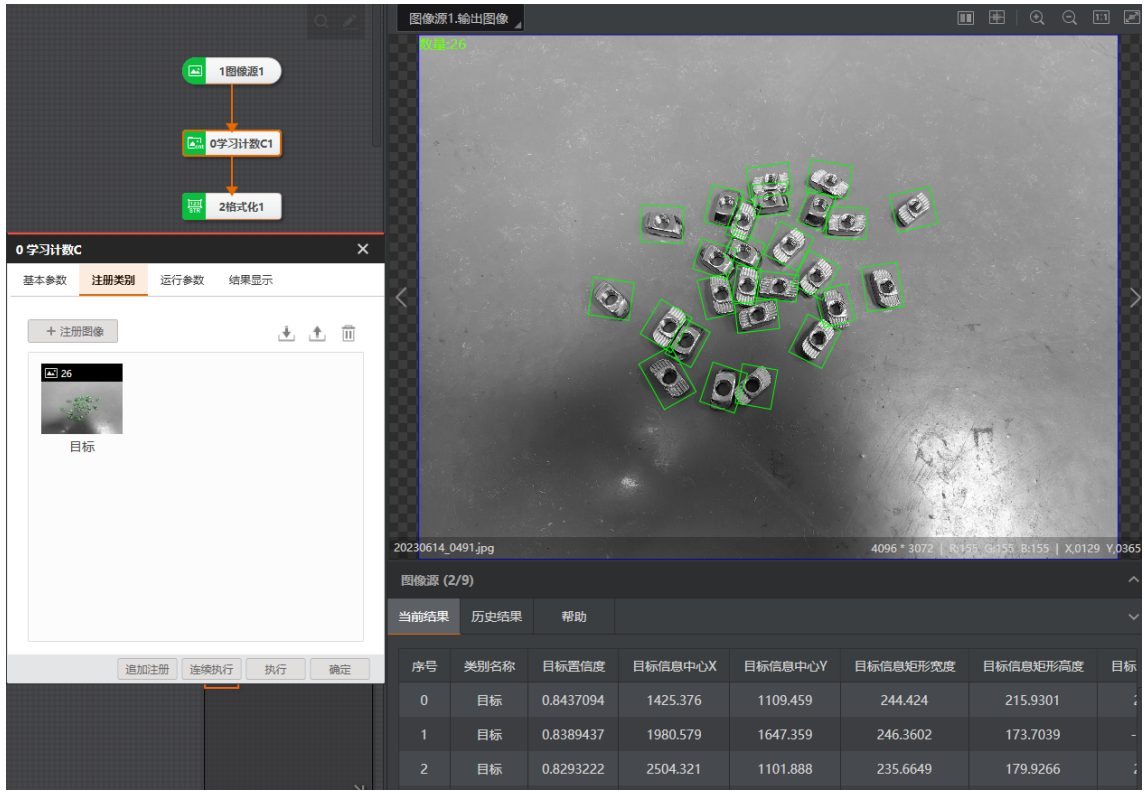


图 13-261 流程示例

注册类别

注册类别可作为*学习计数*的搜索依据，*学习计数*模块根据注册图类别中已注册的目标进行搜索，可在图像中搜索出单一类型的目标并进行计数。

操作步骤

1. 在**注册类别**页签，单击*+ 注册图像*，进入注册图像窗口。

说明

若已完成注册且检测中出现了新样本或异常样本，可单击该页签的*追加注册*继续注册数据并修正检测结果。此时您会进入注册图像窗口，当前模块图像将自动添加至注册图样本中。

2. 在右侧的配置区域，通过以下任一方式添加注册图。

获取流程图像

添加当前流程运行的实时图像作为注册图。

a. 单击*获取流程图像*进入实时图像预览窗口。
b. （可选）通过图像显示区域左上角下拉框设置“图像切换”或“图像叠加”。具体操作请见 *查看模块图像* 。
c. 单击上方的 或 单次执行或连续执行流程，以获取单张或多张当前流程的实时图像。

说明

- 连续执行流程时，当抓取的图像数达到**图像上限**后，流程将自动停止执行。
- 若已存在模型，获取流程图像时支持同步导入注册图的推理结果。
- 若已设置全局触发，当您返回**注册图像**界面时，一旦满足触发条件，流程图像将自动同步至图像列表。

d. 完成流程图像添加后，单击*返回注册图*。

获取当前图像

获取模块当前订阅的输入图像及其绘制的 ROI 区域。

导入图像

可根据实际需求导入缓存图像或本地图像。

- *缓存图像*：当前序模块包含**缓存图像**时，可导入其缓存的所有图像。
- *导入本地图像*：可从本地中导入单个或多个图像。
- *导入文件夹*：选择本地文件夹后，将导入该文件夹下的全部图像。

说明

若勾选图像预览区下方的*图像保存至模型*，可将添加的注册图保存到方案中、在导出模型的时候保存在模型中。

3. **可选操作**: 通过**标签列表**下各标签右侧的 可预览当前分类标签下的注册目标。

说明

将鼠标悬浮至对应标签上即可显示 。

4. **可选操作**: 在左下角的注册样本列表中选中一张图片，单击**标签列表**的**设为背景图**，将该图片设置为背景图。

为保证注册精度，可将一张不含有目标物的图像设置为背景图。

5. 标注图像。

1) 在左下角的注册样本列表中选中需要标注的图像。

2) 在**标签列表**单击选中**目标**。
3) 单击选中***矩形框选***后，在图像中绘制目标区域。绘制完成后，右键单击已绘制 ROI，可复制粘贴选中 ROI 到指定位置，或按需删除选中 ROI。

说明

具体绘制方式，请参见 *模块中模型配置的掩膜区域* 。

4) 重复以上步骤，直到所有图像中的所有目标均已标注。

说明

该模块适用于单一类别下的多目标检测场景，进行图像标注时需将每张图像上的所有目标均标注出，以确保注册模型精度。

6. **可选操作**: 根据已标记目标进行自动推理和标记。
   - 若已标注至少一个目标，可单击***自动追加***对当前注册图上的其它目标进行自动标注。
   - 若已注册至少一张图像，切换到其它未标记注册图后，单击***自动打标***可对其它注册图上的目标进行自动标注。

说明

进行自动追加或自动打标时，当前注册图的推理结果会导入注册界面并自动显示标签，需手动调整目标区域或标签类别将其加入训练样本。

7. 单击**注册图像**窗口右下角的***注册***。
8. 在**注册内容确认**窗口，配置注册相关参数。

   旋转增强

   若启用该参数，学习的特征会对目标角度变化进行适应和增强。当检测目标存在旋转时，建议开启。
9. 单击***开始注册***，开始训练模型。

   注册成功后，可单击**注册图像**左上角的***流程控制***，通过 / 单次执行或连续执行流程以检验注册效果。

   若注册效果不符合预期可继续微调；若符合预期可关闭该窗口返回**注册类别**页签。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

高级参数

目标排序

设置模块输出预测结果的顺序。

- 按中心点 X/Y 坐标升序：按照预测框中心点的 X/Y 坐标值，从小到大排序预测结果。
- 按置信度降序：按照预测结果的确信程度，从大到小排序预测结果。

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

过滤参数

面积使能

用于筛选预测结果。开启后，需设置**面积范围**。若预测结果框面积超出设置的**面积范围**，则模块舍弃该预测结果。

宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

角度使能

用于筛选预测结果。开启时，需设置**角度范围**。若预测结果框相对于水平方向的旋转超出设置的**角度范围**，则模块舍弃该预测结果。此处的旋转指：预测结果框相对于水平线进行顺时针旋转时，角度大于 0；逆时针旋转时，角度小于 0。

模块结果

学习计数模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

结果类别总数

int 型，代表检出目标类别的总数量。

结果类别名称

string[]型，代表检出目标的类别名称。

目标分类计数

int[]型，代表各个类别下的目标数量。

目标总数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

目标状态

int 型，代表每个定位框的状态。1 代表定位框正常，0 代表定位框异常。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

标签颜色

String 型，代表图像中不同类别标签所对应的 RGB 颜色值，该颜色可在注册图像操作中自定义设置。

目标信息矩形

box 型，代表定位框所在的矩形。

目标信息中心点

point 型，代表每个定位框的中心点。

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

目标面积

int 型，代表每个检测目标的面积大小。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.2 有无检测

有无检测模块可判断场景中是否存在特定目标，对其进行定位并输出坐标位置，适用于形态存在变化的柔性物体，例如挤压变形的零食袋、褶皱的快递标签等。该模块所需训练数据量较少，可通过学习少量注册图像和背景图像实现目标检测。

模块原理

有无检测模块通过从注册图像中提取目标的特征数据，并在检测图像中进行特征匹配并定位目标，从而实现有无检测功能。该模块仅支持对图像中的单个目标进行有无判断。

算法分为注册和检测两个步骤，并支持对效果不佳的异常样本追加学习，如下图所示。

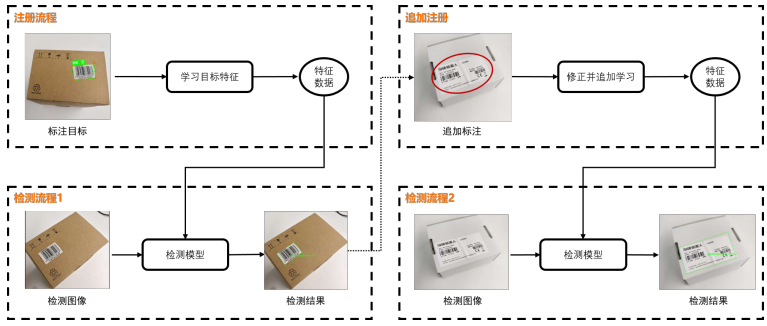


图 13-262 算法流程

1. 注册步骤：算法获取到待注册图像，包含带目标标注的图像和背景图像。基于这些标注，算法通过深度学习提取目标的特征数据。
2. 检测步骤：算法将注册步骤中的特征数据与待检测图像输入至检测模型，在检测图像中进行特征匹配并定位目标。
3. 追加注册与检测步骤：当出现新样本或异常样本时，可修正检测结果并作为追加注册数据。算法获取到追加注册图像和修正结果后，学习这些新样本的特征并更新到检测流程中，从而能够适应新样本并提升检测效果。

使用方法

您可以参照如下内容，使用该模块检测简单场景中是否存在单个特定目标，并定位其位置。在流程中添加该模块后，需完成注册类别，详情请参见 ***注册类别*** 。

前后序模块

使用该模块时，需确保前序模块有 ***图像源*** 模块，采集需进行检测的图像。

配置步骤

1. 在流程中添加***图像源***和***有无检测***，并将***有无检测***模块连接至***图像源***模块。
2. 双击***有无检测***配置相应参数。
3. 在**注册类别**页签，完成注册。
4. 执行流程进行检测。
5. 当出现新样本或未检出目标的异常样本时，在**注册类别**页签单击***追加注册***，将新特征数据或修正后的特征数据更新至检测流程中。
6. 执行流程再次进行检测。

示例方案

如下图所示，注册少量带标签的包装盒图像以及无标签背景图像后，您可通过该模块检测生产环境中包装盒是否存在标签，并定位其位置。

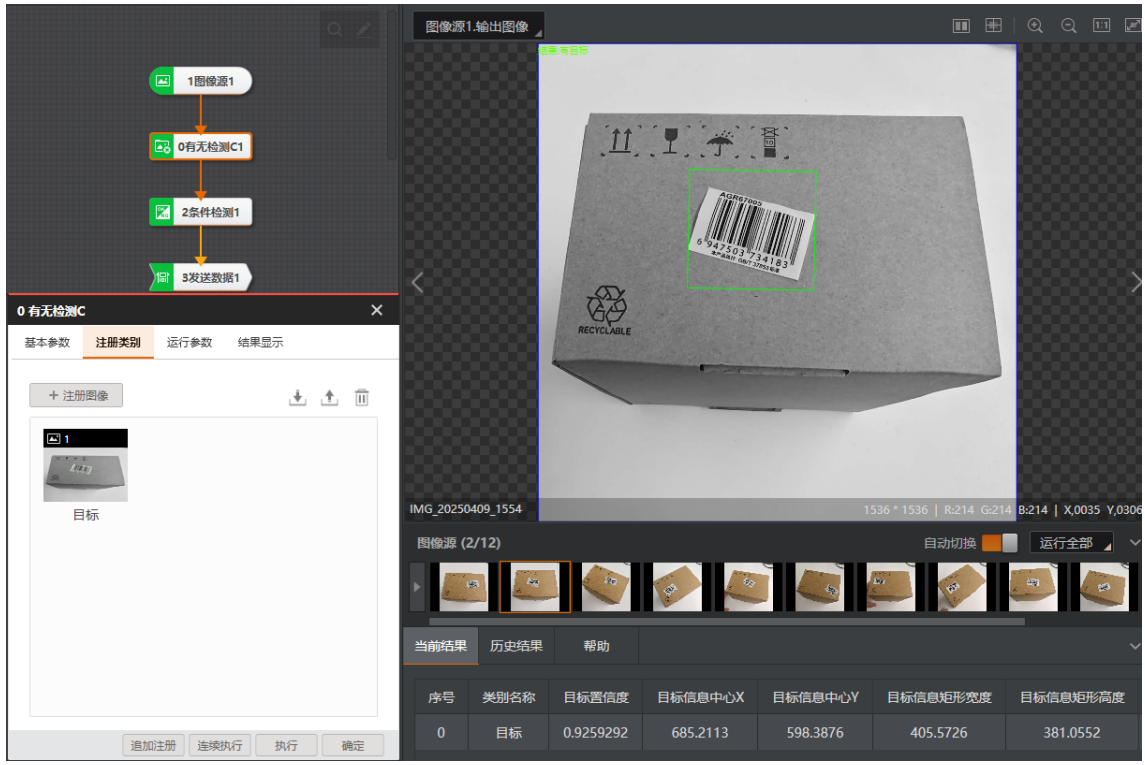


图 13-263 流程示例

注册类别

注册类别可作为*有无检测*的搜索依据，*有无检测*模块根据注册图类别中已注册的目标进行搜索，可在图像中搜索是否存在单个特定目标，并定位其位置。

操作步骤

1. 在**注册类别**页签，单击*+ 注册图像*，进入注册图像窗口。

   说明

   若已完成注册且检测中出现了新样本或异常样本，可单击该页签的*追加注册*继续注册数据并修正检测结果。此时您会进入注册图像窗口，当前模块图像将自动添加至注册图样本中。

2. 在右侧的配置区域，通过以下任一方式添加注册图。

   获取流程图像

   添加当前流程运行的实时图像作为注册图。

   a. 单击*获取流程图像*进入实时图像预览窗口。

   b. （可选）通过图像显示区域左上角下拉框设置“图像切换”或“图像叠加”。具体操作请见 *查看模块图像* 。

c. 单击上方的 或 单次执行或连续执行流程，以获取单张或多张当前流程的实时图像。

说明

- 连续执行流程时，当抓取的图像数达到**图像上限**后，流程将自动停止执行。
- 若已存在模型，获取流程图像时支持同步导入注册图的推理结果。
- 若已设置全局触发，当您返回**注册图像**界面时，一旦满足触发条件，流程图像将自动同步至图像列表。

d. 完成流程图像添加后，单击*返回注册图*。

获取当前图像

获取模块当前订阅的输入图像及其绘制的 ROI 区域。

导入图像

可根据实际需求导入缓存图像或本地图像。

- *缓存图像*：当前序模块包含**缓存图像**时，可导入其缓存的所有图像。
- *导入本地图像*：可从本地中导入单个或多个图像。
- *导入文件夹*：选择本地文件夹后，将导入该文件夹下的全部图像。

说明

若勾选图像预览区下方的*图像保存至模型*，可将添加的注册图保存到方案中、在导出模型的时候保存在模型中。

3. **可选操作**: 通过**标签列表**下各标签右侧的 可预览当前分类标签下的注册目标。

说明

将鼠标悬浮至对应标签上即可显示 。

4. **可选操作**: 在左下角的注册样本列表中选中一张图片，单击**标签列表**的**设为背景图**，将该图片设置为背景图。

为保证注册精度，可将一张不含有目标物的图像设置为背景图。

5. 标注图像。
   1) 在左下角的注册样本列表中选中需要标注的图像。
   2) 在**标签列表**单击选中**目标**。
   3) 单击选中*矩形框选*后，在图像中绘制目标区域。绘制完成后，右键单击已绘制 ROI，可复制粘贴选中 ROI 到指定位置，或按需删除选中 ROI。

   说明

   具体绘制方式，请参见 *模块中模型配置的掩膜区域* 。

   4) 重复以上步骤，直到所有图像中的所有目标均已标注。

   具体操作见如下视频：

6. **可选操作**: 根据已标记目标进行自动标记。

若已注册至少一个目标，切换到其它未标记注册图后，单击*自动打标*可对其它注册图上的目标进行自动标注。

说明

进行自动打标时，当前注册图的推理结果会导入注册界面并自动显示标签，需手动调整目标区域或标签类别将其加入训练样本。

7. 单击**注册图像**窗口右下角的*注册*。
8. 在**注册内容确认**窗口，配置注册相关参数。

旋转增强

若启用该参数，学习的特征会对目标角度变化进行适应和增强。当检测目标存在旋转时，建议开启。

9. 单击*开始注册*，开始训练模型。

注册成功后，可单击**注册图像**左上角的*流程控制*，通过 / 单次执行或连续执行流程以检验注册效果。

若注册效果不符合预期可继续微调；若符合预期可关闭该窗口返回**注册类别**页签。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

算法参数

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

高级参数

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

过滤参数

面积使能

用于筛选预测结果。开启后，需设置**面积范围**。若预测结果框面积超出设置的**面积范围**，则模块舍弃该预测结果。

宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

角度使能

用于筛选预测结果。开启时，需设置**角度范围**。若预测结果框相对于水平方向的旋转超出设置的**角度范围**，则模块舍弃该预测结果。此处的旋转指：预测结果框相对于水平线进行顺时针旋转时，角度大于 0；逆时针旋转时，角度小于 0。

模块结果

有无检测模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

结果类别总数

int 型，代表检出目标类别的总数量。

结果类别名称

string[]型，代表检出目标的类别名称。

目标分类计数

int[]型，代表各个类别下的目标数量。

目标总数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

目标状态

int 型，代表每个定位框的状态。1 代表定位框正常，0 代表定位框异常。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

标签颜色

String 型，代表图像中不同类别标签所对应的 RGB 颜色值，该颜色可在注册图像操作中自定义设置。

目标信息矩形

box 型，代表定位框所在的矩形。

目标信息中心点

point 型，代表每个定位框的中心点。

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

目标面积

int 型，代表每个检测目标的面积大小。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.3 全监督检测

*全监督检测*相比注册检测模块有更高的检出率和准确率，适用于物体和区域形态变化小，并且遮挡较少的简单场景。该模块需求训练数据量较少，可通过学习少量标注样本时实现目标检测。注册图像时，应标注图像中全部的待检测目标。

模块原理

*全监督检测*基于深度学习目标检测算法，可基于少量注册样本，实现检测目标。

算法分为注册和检测两个步骤，如下图所示。

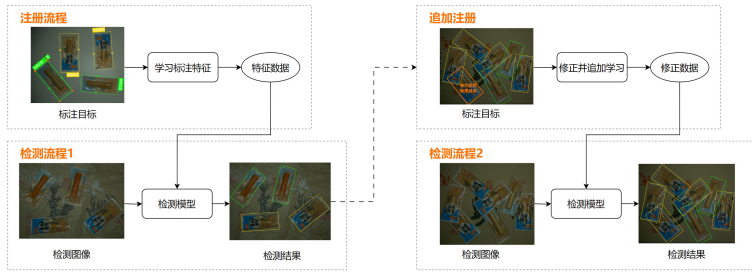


图 13-264 全监督检测算法流程

1. 注册步骤：算法获取到待注册图像。这些图像包含目标标注。基于这些标注，算法学习场景特征，获取对应的特征数据。
2. 检测步骤：算法将注册步骤中的特征数据与待检测图像输入至检测模型，之后，获取最终的检测结果。
3. 追加注册与检测步骤：当出现新样本或异常样本时，可修正检测结果并追加注册数据。算法获取到追加的注册图像和修正结果后，会学习新样本的特征并更新到检测流程中，从而适应新样本并提升检测效果。

DL 目标检测、*注册检测*和*全监督检测*都可用于定位和检测，但三者所需的训练数据（注册图像）和侧重点有差异，因而适用于不同场景。您可以参照下表，选择合适的模块。

表 13-24 适用场景

模块	分类能力	检测目标特点	使用场景	操作要求
注册检测	单个类别	物体，形态变化小	目标间无严重重叠，且背景简单	注册若干图像，仅需标注典型目标
全监督检测	简单分类	物体或区域，形态变化小	目标间无严重重叠	注册若干图像，需标注全部目标
DL 目标检测	复杂分类	物体或区域，形态变化大	目标间有严重重叠	大规模训练数据，需标注全部目标

使用方法

您可通过该模块在简单场景中定位目标。

在流程中添加该模块后，您需通过该模块的*注册类别*页签注册待检测目标及其类别，详情请参见 *注册类别* 。

前后序模块

- 前序模块：需确保前序模块包括用于采集待检测图像的 *图像源* 。
- 后续模块：后续模块可选择 *变量计算* 等运算模块和 *格式化* 等逻辑模块。该模块输出的类别名、目标坐标等检测结果可输出到后续模块进行数据处理，最终输出所需的数据内容。

配置步骤

1. 在流程中添加并连接*图像源*和*全监督检测*模块。
2. 双击*全监督检测*配置相应参数。
3. 在**注册类别**页签，完成注册。
4. 执行流程进行检测。

示例方案

如下图所示，在零食袋分类数量统计场景中，通过*全监督检测*模块对零食袋正反面进行分类后，可通过包含*变量计算*的 *Group* 模块统计正反面的检测结果，再通过*格式化*模块渲染统计结果。

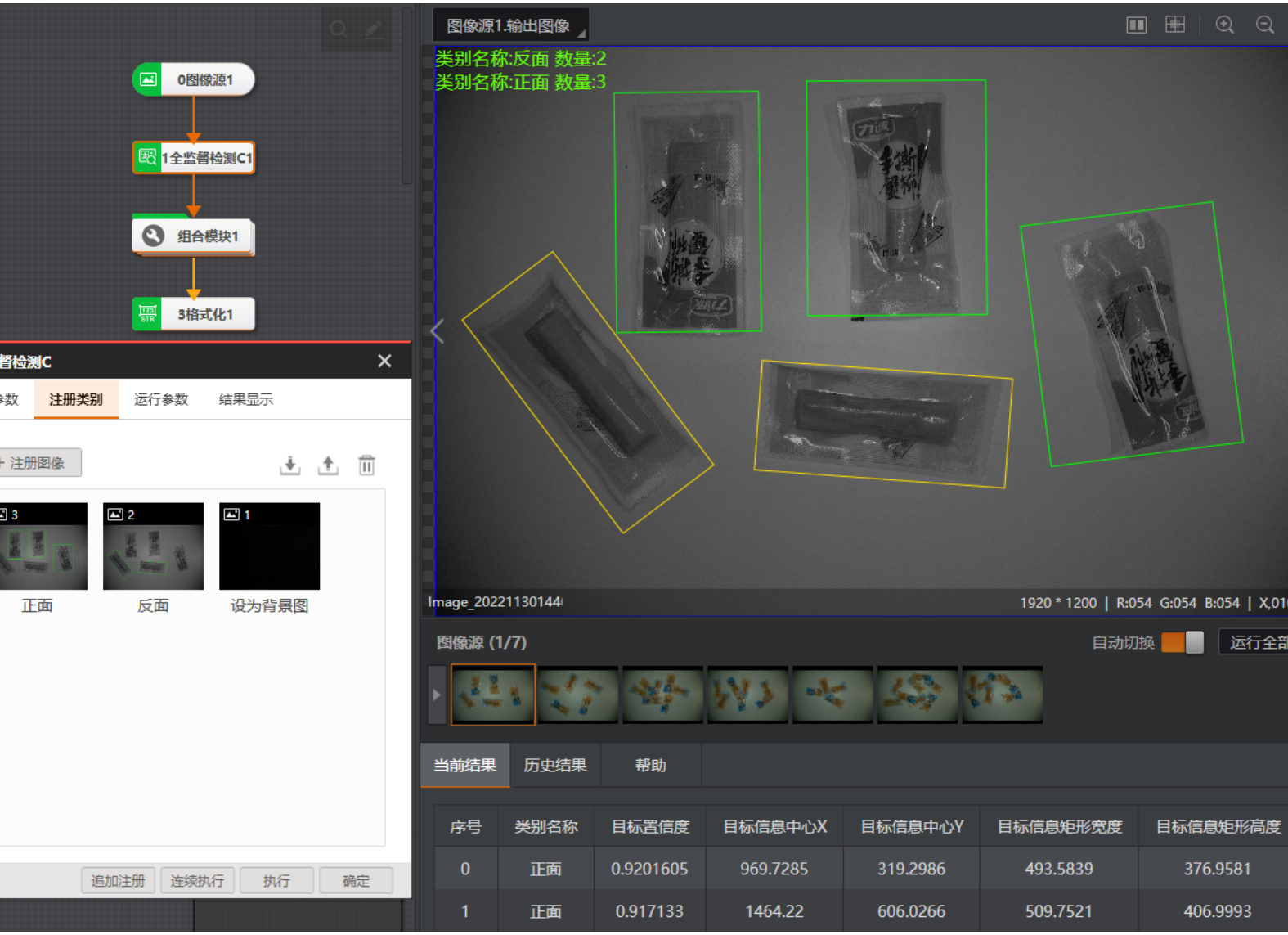


图 13-265 全监督检测流程示例

注册类别

注册类别是*全监督检测*的检测依据。根据注册的目标类别，模块在待检测图像中搜索，并输出符合需求的目标类别。

操作步骤

1. 在**注册类别**页签，单击*+ 注册图像*，进入注册图像窗口。

   说明

   若已完成注册且检测中出现了新样本或异常样本，可单击该页签的*追加注册*继续注册数据并修正检测结果。此时您会进入注册图像窗口，当前模块图像将自动添加至注册图样本中。

2. 在右侧的配置区域，通过以下任一方式添加注册图。

   获取流程图像

   添加当前流程运行的实时图像作为注册图。

   a. 单击*获取流程图像*进入实时图像预览窗口。
   b. （可选）通过图像显示区域左上角下拉框设置“图像切换”或“图像叠加”。具体操作请见 *查看模块图像* 。
   c. 单击上方的 或 单次执行或连续执行流程，以获取单张或多张当前流程的实时图像。

      说明

      - 连续执行流程时，当抓取的图像数达到**图像上限**后，流程将自动停止执行。
      - 若已存在模型，获取流程图像时支持同步导入注册图的推理结果。
      - 若已设置全局触发，当您返回**注册图像**界面时，一旦满足触发条件，流程图像将自动同步至图像列表。

   d. 完成流程图像添加后，单击*返回注册图*。

   获取当前图像

   获取模块当前订阅的输入图像及其绘制的 ROI 区域。

   导入图像

   可根据实际需求导入缓存图像或本地图像。

   - *缓存图像*：当前序模块包含**缓存图像**时，可导入其缓存的所有图像。
   - *导入本地图像*：可从本地中导入单个或多个图像。
   - *导入文件夹*：选择本地文件夹后，将导入该文件夹下的全部图像。

   说明

   若勾选图像预览区下方的*图像保存至模型*，可将添加的注册图保存到方案中、在导出模型的时候保存在模型中。

3. 创建类别标签。

   1) 单击**标签列表**右侧的*+*。
   2) 设置**标签名称**和**标签颜色**，单击*确认*即可添加新类别。

   说明

   - 标签列表默认已包含 1 个基础类别，不支持删除默认类别。
   - 标签列表支持以下快捷操作。

- ：编辑当前分类标签的名称和标签颜色。
- ：预览当前分类标签下的注册目标。
- ：删除当前分类标签以及当前分类标签下的所有注册目标。

4. **可选操作**: 在左下角的注册样本列表中选中一张图片，单击**标签列表**的**设为背景图**，将该图片设置为背景图。

   为保证注册精度，可将一张不含有目标物的图像设置为背景图。

5. 标注图像。
   1) 在左下角的注册样本列表中选中需要标注的图像。
   2) 在**标签列表**单击选中一条标签类别。
   3) 单击选中绘制方式后，在图像中绘制当前类别对应的目标区域。绘制完成后，右键单击已绘制 ROI，可复制粘贴选中 ROI 到指定位置，或按需删除选中 ROI。

      说明

      具体绘制方式，请参见 *模块中模型配置的掩膜区域* 。

   4) 在**标签列表**单击切换为其它标签类别，并绘制该类别对应的目标区域。

      您还可以选择如下任一方式切换标签类别。

      - 通过绘制目标区域左上方的下拉框，快速切换该目标的标签类别。
      - 在*标签列表*，单击对应标签上的小眼睛图标进入 *ROI 列表*。勾选指定图像后，单击*标记图像*，将对应图像切换为指定标签类别。

   5) 重复以上步骤，直到所有图像中的所有目标均已标注。
6. **可选操作**: 根据已标记目标进行自动推理和标记。
   - 若已标注至少一个目标，可单击*自动追加*对当前注册图上的其它目标进行自动标注。
   - 若已注册至少一张图像，切换到其它未标记的注册图后，可单击*自动打标*对其它图像上的目标进行自动标注。

   说明

   进行自动追加或自动打标时，当前注册图的推理结果会导入注册界面并自动显示标签，需手动调整目标区域或标签类别将其加入训练样本。

7. 单击**注册图像**窗口右下角的*注册*。
8. 在**注册内容确认**窗口，配置注册相关参数。

   旋转增强

   若启用该参数，学习的特征会对目标角度变化进行适应和增强。当检测目标存在旋转时，建议开启。

9. 单击*开始注册*，开始训练模型。

   注册成功后，可单击**注册图像**左上角的*流程控制*，通过 / 单次执行或连续执行流程以检验注册效果。

   若注册效果不符合预期可继续微调；若符合预期可关闭该窗口返回**注册类别**页签。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

高级参数

目标排序

设置模块输出预测结果的顺序。

- 按中心点 X/Y 坐标升序：按照预测框中心点的 X/Y 坐标值，从小到大排序预测结果。
- 按置信度降序：按照预测结果的确信程度，从大到小排序预测结果。

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

过滤参数

面积使能

用于筛选预测结果。开启后，需设置**面积范围**。若预测结果框面积超出设置的**面积范围**，则模块舍弃该预测结果。

宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

角度使能

用于筛选预测结果。开启时，需设置**角度范围**。若预测结果框相对于水平方向的旋转超出设置的**角度范围**，则模块舍弃该预测结果。此处的旋转指：预测结果框相对于水平线进行顺时针旋转时，角度大于 0；逆时针旋转时，角度小于 0。

不同类别过滤

开启后，则可过滤不同类别的目标。若不开启，则只会按照同类别过滤。

模块结果

*全监督检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

结果类别总数

int 型，代表检出目标类别的总数量。

结果类别名称

string[]型，代表检出目标的类别名称。

目标分类计数

int[]型，代表各个类别下的目标数量。

目标总数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

目标状态

int 型，代表每个定位框的状态。1 代表定位框正常，0 代表定位框异常。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

标签颜色

String 型，代表图像中不同类别标签所对应的 RGB 颜色值，该颜色可在注册图像操作中自定义设置。

目标信息矩形

box 型，代表定位框所在的矩形。

目标信息中心点

point 型，代表每个定位框的中心点。

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

目标面积

int 型，代表每个检测目标的面积大小。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.4 异常分类

异常分类模块可依据已注册的图像类别将图像分为 OK 和 NG 两类，相比**注册分类**模块具有更高的准确率和分类效率。适用于每个类别样本量较少，且在使用过程中明确不存在第三种类别的场景，例如来料有无、正反检测等。

异常分类的样本特性一般需要满足以下特点：

- 不同类样本类间差异较明显，即 OK 与 NG 两类在关键特征上存在可辨识差异，同时允许存在部分相同特征。
- 允许同类样本间有稍大差异，例如同类样本间存在照明强度、视角、脏污或遮挡等方面的差异。

说明

使用该模块时需确保所有注册图像与后续推理图像的像素格式完全一致。例如，若注册图像采用 RGB24 格式，则所有后续推理图像也必须为 RGB24 格式，否则将无法正常使用。

模块原理

该模块基于深度学习目标检测的注册类算法，分为注册和分类两个流程，并支持对效果不佳的异常样本追加学习，如下图所示。

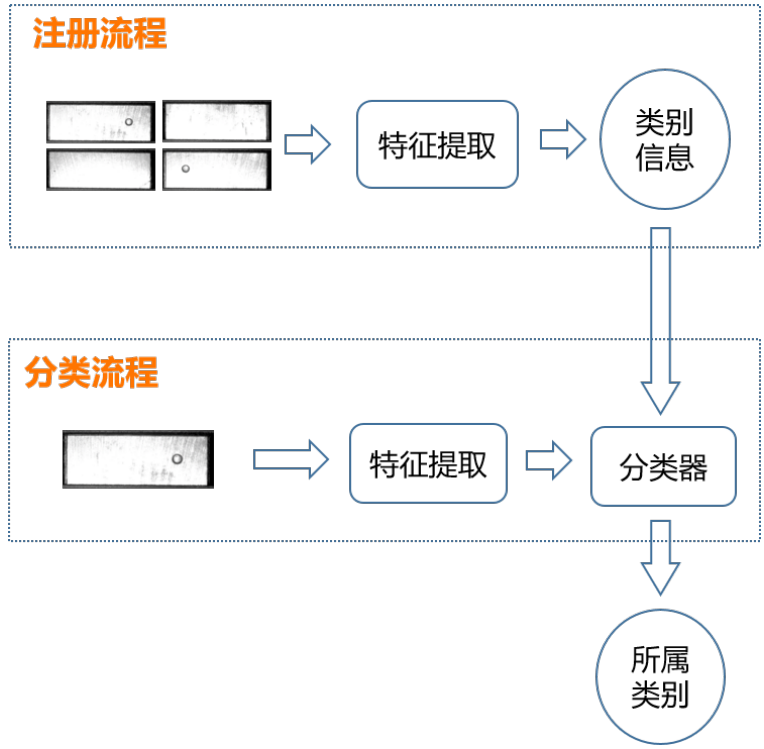


图 13-266 异常分类算法流程图

1. 注册流程：将需要分类的若干示例图像及其所属的真实类别标签（OK 或 NG）一同输入。使用深度学习模型分析上述样本的 OK 或 NG 类别信息并据此构建分类器。
2. 分类流程：输入未知类别的图像，使用深度学习模型分析输入图像，并将分析结果输入分类器。分类器根据分析结果和已注册的 OK 或 NG 类别信息确定输入图像的所属类别（OK 或 NG）。
3. 追加注册与检测步骤：当出现新样本或异常样本时，可修正检测结果并作为追加注册数据。算法获取到追加注册图像和修正结果后，学习这些新样本的特征并更新到检测流程中，从而能够适应新样本并提升检测效果。

以下图为例，导入注册图像并命名为“橘色方块（OK）”和“黑色区域（NG）”两种包含不同特征信息的类别（左图），对待检索图像中的元素进行推理获得 OK/NG 分类结果（右图）。

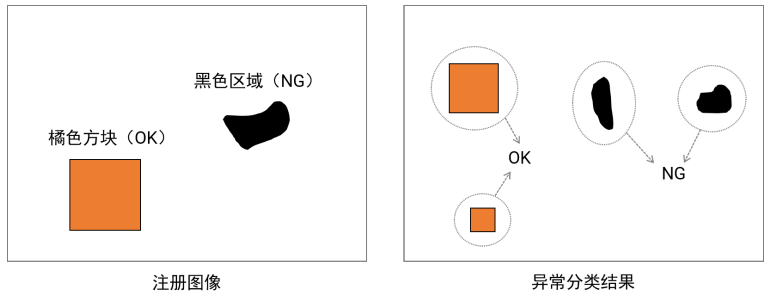


图 13-267 异常分类原理示意图

使用方法

您可以参照如下内容，使用该模块在简单场景中将目标分为 OK 或 NG 两类。在流程中添加该模块后，需完成注册类别，详情请参见 *注册类别* 。

前后序模块

使用该模块时，需确保前序模块有 *图像源* 模块，采集需进行检测的图像。

配置步骤

1. 在流程中添加*图像源*和*异常分类*，并将*异常分类*模块连接至*图像源*模块。
2. 双击*异常分类*配置相应参数。
3. 在**注册类别**页签，完成注册。
4. 执行流程进行检测。
5. 当出现新样本或未检出目标的异常样本时，在**注册类别**页签单击*追加注册*，将新特征数据或修正后的特征数据更新至检测流程中。
6. 执行流程再次进行检测。

示例方案

如下图所示，注册少量控件表面存在凹槽和控件表面不存在凹槽的样本图像后，您可通过该模块实现控件表面凹槽的异常检测与分类，并通过图像显示区查看实际检测结果。

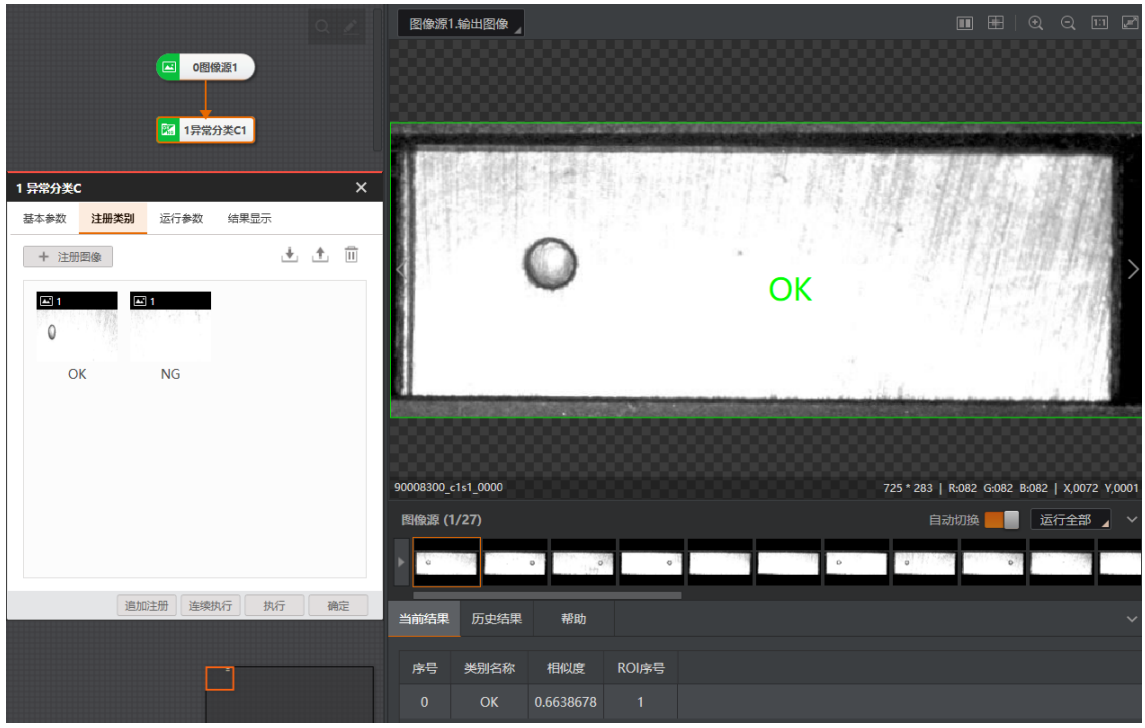


图 13-268 流程示例

注册类别

注册类别可作为*异常分类*的搜索依据，*异常分类*模块根据注册图类别中已注册的 OK 和 NG 图像类别进行搜索，可在多种类型图像中搜索出 OK 或 NG 类别的目标图像。

操作步骤

1. 在**注册类别**页签，单击*+ 注册图像*，进入注册图像窗口。

   说明

   若已完成注册且检测中出现了新样本或异常样本，可单击该页签的*追加注册*继续注册数据并修正检测结果。此时您会进入注册图像窗口，当前模块图像将自动添加至注册图样本中。

2. **可选操作**: 通过**图像上限**下拉框设置支持添加的注册图数量上限。
3. 在右侧的配置区域，通过以下任一方式添加注册图。

   获取流程图像

   添加当前流程运行的实时图像作为注册图。

   a. 单击*获取流程图像*进入实时图像预览窗口。
   b. （可选）通过图像显示区域左上角下拉框设置“图像切换”或“图像叠加”。具体操作请见 *查看模块图像* 。
   c. 单击上方的 或 单次执行或连续执行流程，以获取单张或多张当前流程的实时图像。

      说明

      - 连续执行流程时，当抓取的图像数达到**图像上限**后，流程将自动停止执行。
      - 若已存在模型，获取流程图像时支持同步导入注册图的推理结果。
      - 若已设置全局触发，当您返回**注册图像**界面时，一旦满足触发条件，流程图像将自动同步至图像列表。

   d. 完成流程图像添加后，单击*返回注册图*。

   获取当前图像

   获取模块当前订阅的输入图像及其绘制的 ROI 区域。

   导入图像

   可根据实际需求导入缓存图像或本地图像。

   - *缓存图像*：当前序模块包含**缓存图像**时，可导入其缓存的所有图像。
   - *导入本地图像*：可从本地中导入单个或多个图像。
   - *导入文件夹*：选择本地文件夹后，将导入该文件夹下的全部图像。
   - *按类别导入文件夹*：当图像已按类别整理到不同文件夹，可按分类批量导入。
     在**标签列表**中选中任一类别，单击 *导入图像 → 按类别导入文件夹*，选择对应文件夹，即可导入该文件夹下的全部图像，并自动标记为选定类别。

说明

若勾选图像预览区下方的*图像保存至模型*，可将添加的注册图保存到方案中、在导出模型的时候保存在模型中。

4. **可选操作**: 通过**标签列表**下各标签右侧的 可预览当前分类标签下的注册目标。

说明

将鼠标悬浮至对应标签上即可显示 。

5. 标注图像。
   1) 在左下角的注册样本列表中选中需要标注的图像。
   2) 在**标签列表**单击选中一条标签类别。
   3) 单击选中绘制方式后，在图像中绘制当前类别对应的目标区域。绘制完成后，右键单击已绘制 ROI，可复制粘贴选中 ROI 到指定位置，或按需删除选中 ROI。

      说明

      具体绘制方式，请参见 *模块中模型配置的掩膜区域* 。
   4) 在**标签列表**单击切换为其它标签类别，并绘制该类别对应的目标区域。

      您还可以选择如下任一方式切换标签类别。

      - 通过绘制目标区域左上方的下拉框，快速切换该目标的标签类别。
      - 在*标签列表*，单击对应标签上的小眼睛图标进入 *ROI 列表*。勾选指定图像后，单击*标记图像*，将对应图像切换为指定标签类别。
   5) 重复以上步骤，直到所有图像中的所有目标均已标注。

说明

具体标注操作示例视频，请参见 *注册类别* 。

6. 单击**注册图像**窗口右下角的*注册*。
7. 单击*开始注册*，开始训练模型。

   注册成功后，可单击**注册图像**左上角的*流程控制*，通过 / 单次执行或连续执行流程以检验注册效果。

   若注册效果不符合预期可继续微调；若符合预期可关闭该窗口返回**注册类别**页签。
8. **可选操作**: 在**注册类别**页签，可单击*追加注册*继续注册目标。

   若出现新样本或异常样本，可通过该功能追加注册数据并修正检测结果。此时您会再次进入**注册图像**窗口，当前模块图像将自动添加至注册图样本中。

参数配置

以下介绍该模块的基本参数、运行参数以及结果显示参数详情。通过配置这些模块参数，可定义当前模块如何处理输入的数据以及如何显示结果信息。

基本参数

基本参数详情，请参见 ***基本参数*** 。

运行参数

最小相似度

设置与注册阶段分类为 OK 的样本图像的相似度阈值。当检测样本与 OK 图的相似度评分高于设定阈值时归为 OK 类，低于阈值则判定为 NG 类。阈值设置越高，判定标准越严格，需根据实际检测需求调节该阈值。

结果显示

以下仅介绍**所有类别判断**参数，更多结果显示参数详情，请参见 ***结果显示*** 。

所有类别判断

设置检测结果的判断条件。仅当图像中所有 ROI 区域都满足所设置的判断条件时，模块检测结果为 OK，否则为 NG。单击 ⊞ 可设置单个 ROI 的结果判断条件，结果判断序号与 ROI 序号相对应。最多可添加 16 组结果判断条件。

每组结果判断条件中均包含**相似度判断**和**类别名称判断**，请按需开启并设置上述判断条件。

模块结果

异常分类模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

类别总数

int 型，代表图像中所有 ROI 内检测到的目标类别总数。

类别个数

int 型，代表图像中各个 ROI 内检测到的目标类别个数。

相似度

float 型，代表图像中检测到目标与训练模型中 OK 类别的相似度大小，取值范围为 0~1。越接近 1，代表目标与 OK 类别的特征越相似；若模块 NG 时，则相似度为 0。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

标签颜色

String 型，代表图像中不同类别标签所对应的 RGB 颜色值，该颜色可在注册图像操作中自定义设置。

ROI 序号

int 型，代表当前类别对应的 ROI 序号。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.5 多类别分类

多类别分类模块可依据已注册的图像类别对图像进行分类，相比**注册分类**模块具有更高的准确率和分类效率。该模块允许注册的类别较少，支持最多注册 32 个类别。适用于每个类别样本量较少，且在使用过程中有新增类别的场景，例如来料有无、正反检测、标准产品换新分类。

多类别分类的样本特性一般需要满足以下特点：

- 不同类样本类间差异较明显，即各类别在关键特征上存在可辨识差异，同时允许存在部分相同特征。
- 允许同类样本间有稍大差异，例如同类样本间存在照明强度、视角、脏污或遮挡等方面的差异。

说明

使用该模块时需确保所有注册图像与后续推理图像的像素格式完全一致。例如，若注册图像采用 RGB24 格式，则所有后续推理图像也必须为 RGB24 格式，否则将无法正常使用。

模块原理

该模块基于深度学习目标检测的注册类算法，分为注册和分类两个流程，并支持对效果不佳的异常样本追加学习，如下图所示。

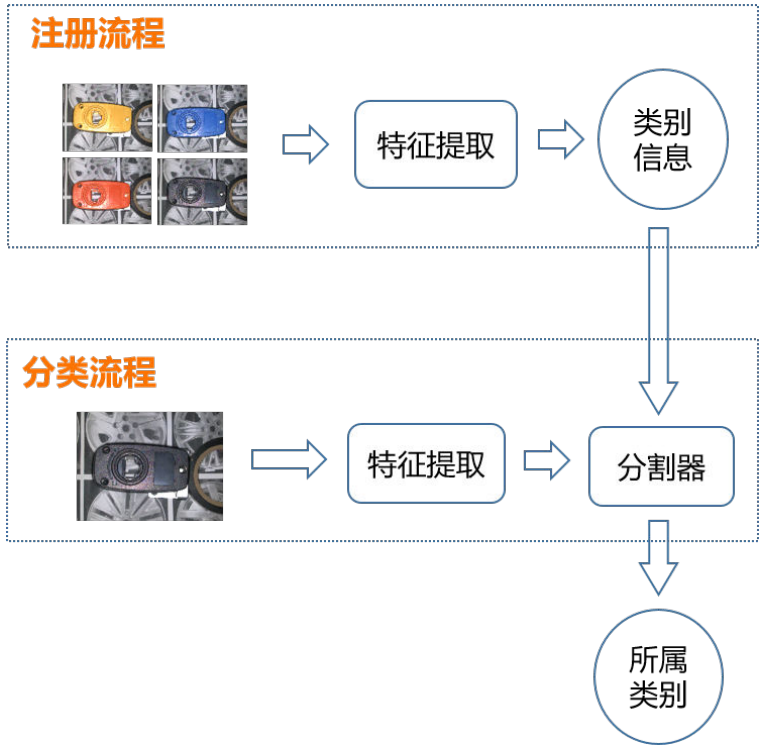


图 13-269 多类别分类算法流程图

1. 注册流程：将需要分类的若干示例图像及其所属的真实类别标签一同输入。使用深度学习模型分析上述样本的类别信息并据此构建分类器。
2. 分类流程：输入未知类别的图像，使用深度学习模型分析输入图像，并将分析结果输入分类器。分类器根据分析结果和已注册的类别信息确定输入图像的所属类别。
3. 追加注册与检测步骤：当出现新样本或异常样本时，可修正检测结果并作为追加注册数据。算法获取到追加注册图像和修正结果后，学习这些新样本的特征并更新到检测流程中，从而能够适应新样本并提升检测效果。

以下图为例，导入注册图像并命名为“橘色方块”、“蓝色圆形”和“黑色区域”三种包含不同特征信息的类别（左图），对待检索图像中的元素进行推理获得分类结果（右图）。

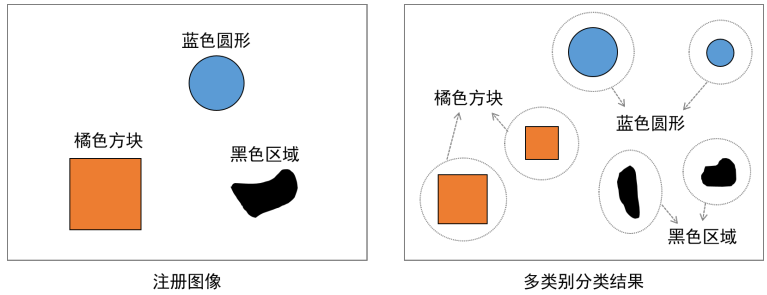


图 13-270 多类别分类原理示意图

使用方法

您可以参照如下内容，使用该模块在简单场景中对目标进行分类。在流程中添加该模块后，需完成注册类别，详情请参见 *注册类别* 。

前后序模块

使用该模块时，需确保前序模块有 *图像源* 模块，采集需进行检测的图像。

配置步骤

1. 在流程中添加*图像源*和*多类别分类*，并将*多类别分类*模块连接至*图像源*模块。
2. 双击*多类别分类*配置相应参数。
3. 在**注册类别**页签，完成注册。
4. 执行流程进行检测。
5. 当出现新样本或未检出目标的异常样本时，在**注册类别**页签单击*追加注册*，将新特征数据或修正后的特征数据更新至检测流程中。
6. 执行流程再次进行检测。

示例方案

如下图所示，注册少量包含多个类别的轮胎样本图像后，您可通过该模块实现对轮胎的分类，并通过图像显示区查看实际分类结果。

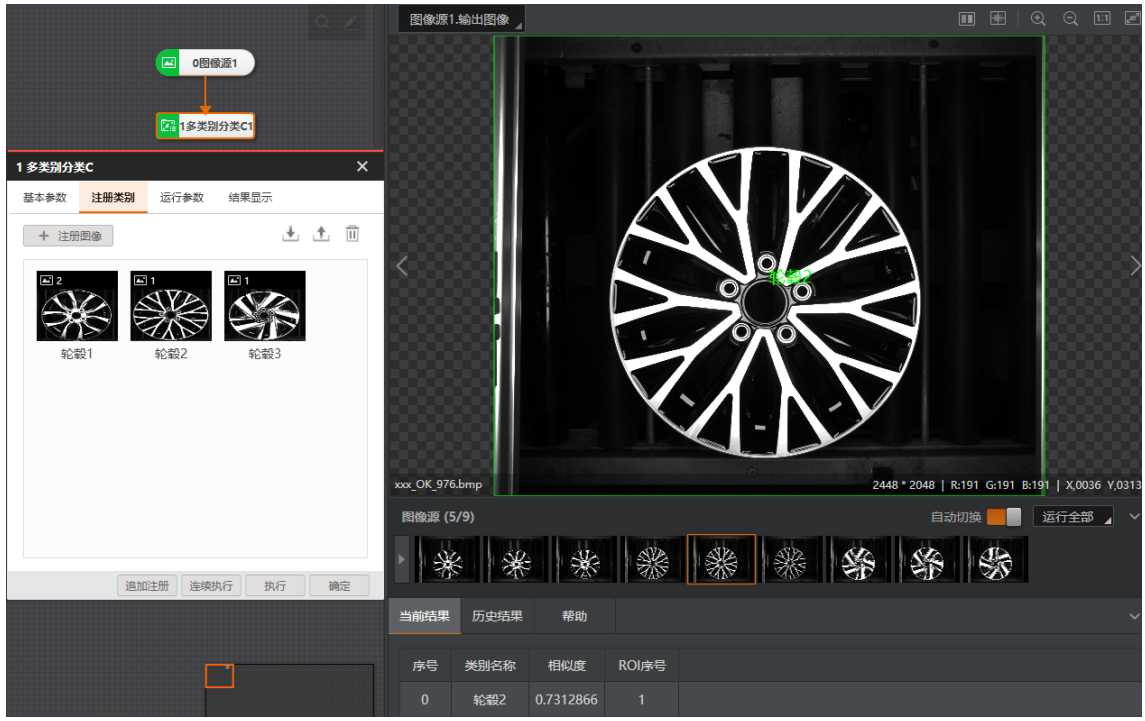


图 13-271 流程示例

注册类别

注册类别可作为*多类别分类*的搜索依据，*多类别分类*模块根据注册图类别中已注册的图像类别进行搜索，可在多种类型图像中搜索出某种类型的目标图像。

操作步骤

1. 在**注册类别**页签，单击*+ 注册图像*，进入注册图像窗口。

   说明

   若已完成注册且检测中出现了新样本或异常样本，可单击该页签的*追加注册*继续注册数据并修正检测结果。此时您会进入注册图像窗口，当前模块图像将自动添加至注册图样本中。

2. **可选操作**: 通过**图像上限**下拉框设置支持添加的注册图数量上限。
3. 在右侧的配置区域，通过以下任一方式添加注册图。

   获取流程图像

   添加当前流程运行的实时图像作为注册图。

   a. 单击*获取流程图像*进入实时图像预览窗口。
   b. （可选）通过图像显示区域左上角下拉框设置“图像切换”或“图像叠加”。具体操作请见 *查看模块图像* 。

c. 单击上方的 或 单次执行或连续执行流程，以获取单张或多张当前流程的实时图像。

说明

- 连续执行流程时，当抓取的图像数达到**图像上限**后，流程将自动停止执行。
- 若已存在模型，获取流程图像时支持同步导入注册图的推理结果。
- 若已设置全局触发，当您返回**注册图像**界面时，一旦满足触发条件，流程图像将自动同步至图像列表。

d. 完成流程图像添加后，单击*返回注册图*。

获取当前图像

获取模块当前订阅的输入图像及其绘制的 ROI 区域。

导入图像

可根据实际需求导入缓存图像或本地图像。

- *缓存图像*：当前序模块包含**缓存图像**时，可导入其缓存的所有图像。
- *导入本地图像*：可从本地中导入单个或多个图像。
- *导入文件夹*：选择本地文件夹后，将导入该文件夹下的全部图像。
- *按类别导入文件夹*：当图像已按类别整理到不同文件夹，可按分类批量导入。
  在**标签列表**中选中任一类别，单击 *导入图像 → 按类别导入文件夹*，选择对应文件夹，即可导入该文件夹下的全部图像，并自动标记为选定类别。

说明

若勾选图像预览区下方的*图像保存至模型*，可将添加的注册图保存到方案中、在导出模型的时候保存在模型中。

4. 创建类别标签。
   1) 单击**标签列表**右侧的*+*。
   2) 设置**标签名称**和**标签颜色**，单击*确认*即可添加新类别。

说明

- 标签列表默认已包含 2 个基础类别，其中“类别 1”为默认标签，不可删除。
- 该模块最少需要创建 2 个类别标签，最多支持创建 32 个类别标签。
- 标签列表支持以下快捷操作。
  - ：编辑当前分类标签的名称和标签颜色。
  - ：预览当前分类标签下的注册目标。
  - ：删除当前分类标签以及当前分类标签下的所有注册目标。

5. 标注图像。
   1) 在左下角的注册样本列表中选中需要标注的图像。
   2) 在**标签列表**单击选中一条标签类别。
   3) 单击选中绘制方式后，在图像中绘制当前类别对应的目标区域。绘制完成后，右键单击已绘制 ROI，可复制粘贴选中 ROI 到指定位置，或按需删除选中 ROI。

说明

具体绘制方式，请参见 *模块中模型配置的掩膜区域* 。

4) 在**标签列表**单击切换为其它标签类别，并绘制该类别对应的目标区域。

您还可以选择如下任一方式切换标签类别。

- 通过绘制目标区域左上方的下拉框，快速切换该目标的标签类别。
- 在*标签列表*，单击对应标签上的小眼睛图标进入 *ROI 列表*。勾选指定图像后，单击*标记图像*，将对应图像切换为指定标签类别。

5) 重复以上步骤，直到所有图像中的所有目标均已标注。

具体操作见如下视频：

6. 单击**注册图像**窗口右下角的*注册*。
7. 单击*开始注册*，开始训练模型。

注册成功后，可单击**注册图像**左上角的*流程控制*，通过 / 单次执行或连续执行流程以检验注册效果。

若注册效果不符合预期可继续微调；若符合预期可关闭该窗口返回**注册类别**页签。

8. **可选操作**: 在**注册类别**页签，可单击*追加注册*继续注册目标。

若出现新样本或异常样本，可通过该功能追加注册数据并修正检测结果。此时您会再次进入**注册图像**窗口，当前模块图像将自动添加至注册图样本中。

参数配置

以下仅介绍该模块的基本参数、运行参数以及结果显示参数详情。通过配置这些模块参数，可定义当前模块如何处理输入的数据以及如何显示结果信息。

基本参数

基本参数详情，请参见 *基本参数* 。

运行参数

最小相似度

设置分类置信度阈值。当分类结果的置信评分高于设定阈值时，按照注册的类别正常分类；若低于阈值则判定为分类失败。当场景中存在错检时，可适当调高该值；当场景中存在漏检时，可适当调低该值。

结果显示

以下仅介绍**结果判断**参数，更多结果显示参数详情，请参见 *结果显示* 。

结果判断

设置检测结果的判断条件。仅当图像中所有 ROI 区域都满足所设置的判断条件时，模块检测结果为 OK，否则为 NG。单击 ⊞ 可设置单个 ROI 的结果判断条件，结果判断序号与 ROI 序号相对应。最多可添加 16 组结果判断条件。

每组结果判断条件中均包含**相似度判断**和**类别名称判断**，请按需开启并设置上述判断条件。

模块结果

多类别分类模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

类别总数

int 型，代表图像中所有 ROI 内检测到的目标类别总数。

类别个数

int 型，代表图像中各个 ROI 内检测到的目标类别个数。

相似度

float 型，代表图像中检测到目标与训练模型中相应类别的相似度大小，越接近 1 代表越相似 。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

标签颜色

String 型，代表图像中不同类别标签所对应的 RGB 颜色值，该颜色可在注册图像操作中自定义设置。

ROI 序号

int 型，代表当前类别对应的 ROI 序号。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.6 注册分类

注册分类模块可依据已注册的图像类别对图像进行分类。

说明

该模块使用时需确保注册图像与后续推理图像的像素格式一致，否则无法使用。例如，注册图像为 RGB24 格式图像，则后续只能推理 RGB24 格式的图像。

模块原理

该模块基于深度学习目标检测的注册类算法，分为注册和分类两个流程，如下图所示。

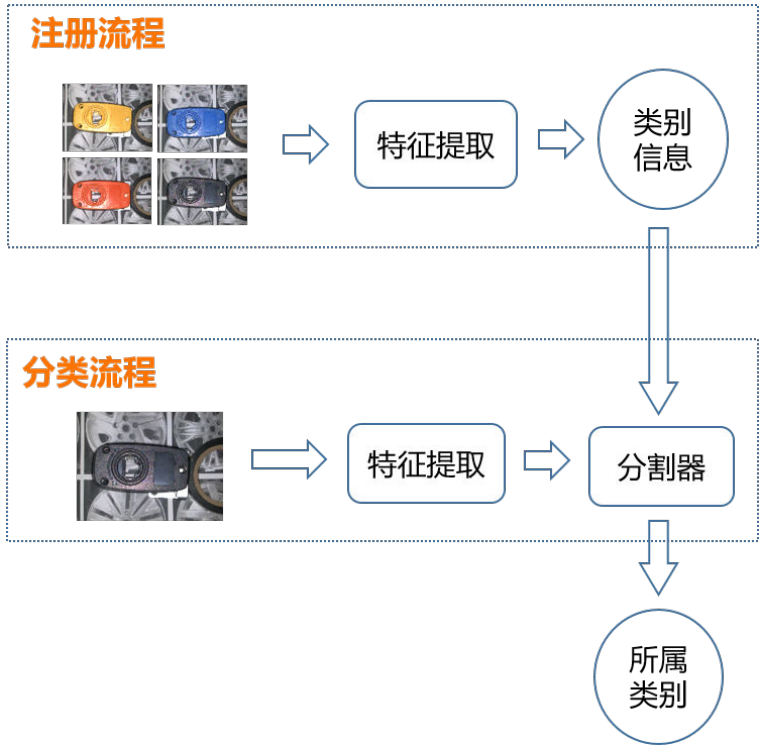


图 13-272 注册分类算法流程图

1. 注册流程：将需要分类的若干示例图像及其所属的真实类别标签一同输入。使用深度学习模型分析上述样本的类别信息并据此构建分类器。
2. 分类流程：输入未知类别的图像，使用深度学习模型分析输入图像，并将分析结果输入分类器。分类器根据分析结果和已注册的类别信息确定输入图像的所属类别。

以下图为例，导入注册图像并命名为“橘色方块”和“黑色区域”两种包含不同特征信息的类别（左图），对待检索图像中的元素进行推理获得分类结果（右图）。

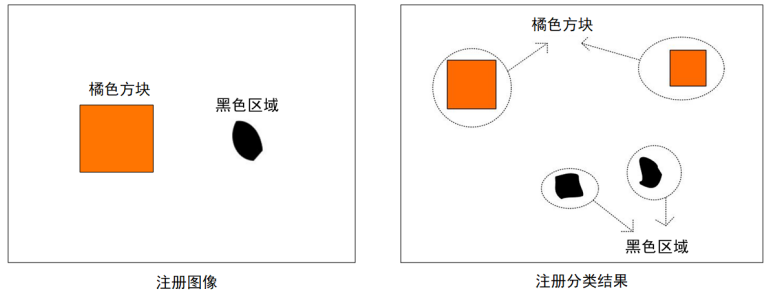


图 13-273 注册分类原理示意图

使用方法

注册分类模块分为注册和分类两步处理步骤。相比于 *DL 分类*，*注册分类*模块更适用于每个类别样本量较少，且在使用过程中有新增类别的场景，例如来料有无、正反检测、标准产品换新分类。

注册分类的样本特征一般需要满足以下特点：

- 不同类样本类间差异明显，即类间差异的特征区别较大。
- 同类样本间一致性好，即同类样本间仅有位置、姿态或者照明强度方面的差异。

注册分类模块的使用步骤如下：

1. 拖拽图像源模块至流程编辑区，并输入图像；
2. 拖拽注册分类模块至流程编辑区并与图像源模块直接连接，双击该模块进行注册图像以及参数配置，注册图像操作请参见 *注册类别* 章节；
3. 点击*执行*即可输出分类检索结果，如下图所示。

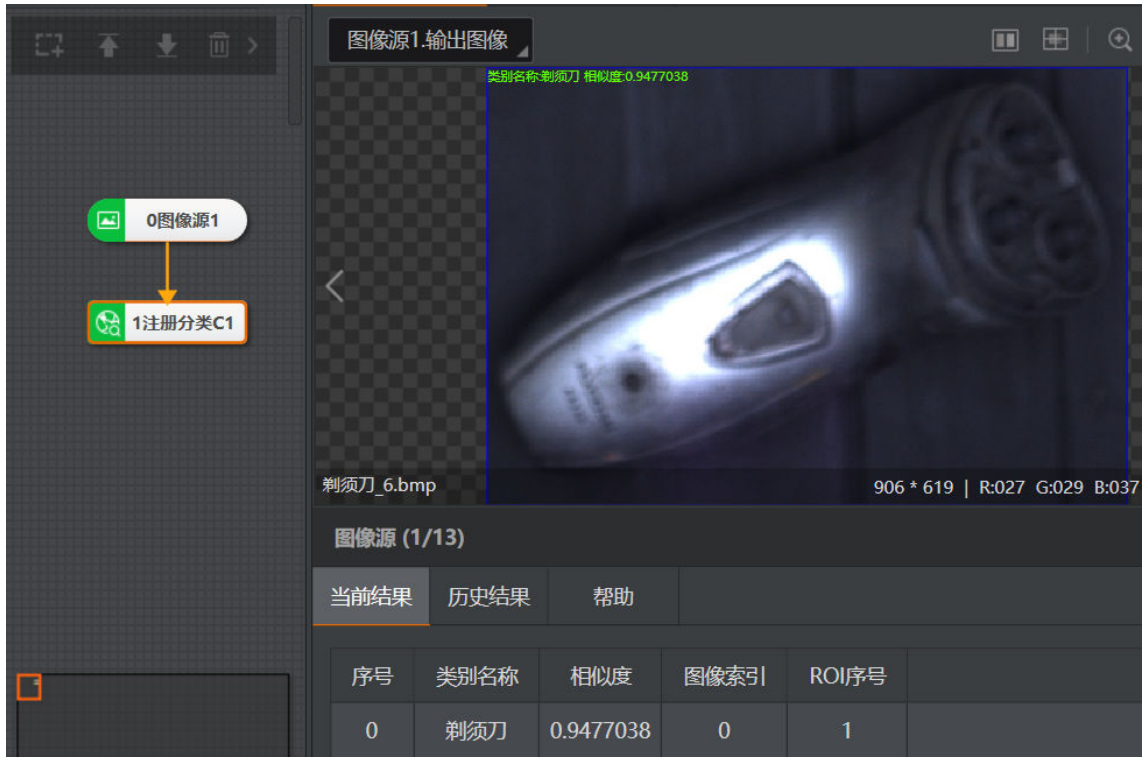


图 13-274 注册分类示例

注册类别

注册类别可作为*注册分类*的搜索依据，*注册分类*模块根据注册图类别中已注册的图像类别进行搜索，可在多种类型图像中搜索出某种类型的目标图像。

操作步骤

1. 在**注册类别**页签，单击*+ 注册图像*，进入注册图像窗口。
2. 在**任务配置**，完成以下参数设置。

   模式类型

   可选**高精度模式**（精度高但耗时）和**快速模式**（快速但精度低）。

   - **高精度模式**：适用于同类样本多样性丰富，类间差异细微的场景。
   - **快速模式**：适用于类内一致性好、类间差异显著且明确的场景。

   任务类型

   可选**非姿态任务**和**姿态任务**。

   若分类依据为目标物的角度，选择**姿态任务**；其他情况均选择**非姿态任务**。

   不同**模式类型**和**任务类型**适用的场景有所差异，如下图所示。

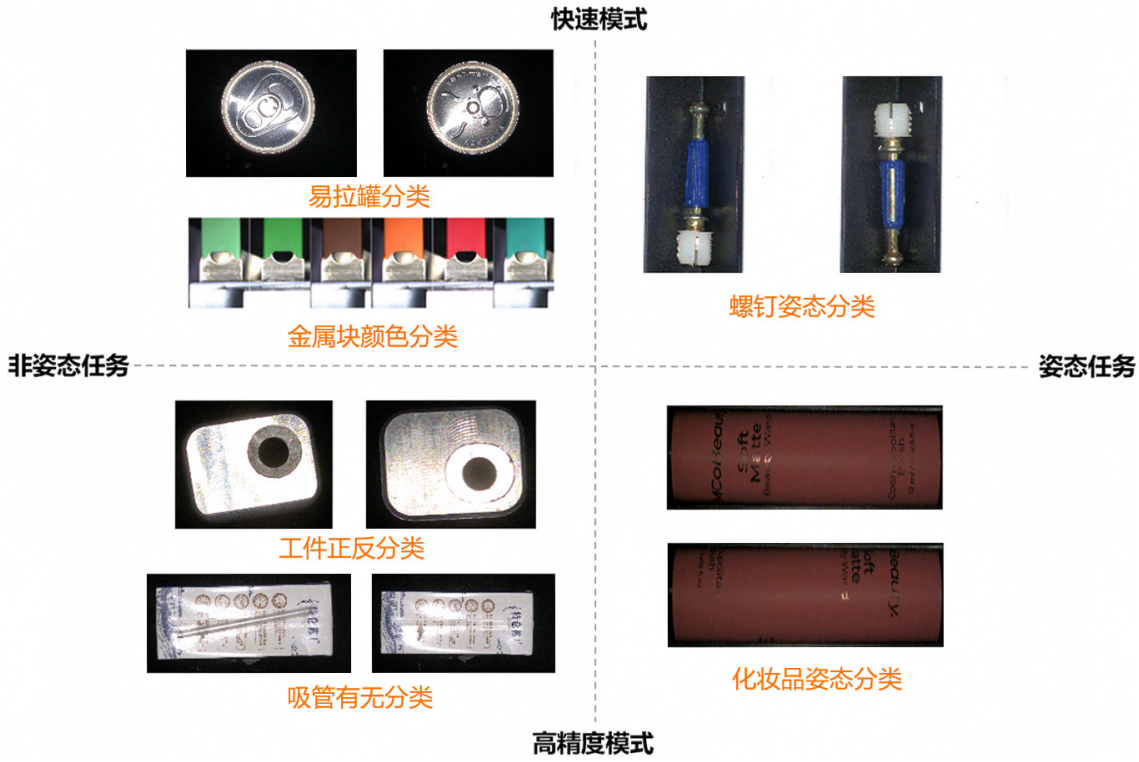


图 13-275 不同类型对应的适用场景

3. 单击**注册类别**下方的*+ 添加*，再点击  并输入**类别名称**，点击*确定*即可添加新类别。

4. 单击 并选择其中一种方式添加图像。
5. 按需设置是整张图像还是仅框选的矩形区域作为检测目标。

> **说明**
>
> 具体如何绘制参见 *模块中模型配置的掩膜区域* 。

6. 选中某张图片，单击对应注册类别即可标注为对应类别。

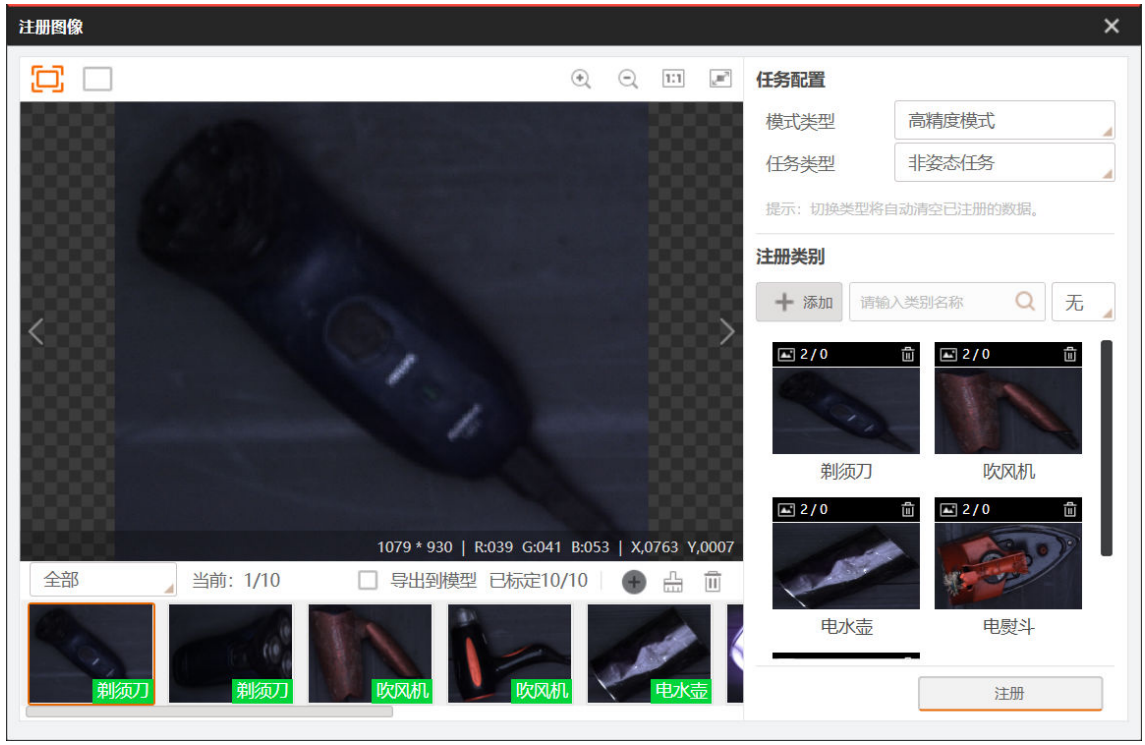


图 13-276 注册图像界面

7. 单击*注册*进行注册。

   注册成功后，注册类别 Tab 页如下图所示。



图 13-277 注册类别面板

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

运行模式

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

算法参数

前 K 个类别

输出置信度得分最高的 k 个类别的索引号和对应的置信度。

最小相似度

设置分类目标的相似程度最小值，过滤相似度小于该参数值的分类结果。当场景中存在漏检时，可适当调低该值；存在错检时，可调高该参数值。

模块结果

*注册分类*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

相似度

float 型，代表图像中检测到目标与训练模型中相应类别的相似度大小，越接近 1 代表越相似 。

图像索引

int 型，代表图像中检测到的目标与注册图像中的第几张图像相似度最高。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

ROI 序号

int 型，代表当前类别对应的 ROI 序号。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.7 注册检测

*注册检测*模块的作用和 *DL 目标检测*模块类似，但该模块无需使用大规模数据进行训练，通过*注册类别*处注册模板即可满足简单场景（目标形态简单、相互无遮挡）的检测需求，适用于产线换型频繁，要求模型迭代速度快的定位检测场景。

说明

该模块只支持单类别检测。若您需要实现简单分类，可调用 *全监督检测* 模块。

模块原理

该模块基于深度学习目标检测的注册类算法，分为注册和检测两个流程，如下图所示。

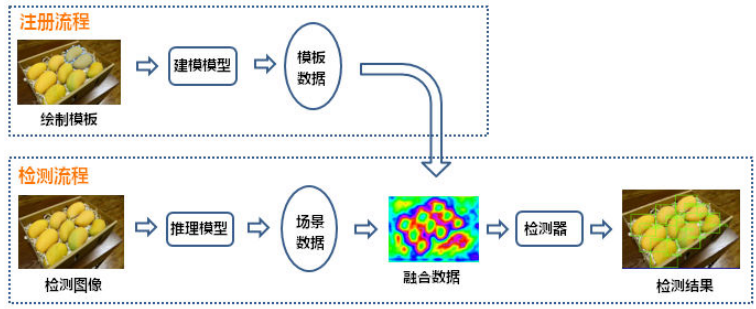


图 13-278 原理示意图

1. 注册流程：首先在图像上标出要检测的目标，再将原始图像和绘制的目标信息通过建模输入到模型中，从而提取所绘制目标的高维特征信息，得到对应的模型数据。
2. 检测流程：将需要检测的图像输入到推理模型中，从而提取检测场景的高维特征信息。再与已经建立的模型数据聚合得到融合数据，最后通过检测器得到最终的检测结果。

使用方法

您可以参照如下内容，使用该模块实现简单场景的目标定位。在流程中添加该模块后，需完成注册类别，详情请参见 *注册类别* 。

前后序模块

使用该模块时，需确保前序模块有 *图像源* 模块，采集需进行检测的图像。

示例方案

如下图所示，在该模块的**注册类别**中注册图像后，可检测简单场景中的对应类别目标。

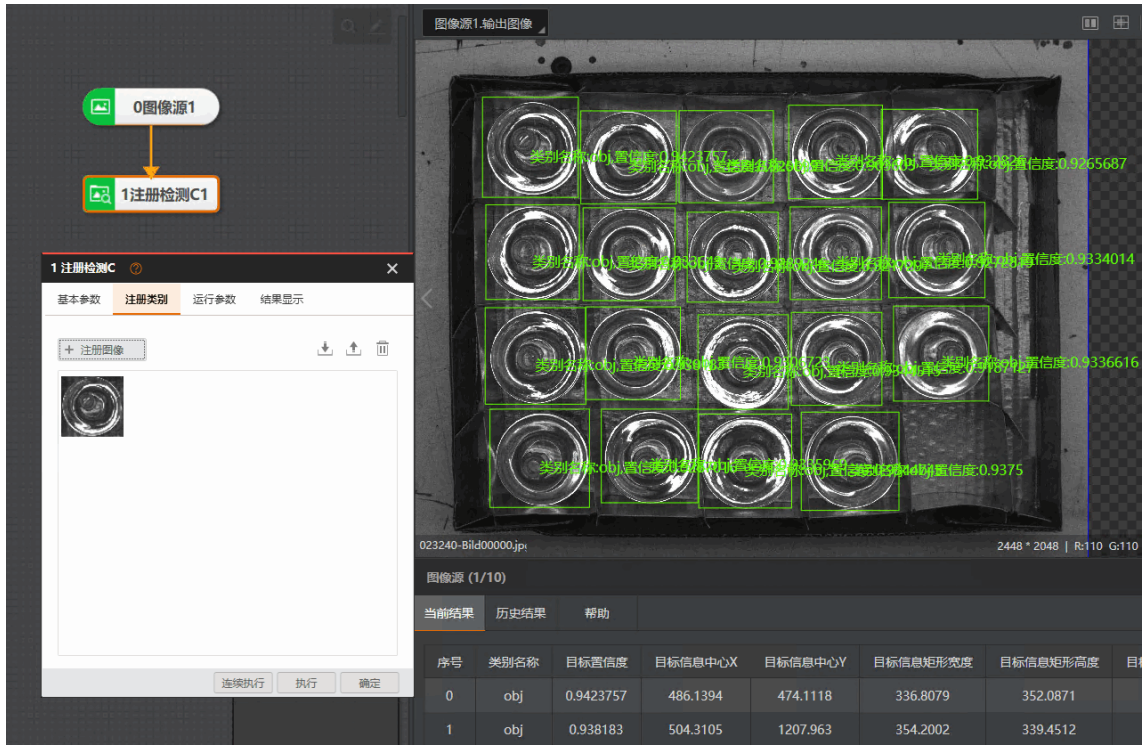


图 13-279 示例流程

注册类别

注册类别可作为*注册检测*的搜索依据，*注册检测*模块根据注册的目标类别进行搜索，在图像中搜索出符合要求的目标类别。

操作步骤

1. 在**注册类别**面板中点击*+ 注册图像*进入注册图像窗口。
2. 完成**参数配置**的参数设置。

类型名称

自定义检测类别的名称。

角度使能

开启该参数后，需根据目标旋转方向调整标注框角度，并对同类别目标采取相同的标注方向。模块将在检测目标时使用带角度的矩形框。该参数适用于检测图像中具有任意方向的对象，可以更精确地定位目标，较少背景干扰。

手动设置分辨率

一般无需设置。自动检测的效果不好时，可开启该参数，通过**分辨率**调整注册图像的分辨率。

分辨率

图像较长一边的分辨率按照设置的数值进行缩放，另一边等比例缩放。

说明

模块会自动推荐一个数值。若仍不满足要求，可基于推荐数值上下微调。

3. 单击 并选择其中一种方式添加图像。
4. 在选中的图像上按需绘制掩膜区域及屏蔽区域。

说明

- 具体如何绘制参见 *模块中模型配置的掩膜区域* 。
- 可在多张图像上绘制掩膜区域，也可在一张图像上绘制多个掩膜区域。建议绘制 3 个以上的掩膜区域，最多可绘制 10 个。

5. 单击*注册*进行注册。

注册成功后，**实例列表**处显示已注册的类别的目标。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

运行模式

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

最大覆盖率

设置区分重叠目标的重叠率最大值，用于控制算法是否将重叠目标拆分成两个目标输出。

在图像中，若两个目标的重叠率大于该参数值，则算法合并两个目标并输出这两个目标总共的预测结果；否则，输出两个目标。

此处的“覆盖率”的计算方式为：“两个目标重叠部分的面积”/“较小目标的面积”。

高级参数

目标排序

设置模块输出预测结果的顺序。

- 按中心点 X/Y 坐标升序：按照预测框中心点的 X/Y 坐标值，从小到大排序预测结果。
- 按置信度降序：按照预测结果的确信程度，从大到小排序预测结果。

无角度建模

使用默认建模模型文件或导入自定义的建模模型文件。该模型文件可基于目标图像的特征信息对目标进行建模处理。如需根据自定义场景需求定制无角度建模模型文件，请联系技术支持。

无角度推理

使用默认推理模型文件或导入自定义的推理模型文件。该模型文件可输出无角度的矩形框，并根据图像的深度特征和建模模型的数据，自动识别与注册模板相似的目标。如需根据自定义场景需求定制无角度推理模型文件，请联系技术支持。

> **说明**
>
> **无角度建模**和**无角度推理**模型文件必须同时替换。在替换预训练模型文件时，当前在*注册类别*页签下已注册的图像实例将被清空，请结合实际情况决定是否替换默认模型文件。

有角度建模

使用默认建模模型文件或导入自定义的建模模型文件。该模型文件可基于目标图像的特征信息对目标进行建模处理。如需根据自定义场景需求定制建模模型文件，请联系技术支持。

有角度推理

使用默认推理模型文件或导入自定义的推理模型文件。该模型文件可输出有角度的矩形框，并根据图像的深度特征和建模模型的数据，自动识别与注册模板相似的目标。如需根据自定义场景需求定制推理模型文件，请联系技术支持。

> **说明**
>
> **有角度建模**和**有角度推理**模型文件必须同时替换。在替换预训练模型文件时，当前在*注册类别*页签下已注册的图像实例将被清空，请结合实际情况决定是否替换默认模型文件。

过滤参数

面积使能

用于筛选预测结果。开启后，需设置**面积范围**。若预测结果框面积超出设置的**面积范围**，则模块舍弃该预测结果。

宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

模块结果

*注册检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

目标个数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

目标信息矩形

目标信息中心点

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

目标面积

int 型，代表每个检测目标的面积大小。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.6.8 注册分割

*注册分割*模块的作用和 *DL 实例分割*模块类似，但该模块无需使用大规模数据进行训练，通过*注册类别*处注册模板即可满足简单场景（目标形态简单、相互无遮挡）的分割需求。

模块原理

该模块基于深度学习实例分割的注册类算法，分为注册和分割两个流程，如下图所示。

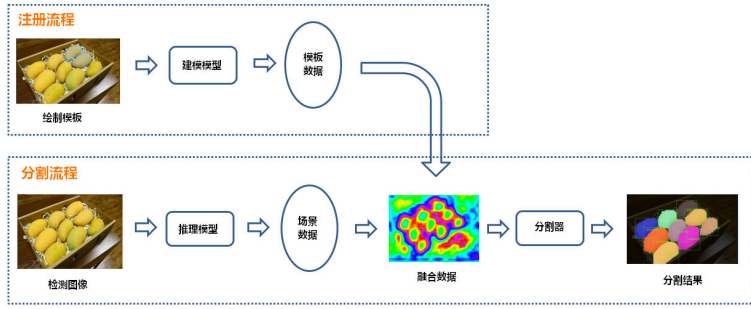


图 13-280 原理示意图

1. 注册流程：首先在图像上标出要分割的目标，再将原始图像和绘制的目标信息通过建模输入到模型中，从而提取所绘制目标的高维特征信息，得到对应的模型数据。
2. 分割流程：将需要检测的图像输入到推理模型中，从而提取检测场景的高维特征信息。再与已经建立的模型数据聚合得到融合数据，最后通过分割器得到最终的分割结果。

使用方法

您可以参照如下内容，使用该模块实现简单场景的目标分割。在流程中添加该模块后，需完成注册类别，详情请参见 *注册类别* 。

前后序模块

使用该模块时，需确保前序模块有 *图像源* 模块，采集需进行检测的图像。

示例方案

如下图所示，在该模块的**注册类别**中注册图像后，可分割简单场景中的对应类别目标。



图 13-281 示例流程

注册类别

注册类别可作为*注册分割*的搜索依据，*注册分割*模块根据注册的目标类别进行搜索，在图像中分割出符合要求的目标类别。

操作步骤

1. 在**注册类别**面板中点击*+ 注册图像*进入注册图像窗口。
2. 完成**参数配置**的参数设置。

类型名称

自定义检测类别的名称。

手动设置分辨率

一般无需设置。自动检测的效果不好时，可开启该参数，通过**分辨率**调整注册图像的分辨率。

分辨率

图像较长一边的分辨率按照设置的数值进行缩放，另一边等比例缩放。

说明

模块会自动推荐一个数值。若仍不满足要求，可基于推荐数值上下微调。

3. 单击 并选择其中一种方式添加图像。
4. 在选中的图像上按需绘制掩膜区域及屏蔽区域。

说明

- 具体如何绘制参见 *模块中模型配置的掩膜区域* 。
- 可在多张图像上绘制掩膜区域，也可在一张图像上绘制多个掩膜区域。建议绘制 3 个以上的掩膜区域，最多可绘制 10 个。

5. 单击*注册*进行注册。

注册成功后，**实例列表**处显示已注册的类别的目标。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

高级参数

掩膜阈值

设置掩膜得分的最小值。当掩膜得分低于该参数值时，算法不输出该掩膜区域。

过滤参数

面积使能

用于筛选预测结果。开启后，需设置**面积范围**。若预测结果框面积超出设置的**面积范围**，则模块舍弃该预测结果。

模块结果

*注册分割*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

目标个数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

目标像素值

int 型，代表输出类别图中分割出的目标区域填充的像素值。

目标面积

int 型，代表每个检测目标的面积大小。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

目标信息矩形

目标信息中心点

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

类别图

image 型，代表类别图。

类别图

image 型，代表类别图的图像数据。

类别图宽度

int 型，代表类别图的宽度。

类别图高度

int 型，代表类别图的高度。

类别图像素格式

int 型，代表类别图的像素格式。

输出渲染图

image 型，代表完成渲染后输出的图像。

输出渲染图

image 型，代表完成渲染后输出的图像。

输出渲染图宽度

int 型，代表输出渲染图的宽度大小。

输出渲染图高度

int 型，代表输出渲染图的高度大小。

输出渲染图像素

int 型，代表输出渲染图的像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

轮廓点

轮廓点 X

float 型，代表每个轮廓点的 X 坐标。

轮廓点 Y

float 型，代表每个轮廓点的 Y 坐标。

轮廓点个数

int 型，代表图像中匹配对象的轮廓点个数。

13.7 深度学习

深度学习是指通过类似脑神经网络的深度学习模型，让计算机像人一样在真实世界中吸收、学习和理解复杂的信息，完成高难度的识别任务，可用于图像分割、图像分类、目标检测、图像检索、无监督分割和实例分割等。

深度学习需要基于一定的数据基础，所以在深度学习前需要对大量的数据集进行训练，参与训练的数据集要完成标签的标注，且应尽可能保证数据的多样性。 深度学习模型文件训练可使用深度学习训练工具（VisionTrain）。

深度学习相关模块均分为 C 和 G 两种类型，这两种模块的原理与参数基本一致，仅在于运行环境不同。后缀为“C”的模块是基于 CPU 运行的，后缀为“G”的模块是基于显卡 GPU 运行的。若使用 CPU 模块，则推荐使用 i7-8 代及以上的 CPU；若使用 GPU 模块，需确保使用显存 6 GB 及以上的独立显卡。显卡相关推荐参见 *推荐硬件* 。

说明

- 其中， *DL 字符识别* 、 *DL 读码* 、 *DL 字符定位* 和 *DL 单字符检测* 等深度学习相关模块放在 *识别* 章节。
- 深度学习相关模块对电脑配置要求较高，具体要求请参见 *运行环境* 章节。

接下来，将从神经网络构成、训练、性能评估指标等方面进行基础介绍，以便更好地理解深度学习相关模块的使用原理。

- *神经网络构成*
- *神经网络模型训练*
- *性能评估指标*
- *各算法模块与对应指标*

神经网络构成

在机器视觉领域，深度学习算法一般是使用卷积神经网络（Convolutional Neural Network, CNN）。一般情况下，一个完整的卷积神经网络是由卷积层、归一化层、池化层、激活层、全连接层和输出层组成。

1. 卷积层：卷积操作主要用于获取更加深层次的特征信息。如下图所示，输入图像是 32*32*3，3 是图像深度（即 R、G、B 三色通道），卷积层是一个 5*5*3 的卷积核，卷积核的深度必须和输入图像的深度相同，比如均为 3。通过一个卷积核与输入图像的卷积可以得到一个 28*28*1 的特征图，下图是用两个卷积核得到了两个特征图。

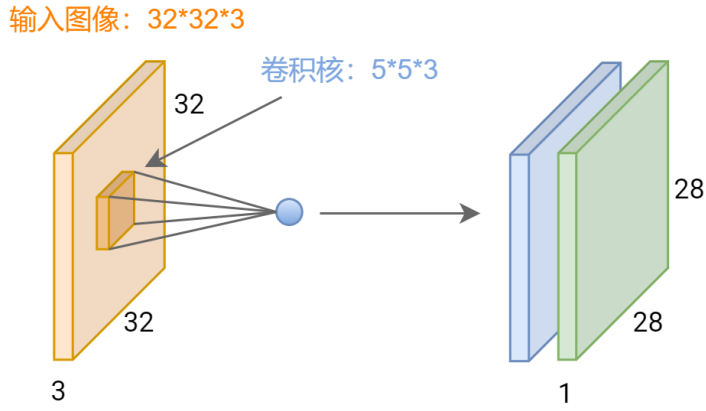


图 13-282 卷积示意图

而深度学习网络是通过多个卷积层串联起来形成的一个网络结构，使用更多的卷积操作可获得更加深层次的特征信息。

2. 池化层：本质上是一种降采样的方式，一般出现在卷积层之后，用于降低因图像尺寸过大而带来的巨大计算量。此外，池化还能够提供平移和旋转不变性，以提升神经网络的鲁棒性。

   目前主要有两种池化方式：均值池化和最大值池化。均值池化时对池化区域内的图像取平均值， 而最大值池化则是将池化区域的像素点取最大值。

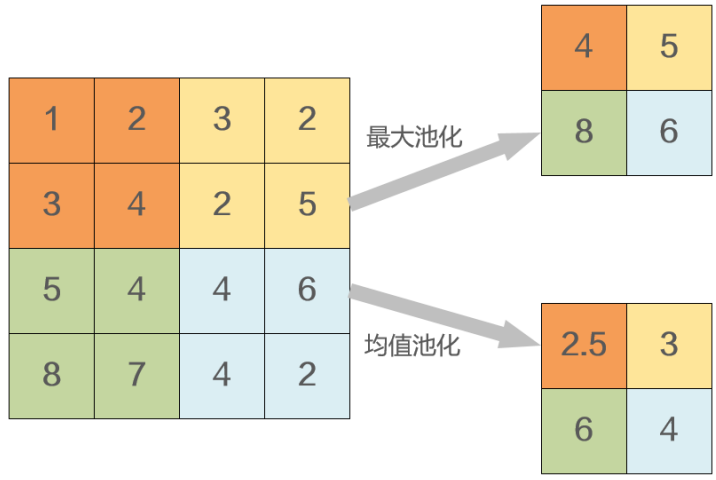


图 13-283 均值池化和最大池化

3. 激活层：激活函数决定了某个神经元是否被激活，以及这个神经元接受到的信息是否有用。

   激活函数的形式如下：

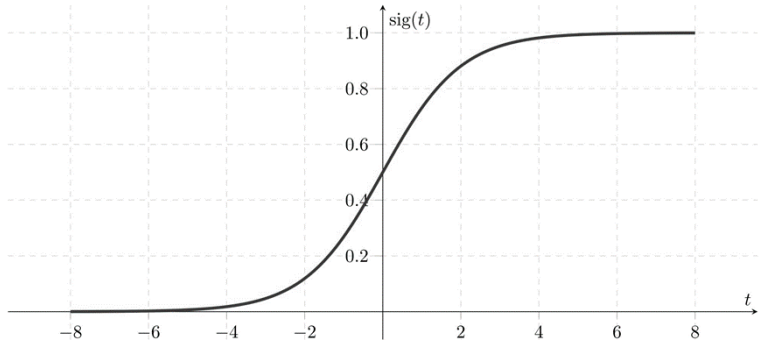


图 13-284 激活函数示意图

对于 y=ax+b 这样的线性函数，当 x 的输入很大时，y 的输出也是无限大或无限小的。如果没有激活层的话，经过多层网络叠加后，值无任何收敛趋势，这显然不符合预期。激活函数的非线性变换则可以使得神经网络能够处理更加复杂的任务。

4. 输出层： 用于将神经网络提取的特征转化为所需结果。比如，对于分类任务，输出层是将特征图或特征向量转化为置信度和类别信息；对于目标检测任务而言，则需将特征图转化为目标的定位框、类别及置信度；对于字符识别任务而言，则是将结果转化为对应的字符串信息。

神经网络模型训练

训练是指按照特定的学习算法将卷积神经网络中的权重参数不断进行调整，以达到局部最优的状态。如下图所示，神经网络的起始点在误差较大的红色区域，随着学习的过程，误差会沿着黑色线条逐步下降到误差较低的蓝色区域，需要注意的是，一般来说，这仅仅是一个局部最优解。在这一过程中，会涉及到几个重要的概念：迭代次数、学习率、预训练模型。

- 迭代次数：表示模型进行训练的次数，即权重更新的次数，对应图中黑色的节点。迭代次数越大，模型权重越接近于局部最优解。
- 学习率：表示每一次权重参数调整的幅度，对应图中黑色节点的距离，学习率越大，权重变化幅度越大，但有一定几率导致模型无法准确收敛；学习率越小，权重变化幅度越小，但可能导致模型学习不够充分，无法达到局部最优解。
- 预训练模型：为了神经网络能够更快进行收敛，提前在大规模数据集上训练好的一组权重参数，相对于权重参数随机初始化，这组权重参数会更加合理，也会使得模型起始点就更接近于局部最优解。

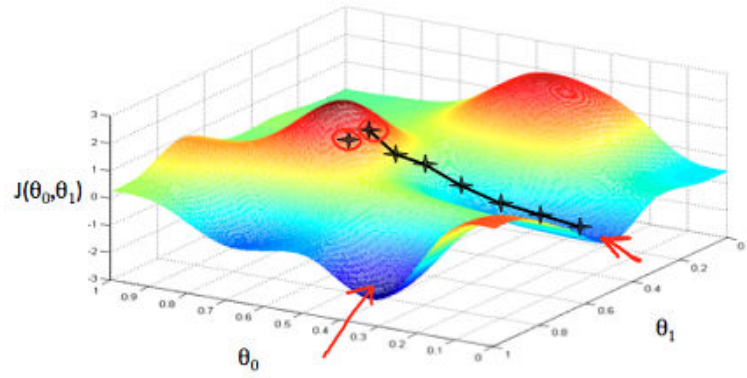


图 13-285 随机梯度下降示意图

性能评估指标

1. 混淆矩阵：即统计分类模型的预测结果，并以矩阵的形式将其展示出来。
   - TP：真实类别为正类，被预测为正类。
   - FP：真实类别为负类，被预测为正类。
   - FN：真实类别是正类，被预测为负类。
   - TN：真实类别是负类，被预测为负类。

混淆矩阵		预测值	
		正类	负类
真实值	正类	TP	FN
	负类	FP	TN

图 13-286 混淆矩阵

2. IoU（交并比）：指预测框与原标记框的交叠率，即它们的交集与并集的比值。若二者完全重叠，则比值为 1。计算公式如下：

$$\mathrm{IoU} = \frac{\mathrm{area(C)} \cap \mathrm{area(G)}}{\mathrm{area(C)} \cup \mathrm{area(G)}}$$

$$\mathrm{CIoU} = sum(IoU_c)/num_c$$

$$\mathrm{MIoU} = sum(CIoU)/class_num$$

其中，CIoU 是指每个类别的平均 IoU，IoU_c 表示 c 类别中的框的 IoU，num_c 表示类别 c 的目标个数；MIoU 表示所有类别的平均 IoU，class_num 表示类别数。

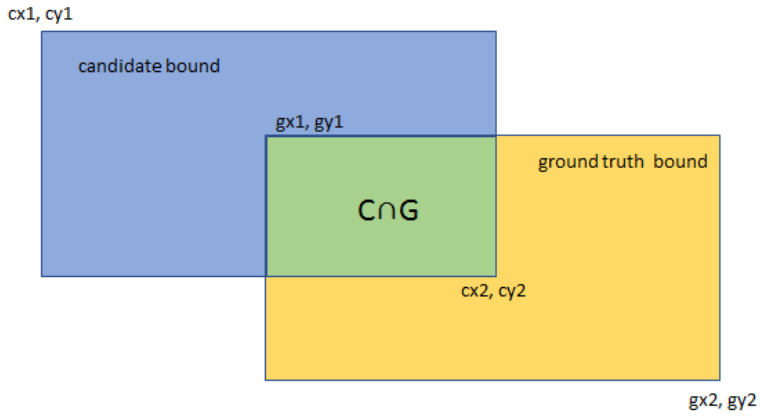


图 13-287 IoU 示意图

说明

这里的预测框和标记框是指多点模式标注的多边形，而不仅仅是矩形框。

3. Precision（精确率）与 Recall（召回率）：Precision 是指预测为正确的数据中真实值为正确的比例，Recall 是指在所有真实值为正确的数据中有多少被预测正确。

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

4. F1_Score：又称平衡 F 分数，同时兼顾了分类模型的精确率和召回率，其最大值是 1，最小值是 0。计算公式如下：

$$\text{F1} = 2 * \frac{Precision*Recall}{Precision+Recall}$$

各算法模块与对应指标

表 13-25 算法模块及评价指标

算法模块	推荐评价指标
分类	混淆矩阵、Precision、Recall、PR 曲线、F1_Score
图像分割	Recall、Precision、M_IoU

算法模块	推荐评价指标
实例分割	Recall、Precision、PR 曲线
目标检测	Recall、Precision、M_IoU、PR 曲线、F1_Score、mAP、AP@0.5:0.95
文本定位	Recall、Precision、PR 曲线、F1_Score
字符识别	行准确率

13.7.1 DL 快速图像分割

*DL 快速图像分割*帮助您检测和分割表面缺陷，如划痕、脏污、裂纹等。该模块可分类多种缺陷，输出缺陷概率图或类别图，常用于缺陷检测、区域定位等场景。

说明

该模块仅可在含独立显卡的工控机上使用。独立显卡性能要求请参见 *推荐硬件* 。

模块原理

图像分割算法基于神经网络提取图像中待测目标掩膜并判断其类别，将相同像素值分配给同一个类别的像素。

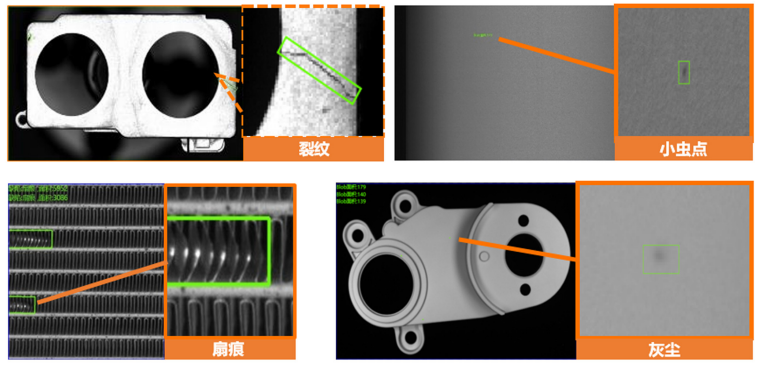
该模块和 *DL 图像分割* 模块存在以下差异：

- 在基本保证检测精度的前提下，*DL 快速图像分割*模块的检测效率有显著提升。
- *DL 快速图像分割*模块不支持输出概率图。

使用场景

*DL 快速图像分割*模块通常应用于目标提取、缺陷检测、场景分割等应用场景，在一些小目标、特征不明显的场景下，具有一定的优势。

- 目标提取指提取图像特征并识别不同实体，从而分割目标与图像背景。
- 缺陷检测指对物体表面缺陷的检测，如斑点、凹坑、划痕、色差、缺损等。例如在工业缺陷检测中，您可用该模块预测和提取前景缺陷，例如检测产品外观的划痕、裂纹等，如下图所示。



假设在检测筷子的缺陷场景中，共包含（1）裂纹、（2）青斑、（3）黑斑和（4）不饱满 4 种缺陷，如下图所示。

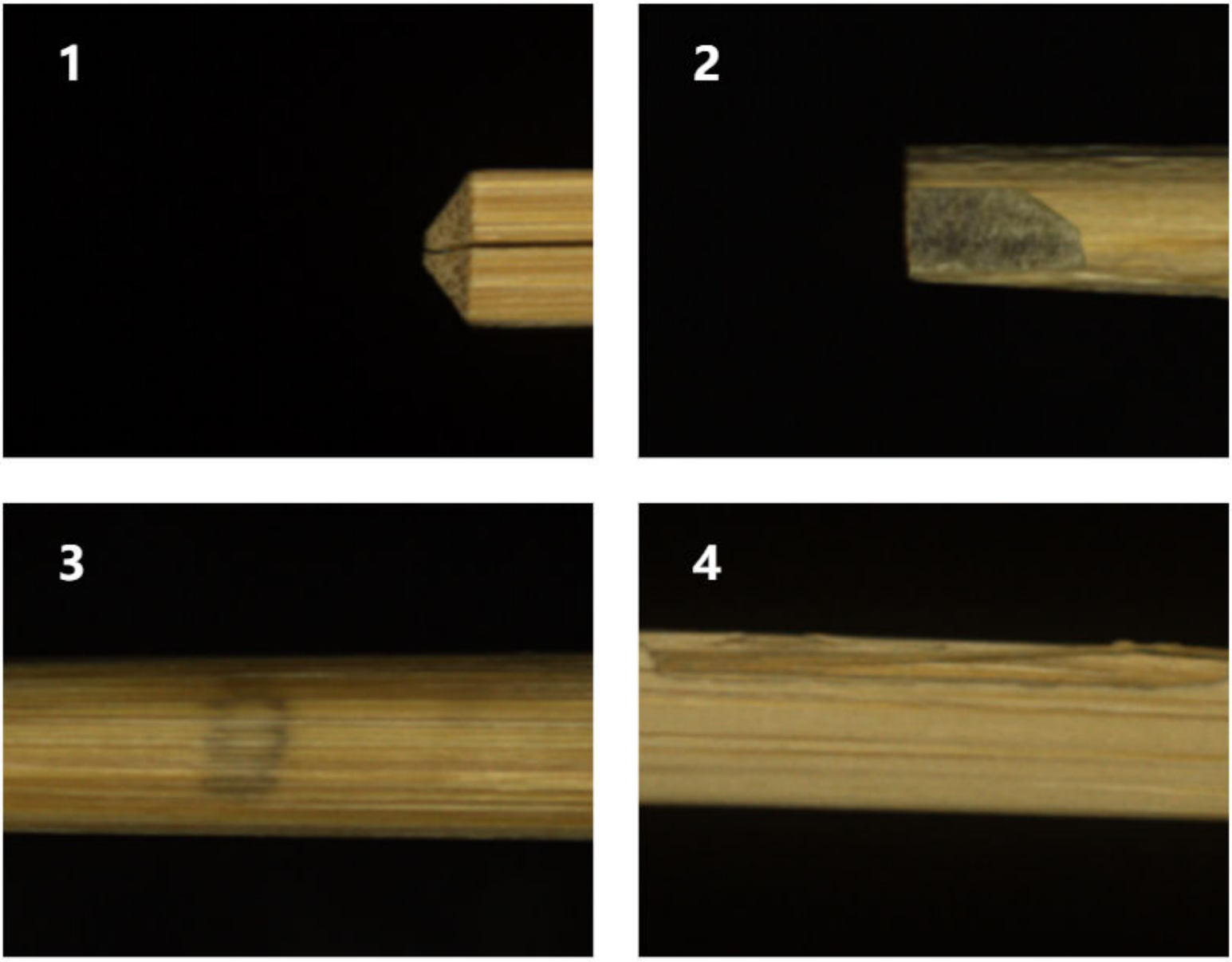


图 13-288 筷子缺陷图

在该场景中，通过 *DL 快速图像分割*识别缺陷时，可通过图像显示，将模块输出图像配置为**缺陷概率图**或**类别图**。

- 缺陷概率图
  概率图可呈现某个点属于某一类别的概率。某点的像素值越低（颜色越深），说明属于该类别的概率越低。
- 类别图

类别图可呈现每个点属于哪种类别，不同类别之间只有颜色的差异，利用不同的颜色来区分不同类别。在该示例中，蓝色代表裂纹，绿色代表青斑，红色代表黑斑，黄色代表不饱满。

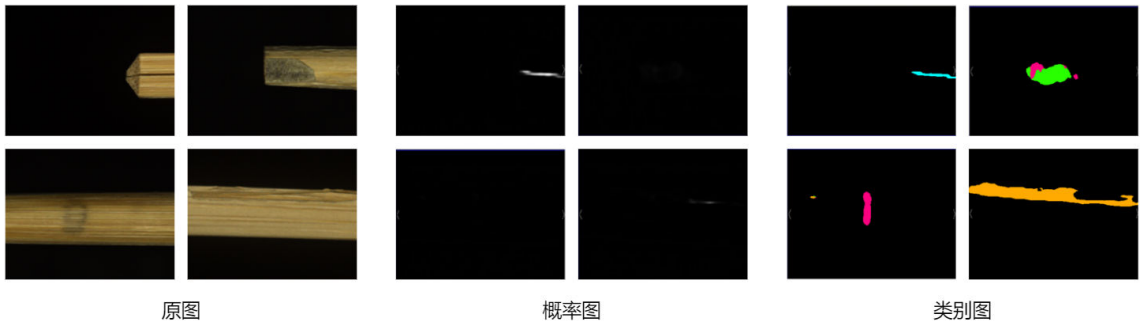


图 13-289 筷子原图和检测结果图

使用方法

根据是否分类缺陷，该模块可用于识别缺陷，或识别并分类缺陷。

模型获取

您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您需通过该模块的**模型文件路径**加载模型，并开启**方案存模型**，将模型文件保存至方案中。

说明

如需分类缺陷，您需要在训练模型时，采用多种标签进行标注。详情请参见对应工具的用户手册。

识别缺陷

当模型内仅存在单个标签时，该模块可用于检测缺陷。

前后序模块

- 前序模块：通常为 *图像源*，为该模块提供图像输入。

  说明

  建议检测目标大于 3×3 像素。

- 后序模块：可搭配 *Blob 分析*，渲染缺陷位置。

示例流程

在该示例中， *Blob 分析* 渲染了“青斑”缺陷的面积。

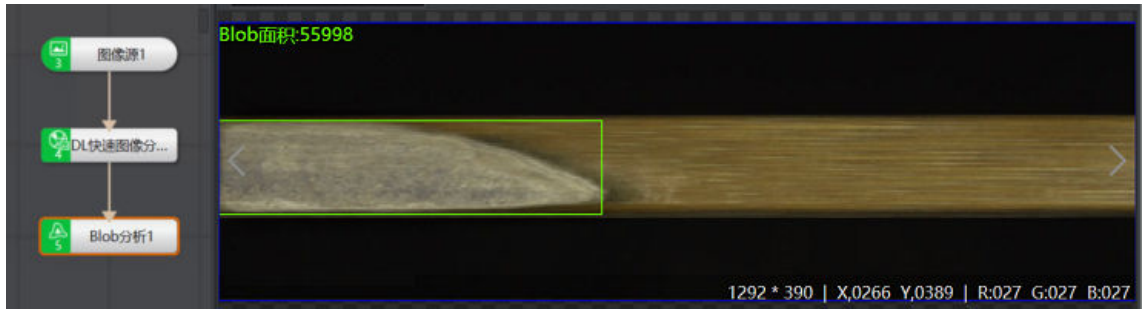


图 13-290 显示缺陷位置

识别并分类缺陷

当模型内仅存在多个标签时，该模块可用于检测缺陷并输出检测缺陷的类别。

前后序模块

- 前序模块：通常为 *图像源*，为该模块提供图像输入。

  说明

  建议检测目标大于 3×3 像素。

- 后序模块：可搭配 *Blob 标签分析*，渲染缺陷位置并输出缺陷类别。

示例流程

在该示例中， *Blob 标签分析* 渲染了“青斑”缺陷的面积，并输出缺陷标签。

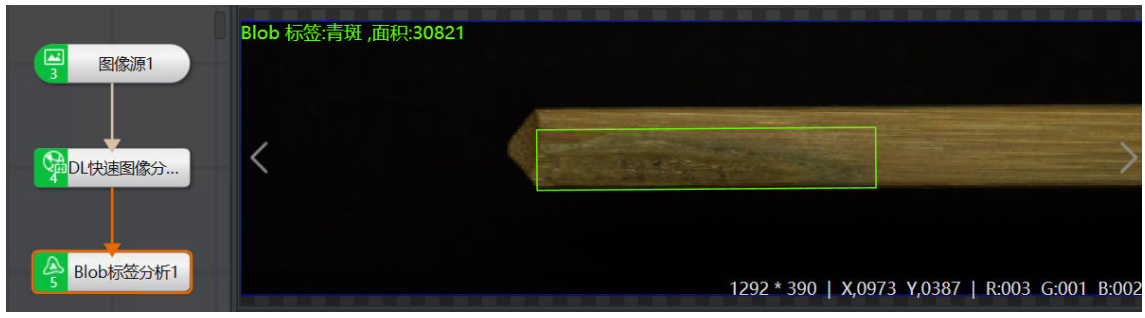


图 13-291 显示并分类青斑缺陷

模型性能调优

实际使用中，模块检测节拍、准确率等受到输入图像质量、模块设置等因素影响。此处列举了几种模型性能不佳的情况以及应对方法。您可以参照以下调整方式，优化模型的表现性能。

当输出结果较差时，您需检查如下几点：

- 模型在训练集上的表现。若模型查准率和查全率不及预期，您可检查模型训练参数和训练集图像标注。
- 训练集和测试集的数据分布。需保持近似的分布。
- 训练集图像，是否包含了全部需分割的标注类型和数据。

若无以上问题，您需根据实际情况在 VisionTrain 或 AI 训练平台中增删训练图像、调整标注或调整标注框。在调整后，您需重新训练模型。

包含较多误检数据

- 在训练模型时，添加误检场景的训练集数据。
- *DL 快速图像分割*中，可适当提高**最小分数**的参数值。

存在较多漏检数据

- 检查输入图像中，待检测目标的像素面积是否占比过小。若过小，建议开启**小目标模式**。
- 可尝试调低**最小分数**参数值。若调低该参数后能够检出相应目标，建议增大训练集数据量。
- 在训练模型时，适当添加漏检场景的训练集数据。

输出结果中掩膜质量较差

检查训练集中标注区域，特别是多边形标注的形状。

说明

通过多边形工具标注时，应紧贴物体轮廓。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

选择该模块运行时使用的模型文件（.bin）。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

按 ROI 裁图

开启时，模块仅预测 ROI 内图像，忽略 ROI 外信息。

关闭时，模块识别整张输入图像，但根据其他设置，如 **ROI 区域**，筛选过滤结果。

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

小目标模式

该模式针对大图小目标场景（待检测目标尺寸占输入图像尺寸低于 0.5%时）优化训练过程，从而提高相应场景下模块的性能。

说明

开启该参数时，无法配置**按 ROI 裁图**。

横向滑窗数
纵向滑窗数

分别设置横向和纵向的图片切割个数。

滑窗重叠率

图片切割时相邻图片的重叠比率。该比率为重叠部分面积和单个滑窗的面积的比值。若设置该参数为 0，相邻滑窗间彼此独立。

算法参数

最小分数

设置预测结果的最小可信度。表示某一像素属于某一类别的最小概率。若该像素的可信度小于该参数值，则模块舍弃该预测结果。减少该参数值时，算法分割的类别区域更小。

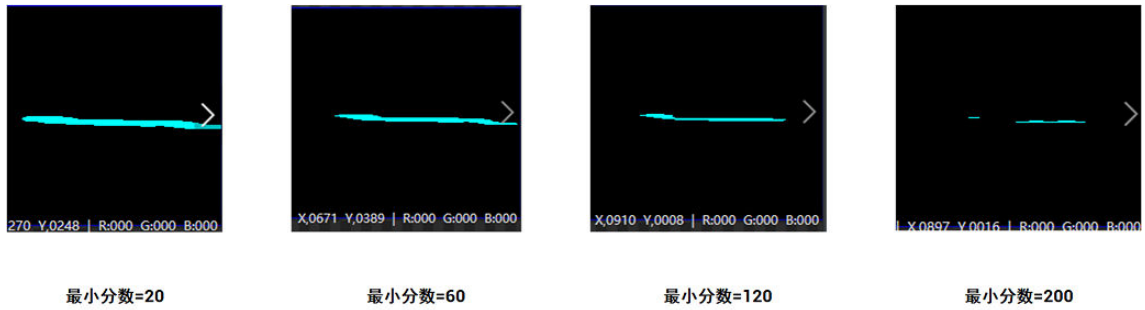


图 13-292 最小分数示意图

模块结果

*DL 快速图像分割*模块的模块结果具体如下。

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

类别信息

image 型，代表类别图信息。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

灰度值

int 型，代表每个类别的灰度值。

最小分数

int 型，代表类别图上属于缺陷的最小概率。

类别图

image 型，代表类别图。

类别图

image 型，代表类别图的图像数据。

类别图宽度

int 型，代表类别图的宽度。

类别图高度

int 型，代表类别图的高度。

类别图像素格式

int 型，代表类别图的像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.7.2 DL 图像分割

*DL 图像分割*帮助您检测表面缺陷，如划痕、脏污、裂纹等。该模块可分类多种缺陷，输出缺陷概率图或类别图，常用于缺陷检测、区域定位等应用。

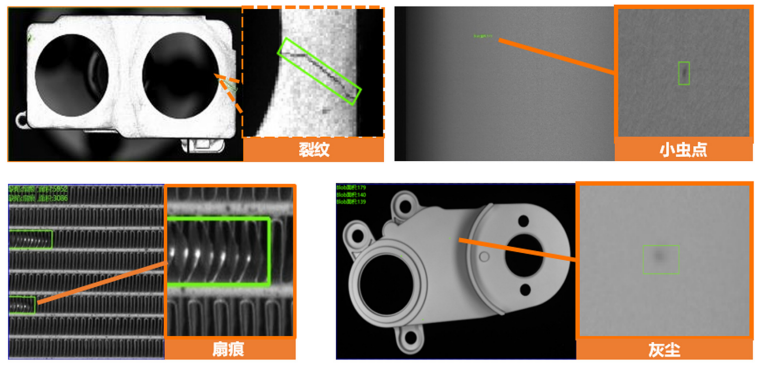
说明

DL 快速图像分割 比该模块效率更高，但仅支持在含独立显卡的工控机上运行。独立显卡性能要求，请参见 *推荐硬件* 。

使用场景

*DL 图像分割*模块通常应用于目标提取、缺陷检测、场景分割等应用场景，在一些小目标、特征不明显的场景下，具有一定的优势。

- 目标提取指提取图像特征并识别不同实体，从而分割目标与图像背景。
- 缺陷检测指对物体表面缺陷的检测，如斑点、凹坑、划痕、色差、缺损等。例如在工业缺陷检测中，您可用该模块预测和提取前景缺陷，例如检测产品外观的划痕、裂纹等，如下图所示。



假设在检测筷子的缺陷场景中，共包含（1）裂纹、（2）青斑、（3）黑斑和（4）不饱满 4 种缺陷，如下图所示。

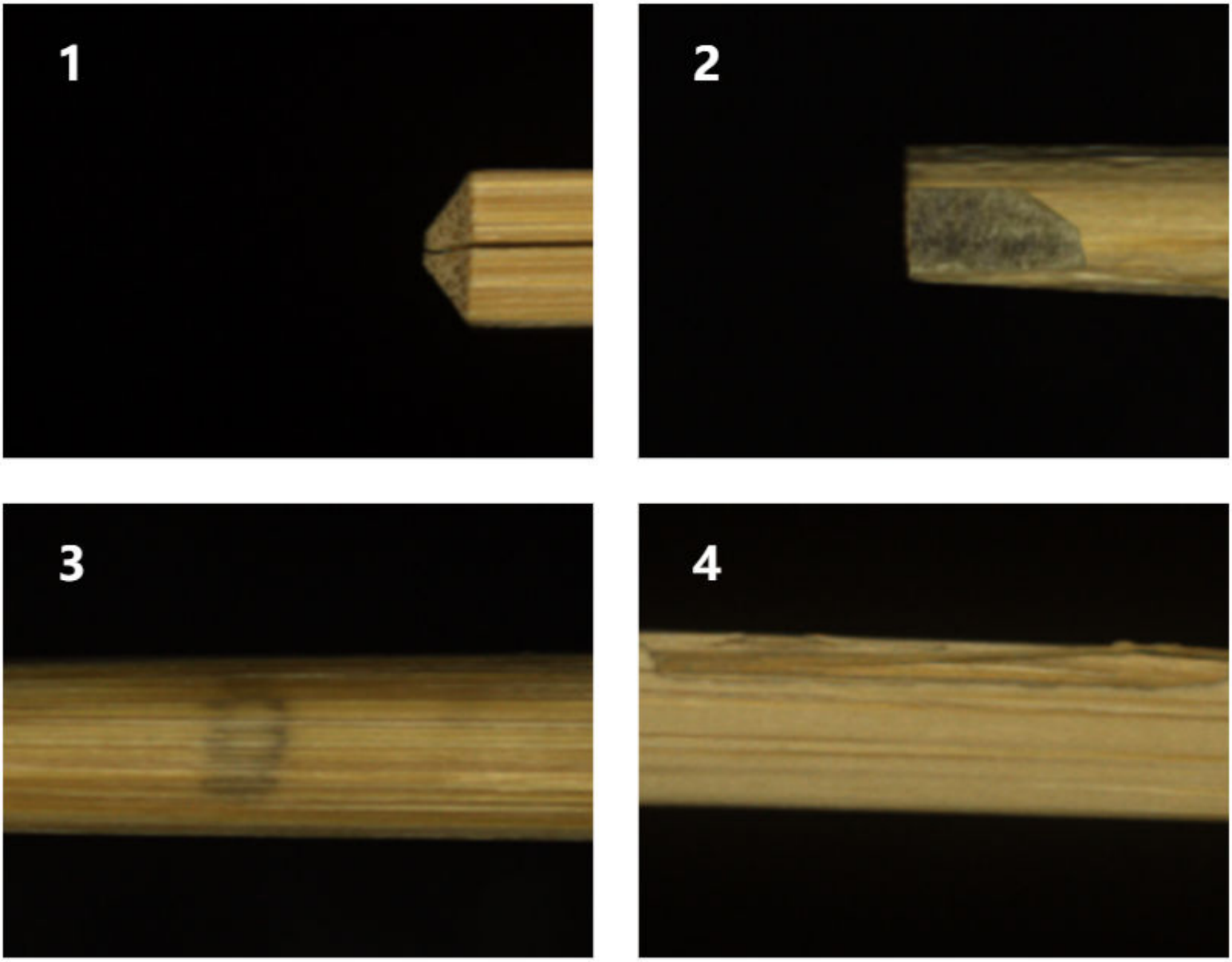


图 13-293 筷子缺陷图

在该场景中，通过 *DL 快速图像分割 G* 识别缺陷时，可通过将图像显示，将模块输出图像配置为**缺陷概率图**或**类别图**。

- 缺陷概率图
  概率图可呈现某个点属于某一类别的概率。某点的像素值越低（颜色越深），说明属于该类别的概率越低。
- 类别图

类别图可呈现每个点属于哪种类别，不同类别之间只有颜色的差异，利用不同的颜色来区分不同类别。在该示例中，蓝色代表裂纹，绿色代表青斑，红色代表黑斑，黄色代表不饱满。

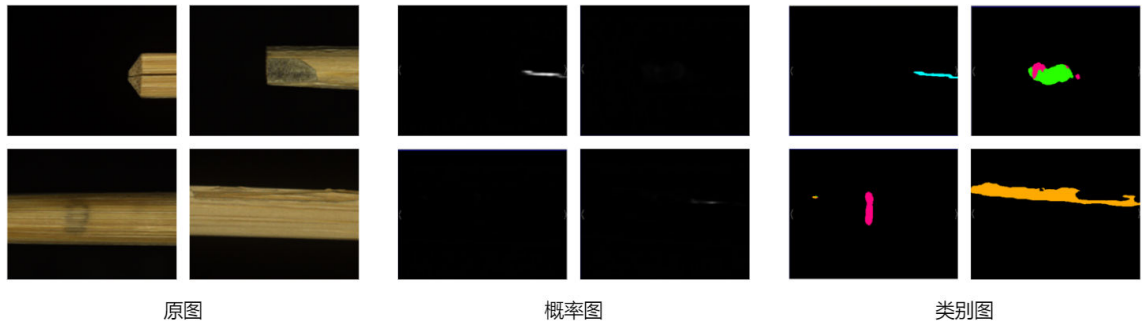


图 13-294 筷子原图和检测结果图

使用方法

根据是否分类缺陷，该模块可用于识别缺陷，或识别并分类缺陷。

模型获取

您需使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您需通过该模块的**模型文件路径**，并开启**方案存模型**，将模型文件保存至方案中。

说明

如需分类缺陷，您需要在训练模型时，采用多种标签进行标注。详情请参见对应工具的用户手册。

识别缺陷

当模型内仅存在单个标签时，该模块可用于检测缺陷。

前后序模块

- 前序模块：通常为 *图像源*，为该模块提供图像输入。
- 后序模块：可搭配 *Blob 分析*，渲染缺陷位置。

示例流程

在该示例中， *Blob 分析* 渲染了“青斑”缺陷的面积。

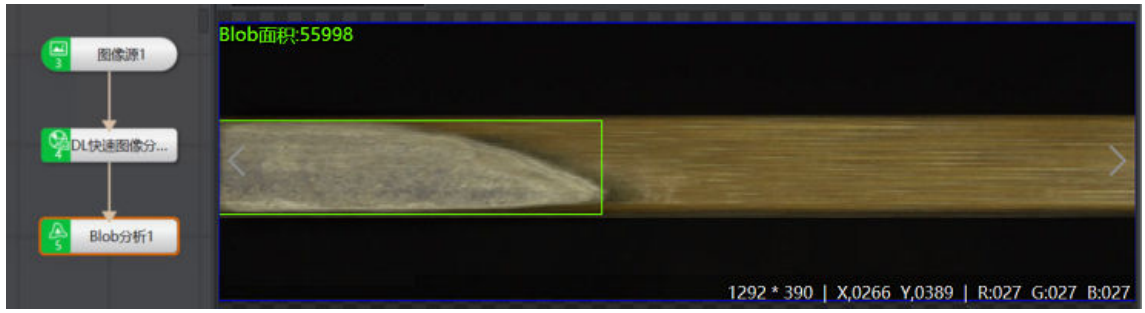


图 13-295 显示缺陷位置

识别并分类缺陷

当模型内仅存在多个标签时，该模块可用于检测缺陷并输出检测缺陷的类别。

前后序模块

- 前序模块：通常为 *图像源*，为该模块提供图像输入。
- 后序模块：可搭配 *Blob 标签分析*，渲染缺陷位置并输出缺陷类别。

示例流程

在该示例中， *Blob 标签分析* 渲染了“青斑”缺陷的面积，并输出缺陷标签。

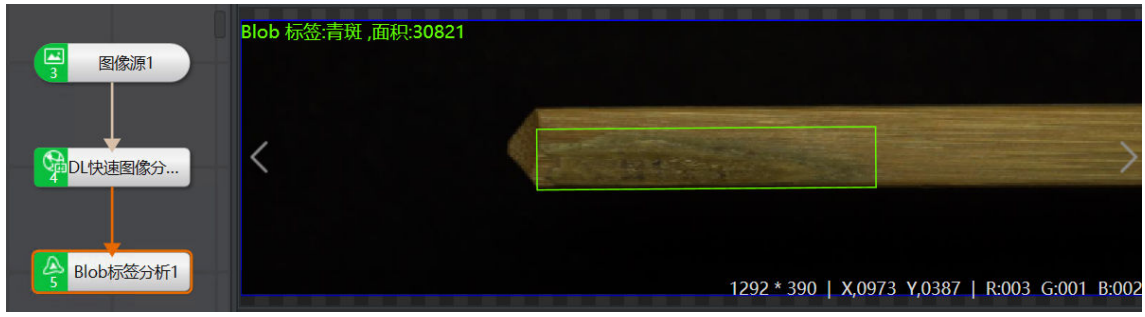


图 13-296 显示并分类青斑缺陷

模型性能调优

实际使用中，模块检测节拍、准确率等受到输入图像质量、模块设置等因素影响。此处列举了几种模型性能不佳的情况以及应对方法。您可以参照以下调整方式，优化模型的表现性能。当输出结果较差时，您需在 VisionTrain 或 AI 训练平台中检查如下几点：

- 模型在训练集上的表现。若模型查准率和查全率不及预期，您可检查模型训练参数和训练集图像标注。
- 训练集和测试集的数据分布。需保持近似的分布。
- 训练集图像，是否包含了全部需分割的标注类型和数据。

若无以上问题，您需根据实际情况在 VisionTrain 或 AI 训练平台中增删训练图像、调整标注或调整标注框。在调整后，您需重新训练模型。

包含较多误检数据

在训练模型时，添加误检场景的训练集数据。

存在较多漏检数据

- 检查输入图像中，待检测目标的像素面积是否占比过小。若过小，建议开启该模块的**小目标模式**。
- 在训练模型时，适当添加漏检场景的训练集数据。

输出结果中掩膜质量较差

检查训练集中标注区域，特别是多边形标注的形状。

说明

通过多边形工具标注时，应紧贴物体轮廓。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

选择该模块运行时使用的模型文件（.bin）。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

获取模型 ROI

开启后，模块自动获取模型文件中的 ROI 信息。此时，您无需配置*基本参数*中的 **ROI 区域**。

按 ROI 裁图

开启时，模块仅预测 ROI 内图像，忽略 ROI 外信息。

关闭时，模块识别整张输入图像，但根据其他设置，如 **ROI 区域**，筛选过滤结果。

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

小目标模式

该模式针对大图小目标场景（待检测目标尺寸占输入图像尺寸低于 0.5%时）优化训练过程，从而提高相应场景下模块的性能。

说明

开启该参数时，无法配置**按 ROI 裁图**。

横向滑窗数
纵向滑窗数

分别设置横向和纵向的图片切割个数。

滑窗重叠率

图片切割时相邻图片的重叠比率。该比率为重叠部分面积和单个滑窗的面积的比值。若设置该参数为 0，相邻滑窗间彼此独立。

模块结果

*DL 图像分割*模块的模块结果具体如下。

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

概率直方图

int 型，代表概率直方图将频数除以总样本数，得到每个区间内数据的概率。

最小分数

int 型，代表类别图上属于缺陷的最小概率。

类别图信息

image 型，代表类别图的图像信息。

类别图

image 型，代表类别图。

类别图

image 型，代表类别图的图像数据。

类别图宽度

int 型，代表类别图的宽度。

类别图高度

int 型，代表类别图的高度。

类别图像素格式

int 型，代表类别图的像素格式。

类别信息

image 型，代表类别图信息。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

灰度值

int 型，代表每个类别的灰度值。

缺陷概率图

image 型，代表每个缺陷概率图。

缺陷概率图

image 型，代表每个缺陷概率图的图像数据。

图像宽度

int 型，代表每个缺陷的概率图宽度。

图像高度

int 型，代表每个缺陷的概率图高度。

图像像素格式

int 型，代表每个缺陷的概率图像素格式。

缺陷*概率图（*代表 1-8）

image 型，代表每个缺陷的概率图的图像。

缺陷*概率图（*代表 1-8）

image 型，代表每个缺陷的概率图的图像数据。

缺陷*概率图宽度（*代表 1-8）

int 型，代表每个缺陷的概率图宽度。

缺陷*概率图高度（*代表 1-8）

int 型，代表每个缺陷的概率图高度。

缺陷*概率图像素格式（*代表 1-8）

int 型，代表每个缺陷的概率图像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.7.3 DL 实例分割

*DL 实例分割*可帮助您标记输入图像中的每个物体实例，并给相同类别的不同物体生成 ID。该模块常用于检测精度要求较高的定位、计数场景，且特别适用于识别目标重叠场景中的上下层目标。

模块原理

实例分割算法整合目标检测算法，并额外提取每个检测框内的特征图以预测对应的目标掩膜。该模块可输出每一个目标的独立掩膜。
模块的工作流程大致如下：

1. 检测目标。
   算子初步定位目标，预测目标定位框。详细算法原理请参见 *DL 目标检测* 的*模块原理*。
2. 提取 ROI 内特征。
   算子结合模型预测的目标定位框，提取每个定位框内的特征信息，并采样至统一大小。
3. 预测掩膜。
   算子基于采样后的内容，预测其掩膜和类别信息。
4. 输出结果。
   算子解析预测结果并按照所需格式输出。

使用方法

您可参照以下介绍，了解如何获取该模块所需的模型、可选前后序模块及如何在流程中使用该模块。
模型获取

您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您需通过该模块的**模型文件路径**，并开启**方案存模型**，将模型文件保存至方案中。

说明

模型训练的具体步骤，请参见对应训练工具的用户手册。

前后序模块

- 前序模块：通常为 ***图像源***，为该模块提供图像输入。
- 后序模块：该模块无指定后序模块，可处理模块输出结果即可。

示例流程

在该示例中，*DL 实例分割*识别了场景中的两个重叠物体，并分别为其渲染了定位框。

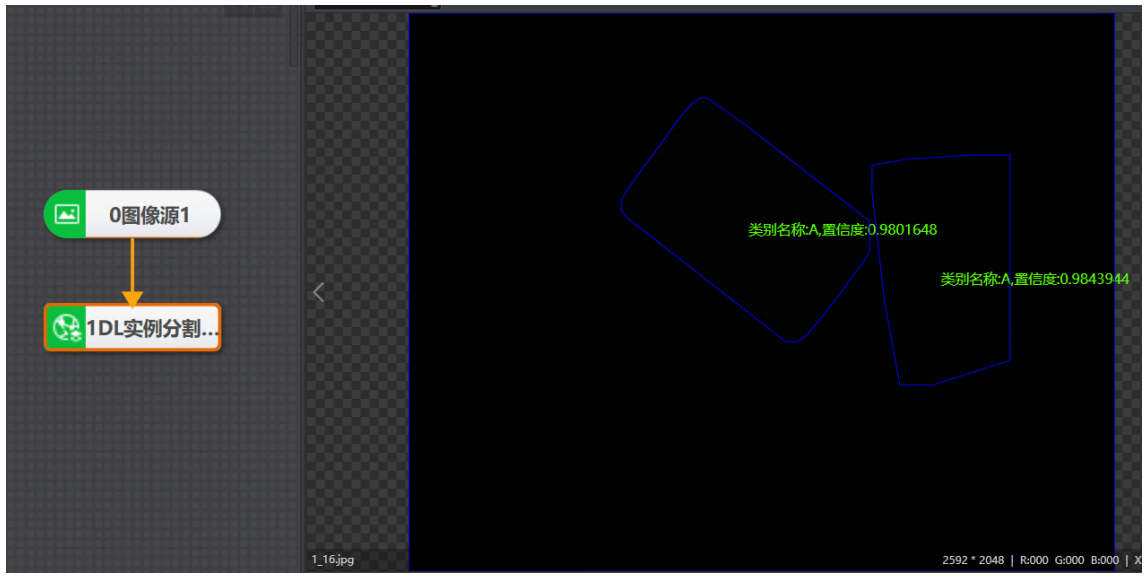


图 13-297 DL 实例分割示例效果图

模块性能调优

实际使用中，模型性能受到输入图像质量、模块设置等因素影响。此处列举了几种模型性能不佳的情况以及应对方法。您可以参照以下调整方式，优化模型的表现性能。

说明

在调整训练模型使用的训练图像、图像标注或训练参数后，您需在 VisionTrain 或 AI 训练平台中重新进行模型训练。在训练结束后，您还需导出和在该模块中加载模型。

当输出结果较差时，您需在 VisionTrain 或 AI 训练平台中检查如下几点：

- 检查训练模型时，是否选择了合适的**训练类型**。
- 检查数据集的标注是否正确。

- 检查模型在训练集上的性能。若模型性能（例如查准率和查全率）不及预期，您可检查模型训练参数和标注是否配置合理。
- 检查训练集和测试集的特征和数据分布，确保具有类似的分布特征。

若无以上问题，您需根据实际情况在 VisionTrain 或 AI 训练平台中增删训练图像、调整标注或调整标注框。在调整后，您需重新训练模型。

包含较多误检数据

向训练集中添加一些易被误检的正常图像。

存在较多漏检数据

- 检查输入图像中，待检测目标的像素面积是否占比过小。若过小，建议增大输入图像的分辨率。
- 尝试调低**目标框置信度**和**掩膜置信度**。若调低该参数后，模块能够检出相应目标，则您需向训练集中添加更多数据并重新训练。
- 若漏检现象仍严重，检查训练集和检测方案的合理性。

输出结果中掩膜质量较差

- 检查训练集中标注区域，特别是多边形标注的形状。

  说明

  通过多边形工具标注时，应紧贴物体轮廓。

- 检查存在掩膜质量问题的目标类型。若为大目标存在该问题，调整标注框的形状；若为小目标，尝试增大目标在图像中的占比。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数*。
- 结果显示参数详情，请参见 *结果显示*。

模型参数

模型文件路径

选择该模块运行时使用的模型文件（.bin）。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

按 ROI 裁图

开启时，模块仅预测 ROI 内图像，忽略 ROI 外信息。

关闭时，模块识别整张输入图像，但根据其他设置，如 **ROI 区域**，筛选过滤结果。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

目标框置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

目标框重叠率

设置预测目标框允许被遮挡的最大比例。当重叠率超过该参数值时，模块舍弃该预测结果。

掩膜置信度

掩膜得分的最小值。当掩膜实际得分低于该参数值时，则不输出该掩膜区域。

掩膜重叠率

两个掩膜区域重叠程度的最大值。当实际重叠率超过该参数值时，则过滤掉置信度较低的掩膜区域。

高级参数

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

渲染输出图像

开启该功能，则运行后图像中检测到的目标以掩膜方式呈现。

不同类别过滤

开启后，则可过滤不同类别的目标。若不开启，则只会按照同类别过滤。

模块结果

*DL 实例分割*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

目标个数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

目标像素值

int 型，代表每个实例分割检测目标的灰度值，用于区分不同的目标。

目标面积

int 型，代表每个检测目标的面积大小。

目标状态

int 型，代表每个定位框的状态。1 代表定位框正常，0 代表定位框异常。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

目标信息矩形

目标信息中心点

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

输出掩膜图

输出掩膜图

image 型，代表完成掩膜后输出的图像。

输出掩膜图宽度

int 型，代表输出掩膜图像的宽度大小。

输出掩膜图高度

int 型，代表输出掩膜图像的高度大小。

输出掩膜图像素

int 型，代表输出掩膜图像的像素格式，如 17301505 代表黑白图片。

输出渲染图

image 型，代表完成渲染后输出的图像。

输出渲染图

image 型，代表完成渲染后输出的图像。

输出渲染图宽度

int 型，代表输出渲染图的宽度大小。

输出渲染图高度

int 型，代表输出渲染图的高度大小。

输出渲染图像素

int 型，代表输出渲染图的像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

多边形轮廓点

point 型，代表检测到的轮廓点。

轮廓点 X

float 型，代表检测到的目标轮廓顶点的 X 坐标。

轮廓点 Y

float 型，代表检测到的目标轮廓顶点的 Y 坐标。

边缘点个数

int 型，代表输出边缘点的个数。

13.7.4 DL 目标检测

*DL 目标检测*用于检测图像中的目标，并确定目标的类别和位置信息。该模块常用于定位、计数和缺陷检测等场景。在这些场景中，您可使用该模块实时定位和识别多个目标，满足视觉应用需求。

模块原理

DL 目标检测是指利用神经网络提取图像中待测目标位置信息的方法。从目标检测算法的实现角度来看，可以分为单阶段和双阶段两种模式。

- 单阶段目标检测：直接输出待测目标的位置和类别信息。
- 双阶段目标检测：第一阶段先进行前背景检测生成候选框，第二个阶段对候选框进行精定位并给出其类别信息。

相较于双阶段目标检测而言，单阶段目标检测在性能相当的基础上，推理效率更高，且更容易训练。因此，在实际场景下大多都是使用单阶段的目标检测算法。这里将结合单阶段目标检测算法进行详细介绍。

单阶段目标检测的算法工作流程大致如下：

1. 提取特征。
   算子通过特征提取单元构成的骨干网络（Backbone）提取图像中不同层级的语义信息。
2. 融合多尺度特征。
   算子通过网络的颈部（Neck）提升检测性能。在该部分，算子采用特征金字塔结构进行上采样，并与通过骨干网络提取到的不同层级的特征融合。融合后，算子可获取检测目标不同层级的特征信息，用于检测不同尺寸的目标。
3. 检测目标特征。

算子通过检测头（Detection Head），从融合的特征中检测目标的位置和类别。

4. 结果输出。
   算子将预测结果按照所需的格式进行解析并输出。

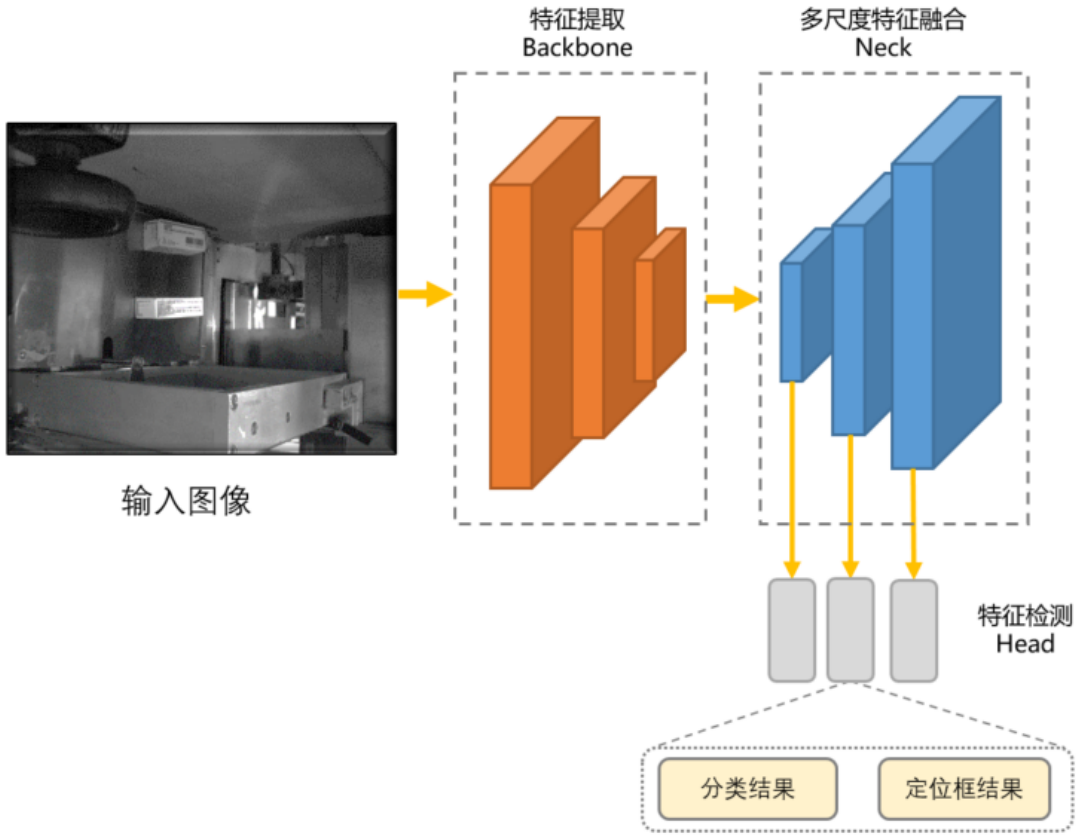


图 13-298 目标检测网络结构示意图

使用方法

您可参照以下介绍，了解如何获取该模块所需的模型、可选前后序模块及如何在流程中使用该模块。

模型获取

您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您需通过该模块的**模型文件路径**，并开启**方案存模型**，将模型文件保存至方案中。

说明

模型训练的具体步骤，请参见对应训练工具的用户手册。

使用限制

使用时，单个待检测目标大小应占输入图像像素面积的 1.31%以上。若不满足需求，您可尝试裁剪图像。

前后序模块

该模块的前后序模块详情如下。

- 前序模块：通常为 *图像源*，为该模块提供图像输入。
- 后序模块：该模块无指定后序模块，可处理模块输出结果即可。

示例流程

在该示例中，*DL 目标检测*识别场景中的三个盒子，并分别输出定位框。

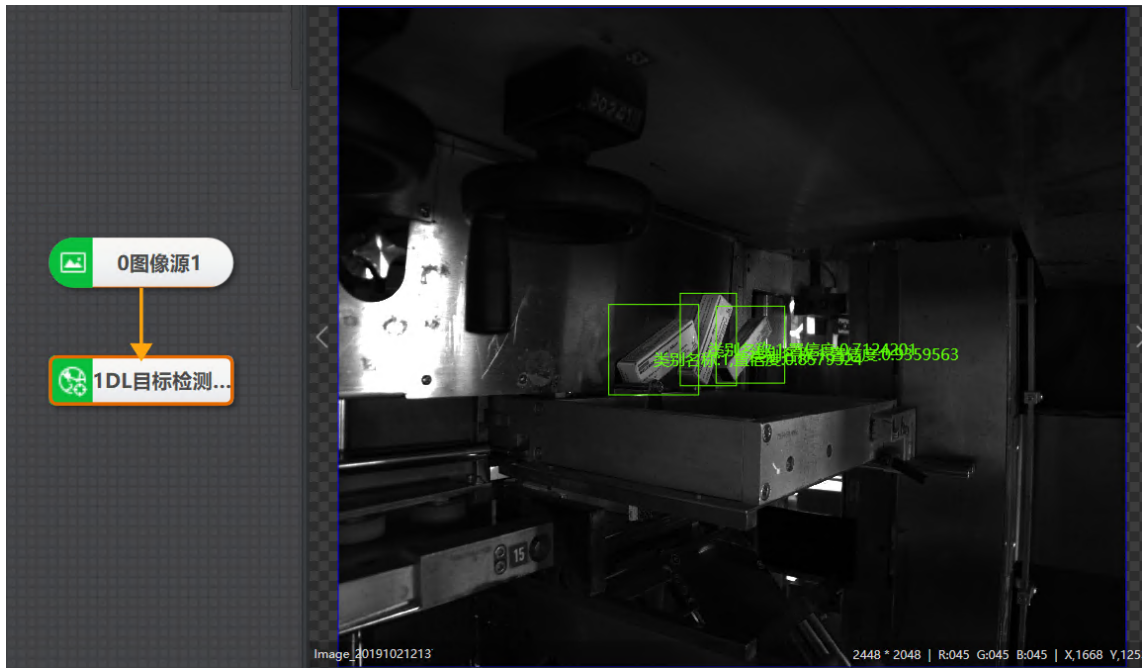


图 13-299 检测示例

模块性能调优

实际使用中，模型性能受到输入图像质量、模块设置等因素影响。此处列举了几种模型性能不佳的情况以及应对方法。您可以参照以下调整方式，优化模型的表现性能。

当输出结果较差时，您需在 VisionTrain 或 AI 训练平台中检查如下几点：

- 检查在 VisionTrain 或 AI 训练平台训练模型时，是否选择了合适的**训练类型**。
- 检查数据集的标注是否正确。
- 检查模型在训练集上的性能。若模型性能（例如查准率和查全率）不及预期，您可检查模型训练参数和标注是否配置合理。
- 检查训练集和测试集的特征和数据分布，确保具有类似的分布特征。

若无以上问题，您需根据实际情况在 VisionTrain 或 AI 训练平台中增删训练图像、调整标注或调整标注框。在调整后，您需重新训练模型。

包含较多误检数据

采用背景图模式，将易被误检的正常图像设置为背景图。

存在较多漏检数据

- 检查输入图像中，待检测目标的像素面积是否占比过小。若过小，建议增大输入图像的分辨率。
- 尝试调低**最小置信度**参数值。若调低后，模块能够检出相应目标，则您需向训练集中添加更多数据并重新训练。
- 若漏检现象仍严重，检查训练集和检测方案的合理性。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

可设置该模块运行时使用的模型文件。默认加载软件自带的模型文件，同时也可以加载自行训练的模型文件。

说明

该模块提供 2 个模型文件，DetectDefault.bin 为默认模型文件，DetectFast.bin 为快速模型文件。如需更换加载的模型，您可将光标悬浮在 ，并在菜单中单击对应模型的 。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行参数

获取模型 ROI

开启后，模块自动获取模型文件中的 ROI 信息。此时，您无需配置*基本参数*中的 **ROI 区域**。

模型 ROI 可在 VisionTrain 或 AI 训练平台训练模型时设置。设置后，训练的模型包含 ROI 信息。

按 ROI 裁图

开启时，模块仅预测 ROI 内图像，忽略 ROI 外信息。

关闭时，模块识别整张输入图像，但根据其他设置，如 **ROI 区域**，筛选过滤结果。

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

小目标模式

该模式针对大图小目标场景（待检测目标尺寸占输入图像尺寸低于 0.5%时）优化训练过程，从而提高相应场景下模块的性能。

说明

开启该参数时，无法配置**按 ROI 裁图**。

横向滑窗数
纵向滑窗数

分别设置横向和纵向的图片切割个数。

滑窗重叠率

图片切割时相邻图片的重叠比率。该比率为重叠部分面积和单个滑窗的面积的比值。若设置该参数为 0，相邻滑窗间彼此独立。

算法参数

最大查找个数

设置模块能够查找到的最大目标数量。若实际查找到的数量大于该参数值，模块舍弃超出参数值的对应目标，仅输出参数值数量的目标。不确定待检测目标数量时，建议调大该参数。

最小置信度

设置预测结果的最小置信度。置信度是模型对其预测结果的确信程度。置信度越高，代表模型越确定预测框正确。若某预测框的置信度小于该参数值，则模块舍弃该预测结果。

若实际检测场景中存在漏检时，可适当降低该参数值，获取更多检测结果；若存在错检时，可提高该参数值，以获取更准确的结果。

最大重叠率

设置两个预测框的最大重叠比例。该重叠比例代表两个预测框的重叠面积与较小预测框的比值。当重叠率超过该参数值时，模块舍弃较小置信度的预测结果。

高级参数

目标排序

设置模块输出预测结果的顺序。

- 按中心点 X/Y 坐标升序：按照预测框中心点的 X/Y 坐标值，从小到大排序预测结果。
- 按置信度降序：按照预测结果的确信程度，从大到小排序预测结果。

边缘筛选使能

开启后，模块筛选超出 ROI 边界的预测结果，需配置**最小边缘分数**。

最小边缘分数

表示预测框在 ROI 内的比例，即预测框在 ROI 内的面积与预测框整体面积的比值。若某一预测框在 ROI 内的比例小于该值，则模块舍弃该预测结果。

不同类别过滤

开启后，则可过滤不同类别的目标。若不开启，则只会按照同类别过滤。

过滤参数

角度使能

用于筛选预测结果。开启时，需设置**角度范围**。若预测结果框相对于水平方向的旋转超出设置的**角度范围**，则模块舍弃该预测结果。此处的旋转指：预测结果框相对于水平线进行顺时针旋转时，角度大于 0；逆时针旋转时，角度小于 0。

宽度使能

用于筛选预测结果。开启后，需设置**宽度范围**。若预测结果框宽度超出设置的**宽度范围**，则模块舍弃该预测结果。

高度使能

用于筛选预测结果。开启后，需设置**高度范围**。若预测结果框高度超出设置的**高度范围**，则模块舍弃该预测结果。

面积使能

用于筛选预测结果。开启后，需设置**面积范围**。若预测结果框面积超出设置的**面积范围**，则模块舍弃该预测结果。

长短边比使能

用于筛选预测结果。开启后，需设置**长短边比范围**。若预测结果框的长短边比值超出设置的**长短边比范围**，则模块舍弃该预测结果。

模块结果

*DL 目标检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

目标个数

int 型，代表检测区域内检测到的定位框个数。

目标类别

int 型，代表检测到的目标实例的种类序号。

说明

需预先训练好模型并标记序号。

目标置信度

float 型，代表每个定位框的置信度。

目标状态

int 型，代表每个定位框的状态。1 代表定位框正常，0 代表定位框异常。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

目标半径

float 型，代表每个检测目标的半径。

目标信息矩形

目标信息中心点

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.7.5 DL 分类

*DL 分类*模块可用于分类目标，适用于不同类别物品之间差异较大、存在明确分类标准的场景。您将该模块用于简单的缺陷检测、产品分拣和分类场景，例如分类工件和分类布匹纹理。

模块原理

基于深度学习的图像分类技术，本质上是利用神经网络提取图像特征，并根据其特征分布情况给出类别信息。一般来说，对于一张图像，分类网络只会给出预测的类别信息以及该图片属于该类别的可能性，即得分或置信度。
典型的分类神经网络工作流程大致如下：

1. 提取特征。
   算子通过卷积操作可以提取图像不同层级的特征，包含轮廓、灰度、结构和关系等特征。
2. 降采样图像。
   算子通过池化层降采样图像并增大卷积的感受野，卷积可以感知更大区域的特征分布。
3. 特征激活。
   算子基于激活层增强网络性能，使网络能够更好地拟合不同的图像特征。
4. 算子重复上述步骤，提取不同层级的图像特征。
5. 分类。
   算子通过全连接层和分类器分类提取到的一组特征，预测当前图像属于不同类别的概率。

使用方法

您可参照以下介绍，了解如何获取该模块所需的模型、可选前后序模块及如何在流程中使用该模块。

模型获取

您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您需通过该模块的**模型文件路径**，并开启**方案存模型**，将模型文件保存至方案中。

说明

模型训练的具体步骤，请参见对应训练工具的用户手册。

使用限制

该模块对输入图像和目标类别有要求。使用该模块分类和分级目标时，您需确保：

- 输入图像可分为多个类别。
- 输入图像包含的类别固定，且目标类间应具有明确的分类标准。否则，分类效果可能不佳。

  说明

  若后续会添加新的目标类别，建议您调用 *DL 图像检索*。
  在添加新的类别后，您需在 VisionTrain 或 AI 训练平台中添加对应数据，并重新训练模型。

- 输入的图像尺寸需至少大于 16×16，建议大于 256×256。并且，目标需要在输入图像中占据较大比例。

  说明

  您可通过 *定位* 和 *图像处理* 中的模块，预处理输入图像，裁剪干扰背景。

前后序模块

该模块的前后序模块详情如下。

- 前序模块：通常为 *图像源*，为该模块提供图像输入。您也可通过 *仿射变换* 等模块裁剪背景干扰。
- 后序模块：该模块无指定后序模块，可处理模块输出结果即可。

示例流程

在该示例中，*DL 分类*用于对玉米粒的完整度进行分类。



图 13-300 DL 分类示例

模型性能调优

实际使用中，模型性能受到输入图像质量、模块设置等因素影响。此处列举了几种模型性能不佳的情况以及应对方法。您可以参照以下调整方式，优化模型的表现性能。

说明

在调整训练模型使用的训练图像、图像标注或训练参数后，您需在 VisionTrain 或 AI 训练平台中重新进行模型训练。在训练结束后，您还需导出和在该模块中加载模型。

模型仅输出一个类别

检查训练时数据集类别分布。若某一类别数据占比过大，模型识别能力会下降。您可向训练集添加其他类别的数据，并重新进行训练。

分类结果准确率较低

- 检查训练集的各类别图像数量，单个类别下，一般建议添加 20 张以上图像。您还可以在 VisionTrain 或 AI 训练平台中适当增加**数据增强**。
- 检查模型在训练集上的性能。若模型性能（例如查准率和查全率）不及预期，您可检查 VisionTrain 或 AI 训练平台中模型训练参数是否合理，例如，**学习率**过大、**迭代轮次**较小等；还可以检查数据标注。
- 若模型性能在训练集上表现良好，您可检查数据集标注正确与否，和训练集与实际检测的数据分布。

模型部分类别准确率低

增加对应类别的训练图像，或检查该类别的标注是否正确。

整体性能较好，但错误分类少量样本

您需要分析误差样本，若：

- 样本特征区分度高。模型泛化能力较弱，您可在 VisionTrain 或 AI 训练平台增加**数据增强**或增加训练集图像数据。
- 样本特征区分度低。您可适当向训练集添加该类图像，您还可以尝试优化标注和使用高精度模型。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

选择该模块运行时使用的模型文件（.bin）。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

获取模型 ROI

开启后，模块自动获取模型文件中的 ROI 信息。此时，您无需配置*基本参数*中的 **ROI 区域**。

模型 ROI 可在 VisionTrain 或 AI 训练平台训练模型时设置。设置后，训练的模型包含 ROI 信息。

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

算法参数

前 K 个类别

输出置信度得分最高的 k 个类别的索引号和对应的置信度。

模块结果

*DL 分类*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

类别个数

int 型，代表图像中检测到目标的类别数量。

类别标签

int 型，代表在分类任务中用来表示不同类别的标签或标识符。

类别概率

float 型，代表属于各个类别的概率。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

置信度

float 型，代表识别的字符的置信度。数组，其长度等于字符个数。

ROI 序号

int 型，代表当前类别对应的 ROI 序号。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.7.6 DL 图像检索

*DL 图像检索*模块可用于物体识别和分拣场景中，将输入图像归类至若干类别的某一类。相较于 *DL 分类*模块，该模块更适用于类内差异较小、类间差异较大的场景。此外，该模块支持扩展检索类别，您可向数据库添加新的类别。

模块原理

该模块主要基于“注册”和“检索”两个流程。在“注册”过程中，您需添加图像并手动归类图像。算子基于输入的图像及类别建立数据库。在“检索”过程中，模块在已注册图像的特征中检索与输入图像特征相似的特征，并根据两者的相似度判定输入图像的类别。

1. 提取特征。
   算子通过神经网络从输入图像中获取样本高维特征。
2. 提取与存储注册图像。
   算子在本地建立数据库，用于存储图像、对应图像特征和两者之间的关联。
3. 提取待检索图像特征。
4. 计算相似度和排序结果。
   算子使用提取到的图像特征，在数据库中检索相似特征。按照相似度降序，算子返回待检索图像的类别。

使用方法

您可参照以下介绍，了解如何获取该模块所需的模型、可选前后序模块及如何在流程中使用该模块。

模型获取

模块内置了通用模型，您可基于此模型直接通过***注册图像***注册图像及其对应类别。

您也可以使用 VisionTrain 或 AI 训练平台自行训练模型。训练并获取模型后，您需通过该模块的**模型文件路径**，并开启**方案存模型**，将模型文件保存至方案中。

说明

模型训练的具体步骤，请参见对应训练工具的用户手册。

使用限制

- 输入的图像尺寸需至少为 16×16 像素，建议大于 256×256 像素。
- 运行流程检索图像时，若通过 ***图像源*** 模块向 *DL 图像检索*输入图像，输入的图像必须为 **RGB24** 格式。

  说明

  注册图像时无此限制。您也可以使用 MONO8 格式的图像注册。

前后序模块

- 在流程中，该模块的前序模块为 ***图像源***，用于提供图像输入。
  该模块还可以搭配 ***定位*** 类模块和 ***仿射变换*** 模块，用于放大目标特征，从而实现更好的检索效果。
- 该模块无指定后序模块。

主要配置步骤

您可以参照如下流程，在流程中调用该模块，完成视觉应用。

1. 拖拽图像源模块至流程编辑区，并输入图像。
2. 拖拽 *DL 图像检索*模块至流程编辑区，双击该模块配置相应配置，加载训练好的图像检索模型文件以及训练得到的 Gallery 文件。

   说明

   您也可加载该模块内置的算法模型。此时，您需注册需分类的图像。
3. 单击*执行*，获取检索结果，如下图所示。

说明

如需添加新的类别，您可以通过 *Gallery 管理* → *注册图像*，添加新的类别。详细操作请参见下文 *Gallery 管理*。



图 13-301 DL 图像检索示例效果图

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 ***基本参数*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

模型参数

模型文件路径

设置该模块运行时使用的模型文件（.bin）。此处默认加载模块内部的对应的模型文件，您也可以加载自行训练的模型文件。

Gallery 路径

Gallery 文件通过**注册图像**获取。在完成图像注册后，您需在此处加载保存的 Gallery 文件。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

批处理使能

仅 GPU 模块才可配置。用于设置模块并行处理多 ROI 区域，提高处理效率。开启后，需配置**批处理等级**，调节并行处理的能力大小。**批处理等级**的，参数值越大，模块并行处理的能力越强，效率越高，同时也会占用更多性能。

算法参数

前 K 个类别

输出置信度得分最高的 k 个类别的索引号和对应的置信度。

最小相似度

设置分类目标的相似程度最小值，过滤相似度小于该参数值的分类结果。当场景中存在漏检时，可适当调低该值；存在错检时，可调高该参数值。

Gallery 管理

查询类别

单击*查询类别*，弹框界面如下左图所示。文本框中会显示当前 Gallery 中总类别数，单击右侧 可以直接导出所有类别信息。在文本框中输入待查询的类别名称后，下方会自动匹配相关类别的缩略图信息、类别名称以及 Gallery 中已注册的图像个数，如下右图所示。



图 13-302 查询类别

说明

您可单击类别缩略图右上的 ，删除某一类别的注册图像。删除后，单击*保存*或*另存为*，保存 gallery 文件。

如通过*另存为*保存 gallery 文件，您需要手动重新配置 **Gallery 路径**参数。

注册图像

单击*注册图像*，弹框界面如下图所示。

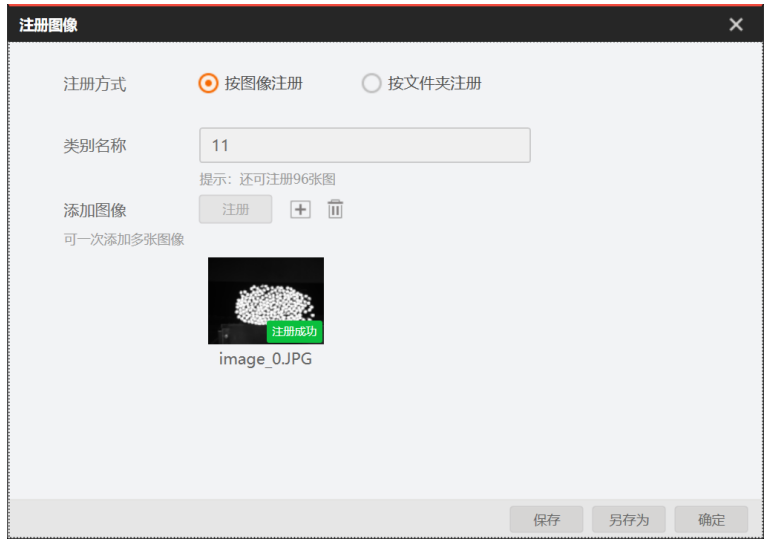


图 13-303 注册图像

注册方式分为按图像注册和按文件夹注册两种。

- 按图像注册：首先输入要注册的类别名称，单击 ⊞ 可以添加待注册的图像，可以一次性选入多个，也可分批次选择图像，再单击***注册***即可注册所有添加的图像。
- 按文件夹注册：单击***选择文件夹***后选择文件夹，此时默认将文件夹名称作为类别名称进行注册，也支持对类别名称进行修改。同时也应注意，一个类别最多可注册 100 张图片。此外，按文件夹注册支持一次注册多个文件夹，即同时注册多个类别。

说明

添加类别后，单击***保存***或***另存为***，保存 gallery 文件。
如通过***另存为***保存 gallery 文件，您需要手动重新配置 **Gallery 路径**参数。

模块结果

*DL 图像检索*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

类别个数

int 型，代表图像中检测到目标的类别数量。

相似度

float 型，代表图像中检测到目标与训练模型中相应类别的相似度大小，越接近 1 代表越相似 。

图像索引

int 型，代表图像中检测到的目标与注册图像中的第几张图像相似度最高。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

ROI 序号

int 型，代表当前类别对应的 ROI 序号。

模型标识

string 型，代表 bin 模型的特定标识，用来判断当前模型版本。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.7.7 DL 无监督分类

*DL 无监督分类*模块可用于提取正常样本图像的特征，帮助您检出与正常样本不一致的异常情况。该模块适用于单类别目标、产品一致性较强、且背景较为单一简单的场景。

说明

该模块仅可在含独立显卡的工控机上使用。独立显卡性能要求请参见 *推荐硬件* 。

模块原理

该模块利用神经网络与特征分布统计技术构建正常样本的全局类别信息，分类正常和异常样本。使用前，您需准备包含多种类别的正常样本图像，并确保每种类别内的数据量较为均衡。算法运行主要分为以下几个步骤：

1. 提取特征。
   算子从正常样本中提取中间层特征。中间层特征包含一定的空间信息和语义信息。
2. 构建标准模板特征库。
   算子过滤全部的正常图像特征，并选择最能代表所有特征信息的核心特征。
3. 算子利用特征判别分类，得到异常得分。

使用方法

您可以参照如下内容，使用该模块分类简单场景中的异常。

使用场景

该模块可用于背景简单、产品一致性高、并且待检测目标在图像中姿态大小较为一致的异常分类场景。

获取模型

在流程中调用该模块前，需在 AI 训练平台或 VisionTrain 中训练模型。训练完成后，在该模块中使用对应的模型文件（.bin）。

训练用于无监督分类的模型时，您可以选择如下 2 种模式：

- 普通模式：用于缺陷样本的判断，例如划痕、脏污等。
- 逻辑模式：用于特征明显的逻辑判断，例如缺失检测、有无检测、数量检测等。

说明

模型训练的具体步骤，请参见对应训练工具的用户手册。

前后序模块

- 使用时，前序模块可为 ***图像源***，向该模块提供图像输入。
- 该模块无指定后序模块。

示例方案

如下图所示，在导入模型后，可实现图像异常分类。

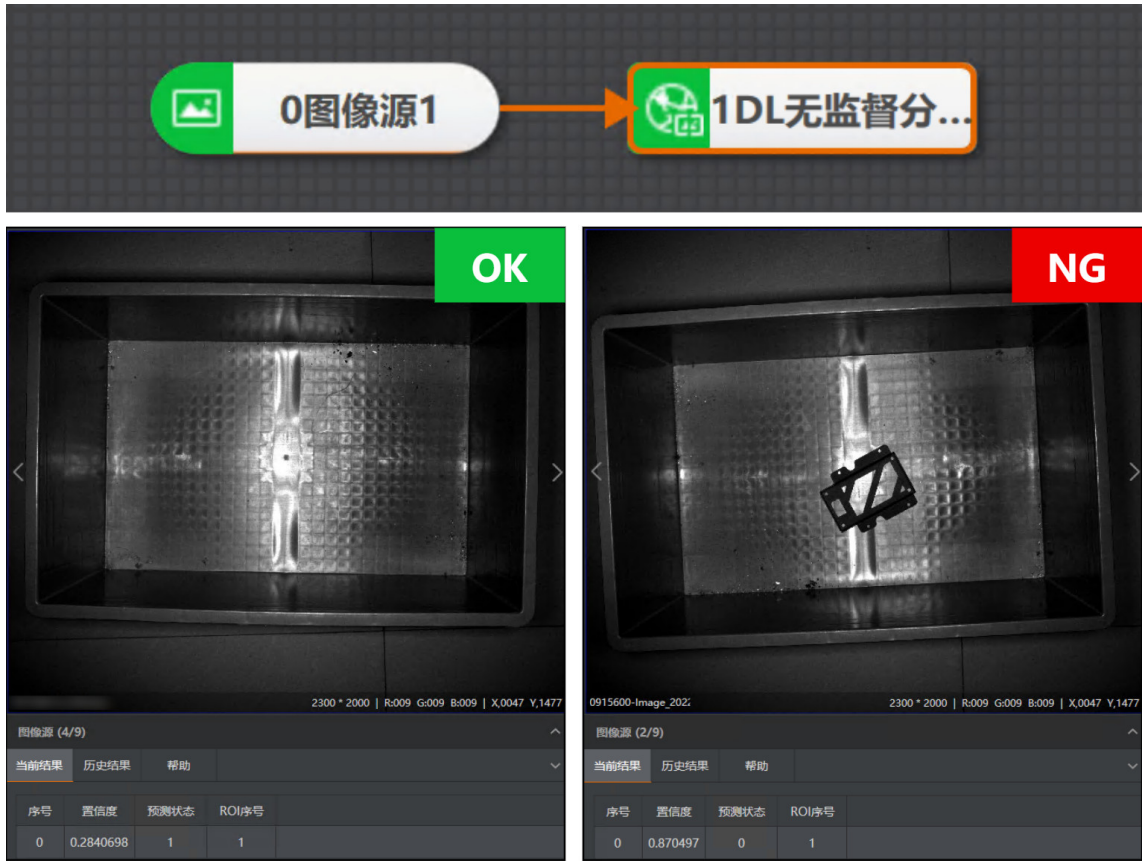


图 13-304 示例流程

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

选择该模块运行时使用的模型文件（.bin）。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行模式

获取模型 ROI

开启后，模块自动获取模型文件中的 ROI 信息。此时，您无需配置*基本参数*中的 **ROI 区域**。

模型 ROI 可在 VisionTrain 或 AI 训练平台训练模型时设置。设置后，训练的模型包含 ROI 信息。

算法参数

置信度阈值

表示检测图片为 NG 图片的最小分数。当图像的置信度低于该数值时，则为 OK；否则为 NG。

模块结果

*DL 无监督分类*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

置信度

float 型，代表识别的字符的置信度。数组，其长度等于字符个数。

预测状态

int 型，代表预测结果，0 为 NG，1 为 OK。

ROI 序号

int 型，代表当前类别对应的 ROI 序号。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.7.8 DL 无监督分割

*DL 无监督分割*可使用基于正常样本训练的模型，推理待检测图像中的异常缺陷。该模块在背景简单、产品一致性高、且待检测目标在图像中姿态较为一致的场景中，可作为主要检测工具，用于识别缺陷图像。其他缺陷检测场景中，该模块常作为辅助工具，用于收集含缺陷图像。

说明

该模块仅可在含独立显卡的工控机上使用。独立显卡性能要求请参见 *推荐硬件* 。

模块原理

无监督分割是指仅利用无缺陷图像进行训练并对缺陷进行检测的算法。无监督分割属于无监督模式，无需标注缺陷样本，只需训练 OK 样本即可。

该算法通过计算原图和重建的相似度推理缺陷区域，算法流程大致如下：

1. 提取特征。
   算子对图像降采样，提取其高层级特征并将特征压缩为一组特征向量。
2. 重建图像。

算子基于上一步得到的特征向量进行上采样，生成重建的图像。

3. 计算相似度。
   算子计算原图与重建图的相似度，并通过相似度获得缺陷区域。

使用方法

使用场景

*DL 无监督分割*适用于背景简单、产品一致性高、可对齐、缺陷明显的场景。在这些简单场景中，您可使用该模块作为主要检测工具。

大部分场景中，您可使用该工具获取含缺陷图像。例如，在无法收集大量含缺陷样本的场景中，您可先基于正常图像训练该模块的模型，然后通过该模块检测和分类出含缺陷的图像，再选择有代表性的含缺陷图像进行有监督的模型训练。

模型获取

您可使用 VisionTrain 或 AI 训练平台自行训练模型。训练完成并获取模型后，您需通过该模块的**模型文件路径**，并开启**方案存模型**，将模型文件保存至方案中。

说明

模型训练的具体步骤，请参见对应训练工具的用户手册。

前后序模块

- 前序模块一般为 *图像源*，为该模块提供图像输入。还可搭配 *定位* 、 *仿射变换* 、 *位置修正* 等模块，对齐检测目标，提高 *DL 无监督分割*模块的表现。
- 后序模块可搭配 *Blob 分析* 模块，可视化显示结果并输出缺陷区域的 Blob 面积。

应用示例

下图示例用于检测机场周转框中是否有物品遗漏。该示例中，该模块检测 *图像源* 模块输入的周转框图像中是否包含遗留物品。若模块认为存在遗留物品，则增大概率图的对应区域灰度值。在获得概率图后，通过 *Blob 分析* 可视化显示概率图，并输出缺陷区域面积。



图 13-305 DL 无监督分割示例

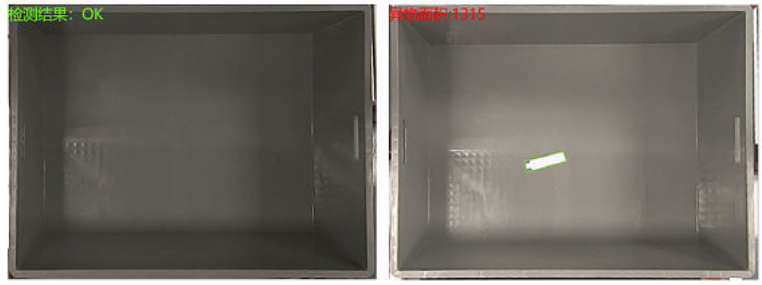


图 13-306 DL 无监督分割示例执行结果

模块性能调优

实际使用中，模型性能受到输入图像质量、模块设置等因素影响。您可以参照以下调整方式，优化模型的表现性能。

- 检查训练集是否包含缺陷图。若包含，则将缺陷图从训练集中移除。
- 检查训练集图像中，待检测目标是否对齐、是否存在复杂纹理变化。若待检测目标存在较多纹理变化，建议您使用 *DL 快速图像分割* 进行有监督训练。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

模型参数

模型文件路径

选择该模块运行时使用的模型文件（.bin）。

方案存模型

开启后，保存方案时，模块将当前使用的模型文件保存到方案或流程文件。

运行参数

获取模型 ROI

开启后，模块自动获取模型文件中的 ROI 信息。此时，您无需配置*基本参数*中的 **ROI 区域**。

模型 ROI 可在 VisionTrain 或 AI 训练平台训练模型时设置。设置后，训练的模型包含 ROI 信息。

模块结果

*DL 无监督分割*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

概率图数据

概率图

image 型，代表概率图。

概率图宽度

int 型，代表概率图的宽度。

概率图高度

int 型，代表概率图的高度。

概率图像素格式

int 型，代表概率图的像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.8 标定

标定分类下的模块主要用于生成两个坐标系之间的转换关系（即标定文件），并进行应用。

不同标定模块的原理和使用场景有所差别：

- ***N 点标定***：通过输入多个一一对应的图像像素点和运动机构物理点，建立图像坐标系和运动机构物理坐标系之间的转换关系。
- ***平移旋转标定***：基于 ***N 点标定***实现“小视野+大角度”类型的旋转标定，自动计算手性一致性和旋转一致性。

> **说明**
>
> 图像坐标系均为左手坐标系，若运动机构为左手坐标系，则手性一致性为 1；若运动机构为右手坐标系，则手性一致性为-1。

- ***旋转标定***：即***平移旋转标定***模块中旋转部分的标定。
- ***N 图像标定***：使用海康自研标定板，建立图像坐标系和运动机构物理坐标系之间的转换关系。
- ***标定板标定***：使用标定板，建立图像坐标系和物理坐标系之间的转换关系。
- ***相机映射***：通过前序模块输出的点集，建立两个图像坐标系之间的转换关系。
- ***映射标定***：通过海康自研标定板，建立两个图像坐标系之间的转换关系。
- ***畸变标定***：对带畸变的标定板图像生成畸变校正文件。供 ***畸变校正*** 模块进行校正。
- ***坐标系***：在图像坐标系中自定义创建一个坐标系，并通过坐标矩阵建立图像坐标系和自定义坐标系之间的关系。
- ***标定加载***：加载部分模块生成的标定文件并输出相关信息。

13.8.1 N 点标定

N 点标定模块主要用于计算相机的图像坐标系和运动机构的物理坐标系之间的转换关系。通过输入多个一一对应的图像像素点和运动机构物理点，计算得出两个坐标系之间的转换关系（即标定文件）。在需要视觉引导运动机构进行抓取、纠偏和对位等动作的应用场景中应用较为广泛。

> **说明**
>
> 该模块使用时需确保 N≥4。较为常见的是 9 点标定和 12 点标定。若运动机构运动是共轴的，需进行 9 次平移，此时使用 9 点标定；若运动机构做不共轴运动，进行 9 次平移后还需进行 3 次旋转，此时使用 12 点标定。

本节内容包含：

- *模块原理*
- *使用方法*
- *调试建议*

- *参数配置*
- *模块结果*

模块原理

在与机器臂相关的视觉任务中，比较关键的是将机器臂的物理坐标与相机特征点的像素坐标进行关联。当机器臂在一个特定的平面运动时，其运动状态主要为平移和旋转这两种。该模块主要就是用来确定平移或旋转的变化在图像坐标系和物理坐标系间的对应关系。

关于平移

对于一组有 N 个点的坐标，可通过一个平移转换关系将这组坐标映射到另一个位置。

对应公式为 $P^w=HP^I$，其中 P^I 表示像素坐标，P^w 表示物理坐标，H 表示转换关系。通过该转换关系可从像素坐标转换到物理坐标。

同理，可得到公式 $P^I=H^{-1}P^w$。其中 H^{-1} 表示与 H 相反的转换关系。通过该转换关系可从物理坐标转换到像素坐标。

实际标定时，一般会存在一个基准量，物理坐标与像素坐标均会减去该基准量，因此 H 和 H^{-1} 代表的是像素坐标变化量和世界坐标变化量之间的转换关系。

关于旋转

当运动机构需进行旋转有关任务时，需使用 N 点标定进行旋转标定，从而确定旋转轴中心的位置，并计算得到旋转的变换关系。

旋转前后的公式如下所示。其中 ΔP^w 表示旋转前后的变化量，T 是旋转前点的位置，R 是旋转的变换矩阵。

$$\Delta P^w = T - R * T$$

对于上述公式中的 T 满足如下公式。其中 $P_{rot}{}^w$ 表示旋转中心。

$$T = HP^I - P_{rot}{}^w$$

通过以上 2 个公式，可解决得到旋转中心的坐标。

使用方法

该模块使用时，需通过前序模块获取图像点和物理点。图像点通过 ***轮廓匹配*** 相关模块、 ***位置修正*** 模块和其他可实现精定位的模块获取。物理点可通过模块自动生成，也可接收通信数据。选择接收通信数据时，需通过 ***接收数据*** 和 ***协议解析*** 模块实现。下图为某个方案中部分模块搭建的示意。

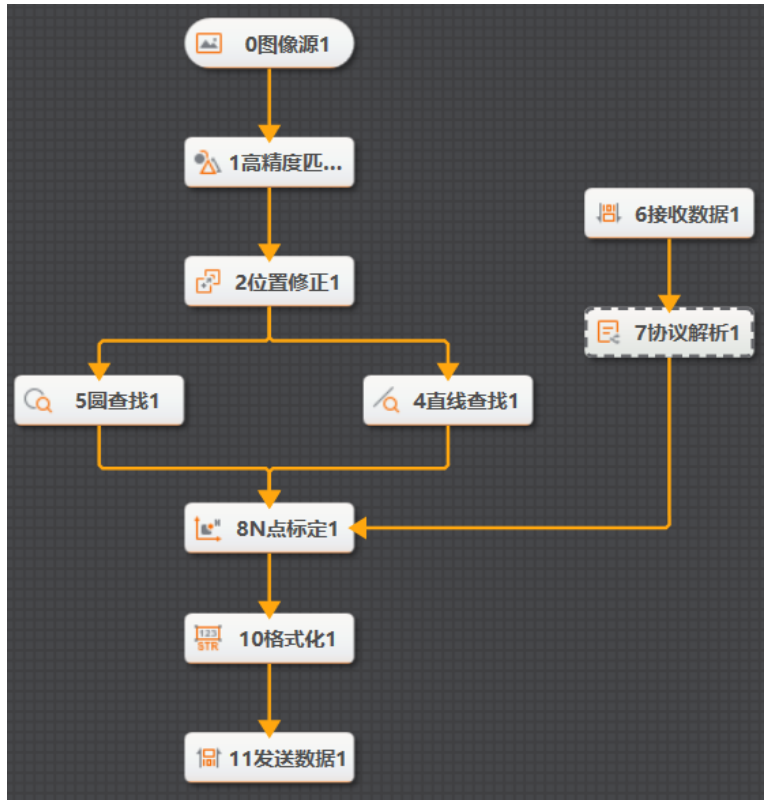


图 13-307 模块使用示例

其次，后序模块一般为 *单点抓取* 、 *单点对位* 、 *单点纠偏* 等模块，引导运动机构进行抓取、纠偏、对位等动作。

调试建议

使用该模块标定时，若反馈标定失败或存在较大误差。可按照以下方面查看能否解决。

- 平移误差较大时，需查看图像中的平移运动轨迹是否正常。通过图像点的 X/Y 方向位移轨迹是否互相平行来判断。若不平行，会导致标定误差增大。此时可查看不平行轨迹对应点的具体情况，可能是机构运动误差导致，也可能是前序模块输出存在误差导致。

  说明

  X 轨迹和 Y 轨迹夹角可以不是 90°，此时可能为相机非垂直拍摄，也可能机构本身 X、Y 并非垂直，该情况不会导致标定误差增加。

- 旋转误差较大时，需查看旋转中心的位置是否合理。若旋转中心位置不正确，检查输入角度是否有误。
  - 某些大半径场景，由于机构限制，需要较大的旋转半径，所以需要提高旋转中心的准确性。建议多旋转几次，增加旋转数据输入，可提高标定结果的准确性。
  - 小角度场景，为得到更准确的旋转中心，一般旋转角度不能太小，用于计算旋转中心的点不能太近。若旋转角度很小，可考虑多运动几次，多个点计算的旋转中心更加准确。

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

标定点获取

可选触发获取和手动输入两种方式。

选择触发获取时，还需设置以下参数：

标定点输入

可选按点、按坐标输入。

图像点
图像坐标 X/Y
图像角度

图像标定点的位置及角度信息，一般从前序模块订阅。

物理点
物理坐标 X/Y
物理角度

运动机构的坐标点的位置及角度信息。每个图像点对应一个运动机构的物理点。

说明

该参数与物理坐标系参数使用时需二选一，无法同时使用。推荐使用物理坐标系参数自动生成。

示教

启用后，可通过相关设置判断外部输入的信号是否为示教信号。

外部输入字符

可订阅外部输入的字符，例如外部通信的 TRIGGER STRING。

外部触发字符

判断**外部输入字符**是否为示教信号的依据。当**外部输入字符**与该参数内容一致时，则判断该信号为示教信号。

平移次数

设置平移运动的输入点数，只针对 X/Y 方向的平移，一般设置为 9。

旋转次数

设置旋转运动的输入点数。旋转轴与图像中心不共轴时，需设置该参数，一般设置为 3，且旋转是在第 5 个点的位置进行。

点击 可在编辑标定点页签查看并编辑各个点对应的图像坐标、物理坐标及角度。

标定点获取选择手动输入时，需通过此处自定义或导入标定点的信息。

自定义

双击数据直接修改即可。

导入

可加载已有的点集对信息文件（txt 格式）。

导出

可将当前显示的点集对信息导出到本地。

清空标定点

清空当前显示的点集对信息。

物理坐标系参数

可通过设置以下参数自定义物理坐标系。

说明

仅在未设置**物理点、物理坐标 X/Y** 或**物理角度**时，该分类的参数可见且可设置。

基准点 X/Y

对应图像标定原点的物理点坐标，单位为 mm。使用相对坐标时，一般设置为（0,0）；使用绝对坐标时，设置为第 5 个点的物理坐标。

偏移 X/Y

运动机构每次运动时 X/Y 方向的物理偏移量，可正可负，单位为 mm。即第 5 个点相对第 1 个点在 X/Y 方向的偏移量。

移动优先

可设置运动机构优先偏移的方向，可选 X 优先、Y 优先。

换向移动次数

可设置运动机构每移动多少次进行换向移动。

基准角度

运动机构旋转的初始角度。

角度偏移

运动机构每次旋转的角度。

举例：假设旋转 3 次，且旋转角度依次为 *-10° → 0° → 10°*，则**基准角度**为 0，**角度偏移**为 10。

下图为部分参数设置的示意图，其中**平移次数**为 9，**旋转次数**为 3，**基准点 X/Y** 均为 0，**偏移量 X/Y** 均为 5，**移动优先**为 X 优先，**换向移动**为 3。

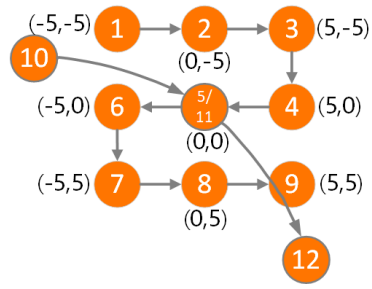


图 13-308 参数效果

使用相对坐标

可设置运动机构的物理坐标系为相对坐标或绝对坐标。开启时，为相对坐标；未开启时，为绝对坐标。默认关闭。开启后，需配置**标定原点**参数，设置第几个点为标定原点。一般设置为 4。因为物理坐标系从 0 开始计数，中间的点为原点。

标定文件路径

点击 可自定义选择标定文件的存储路径，也可通过订阅的方式加载已有的标定文件或新建标定文件。新建标定文件时，在选择的路径下，自定义输入***文件名***并单击***打开***即可，如下图所示。

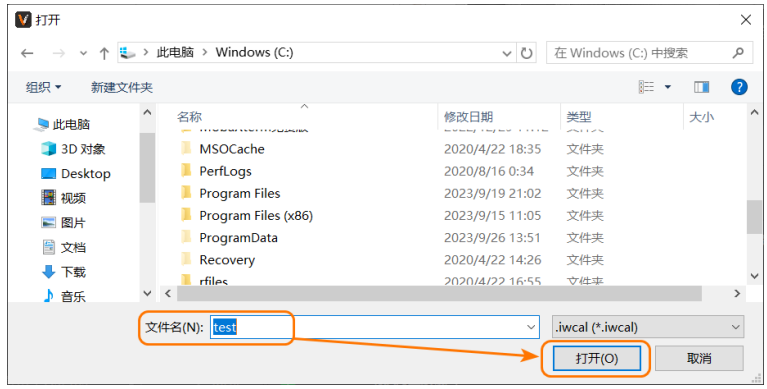


图 13-309 新建标定文件

更新文件

开启该参数后，该模块进行新一轮标定时，会将新的标定结果更新到**标定文件路径**处选择的标定文件中。否则，标定文件不更新。

生成标定文件

点击**生成标定文件**按钮可将该模块生成的标定文件另存为。

- 运行参数：

相机模式

需跟进现场实际情况选择，可选相机静止上相机位（相机固定不动且在被测物上方）、相机静止下相机位（相机固定不动且在被测物下方）、相机运动（相机跟随运动机构运动）。

自由度

可设置拟合 homo 矩阵时的自由度。可选**缩放、旋转、纵横比、倾斜、平移及透射**（透视变换），**缩放、旋转、纵横比、倾斜及平移**（仿射变换）和**缩放、旋转及平移**（相似性变换）这三种，自由度逐步降低，推荐使用默认选项。

举例：正方形通过透视变换可能会变成梯形，通过仿射变换可能会变成矩形，通过相似性变换虽还是正方形但大小会发生变化。

权重函数

可选最小二乘、Huber、Tukey 三种算法权重函数，推荐使用默认选项。

选择 Huber 或 Tukey 时，需设置**权重系数**参数。

权重系数

对应选择函数的削波因子，推荐使用默认值。

模块结果

*N 点标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

标定状态

int 型，0 代表标定失败，1 代表标定成功。

评估标定误差状态

int 型，0 代表误差在正常范围内，1 代表不正常的标定结果。

标定图像点

point 型，代表标定图像点。

图像坐标 X

float 型，代表标定时特征点像素坐标 X 数值。

图像坐标 Y

float 型，代表标定时特征点像素坐标 Y 数值。

标定物理点

point 型，代表标定物理点。

转换坐标 X

float 型，代表标定时机构物理坐标 X 数值。

转换坐标 Y

float 型，代表标定时机构物理坐标 Y 数值。

平移像素平均误差

float 型，代表平移像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

平移像素指将图像中的像素点按照给定的偏移量移动。

旋转像素平均误差

float 型，代表旋转像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转轴图像点

point 型，代表旋转轴图像中心。

旋转轴图像坐标 X

float 型，代表旋转中心在图像坐标系中的坐标 X 。

旋转轴图像坐标 Y

float 型，代表旋转中心在图像坐标系中的坐标 Y 。

旋转中心物理点

point 型，代表旋转中心物理点。

旋转中心物理坐标 X

float 型，代表旋转中心在物理坐标系中的坐标 X 。

旋转中心物理坐标 Y

float 型，代表旋转中心在物理坐标系中的坐标 Y 。

角度旋转一致性

int 型，代表机构物理坐标系角度的变化与像素坐标系角度的变化一致，角度同增或同减。

平移像素最大误差

float 型，代表平移像素过程中，测量值与真实值之间的最大差异。

说明

平移像素指将图像中的像素点按照给定的偏移量移动。

平移像素最大误差对应点数

float 型，代表在平移像素过程中，测量值与真实值之间差异最大时，对应点的序号。

平移估计真实误差

float 型，代表平移像素完成后，测量值与真实值之间的差异。

平移像素真实最大误差

float 型，代表平移像素完成后，测量值与真实值之间的最大差异 。

旋转像素最大误差

float 型，代表旋转像素过程中，测量值与真实值之间的最大差异。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转像素最大误差对应点数

int 型，代表旋转像素过程中，测量值与真实值之间最大差异的对应点数。

旋转真实平均误差

float 型，代表旋转像素完成后，测量值与真实值之间的平均差异。

旋转真实最大误差

float 型，代表旋转像素完成后，测量值与真实值之间的最大差异。

尺度

float 型，代表世界坐标系中单位长度对应图像坐标系中的像素数。

x 偏移

float 型，代表图像坐标系原点到机构物理坐标系原点的偏移值 X，单位为像素。

y 偏移

float 型，代表图像坐标系原点到机构物理坐标系原点的偏移值 Y，单位为像素。

旋转

float 型，代表机构物理坐标系相对于图像坐标系的旋转角度（单位为弧度）。

- 当旋转 θ 为正值时，机构物理坐标系 X 轴沿逆时针方向旋转 θ 后，其 X 轴与图像坐标系 X 轴方向一致；
- 当旋转 θ 为负值时，机构物理坐标系 X 轴沿逆时针方向旋转- θ 后，其 X 轴与图像坐标系 X 轴方向一致。

像素精度

float 型，代表单个像素对应物理坐标系下的尺寸。

13.8.2 N 图像标定

N 图像标定模块的作用和用途与 N 点标定模块相同。两个模块的不同之处在于图像坐标系，N 点标定是订阅前序模块输出的图像点，而 N 图像标定是使用海康自研标定板确定图像中像素点的位置偏移。

说明

N 图像标定相较 *N 点标定* 的精度更高，但必须使用海康自研标定板。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块使用的海康自研标定板已提前标注每个角点的坐标，计算相同标注坐标对应的像素位置就能得到图像中相同特征点在不同图像中的图像坐标。再根据接收的运动机构物理坐标信息，就能得到图像坐标系和运动机构物理坐标系的转换关系（即标定文件）。

说明

- 获取到对应的点信息后，计算转换关系的原理与 N 点标定的 *模块原理* 一致。
- 关于海康自研标定板的介绍，参见标定板标定的 *模块原理*，此处不详细展开。

使用方法

该模块使用时需确保前序模块中有图像源模块。需要通过图像源模块获取海康自研标定板的图像。

说明

图像源模块需确保至少有 4 张以上的标定板图像。

模块使用效果如下图所示。

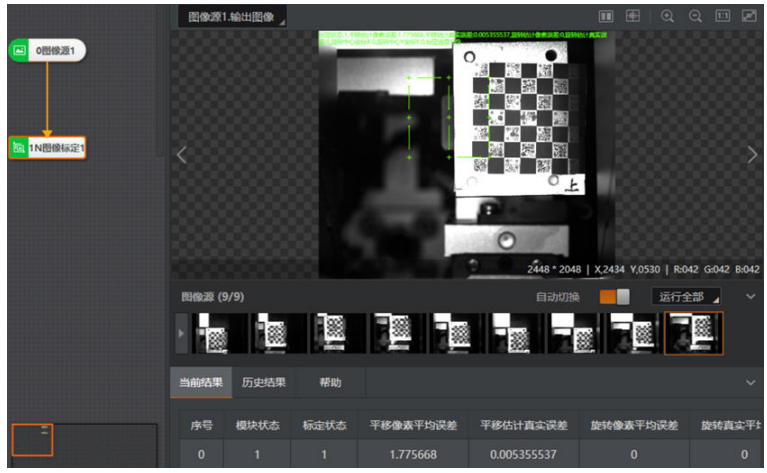


图 13-310 使用效果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

输入源

可设置该模块的图像数据来源。一般直接订阅图像源模块的的图像。

标定点输入

可选按点、按坐标输入。

物理点

物理坐标 X/Y

运动机构坐标点的位置信息。每个图像点对应一个运动机构的物理点。

说明

该参数与物理坐标系参数使用时需二选一，无法同时使用。推荐使用物理坐标系参数自动生成。

旋转角度

运动机构坐标点的角度信息。

说明

该参数与物理坐标系参数使用时需二选一，无法同时使用。推荐使用物理坐标系参数自动生成。

平移次数

设置平移运动的输入点数，只针对 X/Y 方向的平移，一般设置为 9。

旋转次数

设置旋转运动的输入点数。旋转轴与图像中心不共轴时，需设置该参数，一般设置为 3，且旋转是在第 5 个点的位置进行。

物理坐标系参数

此部分实现的功能、涉及的参数以及设置方法均与 N 点标定中的 *物理坐标系参数* 一致，此处不再赘述。

矩阵修正

该模块生成的标定文件中，仅平移标定时，默认将第一张图像中棋盘格左上角角点在图像中的点作为图像坐标系的原点；包含旋转标定时，默认将图像旋转中心作为物理坐标系的原点。

若该对应关系无法满足要求，可开启该参数自行定义图像原点和物理原点的位置。

输入方式

设置**图像点**和**物理点**的获取方式，可选按点、按坐标。

图像点
图像坐标 X/Y

可设置图像坐标系原点的位置信息，一般从前序模块订阅。

物理点
物理坐标 X/Y

可设置运动机构物理坐标原点的位置信息。

说明

每个图像点对应一个运动机构的物理点。

标定文件路径
更新文件
生成标定文件

以上 3 个参数为标定类模块共有参数，功能与操作基本一致，相关介绍参见 *N 点标定中相关参数* 的介绍。

清除图像

可清除已缓存的参与标定的图像。一般需要重新标定用于清除图像。

- 运行参数：

相机移动

相机存在相对运动时，则需要启用该参数。否则，无需开启。会影响旋转一致性。

说明

该参数设置时，需根据现场实际情况选择，否则会影响旋转一致性。

标定板类型

可选海康 I 型标定板、海康 II 型标定板、海康扩展 I 型标定板、海康扩展 II 型标定板。需根据实际使用的标定板类型选择。

中值滤波状态

可设置提取角点前是否执行中值滤波。可选执行滤波和无滤波，推荐使用默认值。

模块结果

*N 图像标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

标定状态

int 型，0 代表标定失败，1 代表标定成功。

评估标定误差状态

int 型，0 代表误差在正常范围内，1 代表不正常的标定结果。

旋转方向

int 型，代表机器臂的旋转方向，可通过调整旋转角度调整现实三维物理坐标系方向。

信息数量

int 型，代表收集、存储、传递的信息数量。

标定图像点

point 型，代表标定图像点。

图像坐标 X

float 型，代表标定时特征点像素坐标 X 数值。

图像坐标 Y

float 型，代表标定时特征点像素坐标 Y 数值。

平移像素平均误差

float 型，代表平移像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

平移像素指将图像中的像素点按照给定的偏移量移动。

平移像素最大误差

float 型，代表平移像素过程中，测量值与真实值之间的最大差异。

说明

平移像素指将图像中的像素点按照给定的偏移量移动。

平移像素最大误差对应点数

float 型，代表在平移像素过程中，测量值与真实值之间差异最大时，对应点的序号。

平移估计真实误差

float 型，代表平移像素完成后，测量值与真实值之间的差异。

平移像素真实最大误差

float 型，代表平移像素完成后，测量值与真实值之间的最大差异 。

旋转像素平均误差

float 型，代表旋转像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转像素最大误差

float 型，代表旋转像素过程中，测量值与真实值之间的最大差异。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转像素最大误差对应点数

int 型，代表旋转像素过程中，测量值与真实值之间最大差异的对应点数。

旋转真实平均误差

float 型，代表旋转像素完成后，测量值与真实值之间的平均差异。

旋转真实最大误差

float 型，代表旋转像素完成后，测量值与真实值之间的最大差异。

机构平移最大误差所在图片索引

int 型，代表进行平移过程中，测量值与真实值之间的最大差异。指出产生最大差异的图片所在位置。

说明

机构平移指机械手按照给定的偏移量移动。

机构平移图像移动量与机构移动量偏差

float 型，代表机构平移过程中，机构平移图像移动量和机构移动量的差异。

机构平移图像尺度变化

float 型，代表机构平移过程中图像的尺度变化。

机构平移图像之间旋转变化量

float 型，代表机构平移过程中，平移图像的旋转变化值。

机构平移最大误差距离误差

int 型，代表机构平移过程中，测量值与真实值之间最大差异的图片距离误差值。

机构平移最大误差图像尺度变化

float 型，代表机构平移过程中，测量值与真实值之间最大差异的图片尺度变化。

机构平移最大误差图像之间旋转变化量

float 型，代表机构平移完成后，测量值与真实值之间最大差异图片之间的旋转变化量。

机构旋转平均误差

float 型，代表机构旋转过程中，计算旋转图像导致计算转化矩阵的平均误差，单位为 mm。

说明

机构旋转指机械手按照给定的偏移量旋转。

机构旋转最大误差

float 型，代表机构旋转过程中，计算旋转图像导致计算转化矩阵的最大误差，单位为 mm。

机构旋转最大误差所在图片索引

float 型，代表机构旋转过程中，测量值与真实值之间最大差异的图片所在位置。

旋转中心坐标

旋转中心坐标 X

float 型，代表通过旋转图像计算出的旋转中心图像坐标 X。

旋转中心坐标 Y

float 型，代表通过旋转图像计算出的旋转中心图像坐标 Y。

x 方向向量

机构坐标系 x 方向 X

float 型，代表标定得到机构物理坐标系 X 轴单位向量的 X 值。

机构坐标系 x 方向 Y

float 型，代表标定得到机构物理坐标系 X 轴单位向量的 Y 值。

y 方向向量

机构坐标系 y 方向 X

float 型，代表标定得到机构物理坐标系 Y 轴单位向量的 X 值。

机构坐标系 y 方向 Y

float 型，代表标定得到机构物理坐标系 Y 轴单位向量的 Y 值。

y 方向和 x 方向移动比

float 型，代表 Y 方向占 X 方向移动比例。

当前图像个数

int 型，代表当前图像是所有图像中的第几个。

图像总个数

int 型，代表图像的总个数。

13.8.3 标定板标定

*标定板标定*模块可根据标定板图像特征点坐标位置和物理坐标位置，计算图像坐标系到物理坐标系的转换关系（即标定文件）。计算得到标定文件后，您可将图像坐标系上的点位置转换到物理坐标系中，从而实现通过测量像素距离测量物理距离等需求。

⚠注意

标定板标定的结果仅可用于转换与标定板处于同一平面的像素。若图像中目标高度不一致，转换后的结果也不准确。

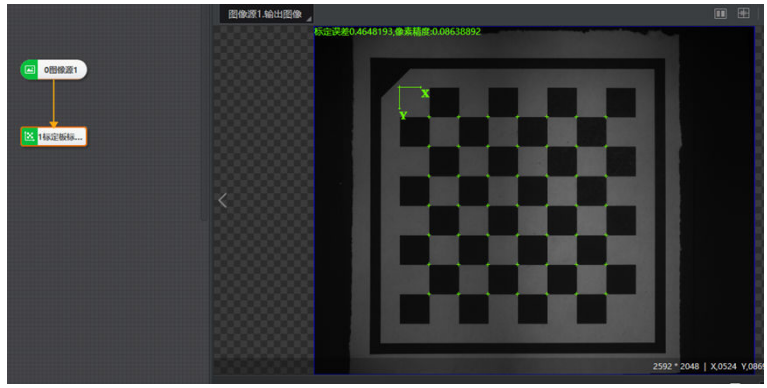


图 13-311 模块运行效果

模块原理

进行标定板标定时，可使用多种不同类型的棋盘格，具体分为棋盘格标定板、圆标定板、海康标定板（包括海康标定板 I 型、海康标定板 II 型、海康扩展标定板 I 型、海康扩展标定板 II 型）。海康标定板的原理为在棋盘格标定板的基础上，部分棋盘格用带位置信息的码替代。其中 I 型标定板为 1 个码占据 4 个棋盘格位置，II 型标定板为码均放置在标定板的白格中。

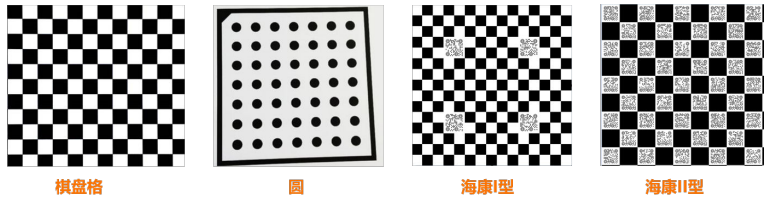


图 13-312 标定板类型

说明

- 上图中的海康 I 型和 II 型标定板为处理后的效果。通过*标定板生成工具*生成可编辑的文件后，您需根据实际场景中的光源填充标定板的颜色。您需确保填充颜色后，在光源照射下，标定板格子
- 海康常规标定板的行数和列数最大支持 255，海康扩展标定板的行数和列数最大支持 1000，其余没差别。
- 海康标定板可通过 VM 自带的标定板生成工具生成。

海康标定板与其他两种标定板相比，由于码已记录角点的物理坐标信息和方向信息，故其自带一个物理坐标系。当该模块选择的标定板类型为海康标定板时，VM 直接使用其自带的物理坐标系。

以上四种类型的标定板其图像坐标系均为：图像左上角为坐标原点，水平向右为 X 轴，垂直向下为 Y 轴。但不同类型标定板的物理坐标系存在差异：

- 棋盘格标定板/圆标定板：以左上角第一个角点为坐标原点，坐标轴与棋盘格的边平行。

说明

当提取到的角点列数大于行数时，坐标系的位置和方向会发生变化，特征点数量较多的方向为 X 轴，如下图所示。

- 海康标定板：由于码记录角点的物理坐标信息和方向信息，因此无论标定板如何放置，都有相同的默认物理坐标系。

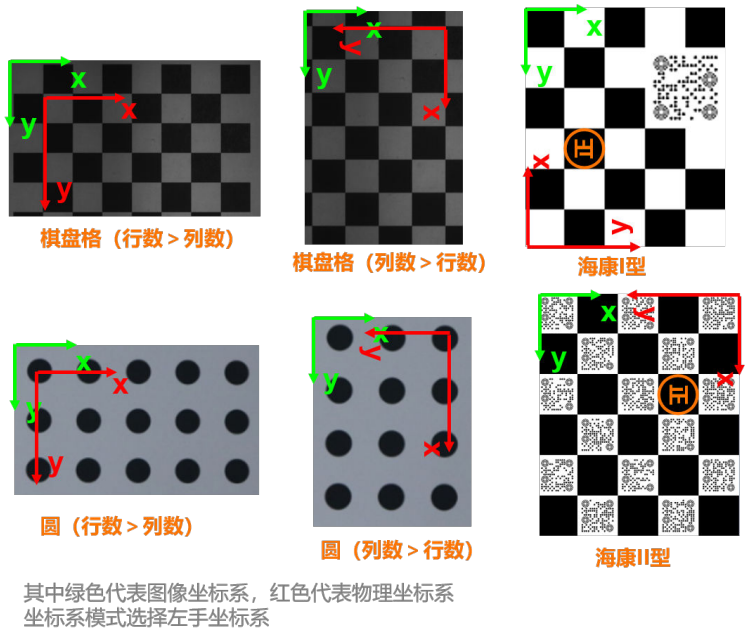


图 13-313 标定板坐标系示意图

不同标定板类型适用的场景有所差异：

- 若仅需使用一个相机，由于此时无需过多关注标定的物理坐标系的原点位置，推荐使用棋盘格标定板或圆标定板。
- 若需要使用多个相机且多个相机需转换到同一个物理坐标系，由于每个相机的视野不同，拍摄的棋盘格区域也有所不同，推荐使用海康标定板。

说明

使用海康标定板时，需解译标定板上的码，此时需保证所拍摄标定板的图像中码足够大且足够清晰（推荐码占据 100 个像素左右），否则可能导致标定失败。

使用方法

该模块主要在测量和定位类项目中使用。

- 测量类项目：可使用该模块的标定结果将图像中的特定点转换到物理坐标系下，然后在物理坐标系进行测量。
- 定位类项目：可借助该模块实现多个相机坐标系的关联。使多个相机标定到同一个物理坐标系下。

该模块使用时，需确保前序模块有 *图像源* 模块。图像源模块用于获取标定板的图片，供该模块标定使用。

调试建议

使用该模块标定时，若反馈标定失败。可按照以下方法操作看能否解决。

1. 确认图像源模块输入的图像是否为标定板图像。
2. 确认**标定板类型**参数的选择是否正确，需和图像源模块输入的标定板图像的类型保持一致。
3. 确认其他参数是否设置合理。例如圆标定板的**圆点类型**是否与实际标定板图像类型一致，**点圆度**参数是否设置的过大，**边缘提取阈值**参数是否设置的不合理。

参数配置

此处仅对该模块基本参数的标定文件以及运行参数进行介绍，基本参数的图像输入和结果显示各模块大同小异，此处不再赘述，具体参见 *基本参数* 和 *结果显示* 。

- 基本参数的标定文件：

  标定文件路径
  更新文件
  生成标定文件

  以上 3 个参数为标定类模块共有参数，功能与操作基本一致，相关介绍参见 *N 点标定中相关参数* 的介绍。

- 运行参数：

  原点 X
  原点 Y

  可设置默认物理坐标系原点进行 X/Y 方向的平移，单位为 mm。该参数数值为负数时，以 X/Y 轴反方向平移；该参数数值为正数时，以 X/Y 轴正方向平移。

  旋转角度

  可设置物理坐标系的旋转方向和角度。该参数数值为负数时，以逆时针方向旋转对应的角度；该参数数值为正数时，以顺时针方向旋转对应的角度。

  坐标系模式

  可设置物理坐标系的类型，可选左手坐标系和右手坐标系。两种坐标系如下图所示。

  说明

  假设下图中棋盘格中各个格子的边长为 1mm，则上述 4 个参数的设置效果如下图所示。

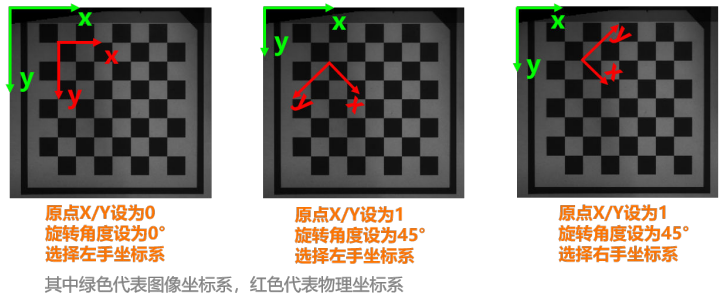


图 13-314 运行参数设置效果

物理尺寸

对应棋盘格各个格子的边长或圆点阵中相邻圆的圆心距，单位为 mm。

说明

该参数需根据标定板上的真实大小填写，否则标定得到的转换关系会失真。

标定板类型

可选棋盘格标定板、圆标定板、海康 I 型标定板、海康 II 型标定板、海康扩展 I 型标定板、海康扩展 II 型标定板。各类型标定板的介绍请参见上文的 ***模块原理*** 。

选择不同类型的标定板，部分需设置的参数有所差异。

选择除圆标定板外的标定板类型时

需设置**中值滤波状态**参数。

中值滤波状态

可设置提取角点前是否执行中值滤波。可选执行滤波和无滤波，推荐使用默认设置。

选择圆标定板时

需设置**点圆度**、**边缘提取阈值**和**圆点类型**参数。

点圆度

可设置圆形度的最小值。识别到的区域圆形度低于设置的数值时，则认为不是有效的圆形区域。

边缘提取阈值

可设置圆边缘两侧的灰度差有效范围。当灰度差不在该范围内时，则认为不是有效圆形区域。

圆点类型

可设置识别的圆点类型，可选白底黑圆和黑底白圆。

自由度

可设置拟合 homo 矩阵时的自由度。可选**缩放、旋转、纵横比、倾斜、平移及透射**（透视变换），**缩放、旋转、纵横比、倾斜及平移**（仿射变换）和**缩放、旋转及平移**（相似性变换）这三种，自由度逐步降低，推荐使用默认选项。

举例：正方形通过透视变换可能会变成梯形，通过仿射变换可能会变成矩形，通过相似性变换虽还是正方形但大小会发生变化。

权重函数

可选最小二乘法、Huber、Tukey 三种算法权重函数，推荐使用默认选项。

选择 Huber 或 Tukey 时，需设置**权重系数**参数。

权重系数

对应选择函数的削波因子，推荐使用默认值。

模块结果

*标定板标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

标定误差

float 型，代表标定过程中，标定测量值与真实值间的差异。 标定误差越大，测量结果越不准确。相机标定允许误差范围在 0.1~0.5 之间。

尺度

float 型，代表世界坐标系中单位长度对应图像坐标系中的像素数。

标定点

point 型，代表标定板角点像素。

标定点 X

float 型，代表标定板角点像素坐标 X。

标定点 Y

float 型，代表标定板角点像素坐标 Y。

标定原点

标定原点 X

float 型，代表标定板左上角第一个角点坐标 X。

标定原点 Y

float 型，代表标定板左上角第一个角点坐标 Y。

坐标 X 向量

坐标 X 向量 X

float 型，代表标定板物理坐标系 X 轴单位向量的 X 值。

坐标 X 向量 Y

float 型，代表标定板物理坐标系 X 轴单位向量的 Y 值。

坐标 Y 向量

坐标 Y 向量 X

float 型，代表标定板物理坐标系 Y 轴单位向量的 X 值。

坐标 Y 向量 Y

float 型，代表标定板物理坐标系 Y 轴单位向量的 Y 值。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

标定点数

int 型，代表所提取标定板上的特征点数量。

平移 X

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 X 。

平移 Y

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 Y 。

旋转

float 型，代表机构物理坐标系相对于图像坐标系的旋转角度（单位为弧度）。

- 当旋转 θ 为正值时，机构物理坐标系 X 轴沿逆时针方向旋转 θ 后，其 X 轴与图像坐标系 X 轴方向一致；
- 当旋转 θ 为负值时，机构物理坐标系 X 轴沿逆时针方向旋转- θ 后，其 X 轴与图像坐标系 X 轴方向一致。

斜切

float 型，代表世界坐标系的 Y 轴旋转角度与 X 轴旋转角度之差（单位为弧度）。

宽高比

float 型，代表世界坐标系的 Y 轴缩放量与 X 轴缩放量的比例。

像素精度

float 型，代表单个像素对应物理坐标系下的尺寸。

13.8.4 相机映射

相机映射模块主要用于计算两个图像坐标系间的映射关系。通过输入一系列一一对应的点计算得出两个图像坐标系的仿射变换矩阵（即标定文件）。该模块在多相机定位引导类场景中应用较为广泛。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

说明

该模块的模块原理与 *映射标定模块的原理* 相同，此处不再赘述。

使用方法

该模块主要配合特征提取类的模块使用。该模块作为后序模块，订阅前序模块提取到的两个图像坐标系下一一对应的点，从而计算仿射变换矩阵。

下图为该模块实际应用的流程示意图，使用的特征提取类模块为边缘交点模块，上相机的边缘交点 1/2/3/4 分别与下相机的边缘交点 5/6/7/8 进行映射。



图 13-315 相机映射使用示意

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数的标定文件：

  输入方式

  可设置**示教点*-运行点***的点信息输入方式，可选按点、按坐标。选择按点时，自定义或订阅的数据为点的坐标；选择按坐标时，自定义或订阅的数据为点 X 和点 Y，从而组成点的坐标。

  示教点*-运行点*

  可设置两个图像坐标系下点与点的映射。至少需设置两组映射关系，也可通过 ⊞ 增加映射关系。

  说明

  设置映射时，需确保每组参数左侧订阅的点属于同一个图像坐标系，右侧订阅的点也需遵循该原则。但左右两侧的点应属于不同的图像坐标系。

  标定文件路径
  更新文件
  生成标定文件

  以上 3 个参数为标定类模块共有参数，功能与操作基本一致，相关介绍参见 *N 点标定中相关参数* 的介绍。

- 运行参数：

  自由度

可设置拟合 homo 矩阵时的自由度。可选**缩放、旋转、纵横比、倾斜、平移及透射**（透视变换），**缩放、旋转、纵横比、倾斜及平移**（仿射变换）和**缩放、旋转及平移**（相似性变换）这三种，自由度逐步降低，推荐使用默认选项。

举例：正方形通过透视变换可能会变成梯形，通过仿射变换可能会变成矩形，通过相似性变换虽还是正方形但大小会发生变化。

权重函数

可选最小二乘法、Huber、Tukey 三种算法权重函数，推荐使用默认选项。

选择 Huber 或 Tukey 时，需设置**权重系数**参数。

权重系数

对应选择函数的削波因子，推荐使用默认值。

模块结果

*相机映射*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

x 方向比例

float 型，代表运行点所在坐标系到示教点所在坐标系的 X 坐标比例。

y 方向比例

float 型，代表运行点所在坐标系到示教点所在坐标系的 Y 坐标比例。

标定状态

int 型，0 代表标定失败，1 代表标定成功。

平移 X

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 X 。

平移 Y

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 Y 。

旋转

float 型，代表机构物理坐标系相对于图像坐标系的旋转角度（单位为弧度）。

- 当旋转 θ 为正值时，机构物理坐标系 X 轴沿逆时针方向旋转 θ 后，其 X 轴与图像坐标系 X 轴方向一致；
- 当旋转 θ 为负值时，机构物理坐标系 X 轴沿逆时针方向旋转-θ 后，其 X 轴与图像坐标系 X 轴方向一致。

尺度

float 型，代表世界坐标系中单位长度对应图像坐标系中的像素数。

斜切

float 型，代表世界坐标系的 Y 轴旋转角度与 X 轴旋转角度之差（单位为弧度）。

宽高比

float 型，代表世界坐标系的 Y 轴缩放量与 X 轴缩放量的比例。

13.8.5 映射标定

映射标定模块的作用和用途与相机映射模块相同。两个模块的不同之处在于获取点集的方式，相机映射是订阅前序特征提取类模块输出的点集，而映射标定是通过海康自研标定版的图像或对应的文件。

说明

关于海康自研标定板的介绍，参见标定板标定的 *模块原理*，此处不详细展开。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

假设有两组点，在坐标系 I_1、I_2 为 P_1^I 、 P_2^I ，在坐标系 W_1、W_2 为 P_1^W 、 P_2^W 。其中坐标系 I_1、I_2 不是同一坐标系（假设为图像坐标系），坐标系 W 是物理坐标系。
则可以分别计算出从 I 坐标系映射到 W 坐标系的关系 H_1、H_2 满足下图所示两个公式。

$$P_1^W = H_1 P_1^I$$

$$P_2^W = H_2 P_2^I$$

由于 P_1^W 与 P_2^W 的坐标系是相同的，对于 I_2 中每一点 P_2^I ，如果将这一点映射到坐标系 W 中，再将其映射到 I_1 中，便可以实现从 I_2 映射到 I_1，对应的公式如下所示，其中 H_1^{-1} 表示 H_1 的逆映射关系。

$$P_{2\to1}^{I} = {H_1}^{-1}H_2P_2^I$$

映射标定就是需要计算最终的映射矩阵 $H_1{}^{-1}H_2$。

使用方法

该模块使用时需确保前序模块中有 2 个图像源模块。需要通过图像源模块分别获取两个相机采集的海康自研标定板图像。

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

输入方式

选择模块所需信息的输入方式，可选图像输入或文件输入。

图像输入

选择图像输入时，需通过前序模块获取带海康自研标定板的图像。

输入源 1/2

分别选择 2 张前序模块输出的图像。

形状

可选择全屏或绘制矩形 ROI，相关操作参见 *绘制 ROI 区域* 。

物理点输入

设置物理点的输入方式，可选按点、按坐标输入。

物理点
物理坐标 X/Y
物理角度

运动机构坐标点的位置及角度信息。每个图像点对应一个运动机构的物理点。

示教

启用后，可通过相关设置判断外部输入的信号是否为示教信号。

外部输入字符

可订阅外部输入的字符，例如外部通信的 TRIGGER STRING。

外部触发字符

判断**外部输入字符**是否为示教信号的依据。当**外部输入字符**与该参数内容一致时，则判断该信号为示教信号。

文件输入

选择文件输入时，需通过前序模块获取海康自研标定板的标定文件。

标定文件 1/2

点击 选择海康自研标定板图像对应的标定文件。

刷新信号

可自定义或订阅数据，作为标定文件是否刷新的信号。

- 参数为 0 或空值时，仅在模块第一次运行时加载选择的标定文件，后续不再更新。
- 参数为非 0 数值且非空值时，每次模块运行时会重新加载选择的标定文件。确保读取的标定文件是最新的。

标定文件路径
更新文件
生成标定文件

以上 3 个参数为标定类模块共有参数，功能与操作基本一致，相关介绍参见 *N 点标定中相关参数* 的介绍。

- 运行参数：

标定板类型

可选海康 I 型标定板、海康 II 型标定板、海康扩展 I 型标定板、海康扩展 II 型标定板。需根据实际使用的标定板类型选择。

自由度

可设置拟合 homo 矩阵时的自由度。可选**缩放、旋转、纵横比、倾斜、平移及透射**（透视变换），**缩放、旋转、纵横比、倾斜及平移**（仿射变换）和**缩放、旋转及平移**（相似性变换）这三种，自由度逐步降低，推荐使用默认选项。

举例：正方形通过透视变换可能会变成梯形，通过仿射变换可能会变成矩形，通过相似性变换虽还是正方形但大小会发生变化。

权重函数

可选最小二乘、Huber、Tukey 三种算法权重函数，推荐使用默认选项。

选择 Huber 或 Tukey 时，需设置**权重系数**参数。

权重系数

对应选择函数的削波因子，推荐使用默认值。

中值滤波状态

需分别对输入源 1 和 2 进行设置。可设置提取角点前是否执行中值滤波。可选执行滤波和无滤波，推荐使用默认值。

模块结果

*映射标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

标定状态

int 型，0 代表标定失败，1 代表标定成功。

X 方向比例

float 型，代表运行点所在坐标系到示教点所在坐标系的 X 坐标比例。

Y 方向比例

float 型，代表运行点所在坐标系到示教点所在坐标系的 Y 坐标比例。

映射误差

float 型，代表进行映射过程中，映射测量值与真实值之间的差异。 映射误差越大，测量结果越不准确 。

当前角点

当前角点 X 坐标

float 型，代表第一幅标定板图像中提取的角点坐标 X 数值。

当前角点 Y 坐标

float 型，代表第一幅标定板图像中提取的角点坐标 Y 数值。

目标角点

目标角点 X 坐标

float 型，代表第二幅标定板图像中提取的角点坐标 X 数值 。

目标角点 Y 坐标

float 型，代表第二幅标定板图像中提取的角点坐标 Y 数值 。

映射角点

映射角点 X 坐标

float 型，代表第一幅图中角点映射到第二幅图像中的 X 坐标。

映射角点 Y 坐标

float 型，代表第一幅图中角点映射到第二幅图像中的 Y 坐标。

当前角点数

int 型，代表当前图像提取角点的个数。

目标角点数

int 型，代表目标图像提取角点的个数。

映射角点数

int 型，代表当前图像提取角点能映射到目标图像角点的数目。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

图像 1/2 标定状态

int 型，0 代表标定失败，1 代表标定成功。

单像素精度 1/2

float 型，代表单个像素对应物理坐标系下的尺寸。

13.8.6 畸变标定

畸变标定模块主要用于对带畸变的标定板图像进行标定，生成相应的畸变校正文件。该模块可应用于定位测量类项目。畸变的主要来源为相机成像时的径向和透视畸变。

说明

实际使用时，通过该模块生成校正文件后，还需使用 ***畸变校正*** 模块对图像进行校正，方可降低畸变所引入的偏差。

本节内容包含：

- *模块原理*
- *使用方法*
- *调试建议*
- *参数配置*
- *模块结果*

模块原理

首先介绍该模块能处理的畸变类型，分为径向畸变、透视畸变以及径向透视畸变三种。

- 径向畸变：以畸变中心（一般为图像中心）为中心点，沿径向方向产生的畸变。一般是由于镜头不同区域的聚光能力不同或其他原因，导致被测物发生扭曲变形。离中心点距离越远，畸变越大。

  说明

  径向畸变校正的目标是使得校正后图像中原本发生扭曲的直线恢复成笔直的状态。

- 透视畸变：即近大远小的畸变。一般是由于被测物与相机之间存在一定倾斜，从而导致图像中被测物边缘夹角与实际相比发生变化的现象。典型现象为棋盘格的格子宽度近大远小。校正后的棋盘格的格子大小完全相同，所有格子宽度以设置的校正点附近格子宽度为基准长度进行校正。

  说明

  透视畸变校正的目标是使得校正后图像中被测物的夹角与实际保持一致。

- 径向透视畸变：顾名思义，既存在径向畸变，又存在透视畸变。

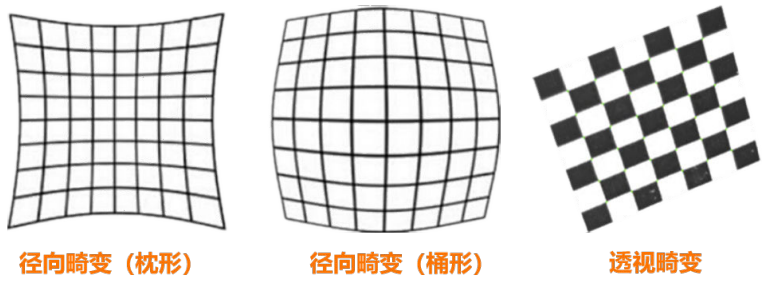


图 13-316 畸变类型

若需要基于发生畸变的图像，得到没有失真的图像以便后续模块（例如拼接、测量类模块）处理，则需要进行畸变标定。该模块算法计算出对应的校正系数后，点击***生成标定文件***可将畸变校正系数以标定文件的方式保存。

说明

其中针对径向透视畸变，会同时得出径向校正系数以及透视校正系数。

畸变标定需借助标定板来完成。标定板可以是棋盘格标定板、圆型标定板或海康自研标定板。

说明

- 关于标定板的介绍，参见标定板标定的 *模块原理*，此处不详细展开。
- 海康自研标定板也可以作为普通棋盘格标定板使用。该模块使用海康自研标定板时，无需译码，查找棋盘格角点即可。

使用方法

该模块主要配合 *图像源* 模块和 *畸变校正* 模块使用。前序模块为图像源模块，使该模块可订阅带畸变的标定板图像；生成的标定文件供畸变校正模块订阅，可对存在相同问题的图像进行校正。

调试建议

使用该模块进行标定时，建议确保以下内容：

- 标定时，建议将标定板占满视野，此时标定效果最好。
- 畸变类型中包含径向畸变时，需要将校正中心点设置到图像中心，此时径向畸变矫正的效果最好。
- 畸变类型中包含透视畸变时，若图像中提取到的特征点行数大于列数，会对图像进行旋转。若不希望旋转，建议拍摄之前保证图像中的特征点行数小于列数。

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示*。

- 基本参数：

  输入源

  选择前序模块输出的带畸变的标定板图像。

  形状

  可选择全屏或绘制矩形 ROI，相关操作参见 *绘制 ROI 区域*。

  标定文件路径
  更新文件
  生成标定文件

  以上 3 个参数为标定类模块共有参数，功能与操作基本一致，相关介绍参见 *N 点标定中相关参数* 的介绍。

- 运行参数：

  校正点输入

  可选按点、按坐标输入。

  校正中心 X/Y
  校正中心点

设置畸变校正的中心点。

说明

畸变类型中包含径向畸变时，需将校正中心点设置到图像中心，此时校正效果最好。

畸变类型

可选透视畸变、径向畸变和径向透视畸变，根据实际情况选择即可。

矫正评估使能

开启该功能后，模块会矫正标定图像。图像上畸变的地方会被尽量处理平直以便降低标定时产生的误差。同时模块会额外输出一张矫正后的图像。

矫正后标定点

开启矫正后，图像上标定点的位置相比原图会有位移。若需要查看矫正后图像所对应每个标定点的位置和误差系数，需开启该参数。

评估误差阈值

每个标定点有一个对应的评估误差，若误差值超过该参数设置的数值，则此标定点在图片上会显示红色，不超过则显示绿色。

说明

在*结果显示*中，可通过**评估误差字号**设置图像上评估误差数值显示的字号，以便标定点过于密集时显示不会杂乱。

标定板类型

可选棋盘格标定板、圆标定板、海康 I 型标定板、海康 II 型标定板、海康扩展 I 型标定板、海康扩展 II 型标定板，根据实际情况选择即可。

自由度

可设置拟合 homo 矩阵时的自由度。可选**缩放、旋转、纵横比、倾斜、平移及透射**（透视变换），**缩放、旋转、纵横比、倾斜及平移**（仿射变换）和**缩放、旋转及平移**（相似性变换）这三种，自由度逐步降低，推荐使用默认选项。

举例：正方形通过透视变换可能会变成梯形，通过仿射变换可能会变成矩形，通过相似性变换虽还是正方形但大小会发生变化。

中值滤波状态

可设置提取角点前是否执行中值滤波。可选执行滤波和无滤波，推荐使用默认值。

权重函数

可选最小二乘、Huber、Tukey 三种算法权重函数，推荐使用默认选项。

选择 Huber 或 Tukey 时，需设置**权重系数**参数。

权重系数

对应选择函数的削波因子，推荐使用默认值。

模块结果

*畸变标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

标定误差

float 型，代表标定过程中，标定测量值与真实值间的差异。 标定误差越大，测量结果越不准确。相机标定允许误差范围在 0.1~0.5 之间。

标定点数

int 型，代表所提取标定板上的特征点数量。

标定点

point 型，代表标定板角点像素。

标定点 X

float 型，代表标定板角点像素坐标 X。

标定点 Y

float 型，代表标定板角点像素坐标 Y。

标定点状态

float 数组型，由“评估误差阈值”和“标定点评估误差”运行参数之间的大小决定。误差大于阈值时取 0，输出图片上对应点显示红色；误差小于等于阈值时取 1，输出图片上对应点显示绿色。

评估误差

float 型，代表总体评估误差。如需查看该参数，需提前开启矫正评估功能。

标定点评估误差

float 型，代表标定点的评估误差。如需查看该参数，需提前开启矫正评估功能。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

矫正后的图像

矫正后的图像数据

image 型，代表开启矫正评估后输出的图像数据。

矫正后的图像宽度

int 型，代表开启矫正评估后输出的图像宽度。

矫正后的图像高度

int 型，代表开启矫正评估后输出的图像高度。

矫正后的图像像素格式

int 型，代表开启矫正评估后输出的图像像素格式。

13.8.7 平移旋转标定

平移旋转标定模块与 N 点标定模块大同小异。

本节内容包含：

- *模块原理&使用方法*
- *参数配置*
- *模块结果*

模块原理&使用方法

该模块的底层标定算法同 ***N 点标定*** 。同时新增组合标定功能，用于小视野旋转标定拟合旋转中心。可通过旋转大角度后再平移回视野范围内提取像素点的方法，推算出大角度旋转后的实际像素点，从而实现小视野大角度类型的旋转标定。
同理，该模块的使用方法与 N 点标定模块也大同小异，此处不再赘述。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见*结果显示*。

标定类型

可选平移标定、平移旋转标定。

选择平移旋转标定时，还需设置以下参数：

旋转次数

设置旋转运动的输入点数。旋转轴与图像中心不共轴时，需设置该参数，一般设置为 3，且旋转是在第 5 点的位置进行。

组合标定

开启该参数后，该模块支持平移并旋转后，再平移回视野的标定方式。该方式对平移精度要求较高。

若小视野场景中机构只能小角度旋转，从而导致旋转中心拟合不准确。此时可启用该参数控制机构旋转较大角度再平移回视野的方式进行旋转中心标定。

标定点获取

可选触发获取、手动输入。一般选择触发获取。选择手动输入时，该模块可单独运行，即手动输入坐标点生成标定矩阵。

选择触发获取时，还需设置以下参数：

标定点输入
图像坐标 X/Y
图像点
图像角度
物理坐标 X/Y
物理点
物理角度
示教

以上参数与 N 点标定模块相同，功能与操作一致，相关介绍参见 *N 点标定中相关参数* 的介绍。

相机模式

可选相机运动、相机静止，根据实际情况选择即可。

自由度

可设置拟合 homo 矩阵时的自由度。可选**缩放、旋转、纵横比、倾斜、平移及透射**（透视变换），**缩放、旋转、纵横比、倾斜及平移**（仿射变换）和**缩放、旋转及平移**（相似性变换）这三种，自由度逐步降低，推荐使用默认选项。

举例：正方形通过透视变换可能会变成梯形，通过仿射变换可能会变成矩形，通过相似性变换虽还是正方形但大小会发生变化。

平移次数

设置平移运动的输入点数，只针对 X/Y 方向的平移，一般设置为 9。

标定文件路径
更新文件
生成标定文件

以上 3 个参数为标定类模块共有参数，功能与操作基本一致，相关介绍参见 ***N 点标定中相关参数*** 的介绍。

清空标定点

可将标定点坐标全部重置为 0。

模块结果

*平移旋转标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

标定状态

int 型，0 代表标定失败，1 代表标定成功。

坐标系左右手一致性

int 型，1 代表机构物理坐标系和像素坐标系手性一致，-1 代表机构物理坐标系和像素坐标系手性不一致。像素坐标系手性为左手系。

标定图像点

point 型，代表标定图像点。

图像坐标 X

float 型，代表标定时特征点像素坐标 X 数值。

图像坐标 Y

float 型，代表标定时特征点像素坐标 Y 数值。

标定物理点

point 型，代表标定物理点。

转换坐标 X

float 型，代表标定时机构物理坐标 X 数值。

转换坐标 Y

float 型，代表标定时机构物理坐标 Y 数值。

平移像素平均误差

float 型，代表平移像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

平移像素指将图像中的像素点按照给定的偏移量移动。

旋转像素平均误差

float 型，代表旋转像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转轴图像点

point 型，代表旋转轴图像中心。

旋转轴图像坐标 X

float 型，代表旋转中心在图像坐标系中的坐标 X 。

旋转轴图像坐标 Y

float 型，代表旋转中心在图像坐标系中的坐标 Y 。

旋转中心物理点

point 型，代表旋转中心物理点。

旋转中心物理坐标 X

float 型，代表旋转中心在物理坐标系中的坐标 X 。

旋转中心物理坐标 Y

float 型，代表旋转中心在物理坐标系中的坐标 Y 。

角度旋转一致性

int 型，代表机构物理坐标系角度的变化与像素坐标系角度的变化一致，角度同增或同减。

平移像素最大误差

float 型，代表平移像素过程中，测量值与真实值之间的最大差异。

说明

平移像素指将图像中的像素点按照给定的偏移量移动。

平移像素最大误差对应点数

float 型，代表在平移像素过程中，测量值与真实值之间差异最大时，对应点的序号。

平移估计真实误差

float 型，代表平移像素完成后，测量值与真实值之间的差异。

平移像素真实最大误差

float 型，代表平移像素完成后，测量值与真实值之间的最大差异 。

旋转像素最大误差

float 型，代表旋转像素过程中，测量值与真实值之间的最大差异。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转像素最大误差对应点数

int 型，代表旋转像素过程中，测量值与真实值之间最大差异的对应点数。

旋转真实平均误差

float 型，代表旋转像素完成后，测量值与真实值之间的平均差异。

旋转真实最大误差

float 型，代表旋转像素完成后，测量值与真实值之间的最大差异。

尺度

float 型，代表世界坐标系中单位长度对应图像坐标系中的像素数。

x 偏移

float 型，代表图像坐标系原点到机构物理坐标系原点的偏移值 X，单位为像素。

y 偏移

float 型，代表图像坐标系原点到机构物理坐标系原点的偏移值 Y，单位为像素。

旋转

float 型，代表机构物理坐标系相对于图像坐标系的旋转角度（单位为弧度）。

- 当旋转 θ 为正值时，机构物理坐标系 X 轴沿逆时针方向旋转 θ 后，其 X 轴与图像坐标系 X 轴方向一致；
- 当旋转 θ 为负值时，机构物理坐标系 X 轴沿逆时针方向旋转- θ 后，其 X 轴与图像坐标系 X 轴方向一致。

像素精度

float 型，代表单个像素对应物理坐标系下的尺寸。

13.8.8 旋转标定

旋转标定模块实现的功能即平移旋转标定模块中旋转部分的功能。通常在单相机与运动机构的旋转中心标定场景使用。

说明

使用该模块时，需确保运动是纯旋转且无平移，每次旋转角度需严格相等，至少旋转 3 次。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的原理与 *平移旋转标定* 中旋转标定部分相同。
平移旋转标定模块的**标定类型**支持选择平移标定或平移旋转标定，对应生成的标定文件为平移的标定文件或平移旋转的标定文件，无法生成只有旋转信息的标定文件。故如需使用只有旋转的标定文件时，可考虑通过该模块生成。

使用方法

该模块使用时，需与平移旋转标定模块配合使用。此时平移旋转标定模块的**标定类型**通常选择平移标定。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

输入源

选择前序模块输出的需标定图像。

输入方式

可选按点、按坐标输入方式明确图像点的来源。

图像坐标 X/Y
图像点

图像标定点的位置信息，一般从前序模块订阅。

物理旋转角度

运动机构每次旋转的物理角度。

旋转次数

设置旋转运动的输入点数。旋转轴与图像中心不共轴时，需设置该参数，一般设置为 3，且旋转是在第 5 个点的位置进行。

反向纠正

旋转半径很大或物理旋转角度很小时，若拟合得到的旋转中心坐标与实际偏差较大，可启用该参数重新进行旋转标定。

加载标定文件

点击 可自定义选择或新建标定文件，也可通过订阅的方式加载已有的标定文件或新建标定文件。

说明

仅支持 xml 和 iwcal 格式的标定文件。

刷新信号

可自定义或订阅数据，作为标定文件是否刷新的信号。

- 参数为 0 或空值时，仅在模块第一次运行时加载选择的标定文件，后续不再更新。
- 参数为非 0 数值且非空值时，每次模块运行时会重新加载选择的标定文件。确保读取的标定文件是最新的。

模块结果

*旋转标定*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

剩余标定次数

int 型，代表标定流程距离标定完成剩余的标定次数。

旋转轴图像点

point 型，代表旋转轴图像中心。

旋转轴图像坐标 X

float 型，代表旋转中心在图像坐标系中的坐标 X 。

旋转轴图像坐标 Y

float 型，代表旋转中心在图像坐标系中的坐标 Y 。

旋转中心物理点

point 型，代表旋转中心物理点。

旋转中心物理坐标 X

float 型，代表旋转中心在物理坐标系中的坐标 X 。

旋转中心物理坐标 Y

float 型，代表旋转中心在物理坐标系中的坐标 Y 。

旋转像素平均误差

float 型，代表旋转像素过程中，测量值与真实值之间的平均差异。 误差越大，测量结果越不准确。

说明

旋转像素指将图像中的像素点按照给定的偏移量旋转。

旋转真实平均误差

float 型，代表旋转像素完成后，测量值与真实值之间的平均差异。

13.8.9 坐标系

*坐标系*模块可在图像坐标系中自定义创建一个坐标系。并通过坐标矩阵建立图像坐标系和自定义坐标系之间的关系。

以下图为例，点 P 在图像坐标系 XY 下的坐标为(x,y)，通过该模块可计算得到点 P 在自定义绘制的坐标系 X'Y'下对应的坐标(m,n)。

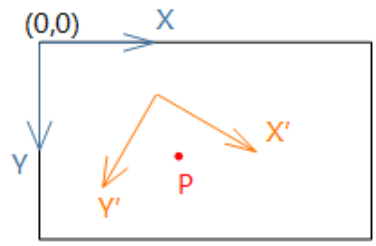


图 13-317 示意图

本节内容包含：

- *使用方法*
- *参数设置*
- *模块结果*

使用方法

您可通过该模块在采集图像上绘制自定义坐标系，或订阅标定矩阵供后序模块使用。两种方式的前序模块及使用方法有所差异。

- 绘制自定义坐标系时，前序模块可为 *图像源* 模块，用于采集图像。后序模块一般为测量类模块，如 *线线测量* 。测量类模块的运行参数按照下图右侧进行配置，可得到自定义坐标系下直线交点的坐标信息。



图 13-318 示例方案

- 订阅标定矩阵时，前序模块需为 *标定加载* 等包含标定矩阵数据的模块，可直接订阅前序模块的标定矩阵，供后序模块使用（例如数据由图像坐标系转换为物理坐标系下的数据）。

参数设置

该模块参数分为**图像输入**和**绘制**两部分。**图像输入**的介绍参见 *模块通用配置中图像输入相关介绍* 。此处主要就**绘制**相关参数进行介绍。

说明

无论选择哪种坐标系绘制方式，支持的自由度均为平移、旋转和缩放。

坐标系创建

可选**绘制**或**订阅**两种方式。

绘制

单击 进行绘制，相关介绍参见 *模块检测的 ROI 区域* 。

订阅

通过订阅前序模块的相关数据，定义坐标系。选择不同的**输入方式**，需设置的参数有所差别。

按点

通过**点**和**角度**订阅坐标系。

按坐标

通过**点 X**、**点 Y** 和**角度**订阅坐标系。

按矩阵

通过**矩阵**直接订阅前序模块的坐标矩阵。

模块结果

*坐标系*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

坐标系矩阵

float 型，代表相对于图像坐标系的坐标系矩阵。

13.8.10 标定加载

标定加载模块可加载部分模块生成的标定文件（xml 格式），并输出相关信息。

说明

当前仅 N 点标定、N 图像标定和映射标定模块生成的标定文件，可通过该模块加载使用。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块通过加载标定文件实现读取并输出标定文件里的标定矩阵、旋转中心、标定误差和像素精度等信息，也可手动输入图像基准坐标、拍照基准坐标、示教坐标等信息供后续订阅。该模块主要用于排查标定问题，也可将加载的标定矩阵输出给计算类模块使用。一般单独使用，用于查看标定文件数据；也可配合 ***N 点标定*** 、 ***平移旋转标定*** 等标定模块生成的标定文件使用。

参数配置

该模块参数主要分为以下几类：

标定文件

需设置**标定文件路径**、**刷新信号**和**方案存矩阵**参数。

标定文件路径

点击 选择需加载的标定文件。

说明

仅支持加载 xml 格式的文件。

刷新信号

可自定义或订阅数据，作为标定文件是否刷新的信号。

- 参数为 0 或空值时，仅在模块第一次运行时加载选择的标定文件，后续不再更新。
- 参数为非 0 数值且非空值时，每次模块运行时会重新加载选择的标定文件。确保读取的标定文件是最新的。

方案存矩阵

启用该参数后，可将标定文件数据存入方案中。即使方案未加载标定文件，也能正常运行。

图像基准坐标

通过 **X**、**Y** 和 **R** 分别设置图像基准坐标系的原点和角度，供后续模块订阅。可自定义设置，也可从前序模块订阅。

拍照基准坐标

通过 **X**、**Y** 和 **R** 分别设置拍照基准坐标系的原点和角度，供后续模块订阅。可自定义设置，也可从前序模块订阅。

示教坐标

通过 **X**、**Y** 和 **R** 分别设置示教坐标系的原点和角度，供后续模块订阅。可自定义设置，也可从前序模块订阅。

标定信息

可查看所加载标定文件的相关信息，如下图所示。

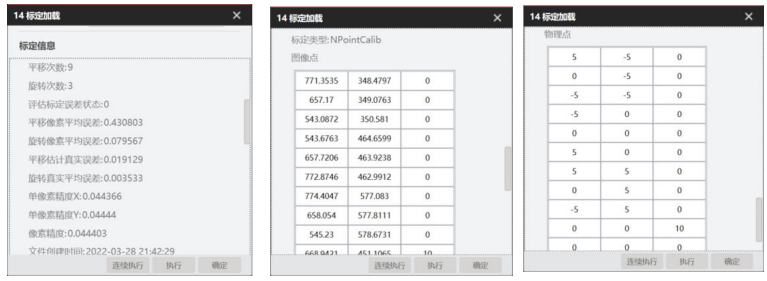


图 13-319 标定信息

模块结果

*标定加载*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

角度旋转一致性

int 型，代表机构物理坐标系角度的变化与像素坐标系角度的变化一致，角度同增或同减。

坐标系左右手一致性

int 型，1 代表机构物理坐标系和像素坐标系手性一致，-1 代表机构物理坐标系和像素坐标系手性不一致。像素坐标系手性为左手系。

旋转中心图像坐标

旋转中心图像坐标点

旋转中心图像坐标 X

float 型，代表旋转中心在图像坐标系中坐标 X。

旋转中心图像坐标 Y

float 型，代表旋转中心在图像坐标系中坐标 Y。

旋转中心图像坐标 R

float 型，代表旋转中心在图像坐标系中角度 R。

旋转中心物理坐标

旋转中心物理坐标点

旋转中心物理坐标 X

float 型，代表旋转中心在物理坐标系中坐标 X。

旋转中心物理坐标 Y

float 型，代表旋转中心在物理坐标系中坐标 Y。

旋转中心物理坐标 R

float 型，代表旋转中心在物理坐标系中角度 R。

图像基准坐标

图像基准坐标点

图像基准坐标 X

float 型，代表目标在基准位姿时特征点的像素坐标 X。

图像基准坐标 Y

float 型，代表目标在基准位姿时特征点的像素坐标 Y。

图像基准坐标 R

float 型，代表目标在基准位姿时特征边角度 R。

拍照基准坐标

拍照基准坐标点

拍照基准坐标 X

float 型，代表通常为运动机构 N 点标定的第 5 点（如果点 位为 9 或 12 点情况）坐标 X。

拍照基准坐标 Y

float 型，代表通常为运动机构 N 点标定的第 5 点（如果点 位为 9 或 12 点情况）坐标 Y。

拍照基准坐标 R

float 型，代表通常为运动机构 N 点标定的第 5 点（如果点位为 9 或 12 点情况）角度 R。

示教坐标

示教坐标点

示教坐标 X

float 型，代表目标在基准位姿时运动机构去抓取或贴合时的坐标 X。

示教坐标 Y

float 型，代表目标在基准位姿时运动机构去抓取或贴合时的坐标 Y。

示教坐标 R

float 型，代表目标在基准位姿时运动机构去抓取或贴合时的角度 R。

目标拍照坐标

目标拍照坐标点

目标拍照坐标 X

float 型，代表映射标定时运动机构在目标拍照位的坐标 X。

目标拍照坐标 Y

float 型，代表映射标定时运动机构在目标拍照位的坐标 Y。

目标拍照坐标 R

float 型，代表映射标定时运动机构在目标拍照位的角度 R。

对象拍照坐标

对象拍照坐标点

对象拍照坐标 X

float 型，代表映射标定时运动机构在对象拍照位的坐标 X。

对象拍照坐标 Y

float 型，代表映射标定时运动机构在对象拍照位的坐标 Y。

对象拍照坐标 R

float 型，代表映射标定时运动机构在对象拍照位的角度 R。

标定矩阵

float 型，代表图像坐标系与物理坐标系的转换关系。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.9 运算

“运算”分类下的模块通过对位运算、转换运算、变量计算等运算方式达到最终需求。

13.9.1 单点对位

该模块可根据输入的目标点坐标和角度、以及对象点的坐标和角度，计算出由对象点对位至目标点所需的偏移量。偏移量包括 X 和 Y 方向的位移以及角度。

说明

- 该模块输入的为目标点和对象点的物理坐标，需配合 *标定转换* 模块使用。
- 模块中的示教点即目标点，运行点即对象点。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的目的是将两个点进行对位，并得到偏移量。

每个物理坐标点均由 x，y 以及 θ 组成。假设有 2 个点 $P_1(x_1,y_1,\theta_1)$和 $P_2(x_2,y_2,\theta_2)$，若需将点 P_2（即运行点）对位至点 P_1（即示教点），则需先将点进行一定角度 Δθ 的旋转，再将旋转后的点按照(Δx，Δy)进行平移即可。其中 Δθ、Δx、Δy 的计算公式如下图所示。

$$\Delta\theta = \theta_1 - \theta_2$$

$$\Delta x = x_1 - (\cos(\Delta\theta) * x_2 - \sin(\Delta\theta) * y_2)$$

$$\Delta y = y_1 - (\sin(\Delta\theta) * x_2 + \cos(\Delta\theta) * y_2)$$

说明

上图中 Δ θ 公式中的角度会被归一化至-π～π 的范围内。

使用方法

该模块需配合 *图像源* 模块、 *定位类模块* 和 *标定转换* 模块使用，如下图所示。

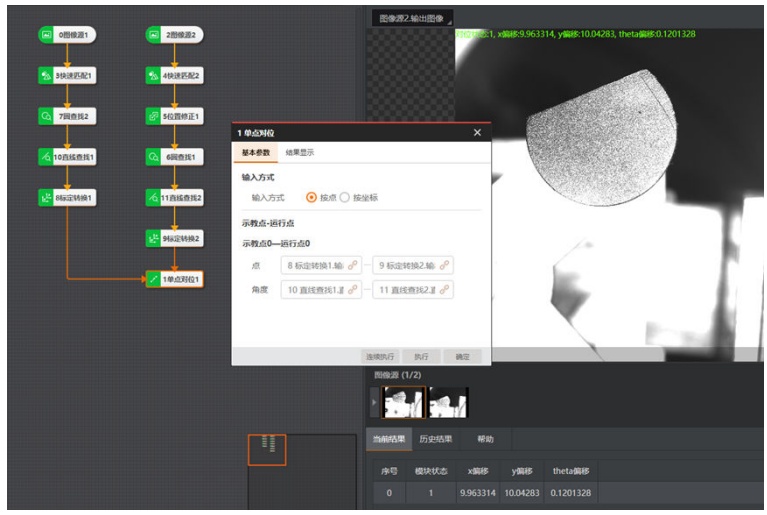


图 13-320 使用示意

上图中的图像源模块用于获取图像，定位类模块用于获取图像坐标系中的特征点及角度信息，标定转换模块用于将图像坐标点转换为物理坐标点，单点对位模块用于在物理坐标系中计算得到运行点到示教点的偏移量。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

输入方式

可选按点、按坐标输入。

示教点 0-运行点 0

设置运行点对位至示教点的点坐标及角度，可自定义输入或订阅前序模块对应的数据。一般从前序模块订阅。

点

点 X/Y

设置点坐标的对位关系。

角度

设置角度的对位关系。

模块结果

*单点对位*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

x 偏移

float 型，代表示教点到运行点 X 方向的位置移动量。

y 偏移

float 型，代表示教点到运行点 Y 方向的位置移动量。

theta 偏移

float 型，代表示教点到运行点的角度偏移量，顺时针为正。

13.9.2 单点抓取

该模块在单相机拍物料的抓取场景中使用，通过该模块可得到基准位移动到运行位的相对偏移量和绝对抓取坐标。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块需与 *平移旋转标定* 模块配合使用，相机和运动机构通过标定得到坐标系转换关系（即标定矩阵 M）。单点抓取再根据订阅的标定矩阵将图像特征点转换到运动机构的物理坐标系下，从而推算出抓取坐标，用于引导运动机构抓取物料。

- 当末端夹具与旋转轴共轴时，无需考虑角度偏移。此时将运动机构在示教拍照位 P1 拍摄的特征点像素坐标定义为 p1，再将运动机构移动至示教物理位 P2，使得末端夹具刚好对准特征点中心。实际生产时，将运动机构运动至示教拍照位 P1 时拍摄的特征点像素坐标定义为 p2。
  先将像素坐标 p1、p2 从图像坐标系转换到运动机构的物理坐标系下，再计算转换后坐标的差值（即相对偏移量）即可。其中相对偏移量与示教物理位之和，就是绝对抓取坐标。
- 当末端夹具与旋转轴不共轴时，由于目标物料会发生角度偏移，运动机构同步旋转，此时会导致末端夹具的 X/Y 坐标发生偏移。为补偿不共轴旋转引起的 x/y 偏移，需引入示教拍照位和示教物理位，计算出不共轴的长度，进而计算出这段不共轴引起的旋转偏差。

因此，当标定类型选择平移旋转标定时，示教拍照位必须填写，模块输出的偏差包含平移偏差和不共轴旋转产生的偏差。使用到的旋转计算公式如下图所示。

$$x = (X - CenterX)\cos(\alpha) - (X - CenterX)\sin(\alpha)$$
$$y = (Y - CenterY)\sin(\alpha) + (Y - CenterY)\cos(\alpha)$$

使用方法

该模块还需与 *图像源* 模块和 *定位类* 模块配合使用。图像源模块用于获取图像，定位类模块用于获取图像特征点。单点抓取模块订阅图像特征点、示教物理点、示教拍照位及标定文件，从而计算出相对偏移量和绝对抓取坐标。如下图所示。



图 13-321 使用示意

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

输入方式

可对**标定类型**和**输入方式**参数进行设置。

标定类型

可选平移标定、平移旋转标定，根据实际需求选择。

输入方式

可选按点、按坐标输入。

像素点

通过**点**（或**坐标 X/Y**）及**角度**设置图像的基准点和运行点。完成像素点订阅后，点击***执行***再点击***创建基准***，若提示“基准点创建成功”，则完成基准点的设置。后续订阅的像素点均为图像运行点。

示教物理点

通过**点**（或**坐标 X/Y**）及**角度**设置物料在基准位置时运动机构抓取物料的绝对物理坐标及角度。

示教拍照物理点

通过**点**（或**坐标 X/Y**）及**角度**设置物料在基准位置时，运动机构携带相机拍照的绝对物理坐标及角度，一般为第 5 个点拍照位的物理坐标。

若生产时，相机相对标定时角度有所变化，还需开启**旋转拍照使能**并设置**旋转相对角度**。角度可根据实际情况自定义填写或从外部通信设备订阅。

说明

标定类型选择平移旋转标定时，必须设置；否则，可不设置。

标定文件

可对加载及刷新标定文件进行相关设置。

加载方式

可设置标定文件的加载方式，可选标定矩阵或加载标定文件。

标定矩阵

通过**标定矩阵**进行订阅。

加载标定文件

点击 可选择需加载的标定文件。

刷新信号

可自定义或订阅数据，作为标定文件是否刷新的信号。

- 参数为 0 或空值时，仅在模块第一次运行时加载选择的标定文件，后续不再更新。
- 参数为非 0 数值且非空值时，每次模块运行时会重新加载选择的标定文件。确保读取的标定文件是最新的。

模块结果

*单点抓取*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

基准点

point 型，代表图像提取到的特征点。

基准点 X

float 型，代表图像提取到的特征点 X。

基准点 Y

float 型，代表图像提取到的特征点 Y。

基准点 R

float 型，代表图像提取到的特征点角度。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

相对坐标 X

float 型，代表机械臂抓取物料的相对基准位置变化的偏移量 X 坐标。

相对坐标 Y

float 型，代表机械臂抓取物料的相对基准位置变化的偏移量 Y 坐标。

相对坐标 R

float 型，代表机械臂抓取物料的相对坐标角度偏移量。

绝对坐标 X

float 型，代表机构抓取的绝对物理位置 X。

绝对坐标 Y

float 型，代表机构抓取的绝对物理位置 Y。

绝对坐标 R

float 型，代表机构抓取的绝对物理位置角度。

13.9.3 单点映射对位

该模块主要用于上相机拍目标、下相机拍对象的对位贴合场景。其作用是通过下相机拍物料、上相机拍料盘，再根据输入对象点的位置和角度、以及目标点的位置和角度，计算出对象点对位贴合到目标点需要的移动量，包括位置移动量和角度移动量。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

单点映射对位模块需与 *平移旋转标定* 模块和 *映射标定* 模块配合使用，该模块适用于上下两个相机做对位的场景。静止下相机和运动机构通过标定得到下相机图像坐标系和运动机构物理坐标系的转换关系（即标定矩阵 M），上下两个相机通过映射标定得到两个图像坐标系之间

的转换关系。单点映射对位再根据两个标定矩阵和输入的特征点位信息将图像特征点转换到运动机构的物理坐标系下，从而推算对位坐标，用于引导运动机构进行对位贴合。
运动机构携带基准物料到静止下相机的示教物理点 P1，此时下相机拍摄特征点像素坐标 p1 和对位直线角度；而上相机回到拍照位 P2 拍摄需要贴合的对象，提取对象点及对位直线的起点和终点。
将上相机对象点及直线起点和终点通过映射标定文件映射到下相机的图像坐标系内，再通过下相机和运动机构的平移旋转标定矩阵将坐标转换到机构坐标系下，从而计算出对位偏差。

使用方法

该模块还需与 *图像源* 模块和 *定位类* 模块配合使用。图像源模块用于获取图像，定位类模块用于获取图像特征点和角度信息。单点映射对位模块订阅对象特征点及角度、目标特征点及直线的起点和终点、示教物理点及标定文件，从而计算出对象到目标的相对偏移量和绝对抓取坐标。如下图所示。

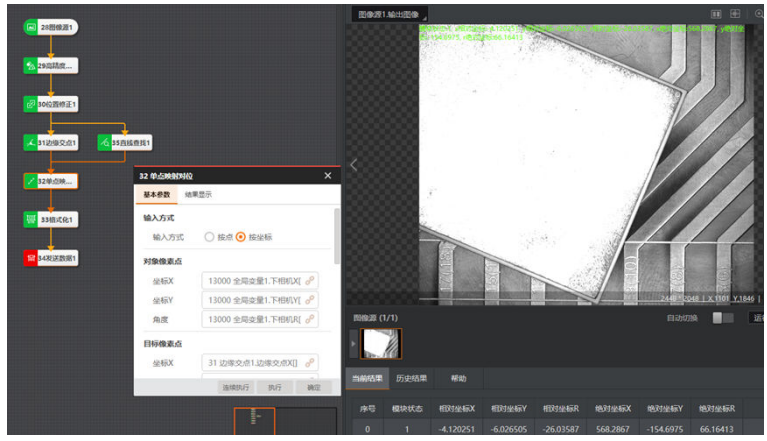


图 13-322 使用示意

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

输入方式

可选按点、按坐标输入。

对象像素点

通过**点**（或**坐标 X/Y**）及**角度**设置对象位（一般指下相机拍照位）的图像像素点及角度。

目标像素点

通过**点**（或**坐标 X/Y**）设置目标位（一般指上相机拍照位）的图像像素点。

目标像素直线起点
目标像素直线终点

通过**点**（或**坐标 X/Y**）设置目标位对位直线的起点/终点。映射到下相机坐标系，确保角度统一。

示教物理点

通过**点**（或**坐标 X/Y**）及**角度**设置相机在做上下相机映射标定时，运动机构抓取标定板时的绝对物理坐标。

标定文件

可对加载及刷新标定文件进行相关设置。

加载方式

可设置标定文件的加载方式，可选标定矩阵或加载标定文件。

N 点标定矩阵
映射标定矩阵

可分别通过订阅 N 点标定或映射标定的矩阵。

N 点标定文件
映射标定文件

点击 可分别选择需加载的标定文件。

刷新信号

可自定义或订阅数据，作为标定文件是否刷新的信号。

- 参数为 0 或空值时，仅在模块第一次运行时加载选择的标定文件，后续不再更新。
- 参数为非 0 数值且非空值时，每次模块运行时会重新加载选择的标定文件。确保读取的标定文件是最新的。

模块结果

*单点映射对位*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

相对坐标 X

float 型，代表机械臂抓取物料的相对基准位置变化的偏移量 X 坐标。

相对坐标 Y

float 型，代表机械臂抓取物料的相对基准位置变化的偏移量 Y 坐标。

相对坐标 R

float 型，代表机械臂抓取物料的相对坐标角度偏移量。

绝对坐标 X

float 型，代表机构抓取的绝对物理位置 X。

绝对坐标 Y

float 型，代表机构抓取的绝对物理位置 Y。

绝对坐标 R

float 型，代表机构抓取的绝对物理位置角度。

13.9.4 单点纠偏

该模块适用于单相机拍物料的纠偏场景。其作用是相机拍物料，再根据输入运行点位置和角度、及基准点位置和角度，计算出运行点移动回基准点需要的移动量，包括位置移动量和角度移动量。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

单点纠偏模块需与 *平移旋转标定* 模块配合使用，相机和运动机构通过标定得到坐标系转化关系（即标定矩阵 M）。单点纠偏可根据标定矩阵将图像特征点转换到运动机构物理坐标系下，从而推算纠偏坐标，用于引导运动机构修正物料姿态。
运动机构携带基准物料到示教物理点 P1，静止相机拍摄特征点像素坐标 p1 创建基准。
生产时，运动机构携带需要纠偏的物料运动到示教拍照位 P1，此时相机拍摄的特征点像素坐标为 p2。将像素坐标 p1、p2 先从图像坐标系转换到运动机构的物理坐标系下，再根据修正角度后的物理坐标计算得出差值，即相对偏移量。相对偏移量与示教物理位之和，就是绝对纠偏坐标。

使用方法

该模块还需与 *图像源* 模块和 *定位类* 模块配合使用。图像源模块用于获取图像，定位类模块用于获取图像特征点及角度信息。单点纠偏模块订阅图像特征点、示教物理点及标定文件，从而计算出相对偏移量和绝对纠偏坐标。如下图所示。



图 13-323 使用示意

参数配置

- 基本参数：与单点抓取模块的参数（除**示教拍照物理点**相关参数）基本一致，此处不再赘述，具体参见 *单点抓取的参数配置* 。
- 结果显示：各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

模块结果

*单点纠偏*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

基准点

point 型，代表图像提取到的特征点。

基准点 X

float 型，代表图像提取到的特征点 X。

基准点 Y

float 型，代表图像提取到的特征点 Y。

基准点 R

float 型，代表图像提取到的特征点角度。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

相对坐标 X

float 型，代表机械臂抓取物料的相对基准位置变化的偏移量 X 坐标。

相对坐标 Y

float 型，代表机械臂抓取物料的相对基准位置变化的偏移量 Y 坐标。

相对坐标 R

float 型，代表机械臂抓取物料的相对坐标角度偏移量。

绝对坐标 X

float 型，代表机构抓取的绝对物理位置 X。

绝对坐标 Y

float 型，代表机构抓取的绝对物理位置 Y。

绝对坐标 R

float 型，代表机构抓取的绝对物理位置角度。

13.9.5 标定转换

该模块可将同一组坐标系下的特征点信息通过加载标定文件转换为另一组坐标系下的特征点。常用于图像坐标系的像素点和物理坐标系的物理点之间进行转换。

输入一系列特征点信息和标定文件后，可以得到每个点转换之后的坐标和角度。输出信息中的**转换坐标 X/Y** 和**转换角度**是转换后的结果，而**单像素精度**、**平移 X/Y**、**旋转**、**尺度**、**斜切**、**宽高比**为标定文件固有的信息属性。

当前结果 | 历史结果 | 帮助

序号	转换坐标X	转换坐标Y	转换角度	单像素精度	平移X	平移Y	旋转	尺度	斜切	宽高比
0	0.002	0.005	0.001	0.017	52.250	69.076	0.000	58.808	0.000	1.000
1	0.998	0.002	0.001	0.017	52.250	69.076	0.000	58.808	0.000	1.000
2	2.002	0.004	0.001	0.017	52.250	69.076	0.000	58.808	0.000	1.000

图 13-324 输出信息

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

对于一组坐标，通过转换关系将这组坐标映射到另一个位置，对应的公式为 $P^w=HP^l$。
当公式中的 P^l 代表图像坐标，而 P^w 代表物理坐标，H 表示转换关系，则通过该转换关系可以从图像坐标转换到物理坐标。

根据上述公式，同理可得 $P^{l}=H^{-1}P^{w}$。其中 H^{-1} 与 H 为相反的转换关系，通过该转换关系可从物理坐标转换到图像坐标。
输出参数中的单像素精度即 H 中的**尺度**，而平移 X/Y、旋转、尺度、斜切和宽高比即 H^{-1} 中的参数分量。

使用方法

该模块一般配合 ***图像源*** 模块和 ***定位类*** 模块使用。图像源模块用于获取图像，定位类模块用于获取图像特征点。标准转换根据输入的图像特征点和标定文件即可获得转换后的物理坐标点。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

图像输入

通过**输入源**订阅图像。

坐标点输入

需设置坐标点的相关信息，并选择坐标类型。

输入方式

可选按点、按坐标输入。

坐标点
坐标 X/Y

需转换的特征点坐标信息，可从前前序模块订阅，也可自定义。

角度

需转换的特征点角度。

坐标类型

可选图像坐标或物理坐标。定义**坐标点输入**处设置的坐标点为图像坐标，还是物理坐标。

标定文件

与单点抓取模块中对应参数基本一致，此处不再赘述，具体参见 ***单点抓取*** 的同名参数。

模块结果

*标定转换*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出点

point 型，代表对输入坐标通过标定转换/逆转换后得到的的点。

转换坐标 X

float 型，代表对输入坐标通过标定转换/逆转换后得到的 X 坐标。

转换坐标 Y

float 型，代表对输入坐标通过标定转换/逆转换后得到的 Y 坐标。

转换角度

float 型，代表对输入角度通过标定转换/逆转换后得到的角度。

单像素精度

float 型，代表单个像素对应物理坐标系下的尺寸。

平移 X

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 X 。

平移 Y

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 Y 。

旋转

float 型，代表机构物理坐标系相对于图像坐标系的旋转角度（单位为弧度）。

- 当旋转 θ 为正值时，机构物理坐标系 X 轴沿逆时针方向旋转 θ 后，其 X 轴与图像坐标系 X 轴方向一致；
- 当旋转 θ 为负值时，机构物理坐标系 X 轴沿逆时针方向旋转-θ 后，其 X 轴与图像坐标系 X 轴方向一致。

尺度

float 型，代表世界坐标系中单位长度对应图像坐标系中的像素数。

斜切

float 型，代表世界坐标系的 Y 轴旋转角度与 X 轴旋转角度之差（单位为弧度）。

宽高比

float 型，代表世界坐标系的 Y 轴缩放量与 X 轴缩放量的比例。

13.9.6 点集对位

该模块与*单点对位*模块功能相似，均是得到对象点到目标点的偏移量。差别在于*单点对位*模块是通过一对点的坐标和角度得到的，而该模块是通过多个点坐标形成的点集得到的。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的原理与*单点对位*的 *模块原理* 大同小异，仅是一对点和多对点的差别，此处不在赘述。

使用方法

该模块必须配合 *图像源* 模块、 *定位类模块* 和 *标定转换* 模块使用，如下图所示。

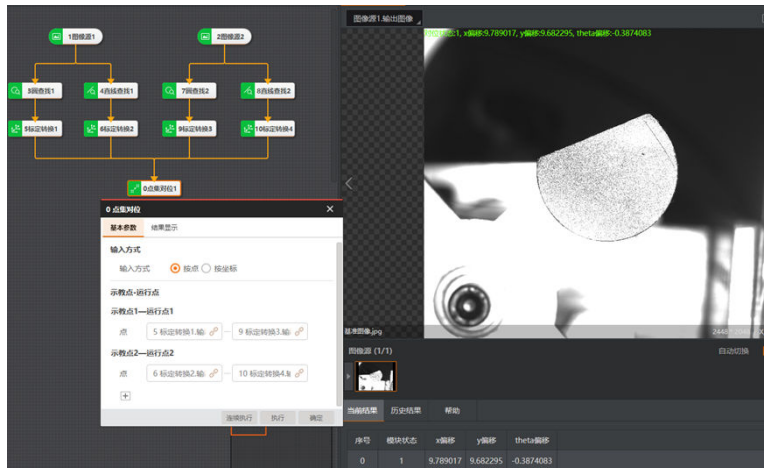


图 13-325 使用示意

上图中的图像源模块用于获取图像，定位类模块用于获取图像坐标系中的多组特征点，标定转换模块用于将图像坐标点转换为物理坐标点，点集对位模块用于在物理坐标系中计算得到运行点到示教点的偏移量。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

输入方式

可选按点、按坐标输入。

示教点-运行点

设置多个运行点对位至示教点的点坐标，可自定义输入或订阅前序模块对应的数据。一般从前序模块订阅。点击 ⊞ 可继续添加，至少 2 个以上，最多不超过 8 个。

点

点 X/Y

设置点坐标的对位关系。

模块结果

*点集对位*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

x 偏移

float 型，代表示教点到运行点 X 方向的位置移动量。

y 偏移

float 型，代表示教点到运行点 Y 方向的位置移动量。

theta 偏移

float 型，代表示教点到运行点的角度偏移量，顺时针为正。

13.9.7 旋转计算

该模块可将点或线绕着旋转中心点按照旋转角度旋转，并计算得到旋转之后该点或线的相关信息。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的原理为下图所示公式。其中 Out X/Y 为输出点的坐标，In X/Y 为输入点的坐标，X_0 和 Y_0 为旋转中心的坐标，θ 为图像坐标系下旋转角度（顺时针为正，逆时针为负）。

$$\begin{pmatrix} Out\ X \\ Out\ Y \end{pmatrix} = \begin{pmatrix} In\ X - X_0 \\ In\ Y - Y_0 \end{pmatrix} * \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} + \begin{pmatrix} X_0 \\ Y_0 \end{pmatrix}$$

使用方法

该模块一般配合 *图像源* 模块和 *定位类模块* 使用，如下图所示。

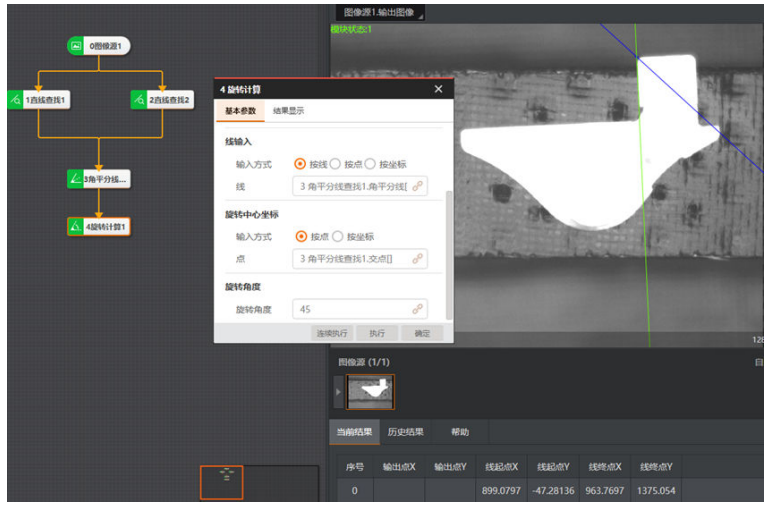


图 13-326 使用示意

上图中的图像源模块用于获取图像，定位类模块用于获取图像坐标系中的特征点或直线，旋转计算模块将将角平分线（蓝色）绕角平分线的交点旋转 45°，得到旋转后的直线（绿色）。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

图像输入

通过**输入源**订阅图像。

输入类型

通过**数据类型**设置旋转计算的目标，可选按点、按线。

点输入

选择旋转点时，设置旋转的点即可。

线输入

选择旋转线时，设置旋转的线即可。

旋转中心坐标

可设置旋转时的中心点。

旋转角度

可设置基于旋转中心点的旋转角度。正为顺时针旋转，负为逆时针旋转。

模块结果

*旋转计算*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出点

point 型，代表输入点旋转后得到的点。

输出点 X

float 型，代表输入点旋转后得到点的 X 坐标。

说明

采用的图像坐标系为左手坐标系。

输出点 Y

float 型，代表输入点旋转后得到点的 Y 坐标。

输出线

线起点

线起点 X

float 型，代表输入的线，旋转后得到线起点的 X 坐标 。

线起点 Y

float 型，代表输入的线，旋转后得到线起点的 Y 坐标 。

线终点

线终点 X

float 型，代表输入的线，旋转后得到线终点的 X 坐标 。

线终点 Y

float 型，代表输入的线，旋转后得到线终点的 Y 坐标 。

13.9.8 线对位

该模块的原理和功能与点集对位模块类似，差别点仅在于点集对位模块针对的是点，而该模块针对的是线。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块必须配合 *图像源* 模块、 *直线查找* 模块和 *标定转换* 模块使用。



图 13-327 使用示意

以上图为例，图像源模块用于获取图像，直线查找模块用于获取图像坐标系中的多组特征线，标定转换模块用于将图像坐标下的直线起点坐标和终点坐标都转换为物理坐标的起点和终点，几何创建模块用于将物理坐标的起点和终点进行连线构成示教线和运行线，线对位模块在物理坐标系中计算得到运行线对位到示教线的偏移量。

参数配置

- 基本参数：与点集对位模块的参数大同小异，此处不再赘述，具体参见点集对位的 *参数配置* 。
- 运行参数：可设置**对位形状**参数，可选开、闭。
  - 选择闭时：特征线必须形成闭合形状，方能检测。例如存在 4 条特征线，必须按顺序形成一个封闭矩形，才能进行检测。
  - 选择开时：特征线无需形成闭合形状也能检测。
- 结果显示：各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

模块结果

*线对位*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

x 偏移

float 型，代表示教点到运行点 X 方向的位置移动量。

y 偏移

float 型，代表示教点到运行点 Y 方向的位置移动量。

theta 偏移

float 型，代表示教点到运行点的角度偏移量，顺时针为正。

13.9.9 单位转换

该模块可将图像中的间距、宽度等距离，通过标定文件由像素距离转换为物理坐标系下的物理距离。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

说明

该模块仅为近似运算，存在一定误差。若需精确计算后得到精确的数值，推荐通过对特征点使用标定转换模块转换后，再通过点点测量模块计算得到。

使用方法

该模块主要配合 *图像源* 模块和 *定位类* 模块配合使用。图像源模块用于获取图像，定位类模块用于获取图像坐标系中特征点之间的距离。单位转换模块将订阅的像素距离转换为物理距离。如下图所示。



图 13-328 使用示意

说明

上图中通过单位转换模块得到的圆心和边缘交点的距离，与右侧点点测量模块得到的距离结果一致。

参数配置

该模块的参数分为**图像输入**、**单位转换**和**标定文件**三部分。
具体参数介绍如下：

图像输入

通过**输入源**订阅图像。

单位转换

通过**像素间距**订阅前序模块输出的距离。

标定文件

需设置**加载标定文件**、**刷新信号**和**像素当量修正**参数。

加载标定文件

点击 🗁 可选择需加载的标定文件。支持 iwcal、txt 和 xml 格式的标定文件。

刷新信号

该参数为必填项，默认设置为 0。通过自定义或订阅的参数值，您可以决定是否刷新标定文件以及**像素当量修正**参数设置是否生效。不同参数值对应的生效规则说明如下。

- 参数为 0 或空值时，仅在模块第一次运行时加载选择的标定文件，后续不再更新，且**像素当量修正**参数设置不生效。
- 参数为非 0 数值且非空值时，每次模块运行时会重新加载选择的标定文件。确保读取的标定文件是最新的，且**像素当量修正**参数设置生效。

像素当量修正

该参数为必填项，默认设置为 1。该参数可用于微调订阅的**像素间距**。设置完成后，模块将结合指定的修正系数，输出校正后的物理距离。***单位转换***的计算公式如下：输出的物理距离=像素间距*单像素精度*像素当量修正。

说明

若您无需修正**像素间距**，建议保持默认值。若您需要修正**像素间距**，可将**刷新信号**设置为非 0 任意数值，且在**像素当量修正**填写指定修正系数，模块将按照指定系数校正订阅数据。

模块结果

单位转换模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

单像素精度

float 型，代表单个像素对应物理坐标系下的尺寸。

转换结果

float 型，代表对输入角度通过单位转换后得到的结果。

像素距离

float 型，代表像素对应物理坐标系下的距离。

平移 X

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 X 。

平移 Y

float 型，代表利用计算得到的标定矩阵，将世界坐标系原点映射到图像坐标系得到的坐标 Y 。

旋转

float 型，代表机构物理坐标系相对于图像坐标系的旋转角度（单位为弧度）。

- 当旋转 θ 为正值时，机构物理坐标系 X 轴沿逆时针方向旋转 θ 后，其 X 轴与图像坐标系 X 轴方向一致；
- 当旋转 θ 为负值时，机构物理坐标系 X 轴沿逆时针方向旋转-θ 后，其 X 轴与图像坐标系 X 轴方向一致。

尺度

float 型，代表世界坐标系中单位长度对应图像坐标系中的像素数。

斜切

float 型，代表世界坐标系的 Y 轴旋转角度与 X 轴旋转角度之差（单位为弧度）。

宽高比

float 型，代表世界坐标系的 Y 轴缩放量与 X 轴缩放量的比例。

13.9.10 变量计算

该模块主要通过设计表达式，完成变量之间的算术或逻辑运算。变量可以时自定义参数，也可以从前序模块订阅输出的模块结果。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *表达式窗口介绍*
- *模块结果*

模块原理

该模块本质是一个计算器，其原理就是计算器的原理。
在计算器中输入一个计算公式（例如 1+3）后， 计算器会解析公式，得到数值 1、3 和运算符+。然后使用运算符从 Math 库里找到对应的运算，并传入数字做为参数，得到运算结果。VM 的变量计算模块在计算器的基础上，还可以订阅前序模块输出的模块解结果。例如“1+<1 圆查找 1.圆心 X>[0]+<1 圆查找 1.圆心 Y>[0]*3”，其中“<1 圆查找 1.圆心 X>[0]” 就是一个订阅数据。故解析时会增加一步，即先把订阅的字符串替换成具体的数字，再进行计算。

使用方法

该模块使用时，可与任意有数据作为模块结果输出的模块配合即可。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见*结果显示*。

点击 ⊞ 可添加变量的计算表达式，默认已添加一个，支持添加多个。

通过*重置*按钮，还可将所有变量重置为初始值。

每个表达式由以下参数组成：

- **名称**：可自定义变量的名称。
- **注释**：可对该条数据通过注释说明。
- **表达式**：点击 可在表达式窗口对具体公式进行配置。具体参见 *表达式窗口介绍*。
- **初始值**：可自定义该变量默认的初始值。
- **输出类型**：可选 int、float 和 POINT。选择 POINT 时，**表达式**仅支持点和点之间的加/减，以及点和常数相乘，不支持设置初始化及初始值，其他表达式窗口不支持的相关功能均为置灰状态。
- **初始化**：启用该参数后，每次流程执行时均会将变量重置为设置的**初始值**。否则，使用上轮运算结果输入到公式中参与运算。
- ：可删除该变量的计算表达式。

表达式窗口介绍

表达式配置窗口如下图所示，相关介绍如下：

- 窗口上方区域：公式显示和编辑区域，点击右侧的 可订阅前序模块。
- 窗口中间区域：标记公式时需使用的常数、运算符号和函数等。此处仅对右侧较复杂函数做介绍，具体参见下表，其他不详细展开。

表 13-26 部分函数使用说明

函数	说明	函数	说明
sinh(x)	x 为角度，返回 x 的双曲正弦值	asinh(x)	反双曲正弦，返回角度值
sin(x)	x 为角度，返回 x 的正弦值	asin(x)	反正弦，-1≤x≤1，返回角度值
cos(x)	x 为角度，返回 x 的余弦值	acos(x)	反余弦，-1≤x≤1，返回角度值
cosh(x)	x 为角度，返回 x 的双曲余弦值	acosh(x)	反双曲余弦，返回角度值
tan(x)	x 为角度，返回 x 的正切值	atan(x)	反正切，返回角度值
tanh(x)	x 为角度，返回 x 的双曲正切值	atanh(x)	反双曲正切，返回角度值
max(x,y)	返回 x 与 y 中的较大值	min(x,y)	返回 x 与 y 中的较小值

函数	说明	函数	说明
round (x)	将 x 的小数位四舍五入后返回整数	trunc(x)	将 x 的小数位取 0 后返回整数
ceil(x)	返回大于等于 x 的最小整数值	floor(x)	返回小于等于 x 的最大整数
log(x)	返回指定数字的自然对数（底为 e）	exp(x)	返回 e 的 x 次幂
pow(x,y)	数字 x 的 y 次幂	log10(x)	返回 x 以 10 为底的对数
sqrt(x)	返回 x 的平方根	abs(x)	返回 x 的绝对值

- **初始值**和**初始化**：功能与基本参数窗口为同一个参数，两边均可设置。
- **校验公式**：可校验设置的公式是否合理。若不合理，会提示失败，建议重新设置。

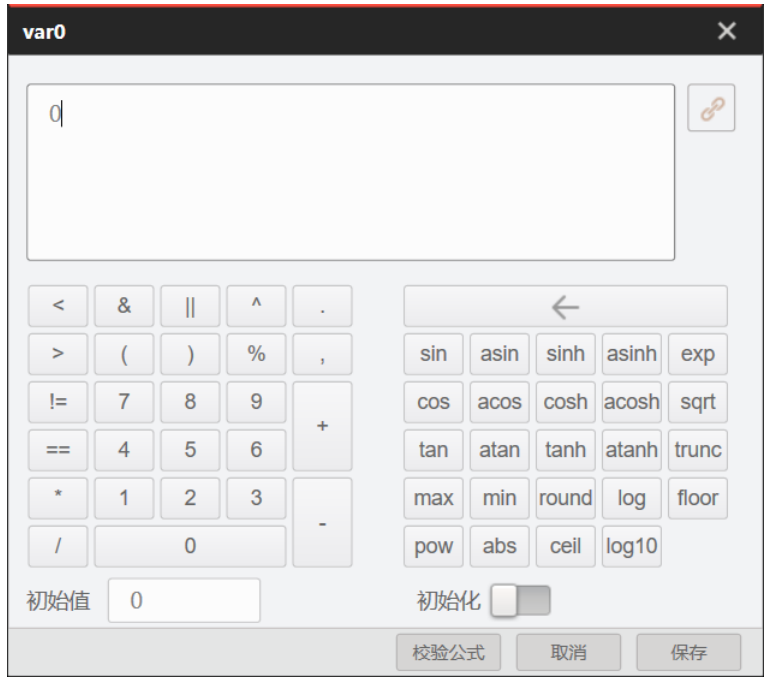


图 13-329 表达式窗口

模块结果

*变量计算*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

var0

float 型，代表默认添加的变量，默认值为 0。

结果显示

string 型，代表显示所有变量名和执行结果值。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.9.11 坐标转换

该模块可将图像上某个点的坐标通过计算转换为其他坐标点信息。一般配合划片拆分模块使用，将划片中的某点坐标转到原图中（即划片前的图像），可获取各个划片区域点在原图中的坐标信息。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块通过获取到的划片区域在原图中的划片中心点坐标、宽、高，从而计算出划片区域左上角在原图中的坐标 P，再将需要转换点的坐标加上坐标 P，即可得到转换点在原图中的坐标信息。

使用方法

该模块必须配合 *图像源* 模块、 *划片拆分* 模块、 *拷贝填充* 模块、 *定位类* 模块使用。
在如下示意图中，图像源模块用于获取图像，划片拆分模块用于将原图像拆分为 2*2 的四个划分区域，拷贝填充模块用于获取第三个划分区域（即原图左下角区域），边缘交点模块提取划片区域的边缘交点做为特征点，最后通过坐标转换模块根据划片信息将划片区域上的特征点坐标进行转换，得到此特征点在原图上的坐标信息。
坐标转换模块的输出点与“边缘点交点 2”模块的结果相同。“边缘点交点 2”模块直接在原图上提取边缘交点坐标信息。



图 13-330 使用示意

参数配置

该模块的参数分为**划片信息**和**输入点**两部分。

划片信息

可设置是否开启**坐标转换使能**，并设置相关参数。

坐标转换使能

该功能默认开启。若关闭，则输入点和输出点的数据完全一致。

输入方式

可设置划片中心点的输入方式，可选按点、按坐标。

划片中心点
划片中心点 X/Y

一般从前序划片拆分模块订阅划片中心点即可。

划片宽
划片高

一般从前序划片拆分模块订阅划片区域的宽/高即可。

输入点

设置输入点坐标的信息来源。

输入方式

可选按点、按坐标。

点
点 X/Y

可自定义或层前序模块订阅点的坐标信息。

模块结果

*坐标转换*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出点

point 型，代表输入点旋转后得到的点。

输出点 X

float 型，代表输入点旋转后得到点的 X 坐标。

说明

采用的图像坐标系为左手坐标系。

输出点 Y

float 型，代表输入点旋转后得到点的 Y 坐标。

13.10 图像处理

图像处理是对目标图像进行图像预处理的过程。当图像对比度较差、毛刺较多、干扰较多、特征不明显时，可考虑使用图像处理模块进行预处理。

13.10.1 图像二值化

图像二值化即将原图像转换为仅包含 0 和 255 两个灰度值的图像。图像二值化的目的是简化图像，突出目标轮廓，方便后续图像处理。例如，在文本识别、边缘检测、图像分割等应用场景中，二值化图像可以帮助算法更准确地识别图像中的边缘和特征。此外，图像二值化还可以帮助减小图像的文件大小，便于存储和传输图像。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*图像二值化*模块采用的二值化方式共有如下五种，各方式的工作原理有所差异。

- *硬阈值二值化*
- *均值二值化*
- *高斯二值化*
- *Sauvola 二值化*
- *自动二值化*

硬阈值二值化

根据预设的低灰度阈值和高灰度阈值进行图像二值化。输入图像像素点的灰度值大于低灰度阈值或小于高灰度阈值时，则为目标（灰度值转换为 255），否则则为背景（灰度值转换为 0）。

均值二值化

根据以下四个参数进行图像二值化。

- 滤波核宽度
- 滤波核高度
- 比较类型
- 阈值偏移量

具体计算过程如下：

1. 在滤波核尺寸范围内对输入原图进行均值滤波，得到背景图像。
2. 计算输出原图和背景图上相同像素点的灰度值之差。
3. 根据差值、预设的比较类型、预设的阈值偏移量，判断像素点为目标还是背景。
   当像素点的差值满足预设的比较类型和阈值偏移量时，则为目标。其中比较类型包括：≥、≤、=、≠。

例子：假设预设的核宽高分别为 15 和 16，比较类型为≥，阈值偏移量为 20，那么这些配置表示：“原图和背景图差值≥20 的像素点为目标，否则为背景。其中背景图通过核尺寸为(15,16)的均值滤波求得。”



图 13-331 输入原图

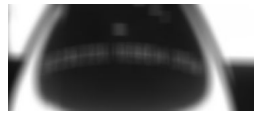


图 13-332 背景图

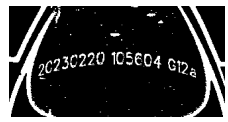


图 13-333 二值化图

高斯二值化

根据以下四个参数进行图像二值化。

- 高斯滤波核尺寸
- 高斯标准差
- 比较类型
- 阈值偏移量

其中高斯滤波核尺寸和高斯标准差用于控制高斯滤波程度。两者越大，滤波效果越强。

具体算法工作流程与上文提及的均值二值化的相同，均为先滤波得到背景图，再将原图和背景图做差，最终得到满足预设关系的二值图。

Sauvola 二值化

根据以下五个参数进行图像二值化。

- 校正系数
- 动态范围
- 分割类型
- 滤波核宽度
- 滤波核高度

具体算法过程为先计算每个像素点在设定核尺寸下的灰度均值 m 和标准差 std，再根据预设的分割类型进行如下判断：

- 分割类型为亮背景时，灰度均值满足如下计算公式则为目标，否则为背景。

$$\mathrm{m}(1+\mathrm{k}\left(\frac{std}{r}-1\right)<\mathrm{pixel}$$

图 13-334 分割类型为亮背景时

- 分割类型为暗背景时，灰度值满足如下计算公式则为目标，否则为背景。

$$\mathrm{m}(1+\mathrm{k}\left(\frac{std}{r}-1\right)>\mathrm{pixel}$$

图 13-335 分割类型为暗背景时

说明

以上公式中的 k 和 r 分别表示校正系数和动态范围。

Sauvola 二值化算法工作流如下图所示。



图 13-336 算法工作流

自动二值化

算法根据最大类间方差法计算自动阈值 T，并根据该阈值进行图像二值化。输入图像中灰度值大于 T 的像素点为目标，小于的为背景。该方式不涉及预设参数。

使用方法

以下对该模块的使用方法分为前后序模块和应用示例两个方面进行介绍。

前后序模块

在流程中，*图像二值化*的：

- 前序模块一般为 *图像源* 。该模块可为*图像二值化*提供原图输入。
- 后序模块一般为其他图像处理模块以及识别、定位、测量等模块。*图像二值化*输出的二值图可用于识别、定位和测量。

应用示例

相关应用示例请参见 *形态学处理* 中的示例。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

ROI 外处理

可设置除 ROI 区域外的图像处理方式，可选黑色或原图。

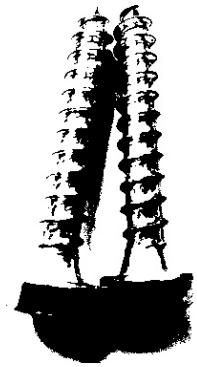
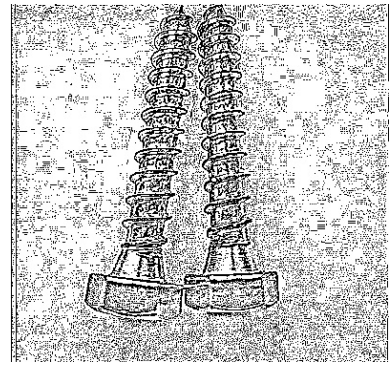
二值化类型

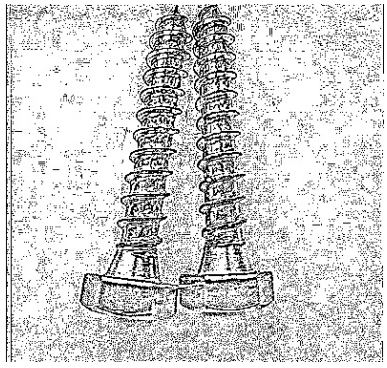
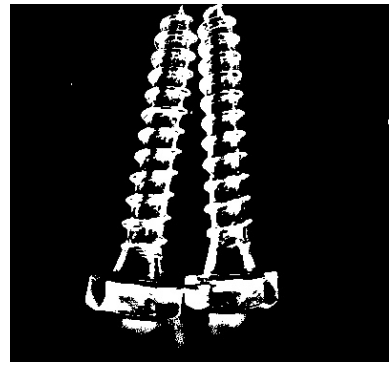
可选择不同的二值化类型。不同类型适用于不同场景。

- **硬阈值二值化**：适用于光影影响较小的环境。
- **均值二值化**：一般用于平滑图像，消除噪声。
- **高斯二值化**：一般用于在保证图像原有信息特征的情况下消除部分噪声。
- **Sauvola 二值化**：适用于对背景单一、光照不均匀的图像进行二值化处理。



图 13-337 原图

硬阈值二值化示例	均值二值化示例
	

高斯二值化示例	Sauvola 效果示例
	

硬阈值二值化

需设置**高阈值**或**低阈值**。具体原理参见上文*模块原理*中提及的 ***硬阈值二值化***

低阈值

用于二值化的低灰度阈值。

灰度指黑白照片上像素点的明暗指数，值为 0 时显示为黑色，为 255 时显示为白色

高阈值

用于二值化的高灰度阈值。

- 当低阈值小于高阈值时，如果像素点灰度值在高低阈值大小范围内，那么该像素点灰度值将被置为 255。
- 当低阈值大于高阈值时，如果像素点灰度值在高低阈值大小范围外，那么该像素点的灰度值将被置为 255。

均值二值化

需设置如下子参数。

滤波核宽度/高度

均值滤波时，矩形滤波窗口宽度与高度。均值滤波即用滤波窗口遍历图像内各点（当前点位于窗口中心），将滤波窗口中邻域点的平均灰度值作为当前点的灰度值。

比较类型

将已经通过滤波核宽度/高度遍历后得到的灰度均值按大于等于、小于等于、等于、不等于四种类型进行处理，满足比较类型条件的图像区域的灰度值置为 255。

阈值偏移量

可容忍的与背景差异的最大阈值。

高斯二值化

需设置如下参数。

高斯滤波核

高斯滤波核的大小，滤波核越大滤波效果越平滑。

高斯标准差

高斯标准差越大，高斯二值化效果越明显。

比较类型

将已经通过高斯滤波核遍历后得到的灰度高斯值按大于等于、小于等于、等于、不等于四种类型进行处理，满足比较类型条件的图像区域的灰度值置为 255。

阈值偏移量

得到高斯值后和偏移量做累加运算得到的结果作为最终高斯值。

Sauvola 二值化

需设置如下参数

校正系数

对应上文*模块原理*中提及的 *图 13-334* 中的 k，校正系数越大，二值化阈值越大。

动态范围

对应上文*模块原理*中提及的 *图 13-334* 中的 r，动态范围越大，二值化阈值越小 。

分割类型

分为暗于背景和亮于背景两种。分割类型的设置影响最终的二值化效果, 具体见上文*模块原理*中提及的 *Sauvola 二值化*

滤波核宽/高度

滤波核的宽度和高度，滤波核主要用于图像遍历。

自动

即对图片进行自动二值化处理，具体原理见上文*模块原理*中提及的 *自动二值化* 。

以下运行参数中的不同**二值化类型**适用于不同场景：

模块结果

*图像二值化*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.10.2 图像滤波

滤波是一类图像处理算法。该类算法通过特定函数对输入图像进行预处理，去除图像中的噪声，进而突出或抑制图像中的特定特征。在不少场景下，图像滤波是图像预处理中不可或缺的操作，其处理效果的好坏将直接影响后续图像处理/图像分析的有效性和可靠性。

说明

图像处理中的噪声指图像采集或传输过程中所受的随机信号干扰。具体表现为图像信息或者像素亮度的随机变化。

模块原理

*图像滤波*模块提供的图像滤波算法包括：

- *高斯滤波*
- *中值滤波*
- *均值滤波*
- *取反滤波*
- *边缘提取*

高斯滤波

滤波时为各像素点逐个生成检测窗口（即下图中的橙框），目标像素点位于窗口中心（即下图中的灰色像素点）。滤波后目标像素点的灰度值，通过对窗口内各像素点的灰度值加权计算得出。

下文 *参数配置* 中提及的**高斯滤波核**可定义上述窗口的尺寸。

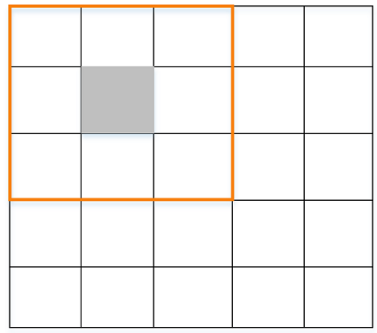


图 13-338 滤波窗口（即滤波核）

加权计算的具体权重根据下图所示的高斯滤波曲线定义。算法会根据窗口尺寸自动调整曲线的高度和坡度，得到窗口内每个像素点权重 σ 。由图中的高斯滤波曲线可知，距离窗口中心的像素点越近，曲线越高，权重越大。

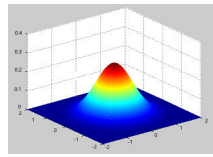


图 13-339 高斯滤波曲线

目标像素点的灰度值计算公式如下。该公式中 $gray_{pixel}$ 表示目标像素点的灰度值， σ_i 表示窗口中每个像素点的权重， $gray_i$ 表示窗口中每个像素点的灰度值。

$$\text{gray}_{\text{pixel}} = \sum \sigma_i \, gray_i$$

图 13-340 高斯滤波计算公式

中值滤波

滤波时为各像素点逐个生成检测窗口（即上文 *图 13-338* 图中的橙框），目标像素点位于窗口中心（即上文 *图 13-338* 图中的灰色像素点）。滤波后目标像素点的灰度值为窗口内像素点灰度值的中值。

下文 *参数配置* 中提及的**滤波核宽度**和**滤波核高度**可定义上述检测窗口的尺寸。

均值滤波

滤波时为各像素点逐个生成检测窗口（即上文 *图 13-338* 图中的橙框），目标像素点位于窗口中心（即上文 *图 13-338* 图中的灰色像素点）。滤波后目标像素点的灰度值为窗口内像素点的灰度均值。

下文 *参数配置* 中提及的**滤波核宽度**和**滤波核高度**可定义上述检测窗口的尺寸。

取反滤波

对每个像素点进行颜色取反，如对灰度范围 0~255 内的像素值 10 取反，取反后像素值为 245。从视觉上看，原黑色的像素取反后变为白色，白色像素变为黑色。该滤波方式常用于切换图像的背景和目标。

边缘提取

边缘提取的算法工作流如下：

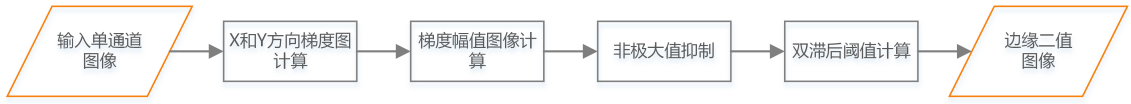


图 13-341 边缘提取算法工作流

1. 计算 X 和 Y 方向的梯度图像。
   算法依据滤波窗口，在 X 方向对窗口中心点的灰度值进行加权赋值，得到 X 方向梯度图；同理，可得到 Y 方向梯度图。梯度越大，则边缘越明显。
2. 合成梯度幅值图像。
   梯度幅值图像由梯度 X 和 Y 图像合成，表示图像梯度边缘强度。该图中各像素点灰度值的计算公式如下。

$$Grad_{(i,j)} = \sqrt{dx_{(i,j)}^2 + dy_{(i,j)}^2}$$

图 13-342 梯度幅值图计算公式

该公式中：

- $Grad_{(i,j)}$表示合成图像中 i 行 j 列像素点的灰度值。
- $dx_{(i,j)}$ 表示 X 方向梯度图像 i 行 j 列像素点的灰度值。
- $dy_{(i,j)}$ 表示 Y 方向梯度图像 i 行 j 列像素点的灰度值。

该步骤获得的边缘，包含非单像素边缘和低对比度边缘。算法需要在此基础上进一步筛选目标边缘。

3. 对梯度幅值图像进行非极大值抑制，保证边缘为单像素点。
   非极大值抑制的目的是获取梯度方向上幅值最大的边缘点。若边缘点不是梯度方向上最大值，则删除该边缘点。
4. 进行双滞后阈值计算，筛选真正的边缘点，对应的灰度值为 255。
   具体筛选规则为：如果像素点梯度幅值大于预设的高阈值，则为强边缘，即一定是边缘点；如果在低阈值和高阈值之间则为弱边缘，即可能为边缘点；当弱边缘的 8 邻域存在强边缘，则将弱边缘变为强边缘，该操作可有效保证边缘的连续性。
   该计算过程涉及的高低阈值，可通过下文 ***参数配置*** 中提及的**边缘阈值范围**定义。

使用方法

以下对该模块的使用方法分为前后序模块、主要配置步骤和应用示例这三个方面进行介绍。

前后序模块

在流程中的，***图像滤波***模块对前后序模块无特定要求。

主要配置步骤

在流程中调用***图像滤波***模块后，该模块的主要配置步骤如下：

1. 执行一次流程，使***图像滤波***模块获取***图像源***模块采集的图像。
2. 在***运行参数***页签选择合适的 ***图像滤波类型*** 。
3. 在***基本参数***页签设置 ***基本参数*** 。
4. （可选）单击该模块配置窗口下方的***执行***或***连续执行***，查看滤波效果。如效果不理想，调整参数取值，直至滤波效果满足需求。



图 13-343 图像滤波配置窗口

应用示例

以下示例通过 *图像增强* 和*图像滤波*模块去除图像噪声，使工件边缘更加清晰平滑。后序通过 *轮廓匹配* 匹配特征圆环，再将匹配特征传递至 *圆查找*，进行同心圆检测。

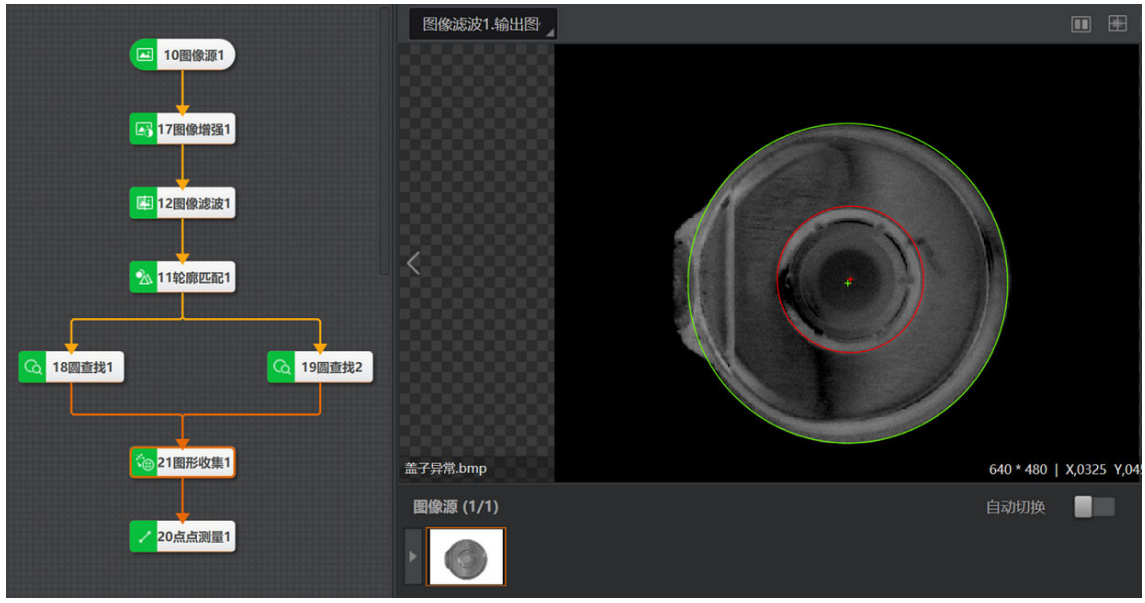


图 13-344 同心圆测量前的图像预处理



图 13-345 示例中的图像预处理

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

该模块的运行参数为**图像滤波类型**，具体选项如下：

高斯滤波

属于低通滤波器。可对图像进行平滑，抑制噪声，尤其是高斯噪声。具体原理参见上文*模块原理*中提及的 *高斯滤波* 。

高斯滤波核

即上文 *模块原理* 中提及的 *图 13-338* 的大小，取值范围 1~101。该值越大，过滤的噪点尺寸越大，丢失的高频信息越多；表现为图像失焦效果越明显，即图像越模糊。



图 13-346 原图

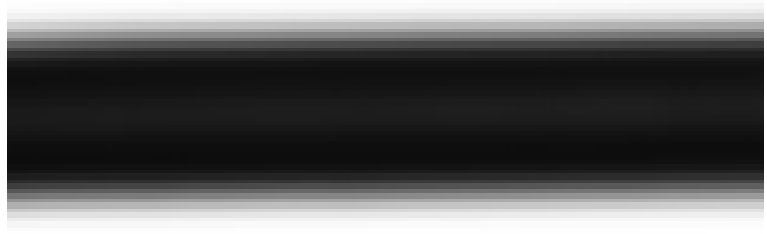


图 13-347 高斯滤波核为 15*15

中值滤波

属于非线性滤波器。可平滑图像，抑制灰度值明显异于背景的噪声。相比均值滤波和高斯滤波，保边能力更强，但耗时更长。

*图 13-338*越大（由滤波核高度和宽度定义），过滤的噪点尺寸越大，丢失的高频信息越多；表现为图像失焦效果越明显，即图像越模糊。

滤波核高度

滤波窗口的高度。

滤波核宽度

滤波窗口的宽度。



图 13-348 原图



图 13-349 中值滤波核为 15*15

均值滤波

属于低通滤波器。可对图像进行平滑，抑制噪声。具体原理见上文*模块原理*中提及的 ***均值滤波*** 。

图 13-338 越大（由滤波核高度和宽度定义），过滤的噪点尺寸越大，丢失的高频信息越多；表现为图像失焦效果越明显，即图像越模糊。

滤波核高度

滤波窗口的高度。

滤波核宽度

滤波窗口的宽度。



图 13-350 原图



图 13-351 均值滤波核为 15*15

取反

该参数详情参见上文*模块原理*中提及的 ***取反滤波*** 。



图 13-352 原图

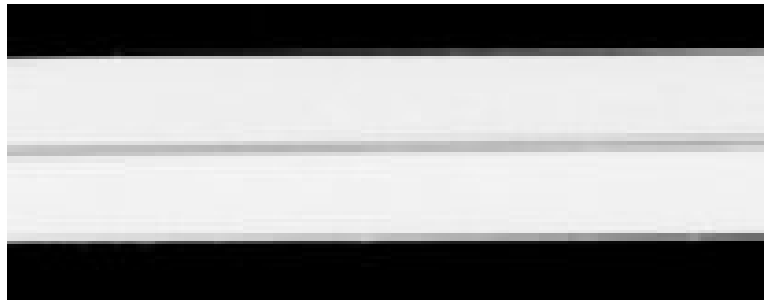


图 13-353 取反滤波效果

边缘提取

将图像边缘点的灰度值转换为 255，非边缘点灰度值转换为 0（即背景）。该算法可提取**边缘阈值范围**内的单像素点边缘。

边缘阈值范围

仅提取边缘梯度阈值大于该值的边缘点。取值范围 0~255。

数值越大，抗噪能力越强，得到的边缘数量越少，甚至导致目标边缘点被筛除。



图 13-354 原图

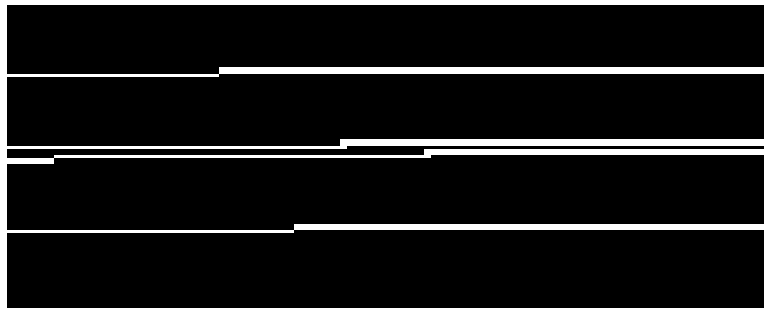


图 13-355 阈值范围 60~125

模块结果

*图像滤波*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.10.3 掩膜工具

*掩膜工具*模块可对输入图像上的指定区域进行形态学处理和掩膜运算。该模块可适应对不规则区域的检测需求，并通过掩膜预处理，为后续的识别、定位、测量和检测等任务提供更准确的图像数据。

模块原理

该模块的核心原理是对输入掩膜图像和订阅掩膜图像中指定像素位置的灰度值作集合运算，并将运算结果渲染成一张新掩膜图像。该模块中涉及形态学处理的原理部分与 ***形态学处理*** 模块原理基本一致，差别仅在于该模块支持叠加多个形态学处理操作且支持多 ROI 绘制。因此，关于形态学处理的原理部分，本节不再重复赘述。本节将基于以下示例介绍掩膜运算相关的模块原理。

假设您在输入图像上绘制了一个圆形及矩形 ROI，并在运行参数的掩膜运算区域订阅了前序模块输出的包含不规则 ROI 的图像。在运算时，掩膜工具将把输入图像及订阅图像渲染为对应的掩膜图，如下图所示。左侧掩膜图像 A 为基于输入图像生成的掩膜图（也即“输入掩膜图”），右侧掩膜图像 B 为基于订阅图像生成的掩膜图（也即“订阅掩膜图”）。



图 13-356 掩膜图像示例

*掩膜工具*目前支持 4 种运算类型：

交集

该运算类型取用掩膜图 A 与 B 中指定区域内对应像素的较小值作为新像素点的值，仅保留 2 张掩膜图中的白色重叠区域。交集运算的公式如下。

$$A \cap B = \min(A, B)$$

图 13-357 交集运算公式

交集运算的处理效果如下。

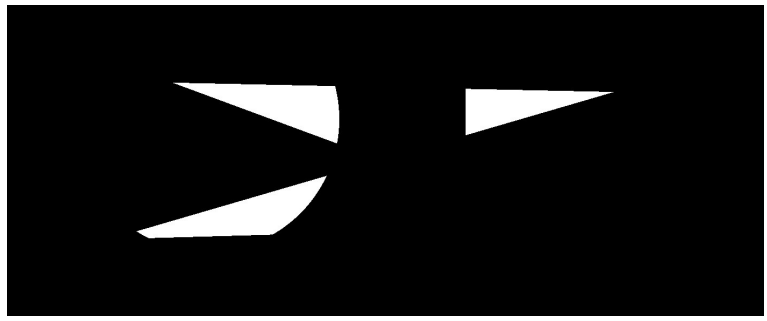


图 13-358 交集运算处理效果

并集

该运算类型取掩膜图 A 与 B 中指定区域内对应像素的较大值作为新像素点的值，保留 2 张掩膜图中所有的白色区域。并集运算的公式如下。

$$A \cap B = \max(A, B)$$

图 13-359 并集运算公式

并集运算的处理效果如下。

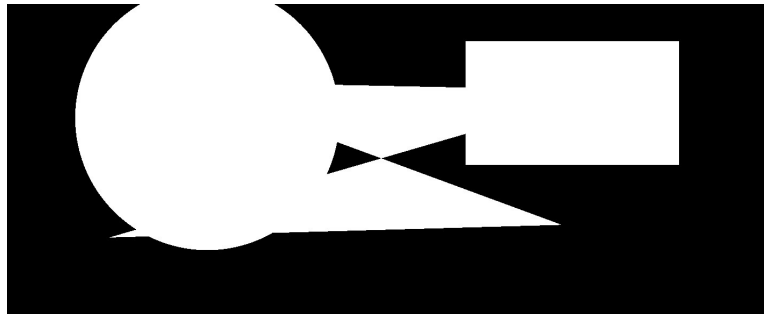


图 13-360 并集运算处理效果

补集

该运算类型将原掩膜图中灰度值为 0 的 ROI 区域和灰度值为 255 的 ROI 区域进行互换，可实现目标掩膜区域的反向转换。补集运算的符号如下。



图 13-361 补集运算符号

¬A 表示取掩模图 A 的反集（也即将掩模图 A 中原灰度值为 255 的 ROI 区域置为 0，原灰度值为 0 的 ROI 区域的灰度值置为 255）。

补集运算的处理效果如下。



图 13-362 补集运算处理效果

差集

该运算类型先取掩膜图 B 的补集，再与掩膜 A 取并集，仅保留掩膜图 B 的补集和掩膜图 A 中的白色重叠区域。差集运算的公式如下。

$$A - B = A \cap \neg B$$

图 13-363 差集运算公式

¬B 表示取掩模图 B 的反集（也即将掩模图 B 中原灰度值为 255 的 ROI 区域置为 0，原灰度值为 0 的 ROI 区域的灰度值置为 255）。

差集运算的处理效果如下。

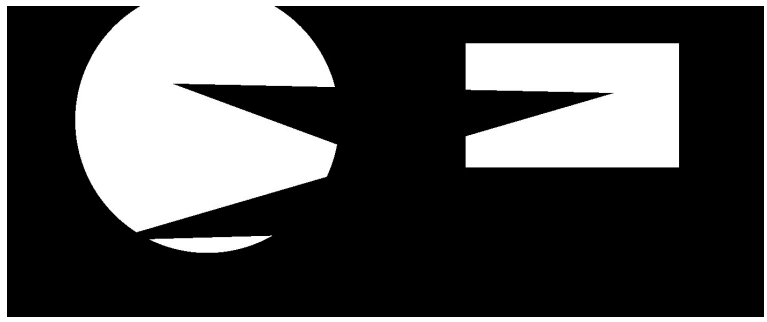


图 13-364 差集运算处理效果

使用方法

下文从前后序模块和应用示例这两个方面对该模块的使用方法进行介绍。

前后序模块

在流程中，*掩膜工具*的：

- 前序模块一般为 *图像源* 或 *图像二值化* 等图像处理模块。这些模块可为*掩膜工具*提供原图输入。
- 后序模块一般为识别、定位、测量以及图像处理等类别的模块。

应用示例

此处以孔洞面积测量为例，介绍如何使用*掩膜工具*提取掩膜图像上的指定孔洞区域并完成 Blob 面积测量。该示例使用*图像二值化*将输入图像转为二值图并绘制屏蔽区域，屏蔽圆形工件中央的三处孔洞；使用*掩膜工具*对屏蔽区和原图进行并集运算，并将运算后的掩膜图像传入 *Blob 分析*进行 Blob 面积计算，最终通过*格式化*模块输出该工件中央三处孔洞的面积数据。

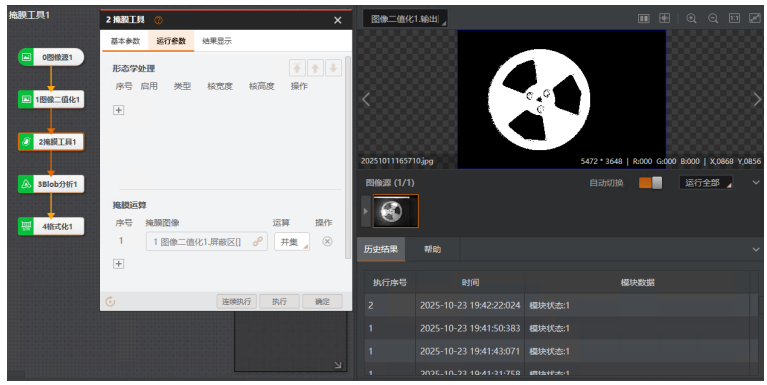


图 13-365 孔洞面积测量

参数配置

以下仅介绍该模块的运行参数配置步骤。通过以下步骤，您可以对输入图像进行各类形态学处理操作和掩膜运算。

说明

- 基本参数详情，请参见 ***基本参数*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

1. 在***掩膜工具***的模块配置窗口，选择***运行参数***页签。
2. 在**形态学处理**区域，单击 ⊞ 添加形态学处理操作。
   完成操作添加后，需设置如下列。

   启用

   按需勾选**启用**单选框，设置模块在执行时对输入图像进行指定形态学处理。未勾选的形态学处理将不执行。

   类型

   添加形态学处理操作时可选择处理操作类型。选择完成后仅支持查看，不支持编辑。可选**膨胀**、**腐蚀**、**开**和**闭**。更多有关处理操作类型的介绍，请参见 ***模块原理*** 。

   核高度
   核宽度

   设置滤波核的宽度和高度，适当增大该值会使形态学处理的效果更明显。

   操作

   单击 ◢ 右下方的小三角，可设置形态学处理相关参数。

   形态学形状

   指滤波核的形状，包括矩形、椭圆形和十字形。设置后，结果图像的轮廓会和形态学形状比较相似。更多有关滤波核形状的介绍，请参见 ***模块原理*** 。

   迭代次数

   指一次形态学处理过程中对原图像进行的操作次数，迭代次数越多，效果越明显，但不宜过多。
3. （可选）完成操作添加后，您还可以进行如下可选操作。
   - 调整选项排序：单击 ⇈⇑⇓ 可置顶/上移/下移当前操作项。
   - 删除操作项：单击 ⊗ 即可删除当前选项。
4. 在**掩膜运算**区域，单击 ⊞ 添加掩膜运算操作。
   完成操作添加后，需设置如下列。

   掩膜图像

   添加掩膜运算操作时可按需选择如下掩膜图像添加方式：

- **直接订阅**：订阅前序模块输出的**屏蔽区**。
- **外部导入**：从本地导入掩膜图像用于运算，支持 PNG、JPG、JPEG、BMP 格式文件。
- **订阅前序图像**：订阅前序模块输出的**输出掩膜**，可直接进行掩膜图像运算，此方式效率优于**直接订阅**。

运算

设置掩膜运算类型，可选**交集**、**并集**、**补集**和**差集**。

操作

单击 ⊗ 即可删除当前选项。

5. 设置完成后，在当前窗口右下角单击*执行*。

模块结果

*掩膜工具*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

掩膜运算图像

image 型，代表模块计算出的掩膜图像，记录了该掩膜图像的宽度、高度和像素格式。

掩膜运算图像数据

image 型，代表经过形态学处理和掩膜计算后的掩膜图像，以二值图的形式输出。

掩膜运算图像宽度

int 型，代表掩膜运算图像的宽度。

掩膜运算图像高度

int 型，代表掩膜运算图像的高度。

掩膜运算图像像素格式

int 型，代表掩膜运算图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区

string 型，代表绘制的屏蔽区位置信息，主要用于后序模块继承该屏蔽区。

13.10.4 形态学处理

*形态学处理*模块主要用来从图像中提取出对描绘区域形状有意义的图像分量，使后续的识别工作能够抓住目标对象最为本质的形状特征，如边界和连通区域等。形态学处理针对的是图像中的白色像素点。

模块原理

形态学处理的数学基础是集合论。对图像的形态学处理通过“结构元”实现。可认为“结构元”是一种特殊的滤波器，其对像素的操作并非灰度运算，而是集合运算。
下图中展示了几种不同形状的结构元（无效区域为白色）。

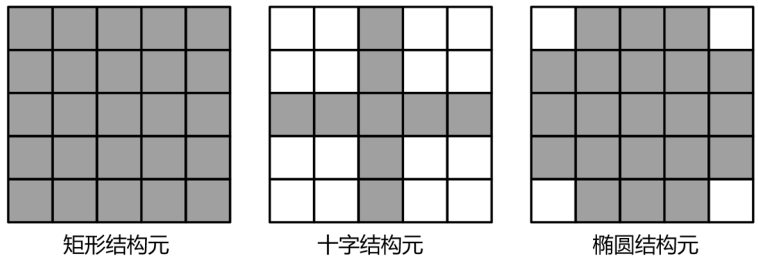


图 13-366 结构元形态

该模块处理二值图和灰度图的算法工作原理有所差异。

二值图像形态学处理

腐蚀

在二维空间中的集合 A 和 B，将 B（结构元）对 A（原始图像）的腐蚀定义为如下公式：

$$A \ominus B = \{z|(B)_z \subseteq A\}$$

图 13-367 二值图像腐蚀处理公式

以上公式可描述为当结构元 B 的锚点在像素 z 的位置时，结构元 B 的所有像素均包含于原始图像 A。

下图描述了二值图像腐蚀操作的样例：使用 3×3 的十字形结构元，锚点为对称中心。其中腐蚀结果图中的橙黄色区域将被填充为背景。

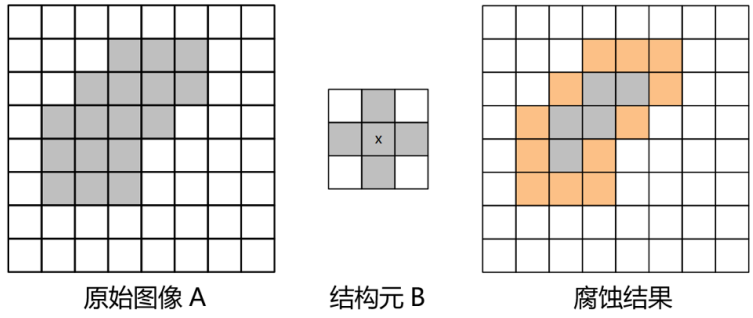


图 13-368 二值图像腐蚀结果

膨胀

在二维空间中的集合 A 和 B，将 B（结构元）对 A（原始图像）的膨胀定义为如下公式：

$$A \oplus B = \left\{ z \middle| \left(\hat{B}\right)_z \cap A \neq \emptyset \right\}$$

图 13-369 二值图像膨胀处理公式

以上公式中， $\hat{B}$ 为原始结构元 B 的反射，当结构元中心对称且锚点位于其集合中心时，B 的反射就是其自身。以上公式可描述为当 $\hat{B}$ 在像素 z 位置时，结构元中至少有一个像素包含于 A。

下图描述了二值图像膨胀操作的样例：使用 3×3 的十字形结构元，锚点为对称中心，反射即为其自身。其中膨胀结果图中的橙黄色区域将被填充为前景。

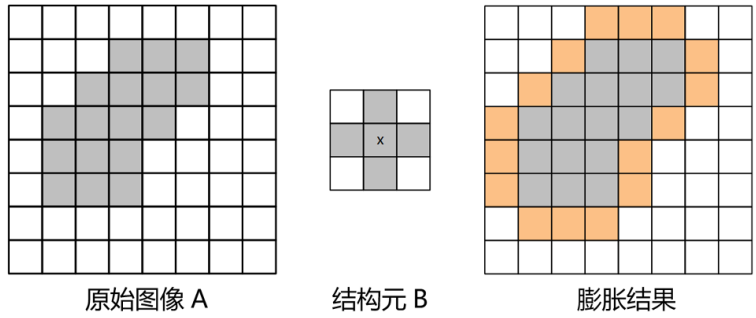


图 13-370 二值图像膨胀结果

开处理与闭处理

开处理和闭处理时腐蚀与膨胀处理的组合，在二维空间中的集合 A 和 B，将 B（结构元）对 A（图像）的开处理定义为如下公式。该公式可理解为先用结构元 B 对 A 进行腐蚀，再用结构元 B 对腐蚀的结果进行膨胀。

$$A \circ B = (A \ominus B) \oplus B$$

图 13-371 二值图像开处理公式

B（结构元）对 A（图像）的闭处理定义为如下公式。该公式可理解为先用结构元 B 对 A 进行膨胀，再用结构元 B 对膨胀的结果进行腐蚀。

$$A \cdot B = (A \oplus B) \ominus B$$

图 13-372 二值图像闭处理公式

开处理和闭处理的效果参考 *图 13-368*。使用的结构元为 5×5 的矩形结构元。在二值图像中使用开/闭操作时，都能平滑边缘。但开操作和闭操作的对细节的处理却相反，开操作将细节前景归纳为背景，闭操作则将细节背景填充为前景。

灰度图像形态学处理

由于灰度图像的灰度值不限于 0 和 1 两个值，将形态学处理从二值图像推广至灰度图像时，其定义发生了一定变化。首先，灰度级的加入使得结构元的有效区域可能存在不同的灰度值，进而衍生出“非平坦”和“平坦”两种结构元。

平坦结构元是指有效结构元区域内所有像素灰度值相等的结构元。如果像素灰度值不相等，则为非平坦结构元。下图展示了两种结构元的灰度分布示例。

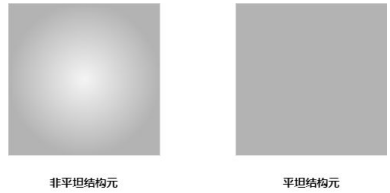


图 13-373 非平坦结构元和平坦结构元

说明

灰度形态学处理主要使用平坦结构元，非平坦结构元应用极少。以下主要介绍平坦结构元的形态学处理。

腐蚀

由于图像和结构元像素灰度值均不限于 0~1。灰度结构元 b 对一幅灰度图像 f 的腐蚀处理定义为如下公式。

$$[f \ominus b](x,y) = \min_{(s,t)\in b}\{f(x+s,y+t) - b(s,t)\}$$

图 13-374 灰度图像腐蚀处理公式

其中 b(s,t)为结构元中坐标为(s,t)处的灰度值。当结构元平坦时，b(s,t)不变，则平坦结构元的腐蚀定义为如下公式。即寻找结构元有效范围内的灰度极小值，用于替换结构元锚点所在的图像坐标的像素灰度值。从实现结果上看，与特殊形状模板的极小值滤波等价。

$$[f \ominus b](x,y) = \min_{(s,t)\in b}\{f(x+s,y+t)\}$$

图 13-375 平坦结构元腐蚀定义

膨胀

与腐蚀相似，灰度结构元 b 对一幅灰度图像 f 的膨胀处理定义为：

$$[f \oplus b](x,y) = \min_{(s,t)\in b}\{f(x-s,y-t)+b(s,t)\}$$

图 13-376 灰度图像膨胀处理公式

其中 b(s,t)为结构元中坐标为(s,t)处的灰度值，当结构元平坦时，b(s,t)不变，则平坦结构元的膨胀定义为如下公式，即寻找结构元有效范围内的灰度极小值，用于替换结构元锚点所在的图像坐标的像素灰度值。从实现结果上看，与特殊形状模板的极大值滤波等价。

$$[f \oplus b](x,y) = \min_{(s,t)\in b}\{f(x-s,y-t)\}$$

图 13-377 灰度图平坦结构膨胀定义

说明

灰度图像的膨胀处理同样需要先对结构元求反射。

开处理与闭处理

灰度图像的开处理和闭处理定义与二值图像的一致，结构元 b 对灰度图像 s 的开处理和闭处理定义分别为：

$$f \circ b = (f \ominus b) \oplus b$$

图 13-378 灰度图像开处理公式

$$f \cdot b = (f \oplus b) \ominus b$$

图 13-379 灰度图像闭处理公式

灰度图像的开处理和闭处理的效果参考 *图 13-373*，使用的结构元为 5×5 的矩形结构元。开处理将削弱亮特征，但不影响暗特征和背景；相反，闭处理则将削弱按特征，但不影响亮特征和背景。开闭处理对亮暗特征的削弱程度取决于平坦结构元的尺寸。

使用方法

前后序模块

在流程中，*形态学处理*的：

- 前序模块一般为 *图像源* 。该模块可为*形态学处理*提供原图输入。也可为 *图像二值化* 等其他图像处理模块。
- 后序模块一般为识别、定位、测量以及图像处理等类别的模块。

应用示例

下图展示了通过组合调用 *图像二值化* 、*形态学处理*、 *DL 读码* 等模块，实现二维码稳定读取的应用示例。



图 13-380 形态学处理应用示例

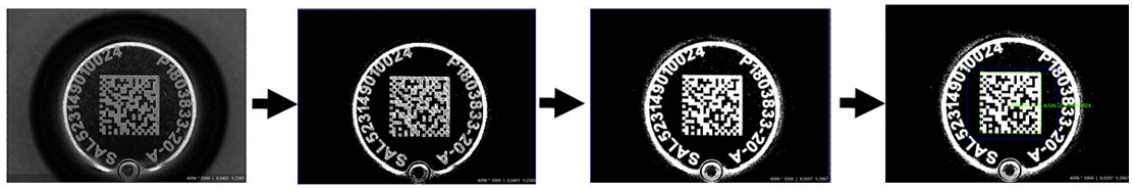


图 13-381 该应用的效果示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

参数	描述
形态学类型	设置形态学处理的类型，包括**膨胀**、**腐蚀**、**开**和**闭**，分别对应上文 *模块原理* 中提及膨胀、腐蚀、开处理和闭处理 效果比对示例，请参见下文 *图 13-382*
形态学形状	具体指滤波核的形状，即上文 *模块原理* 中提及的结构元形态，包括矩形、椭圆形和十字形。 运算结果图像轮廓会和形态学形状比较相似

参数	描述
迭代次数	指一次形态学处理过程中对原图像进行的操作次数，迭代次数越多，效果越明显，但不宜过多 ● 膨胀迭代次数过多容易导致黑色区域过小 ● 腐蚀迭代次数过多容易导致黑色区域过大
核宽度/高度	结构元的宽度和高度，加大该值会使形态学处理的效果更佳明显



图 13-382 不同类型形态学处理的效果对比

模块结果

*形态学处理*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.10.5 图像增强

*图像增强*模块可对退化的图像特征，如边缘、轮廓、对比度等进行预处理，以改善图像的视觉效果，突出图像的整体或局部特征。处理后的图像中的“有用”信息更突出，有利于后序的图像分析处理。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的图像增强算法包括锐化、对比度调节、Gamma 调节和亮度校正。

锐化

计算出图像中每一点与周围领域均值的差异。然后对这种差异使用锐化强度进行放大。最后将这种放大的差异添加至原图上，得到锐化之后的图像。

此处提及的“周围领域”大小，可通过下文*参数配置*中的**锐化核**调整。

对比度

假定图像中的所有像素点的灰度值都在某个均值的附近波动。将波动的幅度变大，图像特征的明暗差异也就更大。明暗差异增大后，更容易区分图像中的信息。

此处提及的“波动”的幅度，可通过下文*参数配置*中的**对比度系数**调整。

Gamma

将图像的像素值归一化到单位 1，并使用伽马曲线对图像进行映射。再反归一化到原始的像素范围。

此处提及的“伽马曲线”的曲度，可通过下文*参数配置*中提及的 **Gamma** 定义。该参数取不同值时的曲线见下图。

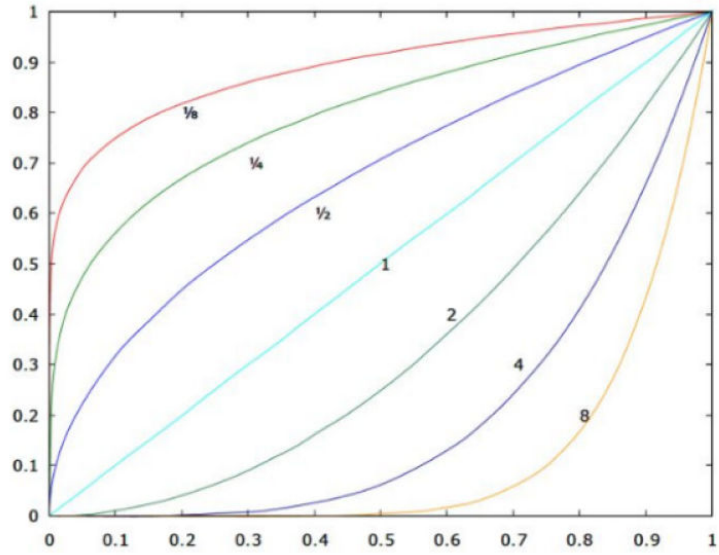


图 13-383 Gamma 不同取值对应的伽马曲线

亮度校正

通过线性变换对图像中的每一点的灰度值进行计算。具体计算公式为：

$$dst[i] = src[i] * gain + offset$$

图 13-384 亮度校正计算公式

该公式中：

- src[i]表示输入图像的当前灰度值
- dst[i]表示输出图像的当前灰度值，其值被界定在[0,255]范围内
- gain 表示亮度校正增益
- offset 表示亮度校正补偿

使用方法

以下对该模块的使用方法分为前后序模块和应用示例这两个方面进行介绍。

前后序模块

在流程中的，*图像增强*模块对前后序模块无特定要求。

应用示例

以下示例通过*图像增强*模块增强原图像对比度，使圆环外的细线圆更清晰。之后将处理后的图片传给 *圆查找*，即可稳定检测细线圆。

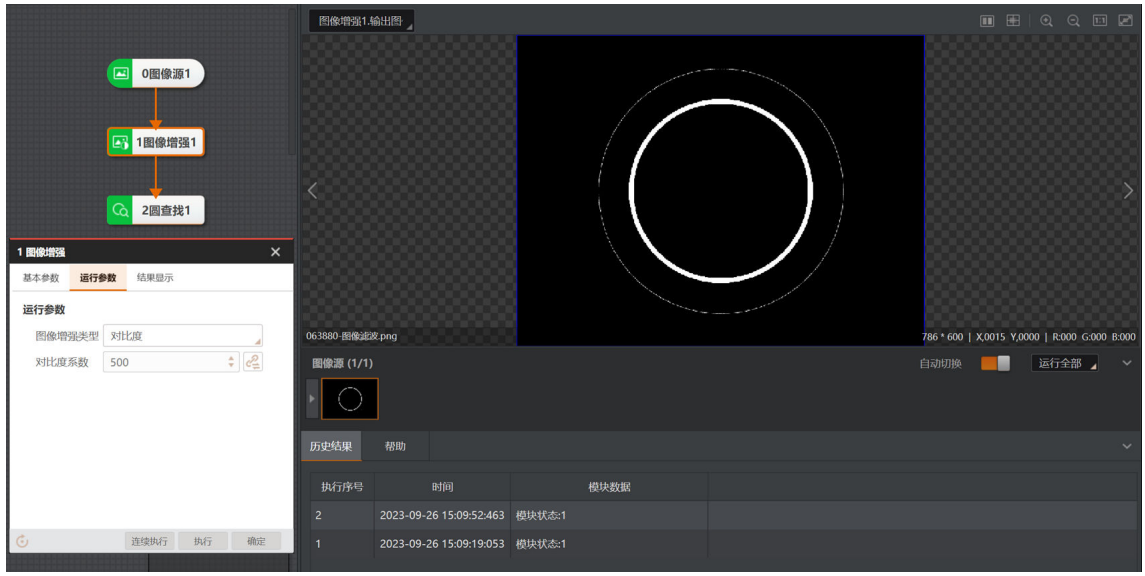
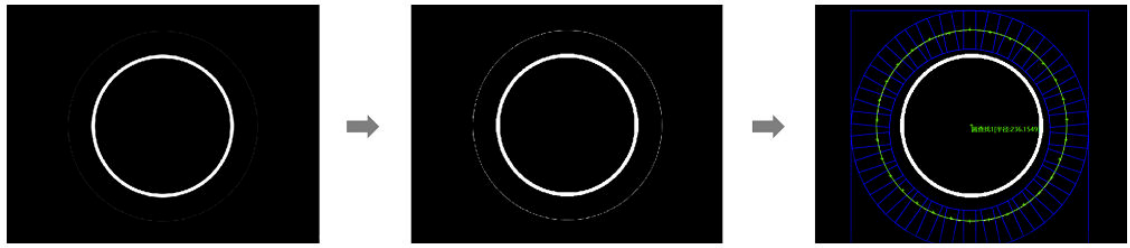


图 13-385 检测外圈细线圆



原图像（外圈圆不明显）　图像增强（提高对比度）　【圆查找】检测外圈圆

图 13-386 效果展示

参数配置

该模块的运行参数为**图像增强类型**。不同增强类型的效果对比请参考下图。

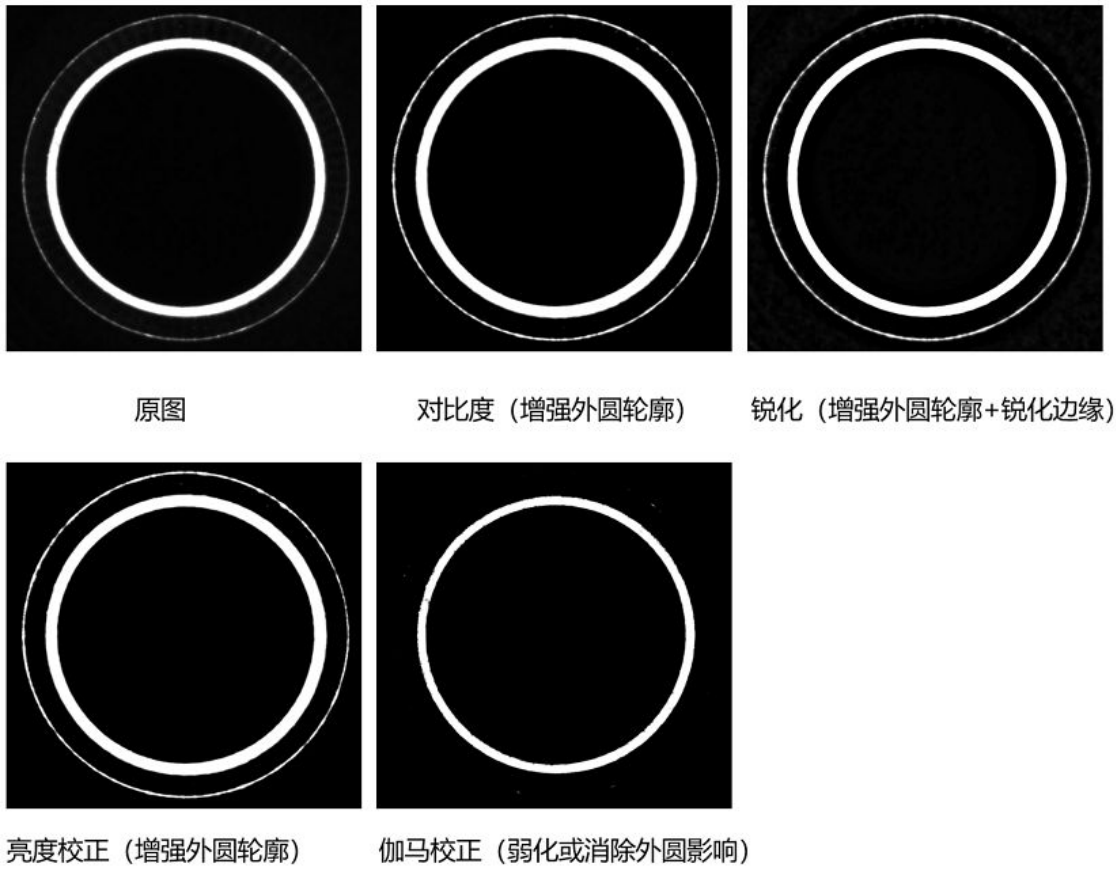


图 13-387 图像增强效果对比

锐化

图像锐化可突出图像上目标物体的边缘轮廓或某些线性目标要素的特征。

锐化强度

锐化系数，100 表示系数为 1，0 表示不进行锐化处理。该值越大，锐化效果越明显。

锐化核大小

对应上文*模块原理*中提及的目标像素点 ***周围领域*** 的大小。取值范围 1~51。

对比度

图像对比度即图像颜色和亮度差异。对比度越大，图像中目标对象与周围差异性越大。

对比度系数

控制对比度的调节系数，100 表示不进行调节；大于 100 对比度增加，小于 100 对比度降低 。

Gamma

对图像进行非线性色调编辑。检出图像信号中的深色部分和浅色部分，并使两者比例增大，从而提高图像对比度效果。

Gamma

Gamma 值在 0~1 之间，图像暗处亮度提升。Gamma 值在 1~4 之间，图像暗处亮度下降 。

亮度校正

如果图像曝光时间过长/不足导致图片过亮或过暗，可进行亮度矫正。

增益

提高图像画面整体像素亮度。默认值为 0，取值范围 0~100。

亮度校正补偿

提高或降低画面整体像素亮度。默认值为 0，取值范围-255~255。

模块结果

*图像增强*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.10.6 阴影校正

*阴影校正*模块可对光照不均匀图像进行光照校正，弱化光照阴影的影响。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

阴影校正是一种计算“输入和背景的差值”图像的算法，同时可选择屏蔽指定方向的纹理背景。下图示例中屏蔽了 Y 方向的栅线纹理。

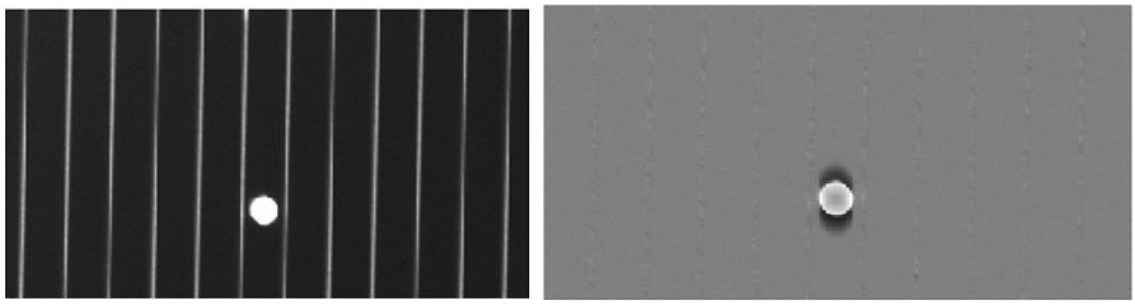


图 13-388 阴影校正效果

该模块的算法工作流程如下：

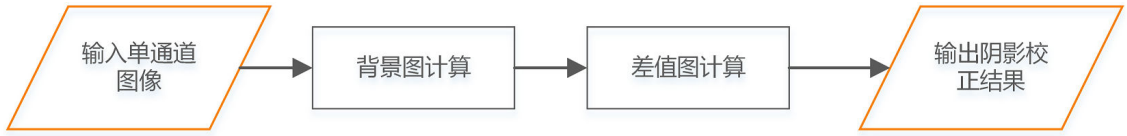


图 13-389 算法工作流程

背景图计算

采用均值滤波方式计算背景图。

- 当方向为 X 方向时，滤波窗口尺寸为(kernel_size,1)。
- 当方向为 Y 方向时，滤波窗口尺寸为(1,kernel_size)。
- 当方向为 XY 方向时，滤波窗口尺寸为(kernel_size, kernel_size)。

此处的变量 kernel_size 表示卷积核尺寸，可通过下文*参数配置*中提及的**滤波尺寸**配置。

差值图计算

- 当目标图像灰度值大于背景灰度值，需降低亮噪声。此时差值图像逐像素计算公式如下：

$$Dst(i,j) = gain * (src(i,j) - back(i,j) - noise) + offset$$

图 13-390 计算公式（目标图像灰度值大于背景）

- 当目标图像灰度值小于背景灰度值，需降低暗噪声。此时差值图像逐像素计算公式如下：

$$Dst(i,j) = gain * (src(i,j) - back(i,j) + noise) + offset$$

图 13-391 计算公式（目标图像灰度值小于背景）

以上两个公式中：

- Dst(i,j)表示结果图像第 i 行 j 列的像素点的灰度值。
- src(i,j)表示原图像第 i 行 j 列像素点的灰度值。

- back(i,j)表示背景图第 i 行 j 列像素点的灰度值。
- gain 表示增益；noise 表示噪声；offset 表示亮度校正补偿。

此处的变量 gain、noise 和 offset，分别可通过下文*参数配置*中提及的**增益/噪声**和**亮度校正补偿**配置。

使用方法

*阴影校正*模块对其在流程中的前后序模块无特定要求，符合图像处理逻辑即可。
以下应用示例中，通过*阴影校正*将光照不均匀且带有字符的图片进行阴影校正，再将校正后图像输出给 *DL 字符识别* 识别字符。

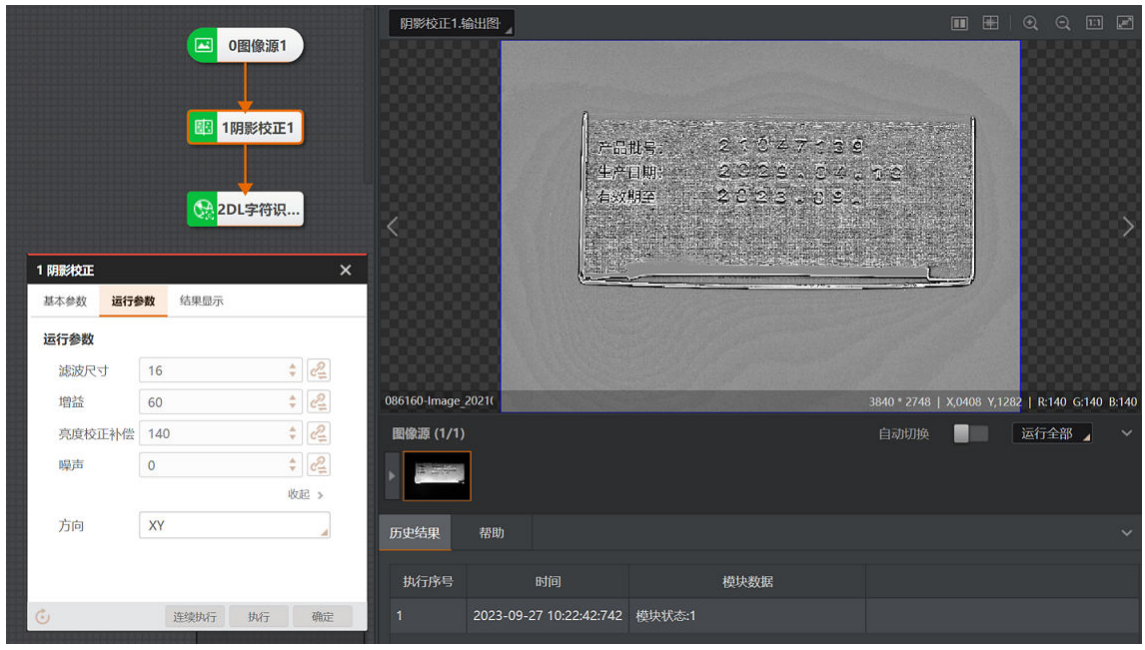


图 13-392 应用示例：字符检测

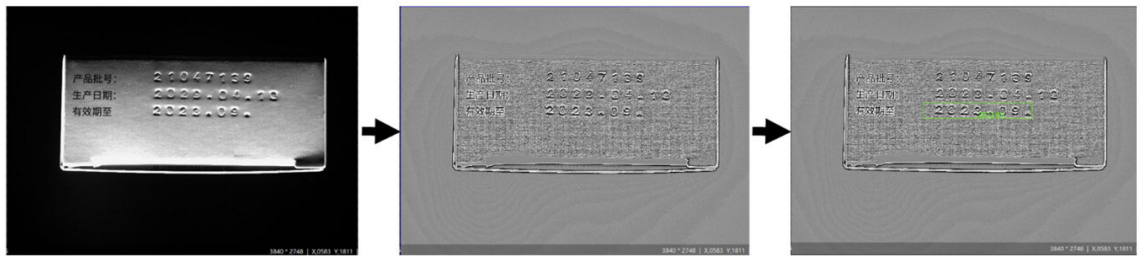


图 13-393 字符图像阴影校正效果

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

表 13-27 运行参数

参数	描述
滤波尺寸	卷积核的大小，取值范围 1~50。卷积核越大，滤波效果越平滑。
增益	增强前景目标，取值范围 0~100。
亮度校正补偿	对图像整体灰度水平进行调整，取值越高图片整体灰度越高，取值范围 0~255。
噪声	干扰像素灰度阈值，取值范围 0~255。灰度值低于该值的像素点，其灰度值将被置为 0。
方向	包括“X”、“Y”以及“XY”3 种方向，表示卷积核的方向。

模块结果

*阴影校正*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.10.7 图像组合

*图像组合*模块主要用于对图像进行一定顺序的复杂处理，使之符合检测、定位、识别和清晰度评估等业务场景的图像要求。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*图像组合*将*形态学处理*、*图像二值化*、*图像滤波*、*图像增强*和*阴影校正*这 5 种图像处理模块任意组合，对图像进行预处理并输出处理后的图像。

使用方法

从*图像源*模块订阅需要进行处理的图像后，按顺序启用适当的图像组合类型，并设定相应的运行参数，并单击*执行*。



图 13-394 图像组合配置

参数配置

- 基本参数详情，请参见 *基本参数* 。
- 运行参数详情，请参见 *图像二值化* 、 *形态学处理* 、 *图像滤波* 、 *图像增强* 、 *阴影校正* 等相关模块。
- 结果显示参数详情，请参见 *结果显示* 。

模块结果

*图像组合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

中间图像

中间图像数据

image 型，代表图像组合过程中产生的中间图像所对应的二进制数据。

中间图像宽度

int 型，代表中间图像的宽度，一般单位为像素。

中间图像高度

int 型，代表中间图像的高度，一般单位为像素。

中间图像像素格式

int 型，代表中间图像的像素格式，17301505 代表黑白图片， 35127316 代表彩色图片。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

13.10.8 图像归一化

*图像归一化*可对图像的灰度分布按照规则进行调整，以增强图像局部细节。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块包含 *直方图均衡化* 、 *直方图归一化* 和 *均值标准差归一化* 三类算法。

直方图均衡化

直方图均衡化假设每一像素值在直方图上分布概率相同, 则每一灰度级经过均衡化后的计算公式如下。其中：

- k=0,1,2,..., L-1，L 表示图像中的灰度级总数。
- n_j 表示当前灰度级的像素总数。

$$S_k = \sum_{j=0}^{k} \frac{n_j}{n}$$

图 13-395 直方图均衡化计算公式

直方图归一化

直方图归一化的算法工作流程如下所示。

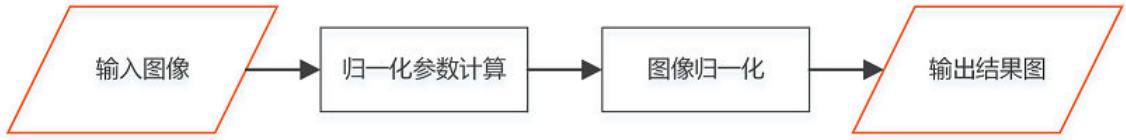


图 13-396 算法工作流程

1. 计算归一化参数。具体分为如下两个子步骤。
   a. 根据预设的灰阶左端像素数量占比（left_scale）和右端像素数量占比（right_scale），计算左端像素级（LeftPos）和右端像素级（RightPos），如下图所示。

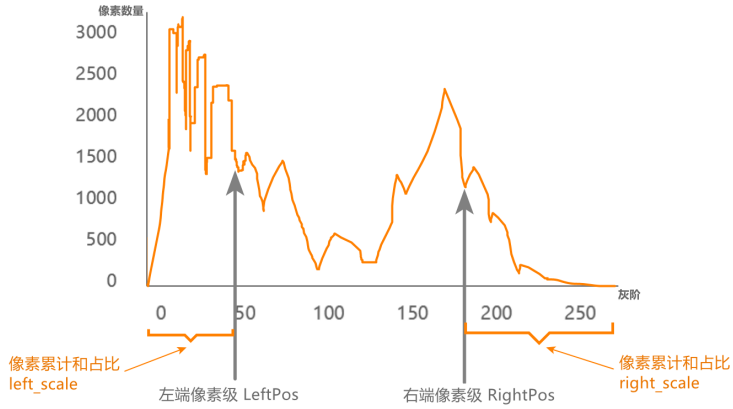


图 13-397 左右端像素级计算

   b. 根据计算的左右端像素值，分别按如下公式计算图像归一化的参数 scale（归一化系数）和 offset（归一化偏移值）。公式中 MinPos 和 MaxPos 分别为预设的灰度值范围的最小值和最大值。
      - scale=(MinPos-MaxPos)/(LeftPos-RightPos)
      - offset= MaxPos- RightPos*scale
2. 进行图像归一化，具体为：
   将计算出的 scale 和 offset 代入如下公式，计算目标图像像素值。其中 DstGray(i,j)表示输出图像第 i 行 j 列像素值，SrcGray(i,j)表示输入图像第 i 行 j 列像素值。
   DstGray(i,j)= SrcGray(i,j)*scale+offset

均值标准差归一化

均值标准差归一化的算法工作流程和 *直方图归一化* 的基本相同，区别在于归一化参数的计算方式。

1. 计算归一化参数 sale（归一化系数）和 offset（归一化偏移值），计算公式分别如下。
   - scale=SetStd/SrcStd
   - offset=SetMean-scale*SrcMean

   公式中，SetStd 和 SetMean 分别表示预设的标准差和均值；SrcStd 和 SrcMean 分别表示输入图像像素的标准差和均值。
2. 进行图像归一化，具体为：
   将计算出的 scale 和 offset 代入如下公式，计算目标图像像素值。公式中，DstGray(i,j)表示输出图像第 i 行 j 列像素值，SrcGray(i,j)表示输入图像第 i 行 j 列像素值。
   DstGray(i,j)= SrcGray(i,j)*scale+offset

使用方法

该模块对前后序模块无特定要求，一般前序模块为*图像源*。

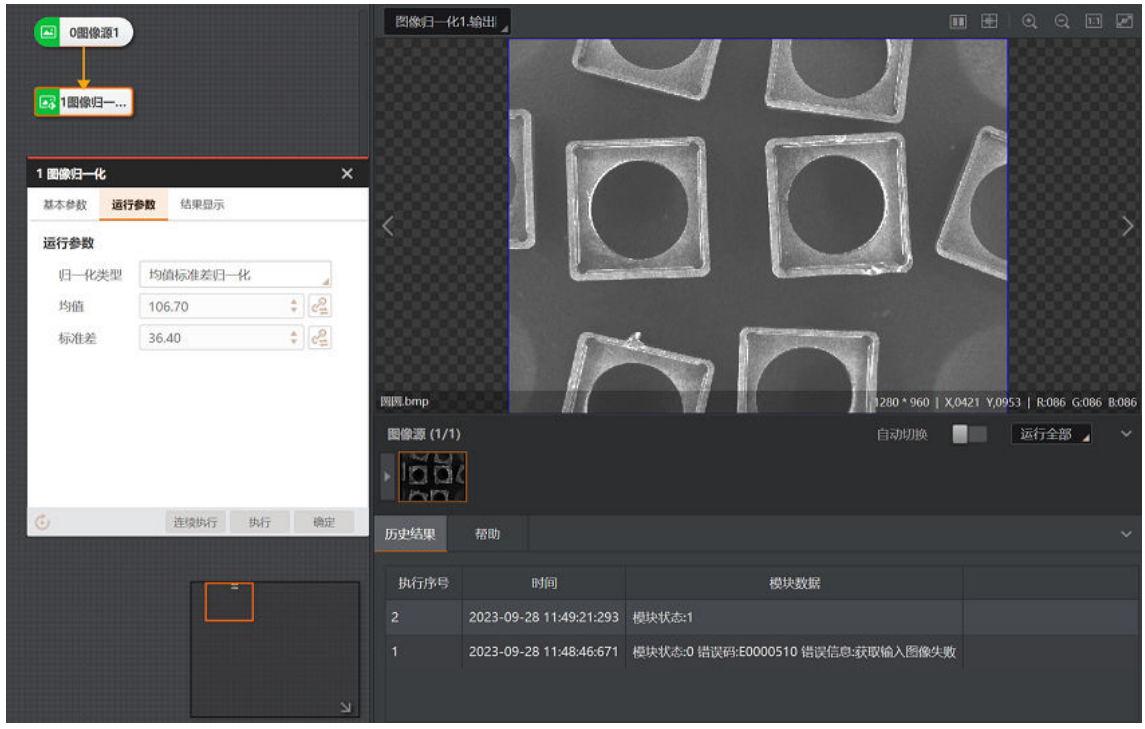


图 13-398 示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

直方图均衡化

通过灰度直方图将图像灰度调整均匀。该方法可增强局部细节和对比度。具体原理见上文*模块原理*中的 *直方图均衡化* 。



图 13-399 原图



图 13-400 直方图均衡化效果

直方图归一化

在指定直方图范围内进行图像归一化。该方法可增强图像局部细节和对比度，相比**直方图均衡化**调参更灵活。具体原理见上文*模块原理*中的、 *直方图归一化* 。

左端比例

即上文*模块原理*中提及的灰阶左端像素数量占比（left_scale）。

右端比例

即上文*模块原理*中提及的灰阶右端像素数量占比（right_scale）。

灰度值范围

对应除去**左端比例**和**右端比例**的直方图灰阶范围。



图 13-401 原图



图 13-402 直方图归一化效果

均值标准差归一化

指定直方图标准差和均值进行图像归一化，增强局部细节和对比度。具体原理见上文*模块原理*中的 *均值标准差归一化* 。

均值

图像数据的平均值，可以影响均衡标准差归一化结果。 目标均值越大，灰度转换系数越大。

标准差

图像数据的标准差，可以影响均衡标准差归一化结果。 目标均值越大，灰度变换偏移系数越大。



图 13-403 原图



图 13-404 均值标准差归一化效果

模块结果

*图像归一化*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

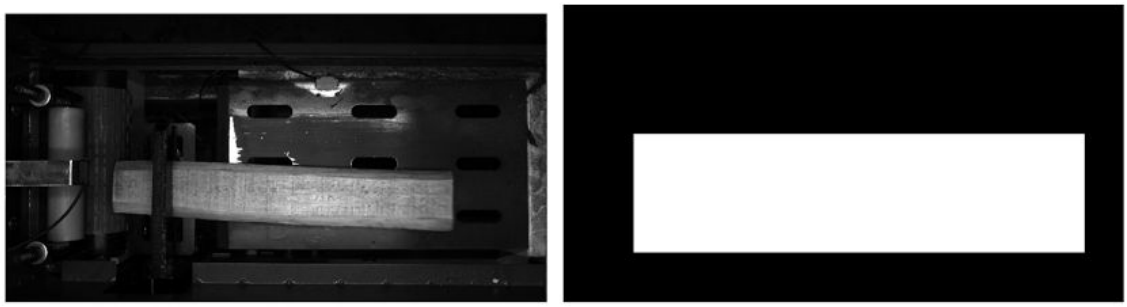
13.10.9 图像运算

图像运算指对输入图像进行常见的逻辑和数据运算。逻辑运算包括与、或、异或等。数据运算包括加、减、乘、除、取最大、取最小等。图像运算在图像处理领域具有广泛的运用。例如，图像减操作可以检测同一场景或物体的不同图片之间的误差，图像加操作可降低图像中的随机噪声。

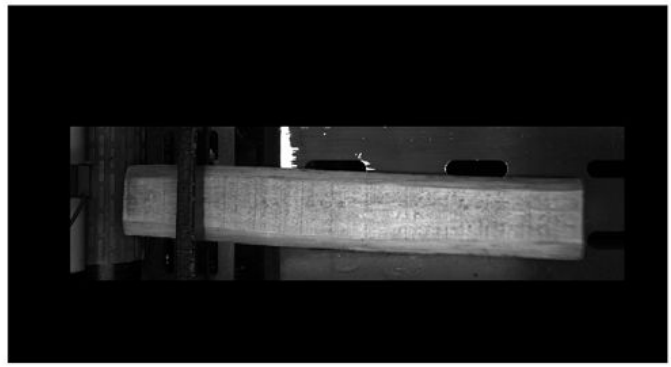
模块原理

针对两张大小和分辨率相同的输入图像，该模块可对相同像素位置的两个灰度值进行算术运算，全图运算后得到新的图像。

以下示例中，该模块对输入的两张图像（一张灰度图，一张掩膜图）进行**两者取最小**运算，得到一张只保留感兴趣区域的灰度图。



(a)输入灰度图　　(b)输入掩膜图



(c)取最小值计算后结果

图 13-405 “两者取最小”效果示例

使用限制

表 13-28 使用限制

限制类型	描述
运算范围	该模块仅支持对图像进行全图运算，即无法指定 ROI 处理 ROI 范围内的图像。
图像类型	• V4.3.1 之前版本：该模块仅可输入两张灰度图。 • V4.3.1 起：该模块可输入的图像类型取决于**运算类型**参数取值（具体参见下文 *表 13-30*）。 　○ 如果将该参数设置为**两者最小值**，则可输入一张 Mono8 灰度图和一张 RGB24 彩色图。 　○ 如果将该参数设置为其他取值，则仅可输入两张灰度图。

使用方法

前后序模块

在流程中，该模块的：

- 前序模块一般为两个 ***图像源***，提供两张图像输入。
- 后序模块无特定要求。

应用示例

以下应用示例将输入的两张图像进行**加运算**，使外圆与内圆环更清晰。

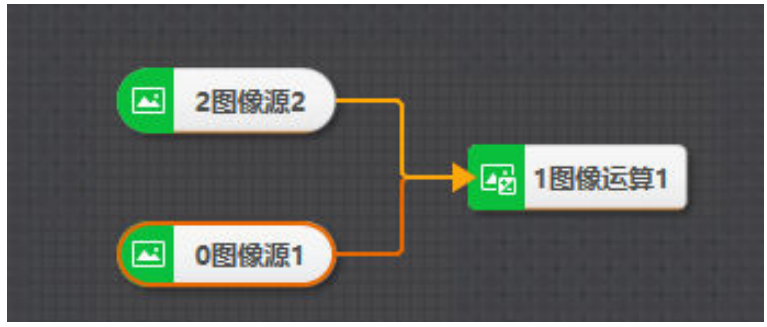


图 13-406 流程示例

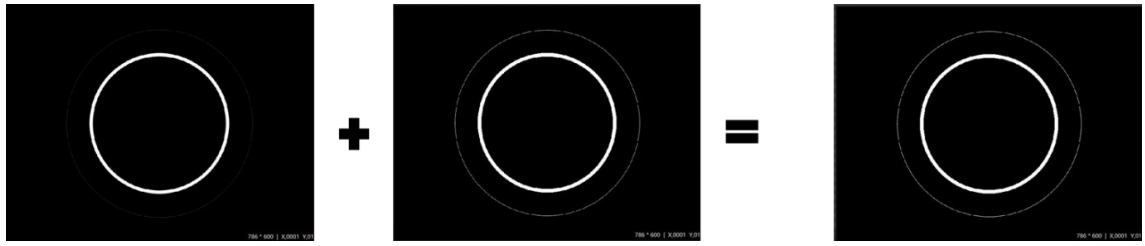


图 13-407 加运算效果

参数配置

以下仅介绍该模块的**图像输入**参数和***图像运算类型***参数。其他参数详情请参见 ***基本参数*** 。

表 13-29 图像输入参数

参数	描述
输入源 1/2	订阅图像源

参数	描述
	⚠注意 请确保**输入源 1** 和**输入源 2** 所设置的图片大小一致，即分辨率一致。否则将导致模块运行失败。
图像权重 1/2	当进行灰度值乘法计算时，**输入源 1** 与**输入源 2** 的灰度值各自占的权重系数，分别对应下表“运算类型”列中的 k1 和 k2。仅在选择对应的运算类型时显示
图像补偿 1/2	对应下表“运算类型”列中的 C。仅在选择对应的运算类型时显示

表 13-30 图像运算类型

序号	运算类型	描述
1	图像加	输入两张图逐像素相加得到目标图像。若运算结果超过 255，则目标图像上对应像素的灰度值为 255
2	图像减	将**输入源 1** 各像素的灰度值减去**输入源 2** 对应像素的灰度值，得到目标图像。若运算结果小于 0，则目标图像上对应像素的灰度值为 0
3	图像绝对差	输入两张图逐像素差值取绝对值得到目标图
4	两者最大值	输入两张图逐像素取最大值得到目标图
5	两者最小值	输入两张图逐像素取最小值得到目标图
6	两者均值	输入两张图像逐像素计算均值得到目标图
7	图像与	对输入两张图像逐像素进行“与运算”，得到目标图像。该运算的规则如下（用∧表示该运算）： • 图像 1 灰度值非 0∧图像 2 灰度值 0=目标图像灰度值 255 • 图像 1 灰度值 0∧图像 2 灰度值非 0=目标图像灰度值 255 • 图像 1 灰度值非 0∧图像 2 灰度值非 0=目标图像灰度值 255 • 图像 1 灰度值 0∧图像 2 灰度值 0=目标图像灰度值 0

序号	运算类型	描述
8	图像或	对输入两张图像逐像素进行“或运算”，该运算的规则如下（用∨表示该运算）： • 图像 1 灰度值非 0∨图像 2 灰度值 0=目标图像灰度值 0 • 图像 1 灰度值 0∨图像 2 灰度值非 0=目标图像灰度值 0 • 图像 1 灰度值非 0∨图像 2 灰度值非 0=目标图像灰度值 255 • 图像 1 灰度值 0∨图像 2 灰度值 0=目标图像灰度值 0
9	图像异或	对输入两张图像逐像素进行“异或运算”，该运算的规则如下： • 若两张图像灰度不一致，运算得到的目标图像灰度值为 0 • 若两张图像灰度一致，运算得到的目标图像灰度值为 255
10	k1*I1+k2*I2+C	逐像素对图像 1 乘以 K1，图像 2 乘以 K2 后相加再加常数 C 得到目标图像
11	k1*I1+C	逐像素对输入图像 1 乘以 K1 再加常数 C 得到目标图像
12	I1+C	逐像素对输入图像 1 加常数 C 得到目标图像
13	I1-C	逐像素对输入图像 1 减常数 C 得到目标图像
14	C-I1	逐像素使用常数 C 减图像 1 得到目标图像
15	两幅图像乘	输入两张图逐像素相乘得到目标图像
16	两幅图像除	输入两张图逐像素相除得到目标图像
17	I1/C	逐像素对输入图像 1 除以常数 C 得到目标图像
18	C/I1	逐像素使用常数 C 除以图像 1 得到目标图像
19	I1&&C	逐像素对输入图像 1 和常数 C 与运算，结果大于 0 为 255，否则为 0，得到目标图像
20	I1||C	逐像素对输入图像 1 和常数 C 或运算，结果大于 0 为 255，否则为 0，得到目标图像
21	I1^C	逐像素对输入图像 1 和常数 C 异或运算，结果大于 0 为 255，否则为 0，得到目标图像

序号	运算类型	描述
22	图像和常数最大值	逐像素对输入图像 1 和常数 C 取最大值得到目标图像
23	图像和常数最小值	逐像素对输入图像 1 和常数 C 取最小值得到目标图像
24	图像和常数均值	逐像素对输入图像 1 和常数 C 取最均值得到目标图像
25	|I1-C|	逐像素对输入图像 1 和常数 C 差值取绝对值得到目标图像
26	两幅图像与非	输入两张图像逐像素与运算，结果大于 0 为 0，否则为 255，得到目标图像
27	两幅图像或非	输入两张图像逐像素或运算，结果大于 0 为 0，否则为 255，得到目标图像
28	!(I1&&C)	逐像素对输入图像 1 和常数 C 与运算，结果大于 0 为 0，否则为 255，得到目标图像
29	!(I1||C)	逐像素对输入图像 1 和常数 C 或运算，结果大于 0 为 0，否则为 255，得到目标图像
30	(I1-(I2+C))>=0?255:0	逐像素对图像 2 加常数 C，若小于图像 1 像素值，则目标像素为 255，否则为 0
31	(I1-(I2+C))>=0?0:255	逐像素对图像 2 加常数 C，若小于图像 1 像素值，则目标像素为 0，否则为 255

模块结果

*图像运算*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

13.10.10 图像修正

*图像修正*模块可搭配*轮廓匹配/ 灰度匹配*和*位置修正*模块使用，对产线上摆放位姿存在变动的物体在图像上进行位姿纠偏。

使用场景

产线上被传送的物体可能发生位置偏移，如下图所示。



图 13-408 场景示例

拍摄到的物体在图像中发生位置偏移时，需要对物体图像进行纠偏，保证后续定位处理的准确性。

使用方法

*图像修正*模块的前序模块为 *图像源* 和 *位置修正*，后者为*图像修正*输入位置修正信息。
下图示例中：

- *轮廓匹配* 定位样本，并将位置信息输出至*位置修正*。

  说明

  *轮廓匹配*中建议通过标准图像创建模板，以保证后序*位置修正*输出准确的位置修正信息。
- *位置修正*输出位置修正信息至*图像修正*。
- *图像修正*根据位置修正信息将图像中的物体进行位置修正。

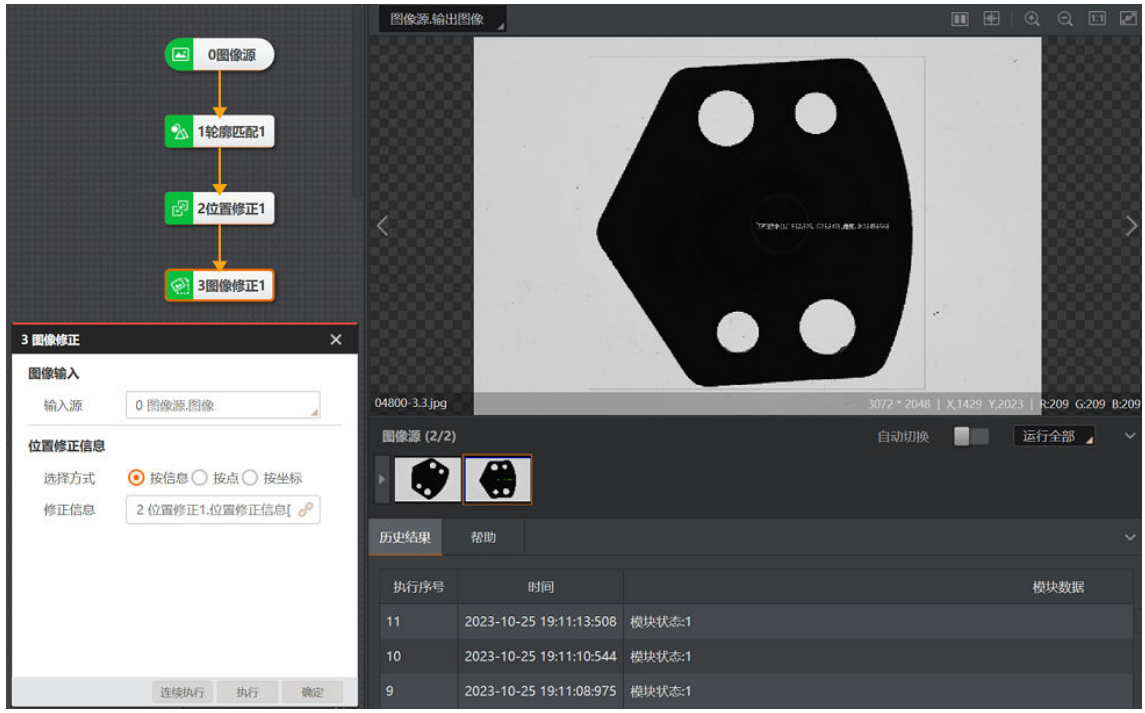
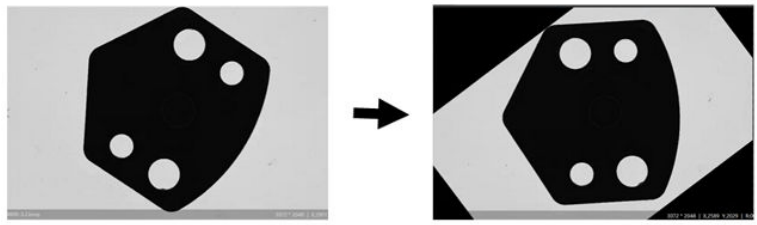


图 13-409 应用示例



修正样本



修正前　　修正后

图 13-410 图像修正效果示例

参数配置

表 13-31 参数详情

参数	描述
按信息	通常直接绑定*位置修正*中的修正信息
按点	订阅前序模块输入的基准点、基准角度、运行点、运行角度
按坐标	基准点与运行点的坐标都可自定义。当需要对图像进行旋转时可以选择该方式，例如需将一张分辨率为 2000*1000 的图像顺时针旋转 90°时，可参照下图设置



图 13-411 按坐标修正配置示例

模块结果

*图像修正*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.10.11 仿射变换

通过*仿射变换*模块可对图像进行剪裁缩放、镜像翻转和平移处理。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*仿射变换*模块的仿射算法类型包括 *剪裁缩放* 、 *平移* 和 *镜像* 。

剪裁缩放

剪裁缩放算法工作流程如下：

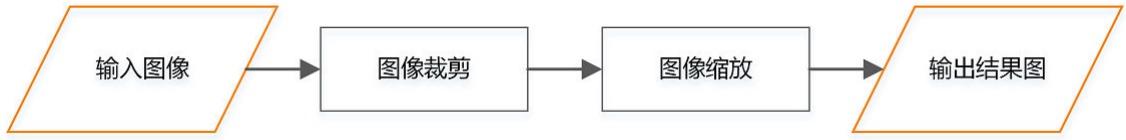


图 13-412 算法工作流程

1. 进行图像剪裁。即先计算 ROI 内目标物体的最小外接矩形，再将外接矩内的图像裁剪出来，如下图所示。
   裁剪后图像的宽高和最小外接矩形宽高对应。若外接矩形一部分在图像外部，则需要根据预设的填充方式和填充值进行填充。具体填充方式见下文的 *参数配置* 。

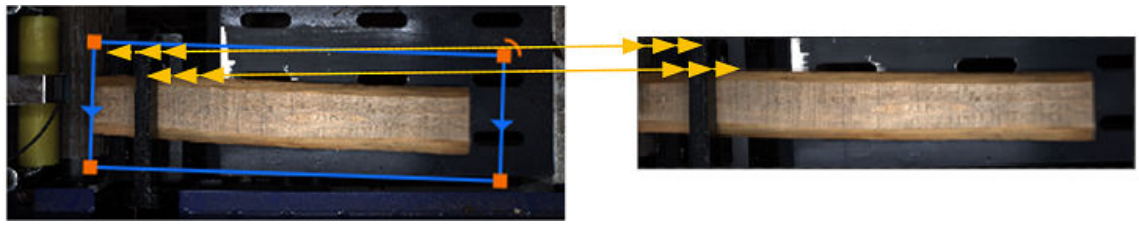


图 13-413 像素对应示意

2. 进行图像缩放。即缩放需要输出的目标图像尺寸，计算公式如下。

其中 CropH、CropW 表示裁剪后图像的高度和宽度；DstH、DstW 表示输出目标图像的高度和宽度；scale 表示设置裁剪缩放的尺度；aspect 表示设置的宽高比。

$$DstH = CropH * scale$$
$$DstW = CropW * scale * aspect$$

图 13-414 图像缩放计算公式

平移

平移包括 X 和 Y 方向平移，实现原理为将像素矩阵整体向 X 或 Y 方向平移，移动区域使用 0 填充。

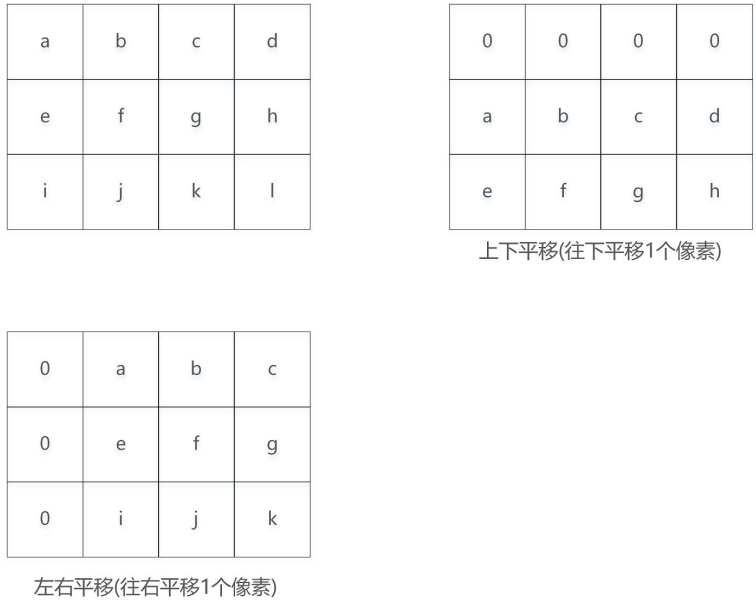


图 13-415 图像平移原理

镜像

包括水平镜像、垂直镜像、水平垂直镜像。

说明

水平垂直镜像表示先水平镜像后再垂直镜像，故本文仅介绍水平镜像和垂直镜像。

水平镜像指将输入图像的像素矩阵左右翻转；垂直镜像指将像素矩阵上下翻转，翻转原理示意图如下。

a	b	c	d
e	f	g	h
i	j	k	l

d	c	b	a
h	g	f	e
l	k	j	i

左右翻转

i	j	k	l
e	f	g	h
a	b	b	d

上下翻转

图 13-416 镜像原理示意

使用方法

*仿射变换*在流程中的前序模块为 *图像源* 。执行一次流程后即可对*图像源*模块输入的图像进行仿射变换。

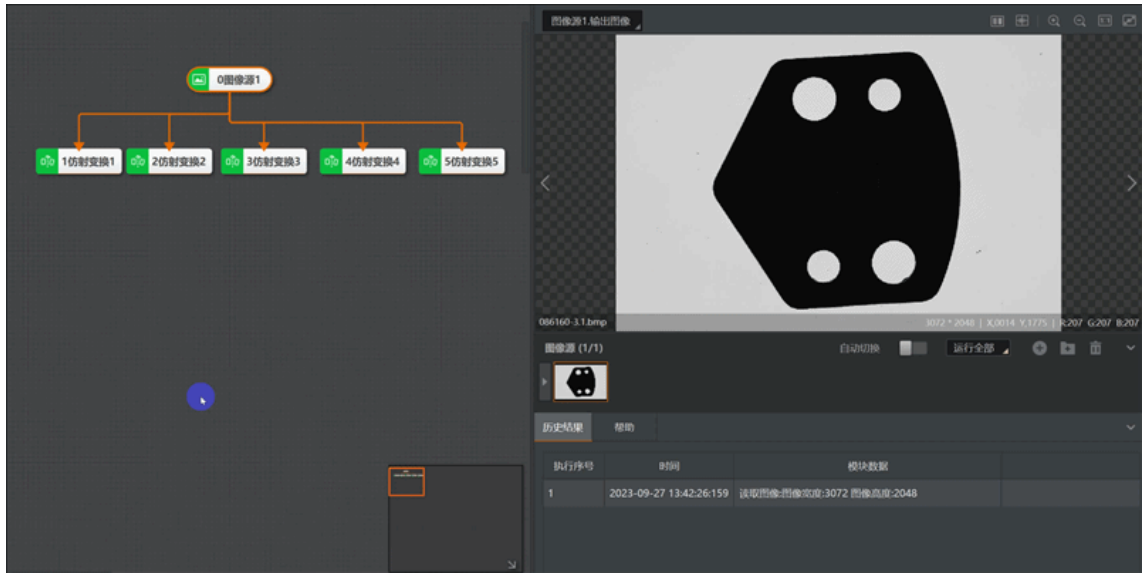


图 13-417 效果示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

可配置的运行参数随**仿射变换类型**选择而变化，具体见如下表格。

说明

无论选择哪种**仿射变换类型**，均可启用/关闭**锁定输出尺寸**。启用后，可设置宽高限制输出图像的尺寸。

表 13-32 仿射变换类型为剪裁缩放时

参数	描述
尺度	图像缩放系数
宽高比	图像宽度与高度的比值
插值方式	可选**最邻近 和双线性** • **最邻近**：选取最邻近的像素点进行插值计算 • **双线性**：先对 X 轴左右最邻近的像素点进行差值计算，然后对 Y 轴上下最邻近的像素点进行差值计算
填充方式	旋转矩阵超出图像边界部分的灰度值填充方式，可选**常数**或**临近复制**，**临近复制**表示使用相邻像素点的灰度值进行填充
填充值	**填充方式**设置为**常数**时，通过该参数设置具体填充的灰度值

表 13-33 仿射变换类型为镜像时

参数	描述
镜像方向	可选择**水平**、**垂直**或**水平垂直**

表 13-34 仿射变换类型为平移时

参数	描述
X 移动距离	取值为正数时图像向左平移，负数时图片向右平移
Y 移动距离	取值为正数时图片向上平移，负数时图片向下平移

模块结果

*仿射变换*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.10.12 逆仿射变换

*逆仿射变换*模块可将输入图像映射到特定底图的矩形 ROI 之中。该模块用于将*仿射变换*剪裁之后的图像放回至原图对应位置。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*逆仿射变换*模块的算法工作流程主要包括如下两步。

1. 根据逆仿射输入的图像的区域（相当于一个矩形）与底图 ROI（另一个矩形）可以计算从输入图像映射到 ROI 区域的变换尺度、角度以及位置偏移。
2. 通过这些变换关系可以描述从输入图像中的点到 ROI 中的点的映射关系，即 $P^{ROI}=HP^{I}$。其中 P^{ROI} 是 ROI 中的点的坐标，P^{I} 是输入图像中点的坐标，H 即两者的映射关系。
   ROI 内各点的灰度值与输入图像中对应点的灰度值相同，即 ROI 中各点坐标对应的灰度值与输入图像中对应点坐标的灰度值相同。根据该关系计算 ROI 区域的所有坐标的灰度值，便可实现将输入图像映射到底图 ROI 中。

使用方法

*逆仿射变换*模块的前序模块为 *仿射变换* 。两者搭配可将*仿射变换*裁剪的图像放到空白图片上或原图对应位置，以便后序模块更好地处理图像。

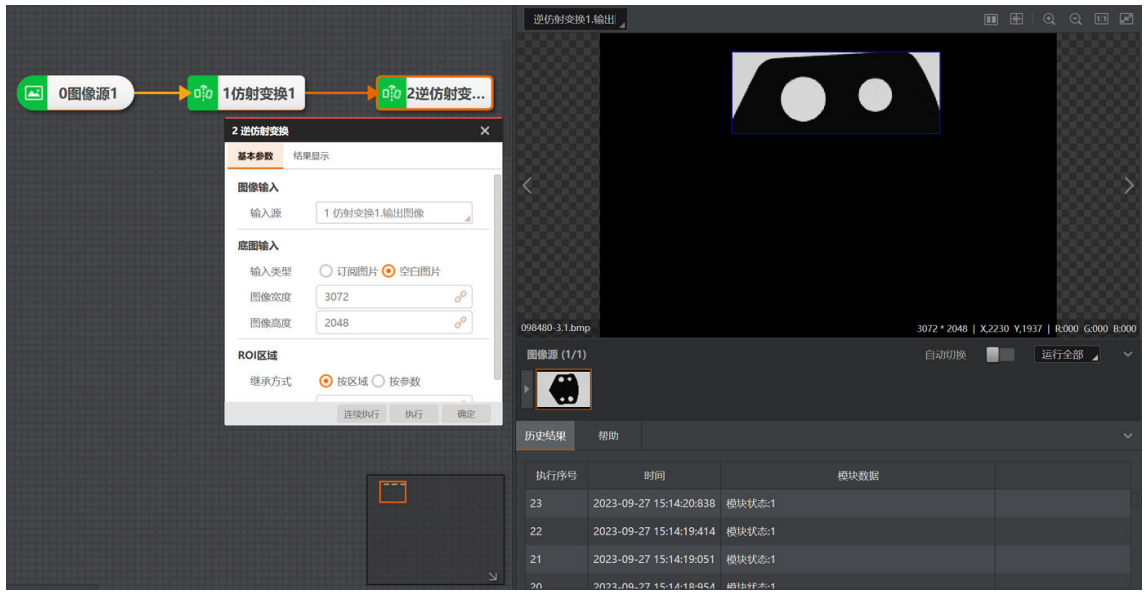


图 13-418 逆仿射变换应用示例

如下效果示例中：

- 左图中设置了仿射变换的 ROI，根据预设的仿射变换参数便可得到仿射变换结果，如中图所示。
- 使用逆仿射变换可以将仿射变换结果映射到原图（与左图基本一致）或空白图像中（如右图所示）。

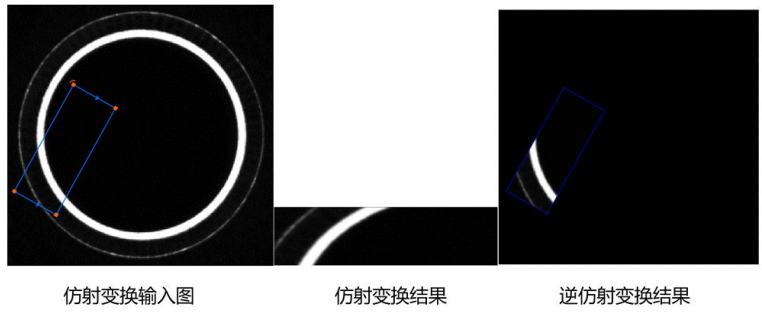


图 13-419 逆仿射变换效果示例

参数配置

以下仅介绍该模块的基本参数详情。

说明

结果显示参数详情，请参见 *结果显示* 。

表 13-35 基本参数

参数	描述
图像输入	仿射变换输出的图像，逆仿射变换待被填充的图像，是小图
ROI 区域	ROI 区域与仿射变换的 ROI 一致，位于底图中，是叠加图像的区域。订阅方式可选按区域或按参数两种方式
底图输入	仿射变换输入的图像或者是同样宽高的空白图像，是大图 **说明** 宽高一致时便于观察，实际操作中可以选择其他宽高的图像。

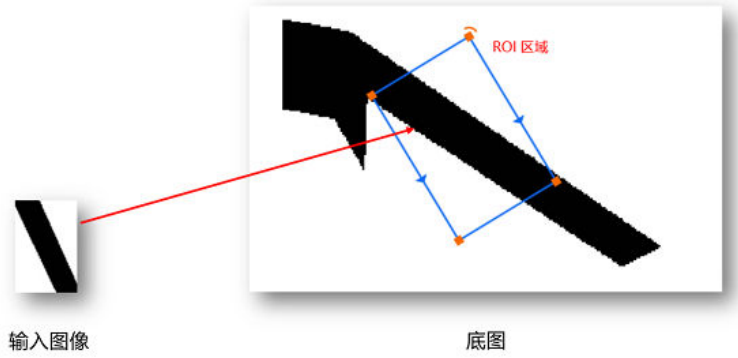


图 13-420 输入图像与底图

模块结果

*逆仿射变换*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.10.13 图像缩放

*图像缩放*模块可将输入图像缩放到指定尺寸，使其适应不同大小的特征模式，提高特征匹配的准确性。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的算法工作流程如下：

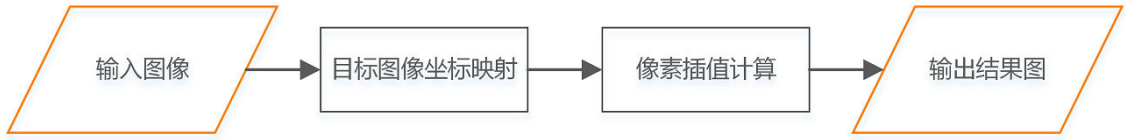


图 13-421 算法工作流程

目标图像坐标映射

输入图像坐标映射到目标图像上的计算公式如下图所示。

前两个公式中，DstW、DstH 表示目标图像的宽度和高度；srcW、srcH 表示输入图像的宽度和高度；两者均已知，则可计算 scale1 和 scale2 的值，表示输入图像的缩放比例。

计算得出 scale1 和 scale2 的值后，再根据后两个公式可得到“目标图像上像素”到“输入图像”的转换关系，最后进行插值即可。

$$DstW = scale1 * srcW$$

$$DstH = scale2 * srcH$$

$$Dstcol = srcCol/scale1$$

$$DstRow = srcRow/scale2$$

图 13-422 计算公式

像素插值计算

插值计算方式包括最近邻、双线性和双三次，实现原理如下图所示（红框为插值目标）：

- 如果使用最近邻插值，则红框的灰度值将等于第 0 行 2 列的灰度值，因为红框中心距离其最近。
- 如果使用双线性插值，则红框灰度值将由其覆盖的 4 个像素点决定。每个点贡献的权重和其到红框中心的距离成反比。
- 如果使用双三次插值，则使用红框周边 4×4 的 16 个像素点的灰度值计算。每个点的权重和其到红框中心的距离同样呈反比。

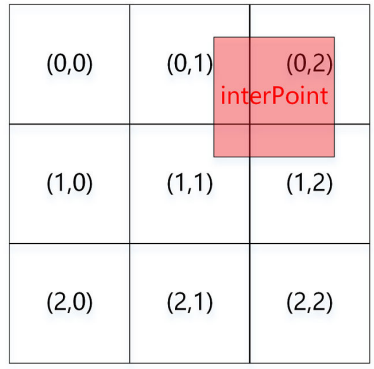


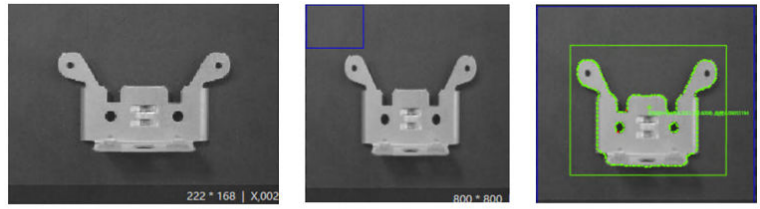
图 13-423 插值计算原理示意图

使用方法

在流程中，*图像缩放*的前序模块一般为 *图像源*，后序模块一般为 *轮廓匹配* 相关模块。
如下应用示例通过*图像缩放*模块将分辨率为 222×168 的图像转换为 800×800 的图像，使其特征更符合实际部件。最后将转换后的图像输出至 *快速匹配* 进行特征匹配。



图 13-424 应用示例



原图像　　缩放后　　特征匹配

图 13-425 图像缩放效果

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

输出图像宽度/高度

设置输出图像高度及宽度。

插值类型

最临近插值法

将目标像素点的四个相邻像素点中最近邻像素点的灰度值赋予目标像素点。更多原理详情参见上文*模块原理*提及的**像素插值计算**。

该插值类型计算量小，耗时短，但生成的灰度图像可能不连续。

双线性插值法

在像素矩阵上进行 x 和 y 两个方向线性插值求得目标像素点的灰度值。更多原理详情参见上文*模块原理*提及的**像素插值计算**。

该插值类型计算量较大，耗时较长，生成的灰度图像连续，但轮廓可能略微模糊。

双三次插值法

通过像素矩阵中最邻近的 16 个采样点加权平均得到目标像素点的灰度值。更多原理详情参见上文*模块原理*提及的**像素插值计算**。

双三次插值法精度高，生成的灰度图具有平滑的边缘，图像质量损失少，但计算量较大，耗时较长。

抗混叠

启用/关闭抗混叠滤波器。启用后，可改善缩放后出现的锯齿状像素点。原理是裁剪高频信号，对图像的插值计算使用适当的采样速率。

启用抗混叠前后的效果对比参见下图。



图 13-426 启用前后效果对比

模块结果

*图像缩放*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.10.14 几何变换

*几何变换*模块将图像进行水平、垂直、水平垂直方向的镜像变换，且可配合一定角度在镜像变换后再按照一定角度旋转图像。该模块适用于消除图像因被拍摄物摆放角度、透视关系、拍摄等原因造成的几何失真，进而避免图像无法识别。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块的算法工作流程如下。

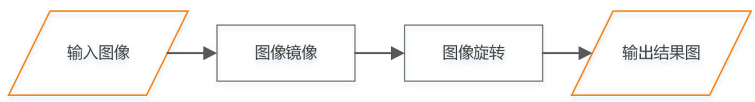
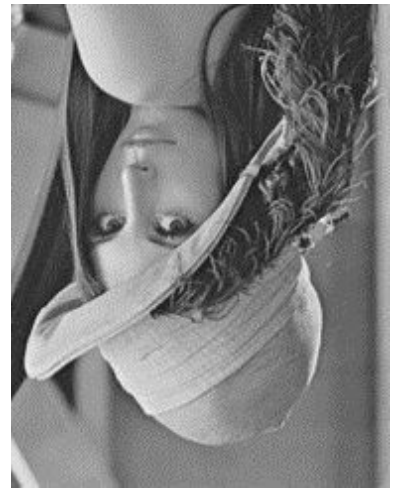


图 13-427 算法工作流

以上流程中，图像镜像包括如下三种镜像变换方式。

原图	水平镜像
	

垂直镜像	水平垂直变换（即对角镜像）
	

使用方法

该模块对前后序模块无特定要求。获取前序模块输入的图像即可通过调整参数进行图像矫正。

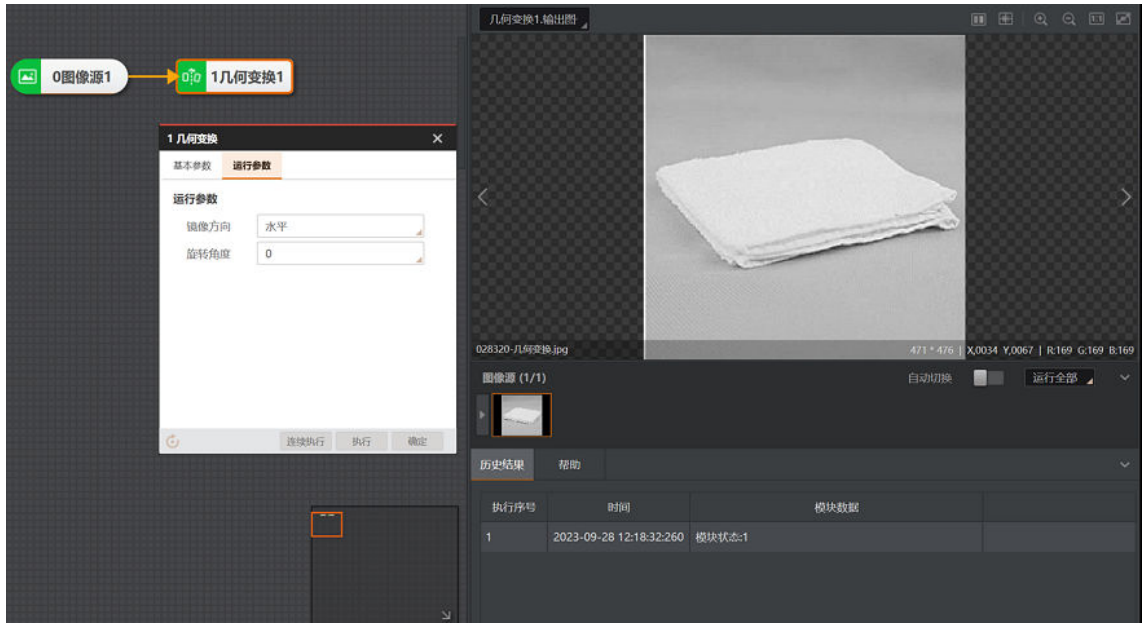


图 13-428 示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

镜像方向

可选水平、垂直、水平垂直。

水平

将图像的左右部分以图像垂直中轴线为中心进行镜像对换。

垂直

将图像的上下两部分以图像水平中轴线为中心进行镜像对换。

水平垂直

以图像水平中轴线和垂直中轴线的交点为中心对图像进行镜像对换。相当于先后进行水平镜像和垂直镜像。

旋转角度

图像以某一点为中心，旋转某一角度，形成一副新的图像。在原图像中某一像素点（x，y）根据旋转算法变换为结果图像中的（x’，y’）。

模块结果

*几何变换*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.10.15 圆环展开

*圆环展开*模块可从特定起始角度开始将圆环图像展开为矩形图像。该模块适用于圆环零部件检测、瓶口缺陷检测、圆环部件字符检测等场景。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*圆环展开*模块的算法可将圆环 ROI 中的图像平展开，其工作流程如下：

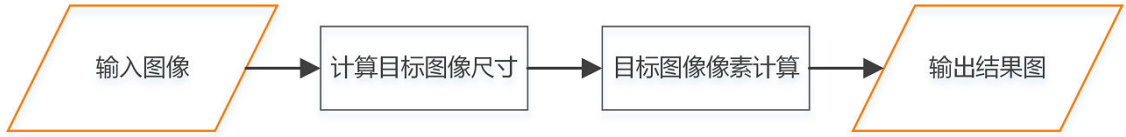


图 13-429 算法工作流程

计算目标图像尺寸

目标图像尺寸的计算公式如下。其中 DstW 和 DstH 分别表示目标图像宽高，PI 表示圆环的角度范围，OutRadius 和 InRadius 分别表示圆环的外径和内径，PI*OutRadius 表示外径的弧长，PI*InRadius 表示外径的弧长。

DstW=(PI*OutRadius+PI*InRadius)*0.5 (2.1)

DstH=OutRadius-InRadius

目标图像像素计算

目标图像第 i 行 j 列对应输入图像中的像素坐标的计算公式如下。

srcCol=centerX+rho*cos θ (2.3)

srcRow=centerY-rho*sin θ (2.4)

上述两个公式中：

- srcCol 和 srcRow 分别为输入图像中列和行坐标。
- centerX 和 centerY 表示圆环的中心点坐标。
- θ 表示圆环的起始偏移角度。
- rho 表示从内往外或从外往内逐像素点遍历时的圆环半径，遍历步长为 1。
  以从内往外展开圆环为例（如下图所示），rho 值可理解为：“从 InRadius（即圆环内径的值）开始，以步长 1 累加 DstH（即目标图像高度）次直至到达外环。”

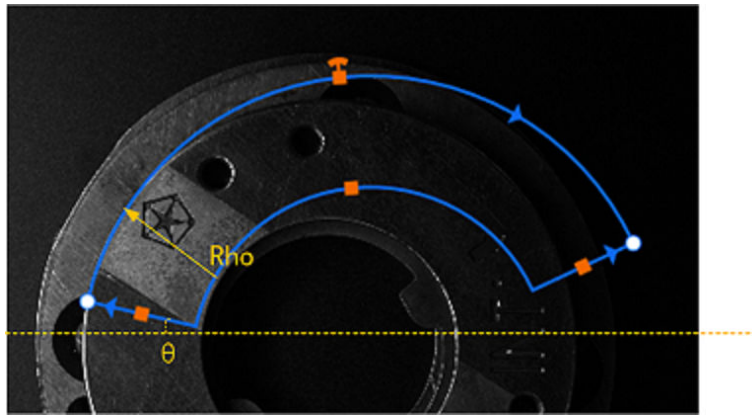


图 13-430 rho 值计算示意

注意事项

- 该模块输出图像仅支持 HKA_IMG_MONO_08 格式。
- 该模块输出图像中 data 数据内存由外部计算并分配。

使用方法

*圆环展开*的前序模块通常为*图像源*，后序模块通常为 *几何变换* 。一般圆环展开的图像需要通过*几何变换*进行方向转换。

如下应用示例中，*圆环展开*从*图像源*获取到圆环部件图像后将图像进行圆环展开处理，消除部件上字符因环状呈现而引发的字符识别问题。最终 *DL 字符识别* 成功识别部件上的字符。

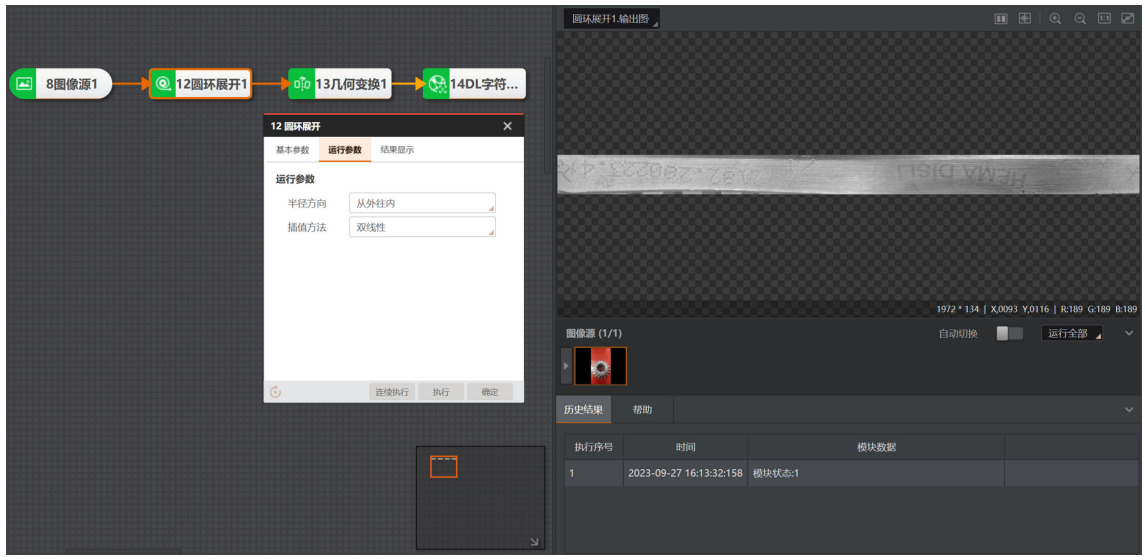


图 13-431 应用示例

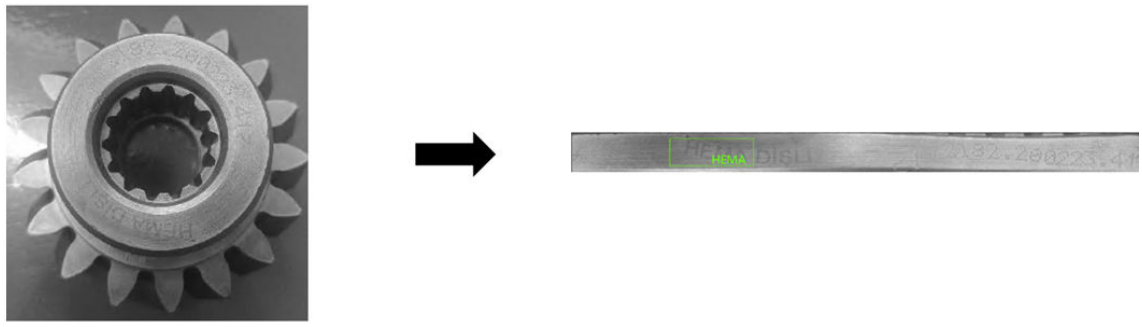


图 13-432 应用效果

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

半径方向

由内向外

圆环展开方向为由内向外。具体效果见如下两张示意图。



图 13-433 原图像



图 13-434 由内向外展开

由外向内

圆环展开方向为由外向内。具体效果见如下两张示意图。



图 13-435 原图像



图 13-436 由外向内展开

插值方法

分为**双线性**和**最邻近**。

双线性

先对 X 轴方向左右最邻近的像素点进行差值计算, 然后对 Y 轴方向上下最邻近的像素点进行差值计算。

最邻近

以最邻近的像素点进行差值计算。

模块结果

*圆环展开*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

检测圆弧

ANNULUS 型，代表检测圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点。

检测圆弧中心 X

float 型，代表圆弧 ROI 中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表圆弧 ROI 中心点的 Y 坐标。

检测圆弧内径

float 型，代表圆弧 ROI 的内径。

检测圆弧外径

float 型，代表圆弧 ROI 的外径。

检测圆弧起始角度

float 型，代表圆弧 ROI 起始点相对于圆心的角度。如果起始点位于 X 轴正方向，则为 0 度，顺时针旋转则角度随之增大。

检测圆弧角度范围

float 型，代表圆弧 ROI 从起始边开始至终止边所跨过的角度范围。

13.10.16 拷贝填充

*拷贝填充*模块提供图像拷贝功能和填充功能。简单来说，拷贝功能即把原图 ROI 内的图像复制出来，填充则是将复制出的图像填充指定灰度值。该模块多用于摘选突出有效特征，以便对特征进行精准处理。

本节内容包含：

- *模块原理*
- *注意事项*

- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*拷贝填充*模块算法功能分为拷贝和填充两种。

- 拷贝指将 ROI 内的图像拷贝至目标图像区域中，背景可指定填充 0~255 任意像素值。
- 填充指将 ROI 内和 ROI 外的像素分别进行 0~255 指定值填充。ROI 可以是任何几何形状。

如下图所示，绿色框为 ROI，红色框为 ROI 的最小外接矩形。最小外接矩形即输出的目标图像区域。

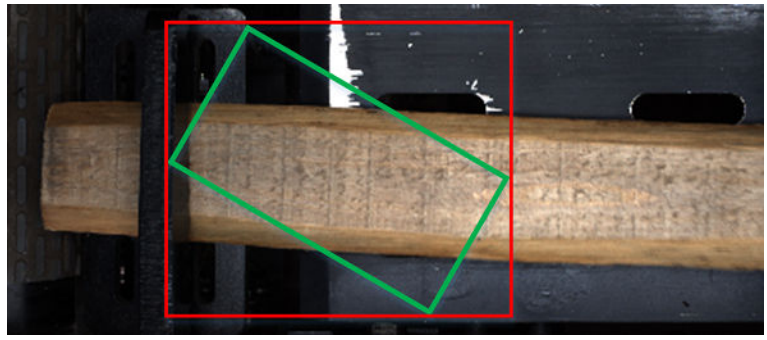


图 13-437 ROI 与最小外接矩形

具体算法工作流程如下图所示。



图 13-438 算法工作流程

注意事项

- 该模块输出图像目前仅支持 HKA_IMG_MONO_08、HKA_IMG_RGB_RGB24_C3 和 HKA_IMG_RGB_RGB24_P3 格式。
- 该模块输出图像中，data 数据内存由外部计算并分配，且内含其他输入。建议必要时进行初始化。

使用方法

该模块的前序模块为*图像源*，后序模块无特定要求。

该模块的配置操作请参考如下视频。

以下应用示例在图片中选取了一个扇形 ROI，*拷贝填充*模块输出一张 ROI 及其最小外接矩形。

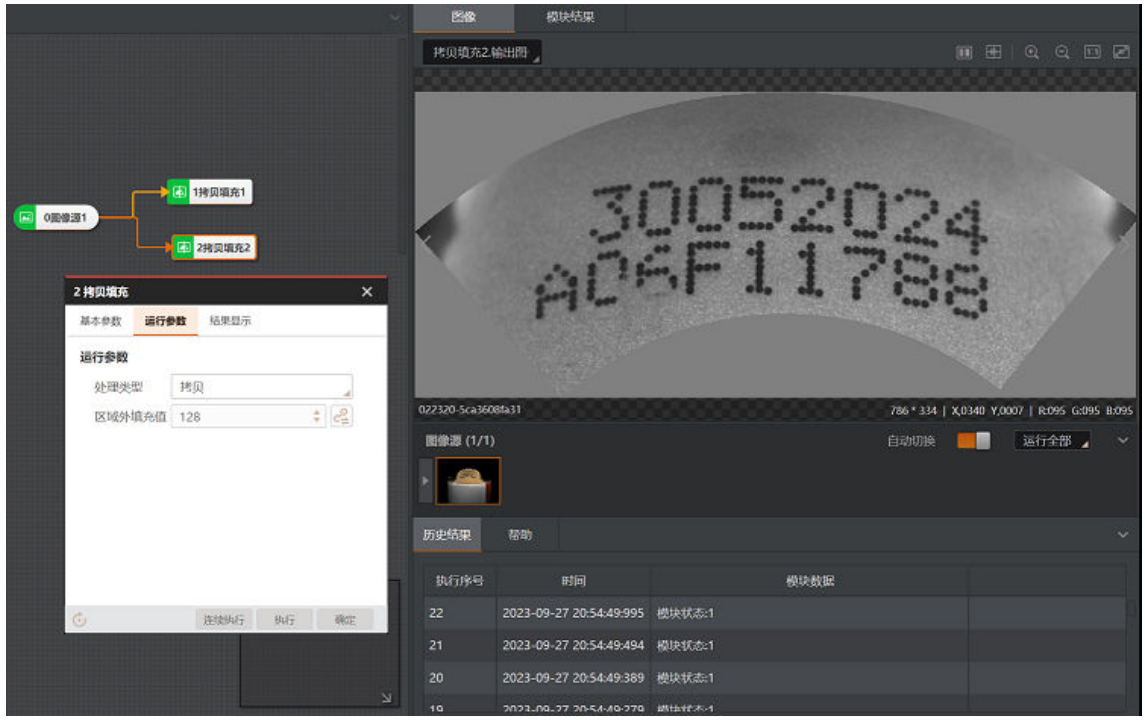


图 13-439 拷贝填充应用示例

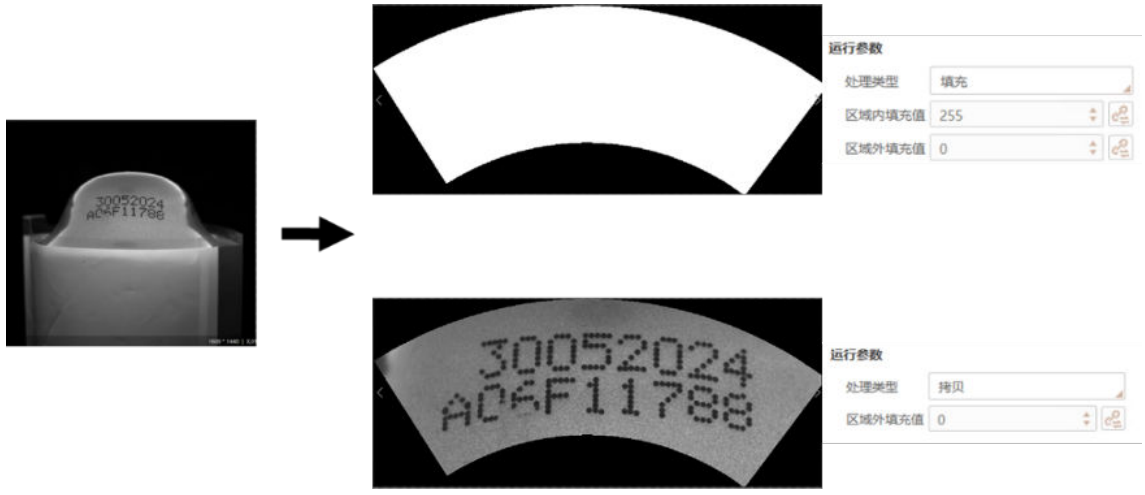


图 13-440 应用示例对应的效果

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

表 13-36 运行参数

参数		描述
拷贝	区域外填充值	在“ROI 以外，最小外接矩形以内”区域填充该灰度值
填充	区域内填充值	在 ROI 内部填充，填充的灰度值的范围为 0~255
	区域外填充值	在“ROI 以外，最小外接矩形以内”区域填充该灰度值

模块结果

*拷贝填充*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

13.10.17 畸变校正

*畸变校正*模块通过获取的图像坐标，校正图像中的畸变，使目标呈现更真实。该模块用于涉及标定的机械臂应用场景，如抓取、定位、贴合等。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块可解决图像的透视畸变、径向畸变和径向透视畸变问题。主要通过加载 *畸变标定* 生成的标定文件来实现。

使用方法

一般需将 *畸变标定* 作为*畸变校正*的前序模块使用。*畸变标定*可生成标定文件。
以下示例中，*图像源*输入相机畸变图像源并调整图像源的规格尺寸参数；*畸变标定*模块从相机畸变图像源中提取图像特征，计算出现实三维世界中对应的空间位姿的映射矩阵，输出标定图像源的状态和误差信息，最终生成标定文件；*畸变校正*模块依据*畸变标定*的生成的标定信息，输出校正后的相机图像。



图 13-441 流程示例

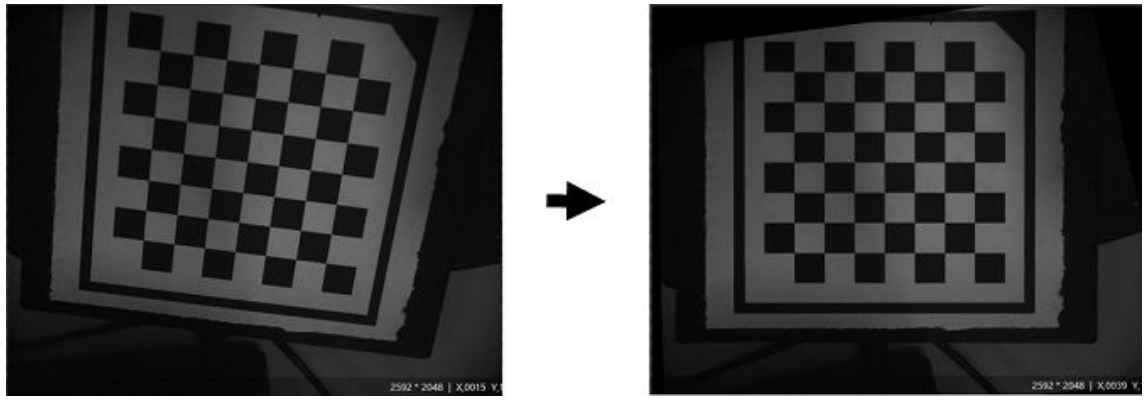


图 13-442 畸变校正效果

参数配置

表 13-37 标定参数

参数	描述
标定文件路径	标定文件通过 ***畸变标定*** 模块生成 标定文件在所选路径下才可成功加载。当文件不存在或者加载失败，运行时将报错
刷新信号	当 int 数值为非 0 数值时，会重新加载选定的标定文件

表 13-38 运行参数

参数	描述
透视畸变校正	透视畸变是广角镜头拍摄的图像具有的一种畸变特征。被拍摄体越远，显得越小。镜头距离被拍摄物体越近，被拍摄物体外观大小变化越大 透视畸变校正计算图像的透视变换矩阵进行畸变校正。该模式适用于标定板平面存在倾斜且镜头畸变较小的情况
径向畸变校正	径向畸变图像中，图像像素点以畸变中心为中心，沿着径向产生位置偏差

参数	描述
	径向畸变校正通过计算图像的径向畸变参数，去除图像的径向畸变。该模式适用于无需去除图像的透视畸变，只需去除径向畸变的场景
径向透视畸变校正	径向透视畸变图像中，图像像素点以畸变中心为中心，沿着径向产生位置偏差。被拍摄物体具体镜头越远，显得越小。镜头距离被拍摄物体越近，被拍摄物体外观大小变化越大 该模式适用于一般场景

模块结果

*畸变校正*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.10.18 图像矫正

由于光学透镜的固有特性，广角镜头拍摄的照片具有许多种畸变特征。*图像矫正*模块可将相机拍摄的径向畸变图像源转换为正常的图像源，使后续定位、分析、测量等应用输出更准确的数据。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块的前序模块为*图像源*，对后续模块无特定要求。
获取*图像源*输入的图像后，按需调整运行参数并执行该模块查看矫正效果，直至效果理想。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

曲张量

用于调整图像畸变。取值大于 0 时矫正桶形畸变，小于 0 矫正枕形畸变。

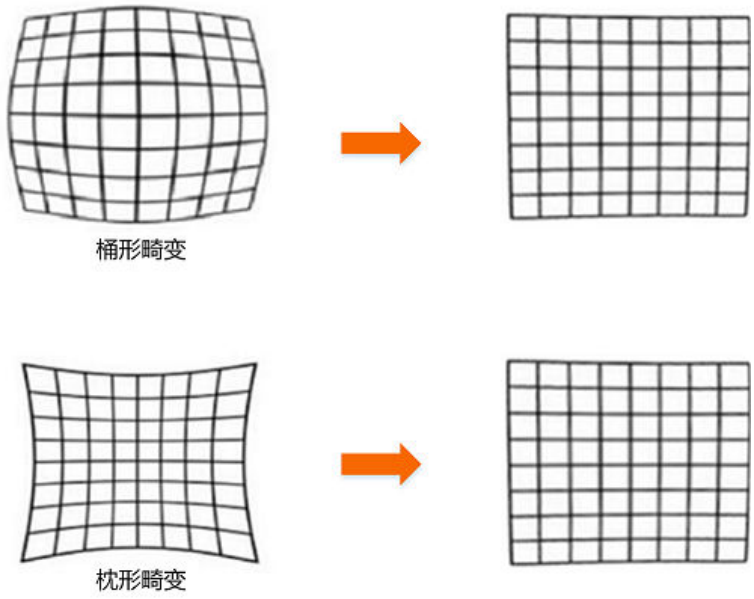


图 13-443 桶形和枕形畸变

缩放量

在不改变图像分辨率情况下调整图像大小。取值大于 0 时放大图像，小于 0 时缩小图像。

模块结果

*图像矫正*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.10.19 图像拼接

部分业务场景下，相机视野无法覆盖实物整体，但需要用一张图像将实物整体显示出来。此时可移动实物，分别拍摄实物不同部位的图像，并将其通过*图像拼接*模块拼接为大图，用于后续的定位、测量、识别等应用。

本节内容包含：

- *模块原理*
- *使用方法*
- *拼接模型*
- *参数配置*
- *模块结果*

模块原理

该模块的拼接功能分为硬拼接和模型拼接两种。硬拼接即将待拼接图像依次排列，此处不做赘述。

模型拼接算法的工作流程主要包括两个步骤，即建模和拼接。

说明

以下步骤以“拼接两张图片”为例进行介绍。

步骤 1：建模

1. 从标定板提取角点。
   每个角点都包含如下两类坐标信息：
   - 对应的图像上提取的像素坐标。
   - 由标定板上信息确定的物理坐标。
2. 计算从物理坐标映射到图像坐标的映射关系，并基于该映射关系计算最终映射点的坐标。
   a. 假定两组标定板角点的图像坐标分别为 P_2^I 和 P_2^I，物理坐标分别为 P_1^W 和 P_2^W，那么可分别计算从物理坐标映射到图像坐标的关系 H_1 和 H_2。计算公式如下：

$$P_1^I = H_1 P_1^W$$

$$P_2^I = H_2 P_2^W$$

图 13-444 物理坐标到图像坐标映射关系计算公式

   b. 由于 P_1^W 和 P_2^W 的坐标系相同，将第二图像中的点映射到物理坐标系，再将映射到物理坐标系的点映射到第一图像中，得到最终映射点的坐标。
   计算公式如下，公式中 H_2^{-1} 表示 H_2 的逆映射关系。

$$P_{2\to1}^I = H_1 H_2^{-1} P_2^I$$

   建模阶段的核心工作即计算 $H_1H_2^{-1}$

步骤 2：拼接

1. 通过 $H_1H_2^{-1}$ 将第二张图像中所有的像素点全部映射到第一图像
2. 与第一图像中的原始素点合并，得到最终拼接的图像。

使用方法

前后序模块

该模块的前序模块为***图像源***，后序模块可为识别、定位、测量、图像处理等模块。

主要配置步骤

在流程中调用***图像拼接***模块后，该模块的主要配置步骤如下。

1. 执行一次流程，使*图像拼接*获取*图像源*输入的图像。
2. 双击流程中的*图像拼接*，打开该模块的配置窗口，配置 *基本参数* 、 *拼接模型* （仅**拼接类型**设置为**模型拼接**需配置）、 *运行参数* 和 *结果显示* 。
3. 单击*执行*，查看该模块的运行结果。

拼接模型

该模块基本参数中的**拼接类型**设置为**模型拼接**时，必须创建拼接模型方可进行图像拼接。

前提条件

已准备海康标定板图像（I 型、II 型、扩展 I 型、扩展 II 型）。标定板图像的获取方式详情，请参见 *标定板生成* 。

说明

每一张标定板图像都需与待拼接图像一一对应；换言之，即需使用同一点位的相机拍摄相同位置的图像，保证创建的标定板空间位置关系模型与待拼接图像的模型一致。否则将导致图像拼接失败。

请参照如下步骤创建拼接模型。

操作步骤

1. 在图像拼接配置窗口选择**拼接模型**页签。
2. 单击*新建模型*，打开*模板配置*窗口。
3. 在该窗口单击 ＋ ，导入标定板图像。

   说明

   - 仅支持导入海康标定板的图像。
   - 请确保导入的图像无畸变。为了以防万一，建议标定板图像和待拼接图像全部先进行 *畸变校正* 。

4. **可选操作**: 如有需要，单击 清空输入的图像。
5. 按需配置是否开启**拼接修正**。

   未开启时，只能进行拼接；开启后，还支持进行修正。
6. 按需配置拼接相关参数**标定板类型**。

   根据实际使用的标定板类型设置。可选海康 I 型标定板、海康 II 型标定板、海康扩展 I 型标定板、海康扩展 II 型标定板。
7. **可选操作**: 若开启**拼接修正**，需配置修正相关参数。

   优化方法

   可选择优化模型的方法，可选快速方法和精确方法。精确方法计算优化位置更准确，快速方法计算优化位置更快速。按需选择即可。

   采样点数

可设置优化过程中使用的点数。点数越多，计算越准，耗时越长。

X 负方向范围
X 正方向范围
Y 负方向范围
Y 正方向范围

可设置模型优化调整的像素范围。若模型优化前，拼接位置与预期相差较大，可调大范围，但也会造成耗时边长。

偏移 X
偏移 Y

自动计算的位置准确时，可手动增加偏移数值进行调整。

说明

此参数需选中单张图像才会显示。每张图像的数值需单独设置。

8. 单击*提取特征*。
9. **可选操作**: 单击*拼接预览*预览拼接图像。

   如果拼接图像效果欠理想，调整模型参数并再次预览，直至效果符合预期。
10. 单击*创建模型*，完成模型创建并返回*拼接模型*页签。
11. **可选操作**: 在*拼接模型*页签进行如下可选操作。

编辑模型	单击*编辑模型*继续编辑模型。
清空模型	单击 。
清空图像	单击*清空图像*。
导出模型	单击 将当前模型导出至本地路径。
导入模型	单击 从本地路径导入模型。

参数配置

以下仅介绍该模块的 ***基本参数*** 和 ***运行参数*** 详情。

说明

- 拼接模型相关参数详情，请参见 ***拼接模型*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

基本参数

输入方式

分**单来源**和**多来源**两种方式。

单来源

选择一个图像源即可。对应单相机多次移动的业务场景。

多来源

选择拼接数目个图像源。对应多相机固定位置采图的场景。

拼接类型

可选**模型拼接**和**硬拼接**。

模型拼接

图像拼接通过拼接模型执行。

此时需设置**拼接方式**参数。

拼接方式

A×B 即 A 行 B 列图像。该参数配置会影响拼接模型里面的图像分布。

硬拼接

图像拼接直接执行。

此时需设置**拼接方向**参数。

拼接方向

可选 **X 方向**或 **Y 方向**，定义拼接方向为“从左往后”或“从上往下”。

拼接数目

自定义需要拼接的图像数目。

- **输入方式**为**单来源**时，运行次数达到本参数取值即可完成一次拼接。
- **输入方式**为**多来源**时，图像源个数应与拼接数目保持一致，单次运行完成拼接。

首图清空

通过该参数可判断当前输入的图像是否为拼接的第一张图像。若是，则清空前面输入的图像。

应订阅 int 型数据，订阅的数值为 1 时，则清空前面输入的图像；其他数值，则不清空，继续进行拼接。

图像输入

- **输入方式**设置为**单来源**时，选择一个图像源即可。通过**输入源**参数设置。
- **输入方式**设置为**多来源**时，需选择的图像源个数与**拼接数目**取值一致。通过**输入图像***参数分别设置。

运行参数

融合模式

可选**均值融合**、**最大值融合**、**最小值融合**以及**接尾融合**。

说明

该参数仅在**拼接类型**选择**模型拼接**时可设置。

均值融合

默认方式，对重复区域取均值然后完成融合，通常效果较好。

最大值融合

获取重合图像的最大值，将其添加到拼接图像中。适用于拼接“中间亮、死角暗”的图像。

最小值融合

在重合图像区域中取最小值，将最小值添加到融合图像中。

接尾融合

对已完成拼接的图像不再进行重复拼接，仅处理未拼接的区域。适用于解决拼接中的重影现象，但图像连接处渐变性较差。

裁剪 X 方向
裁剪 Y 方向

可分别设置图像水平/垂直方向外圈的裁剪比例，参数对应裁剪的百分比。取值越大处理区域越少，模块运行耗时越少。适用于拼接重合较大的图像，可降低拼接重影。

说明

该参数仅在**拼接类型**选择**模型拼接**时可设置。

自动清空

与**自动填充**互斥，只能启用其中一个。

- 如果启用，完成一次图像拼接后自动删除本次输入图像
- 如果关闭，完成一次图像拼接并再次输入一张图像时，新输入的图像将替换之前拼接图像中的一张。

自动填充

与**自动清空**互斥，只能启用其中一个。

- 若开启，获取的图像数量未达到预设的数量时，用全黑图像替代欠缺的图像进行拼接。
- 若关闭，仅在待拼接图像达到预设数量时进行拼接。

ROI 列表

可设置各个输入图像的 ROI 参数。

设置时需确保各个 ROI 区域 X 方向的宽度、或 Y 方向的高度是否完全一致，否则会影响拼接效果。至于需确保 X 方向还是 Y 方向由*基本参数*中的**拼接方向**参数决定。

说明

该参数仅在**拼接类型**选择**硬拼接**时可设置。

模块结果

*图像拼接*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

当前图像个数

int 型，代表当前图像是所有图像中的第几个。

图像总个数

int 型，代表图像的总个数。

拼接图像数据

拼接图像

image 型，代表需要拼接的所有图像。

拼接图像宽度

int 型，代表拼接图像的宽度。

拼接图像高度

int 型，代表拼接图像的高度。

拼接图像格式

int 型，代表拼接图像的图像格式。

最小外接矩形

box 型，表示能完整包含一个图形或目标区域的最小矩形。

中心点

point 型，表示最小外接矩形的中心点。

中心 X

float 型，代表最小外接矩形中心点的 X 坐标。

中心 Y

float 型，代表最小外接矩形中心点的 Y 坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

角度

float 型，代表最小外接矩形的较长边基于水平线旋转的角度。其中，顺时针旋转为正，逆时针旋转为负。

13.10.20 清晰度评估

通过*清晰度评估*模块，可对指定图片的清晰度进行量化评估，进而评判相机是否聚焦清晰。该模块多用于相机调焦过程中做图像清晰度评估。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块的前序模块为 *图像源* 。*图像源*采集图像给到*清晰度评估*模块。后序模块可根据实际场景搭配，例如可搭配 *条件检测* 和 *条件分支* 等逻辑模块进行逻辑处理。

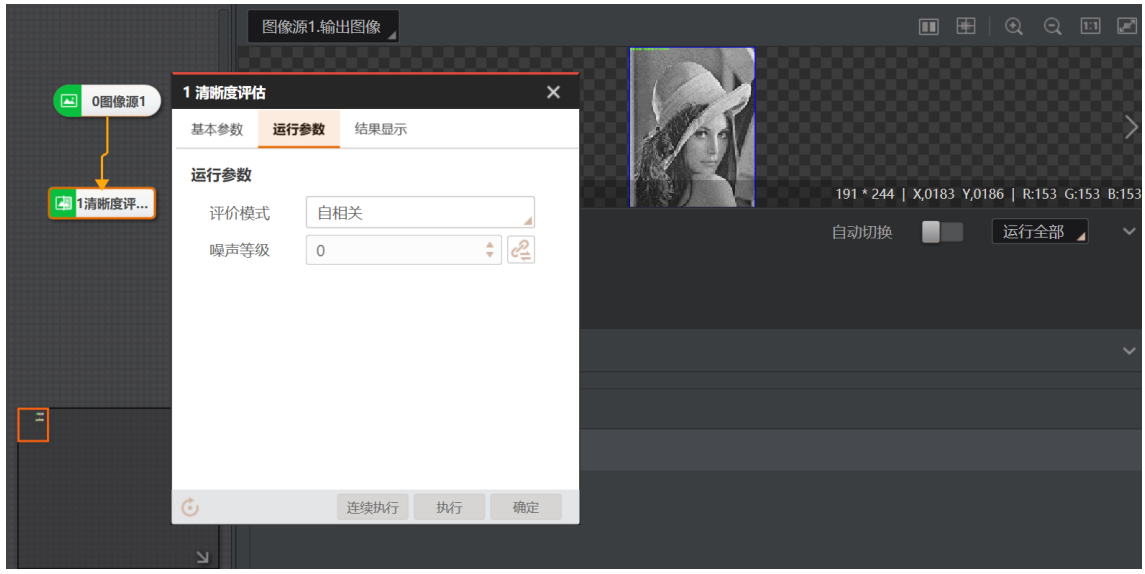


图 13-445 清晰度评估

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

该模块的运行参数为**评价模式**，具体选项如下：

自相关

自相关适用于纹理信息较少的场合，不适用于噪声较大场景。

噪声等级

用于提高抗噪能力，数值大小表示当前图像中的灰度噪声标准差。噪声越大，数值越大，一般为 0~32。

梯度平方

适用于图像内梯度信息比较丰富的场景。

模块结果

*清晰度评估*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

清晰度

float 型，代表即对处理图像的清晰度评估结果，数值越大代表越清晰。

输出掩膜

IMAGE 型，表示模块输出的掩膜图像。

输出掩膜图像

image 型，代表根据检测区域和屏蔽区域计算出的掩膜图像，以二值图的形式输出。

输出掩膜宽度

int 型，代表输出掩膜图像的宽度。

输出掩膜高度

int 型，代表输出掩膜图像的高度。

输出掩膜像素格式

int 型，代表输出掩膜图像的像素格式。一般为 17301505，表示 Mono8 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

多边形字符串

string 型，代表以字符串形式输出的多边形信息。字符串中的数值表示 ROI 各顶点的 X 坐标和 Y 坐标，每个坐标值中间用英文逗号隔开。

屏蔽区字符串

string 型，代表屏蔽区位置的字符串信息，主要用于后序模块继承屏蔽区。

13.10.21 帧平均

*帧平均*通过对多帧图像相同像素位置的灰度值进行均值计算处理，可获取一系列图片的平均像素灰度图像。该模块多应用于图像降噪场景。

本节内容包含：

- *注意事项*
- *使用方法*
- *参数配置*
- *模块结果*

注意事项

- 该模块输出图像仅支持 HKA_IMG_MONO_08 格式。
- 该模块输出图像中 data 数据内存由外部计算并分配。

使用方法

该模块的前序模块一般为*图像源*，后序模块无特定要求。
*图像源*为该模块提供多帧图像。该模块输出的均值图像给后序模块处理。

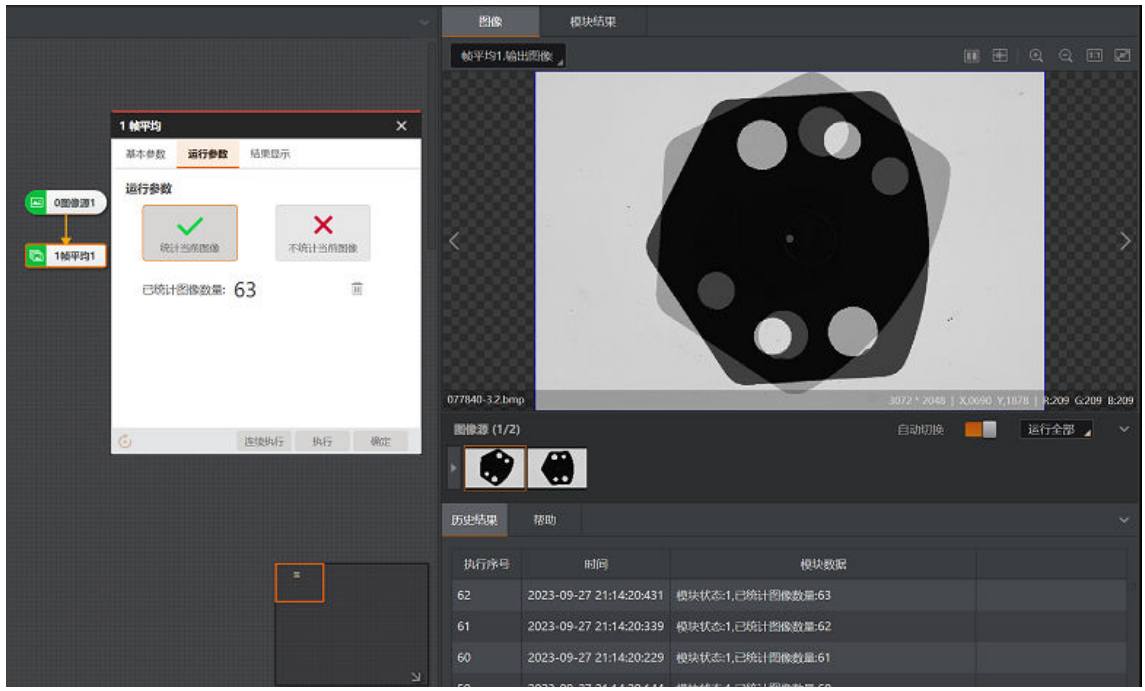


图 13-446 应用示例

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

表 13-39 运行参数

参数	描述
统计当前图像	对当前最多 100 张输入图像进行累加像素灰度平均统计，输出平均过后的均值图像
不统计当前图像	跳过对当前输入图像的累加统计
清空统计图像	将当前统计得到的均值图像清零

模块结果

*帧平均*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

已统计图像数

int 型，代表参与帧平均统计的图像总数。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.10.22 多图融合

*多图融合*模块可融合不同光照角度和亮度的多路图像，获取更明显的图像特征和纹理。该模块适用于在“有无辨别”、“瑕疵污点”、“字符识别（凹/凸字）”等场景中，消除多角度光照对目标物体造成的亮度不均和反光等影响。

本节内容包含：

- *模块原理*
- *使用限制*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

该模块相关的主要工作流程为：

1. 利用 1 个相机和 3 个及以上的多角度同光谱光源（常用为 4 或 8 个角度的光源），通过分别开启光源，获取同光源数量的图像。
2. 将上述多路图像融合为一路图像。

使用限制

- 该模块暂时只支持灰度图。
- 请务必输入 3 至 8 张多个不同角度光线采集的图像，否则图像融合效果可能受影响。

使用方法

*多图融合*的前序模块为 *多图采集* 。*多图采集*同时输出 3 至 8 路多角度同光谱图像至*多图融合*，后者对这些图像进行融合。

如下应用示例中：

1. *多图采集*输出 4 路图像至*多图融合*。
2. *多图融合*输出“突出划痕”的阴影图像至*图像滤波*。
3. *图像滤波*对阴影图像进行平滑处理，弱化与划痕无关的特征后，将图像输出至 *表面缺陷滤波* 。
4. *表面缺陷滤波* 输出划痕特征显著的图像，用于后续的划痕检测。

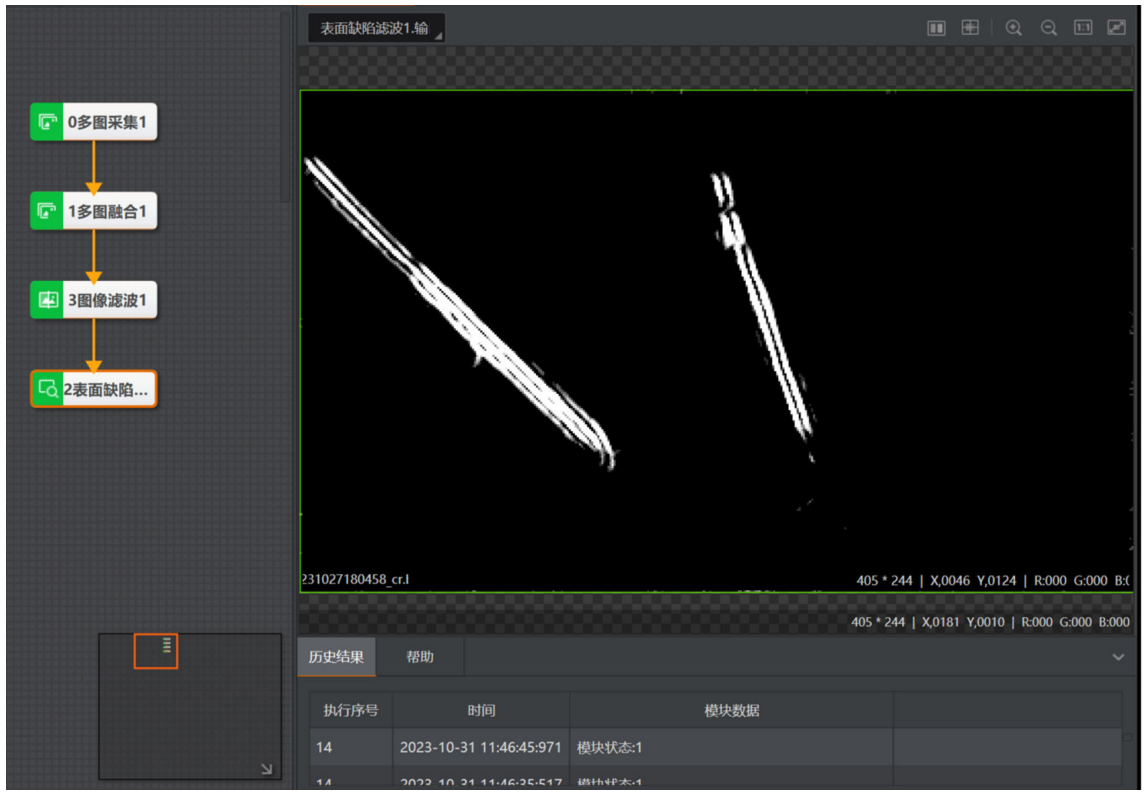


图 13-447 应用示例：划痕检测

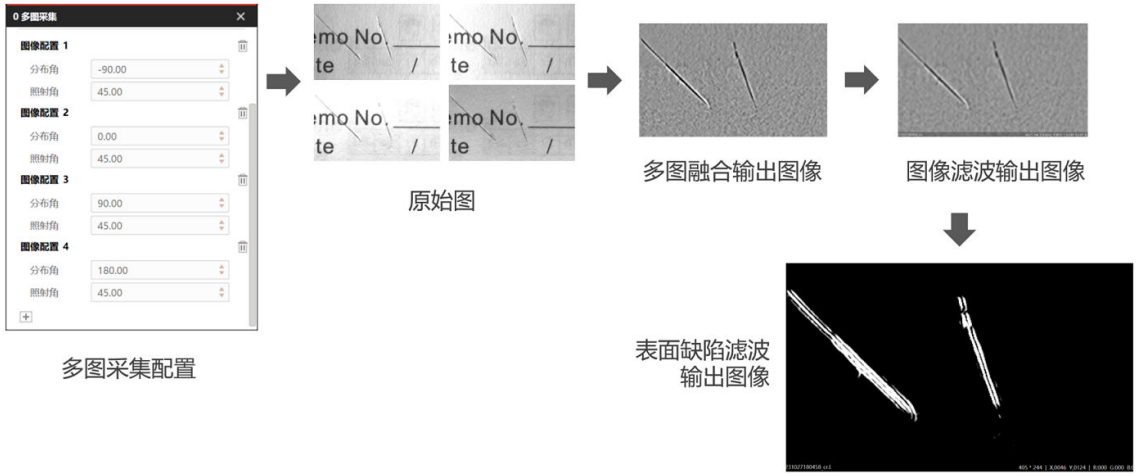


图 13-448 示例中的图像处理

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 ***基本参数*** 。
- 结果显示参数详情，请参见 ***结果显示*** 。

输出图像类型

包括**全部**、**反射**（反射图像）和**阴影**（阴影图像）三种类型。

滤波尺寸

滤波器大小，可用于增强边缘和抑制噪声，最小值为 1。

当边缘模糊或有噪声干扰时，增大该值有利于使得检测结果更加稳定，但如果边缘与边缘之间距离小于滤波尺寸时反而会影响边缘位置的精度甚至丢失边缘。边缘过渡带较小时使用较小滤波尺寸，边缘过渡带较大时使用较大滤波尺寸。

滤波尺寸越大边缘越模糊，滤波尺寸过小达不到滤波效果。

增强使能

用于增强阴影图像。如果关闭，则该模块自适应给出阴影图像。如果自适应的阴影图像无法满足需求，可开启并配置如下参数，手动调节阴影图像效果。

背景亮度

取值越大，阴影图像的背景越黑；取值越小，背景越白。

对比系数

取值越大，图像阴影越明显。

光晕去除等级

取值越大，光晕消除能力越大，输出的反射图像更暗。如果输入图像为 4 幅，建议配置为 3；如果为 8 幅，建议为 5。

方向增强类型

仅对阴影图像有效，默认值为空。将其设置为 **X 方向**或 **Y 方向**时，对应方向的凹凸信息将更明显。

方向增强等级

方向增强等级越大，凹凸信息更明显。

模块结果

*多图融合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

图像数

int 型，代表需要融合的图像总数。

图像宽度

int 型，代表图像的宽度。

图像高度

int 型，代表图像的高度。

反射图像数据

反射图像

image 型，代表输出图像类型为反射图像。

反射图像高度

int 型，代表输出反射图像的高度值。

反射图像宽度

int 型，代表输出反射图像的宽度值。

反射图像像素格式

int 型，代表输出反射图像的图像像素格式。

阴影图像数据

阴影图像

image 型，代表输出图像类型为阴影图像。

阴影图像宽度

int 型，代表输出阴影图像的宽度值。

阴影图像高度

int 型，代表输出阴影图像的高度值。

阴影图像像素格式

int 型，代表输出阴影图像的图像像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

13.11 颜色处理

颜色处理分类下的模块可对彩色图像进行处理，包括**颜色抽取**、**颜色测量**、**颜色转换**、**颜色识别**、**颜色分割**以及**彩图生成**模块。

颜色处理相关模块中关于几种颜色空间的介绍详情请参见 *术语介绍* 。

13.11.1 颜色抽取

颜色抽取模块以彩色图像作为输入，根据指定颜色空间中设置的三通道阈值，抽取彩色图像中的目标区域。最终得到仅含目标区域的二值图（目标区域像素值为 255，其它像素值为 0），同时输出目标区域的面积。该模块通常用于目标区域和非目标区域色彩区分明显的场景，例如“颜色面积检测”等场景。

使用方法

该模块的前序模块一般为**图像源**模块，并设置为输出 RGB 图像，后序模块一般可搭配 **BLOB 分析**模块，可用于定位抽取颜色的目标。

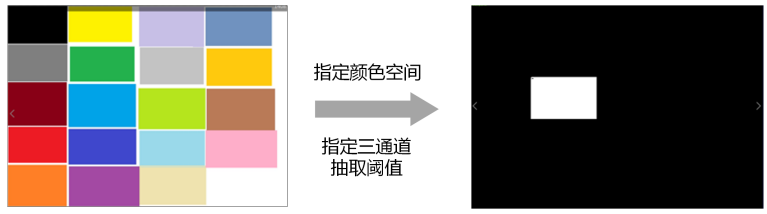


图 13-449 颜色抽取模块功能示意图



图 13-450 颜色抽取搭配使用示例



图 13-451 颜色抽取使用效果图

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

抽取列表

显示当前已设置的颜色范围，点击一个颜色范围后可编辑相应的运行参数。通过点击加号，可以设置多个颜色范围，所有范围取并集作为目标的颜色范围。

说明

- 当鼠标悬停在颜色范围上时，颜色范围右上角显示删除符号，若点击该符号，则删除该颜色范围。
- 若颜色范围左上角的复选框没有被勾选，则该颜色范围处于未使能状态，不会被应用于颜色抽取，仅有勾选的颜色范围才被用于颜色抽取。

颜色空间

支持 RGB、HSV、HSI 三种颜色空间，具体介绍请见 *术语介绍* 章节。

通道一/二/三

当颜色空间选择 RGB 颜色空间时，通道一、二、三分别对应 R、G、B 三通道；当选择 HSV 颜色空间时，通道一、二、三分别对应 H、S、V 通道；当选择 HSI 颜色空间时，通道一、二、三分别对应 H、S、I 通道。

当**颜色空间**分别选择 RGB/HSV/HSI 颜色空间，**通道一/二/三**的上下限值为默认的 0 和 255 时，抽取列表中的颜色范围示例如下所示。



图 13-452 颜色范围示例

通道下限

指定颜色空间内，用来存储、传递与颜色相关的图像信息的通道抽取像素的最小值。

通道上限

指定颜色空间内，用来存储、传递与颜色相关的图像信息的通道抽取像素的最大值。

颜色反转

若开启颜色反转使能，则将当前颜色空间中设置的颜色范围的补集作为实际抽取范围。

模块结果

*颜色抽取*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

总面积

float 型，代表所抽取颜色在输入图像中所占据的总面积，也就是输出图像中白色区域的面积。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

13.11.2 颜色测量

颜色测量模块以彩色图像作为输入，在指定颜色空间下，计算图像 ROI 区域的三通道均值、最大值、最小值、标准差和直方图等数据。该模块通常用于检测当前图像的 RGB 值，将所得数据与设定值进行比较，进而输出判断结果。

使用方法

该模块的前序模块一般为**图像源**模块，并设置为输出 RGB 图像，后序模块一般可搭配**逻辑**模块，可用于结果的判断和比较。



图 13-453 颜色测量模块功能示意图

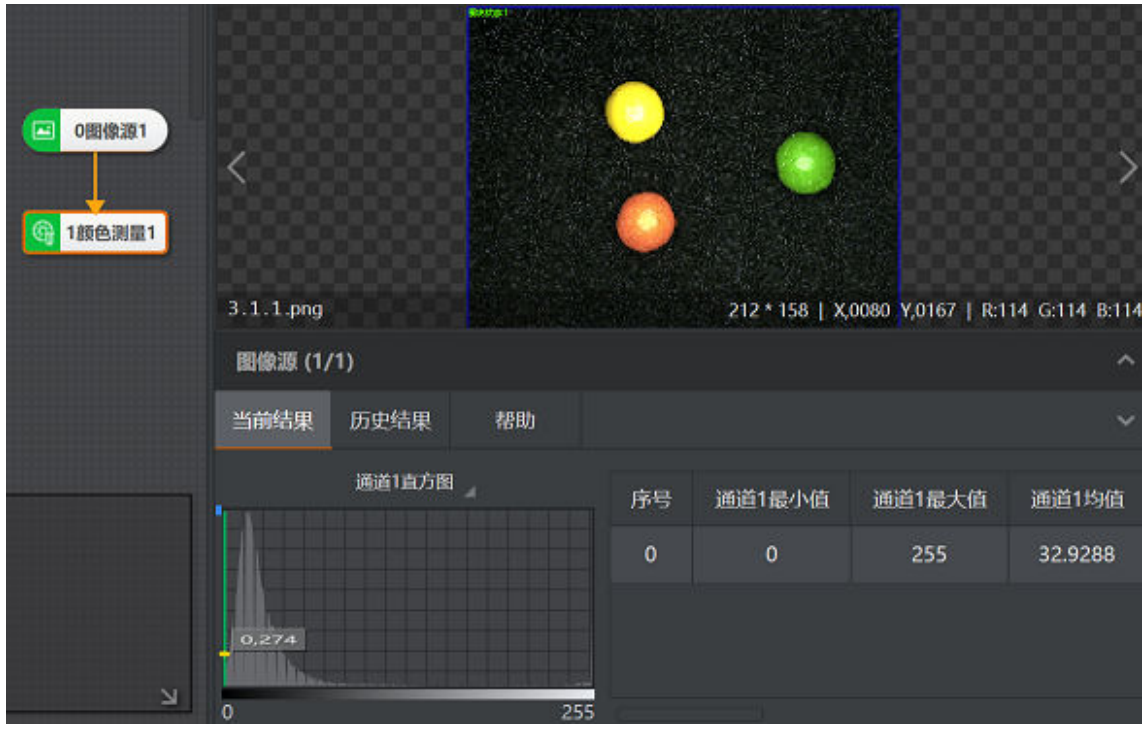


图 13-454 颜色测量示例效果图

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

颜色空间

支持 RGB、HSV、HSI 三种颜色空间，具体介绍请见 *术语介绍* 章节。

模块结果

*颜色测量*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

通道*最小值（*代表 1-3）

int 型，代表用来存储、传递与颜色相关的图像信息的通道*最小值。

通道*最大值（*代表 1-3）

int 型，代表用来存储、传递与颜色相关的图像信息的通道*最大值。

通道*均值（*代表 1-3）

float 型，代表用来存储、传递与颜色相关的图像信息的通道*平均值。

通道*标准差（*代表 1-3）

float 型，代表用来存储、传递与颜色相关的图像信息的通道*标准差值。

通道*直方图（*代表 1-3）

int 型，代表将存储、传递与颜色相关的图像信息的通道*用直方图表示的结果。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

13.11.3 颜色转换

模块原理

颜色转换模块是对输入的 RGB 图像中指定彩色区域的颜色信息进行测量，并将图像转换到指定的颜色空间（支持灰度、HSV、HSI、YUV），并输出指定颜色通道的灰度图像。



图 13-455 颜色转换模块功能效果图

使用方法

颜色转换模块一般用于需要将原本的彩色图像转换成其他颜色空间并进行计算处理的场景。颜色转换模块需要输入彩色图像（比如：图像源模块设置为输出 RGB 图像），后续一般是搭配需要使用的检测模块，对指定颜色空间的图片进行处理。



图 13-456 颜色转换使用效果图

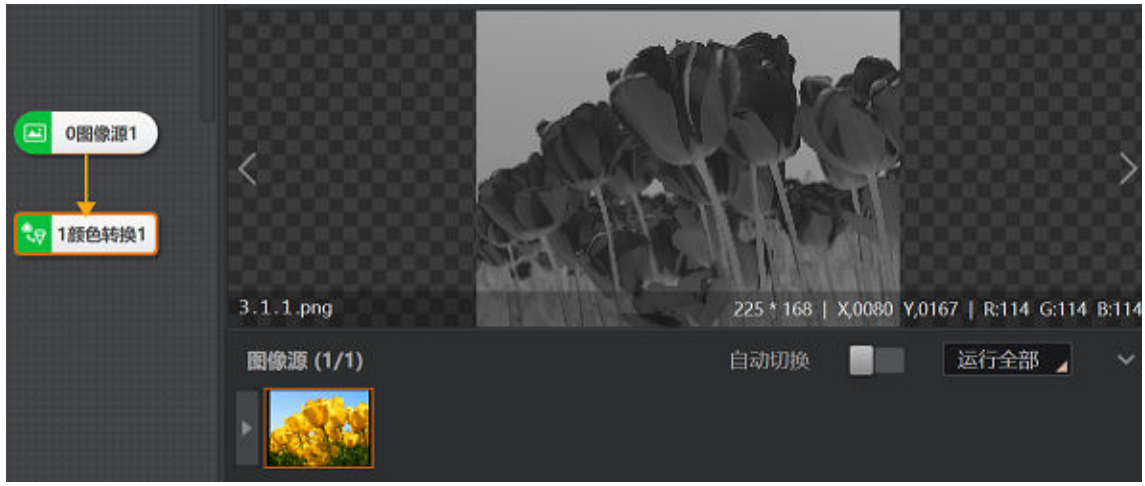


图 13-457 颜色转换示例效果图

参数配置

转换类型

包含 RGB 转灰度、RGB 转 HSV、RGB 转 HSI、RGB 转 YUV。选择不同的转换类型需设置的参数有所差别。

选择 **RGB 转灰度**时，需设置**转换比例**。

转换比例

说明

以下参数介绍中涉及的：小写的 r、g、b 分别代表图像中实际的 R、G、B 通道分量。

- 通用转换比例：0.299r + 0.587g + 0.114b。



图 13-458 通用转换比例

- 平均转换比例：(r + g + b) / 3。



图 13-459 平均转换比例

- 通道最小值：min(r, g, b)。



图 13-460 通道最小值示意图

- 通道最大值：max(r, g, b)。



图 13-461 通道最大值示意图

- 自设转换比例： (R*r+G*g+B*b) / (R+G+B) 。

  说明

  R 转换比例、**G 转换比例**、**B 转换比例**分别对应公式中大写的的 R、G、B。

- R 通道：r + 0*g + 0*b，即 R 通道灰度值。
- G 通道：0*r + g + 0*b，即 G 通道灰度值。
- B 通道：0*r + 0*g + b，即 B 通道灰度值。
- 无约束转换比例：R*r+G*g+B*b。

  说明

  R 转换比例、**G 转换比例**、**B 转换比例**分别对应公式中大写的的 R、G、B。

选择 **RGB 转 HSV/HSI/YUV** 时，需设置**显示通道**。

显示通道

- 第一通道：按照第一通道转换。
- 第二通道：按照第二通道转换。
- 第三通道：按照第三通道转换。

模块结果

*颜色转换*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.11.4 颜色识别

颜色识别模块通过对图像样本建立颜色模型，识别并计算目标与模型的相似程度。主要用于通过颜色对目标图像进行分类。

模块原理

颜色识别模块是对目标图像进行分类识别的工具，包含建模和识别两部分。

- 建模：通过输入一张或多张图像，选择图像中的 ROI 区域作为样本，设置样本标签，建立颜色模型。
- 识别：输入待识别图像的 ROI 区域，根据颜色模型中的样本与待识别区域的相似程度识别出最接近的标签类名称。

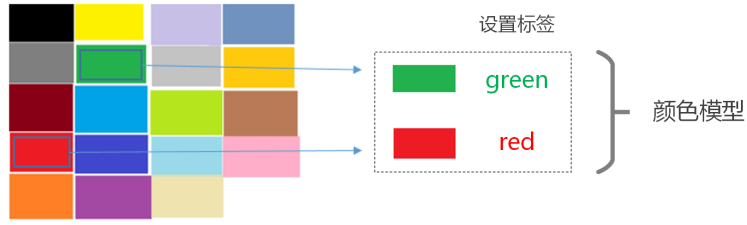


图 13-462 颜色识别模块建模功能示意图



图 13-463 颜色识别模块识别功能示意图

使用方法

颜色识别通过对图像样本建立 ***颜色模型***，识别计算目标与模型的相似程度，主要是对目标图像进行分类的工具。工具适用于目标物体颜色变化较为明显的场景，常用在“有无检测”场景，对于体积变化、形状变化等不适用。

颜色识别模块需要输入彩色图像（比如：图像源模块设置为输出 RGB 图像），后续一般是搭配需要使用的逻辑模块，对颜色识别模块的结果进行判断处理。

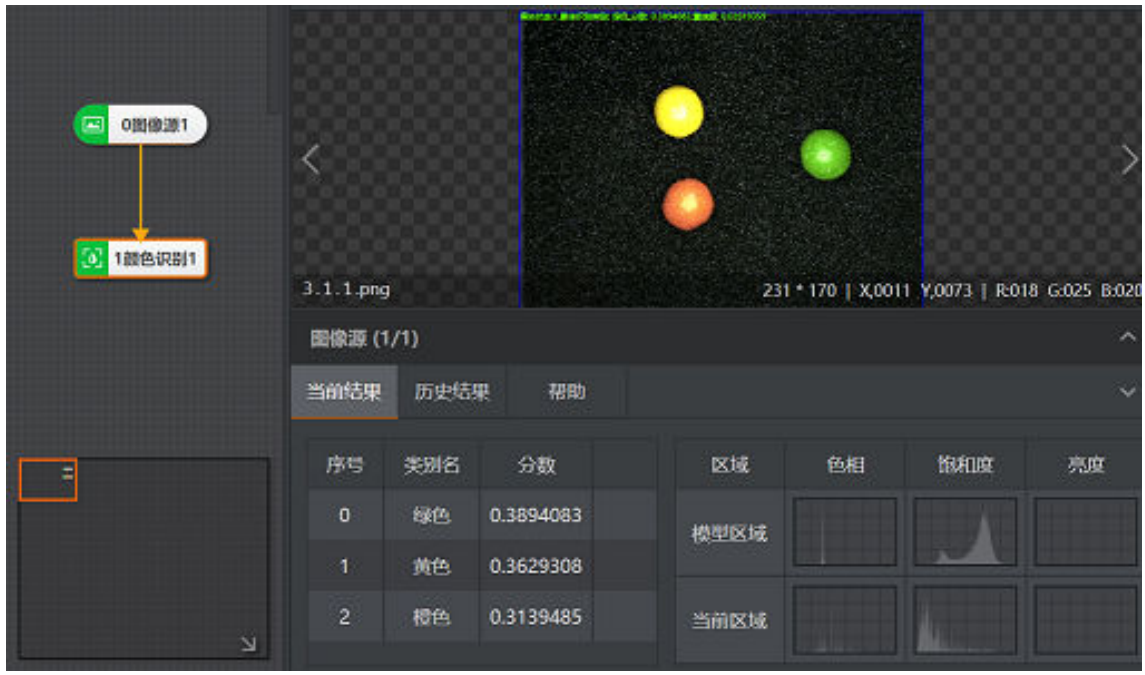


图 13-464 颜色识别示例效果图

颜色模型

颜色模型是通过输入一张或多张图像，选择图像中的 ROI 区域作为样本，设置样本标签并进行创建。

操作步骤

1. 在**颜色模型**面板中点击 ⊞ 可进入模板配置界面进行创建模型。
   1) 加载图像。点击下方**图像列表**处的 添加当前图像源模块中导入的图片，也可点击 ⊞ 或 添加本地图片。
   2) 添加标签。点击**标签类列表**右侧的 ⊞ 创建标签，点击标签右侧的 可对标签名称进行编辑。
   3) 框选掩膜。使用 、 或 对标注区域绘制掩膜 ROI。

4) 添加至标签。点击*添加至标签*可将绘制的掩膜样本加入对应标签类别，标签名称右侧会同步显示对应的样本数量。

说明

当样本打标错误时，可将样本移动至正确的标签列表中。

5) 配置参数。

敏感度

表示对图像颜色变化的敏感程度。分为高、中、低三种敏感模式，当图像对颜色变化的敏感程度较高时，建议选择高敏感模式。

特征类型

包含直方图特征和色谱特征两种类型。相较而言，直方图特征分布更细，对图像变化更加敏感。

不同参数对颜色变化的敏感程度如下图所示。建议选择直方图特征高敏感模式，若识别效果不满足要求，可根据实际效果修改参数。

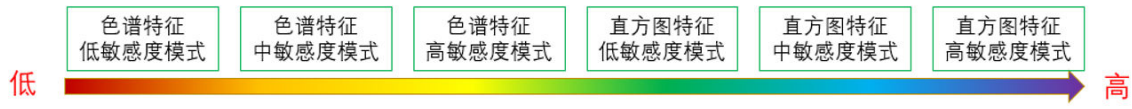


图 13-465 不同参数对颜色变化的灵敏度

亮度

反映光照对图像的影响程度。若需要在光照变化的情况下保持识别结果更加稳定，可关闭亮度特征。

说明

当选择直方图特征时可选择启用或关闭亮度特征，选择色谱特征时推荐始终启用亮度特征。



图 13-466 模板配置界面

6) 点击*保存模型*即可将配置的模型生成.bin 格式模型文件，点击*确定*即配置成功并返回颜色模型面板。



图 13-467 颜色模型面板

2. 在**颜色模型**面板中点击右上角的*编辑模型*可进入模板配置界面对当前已有的模型进行编辑。
3. 在**颜色模型**面板中点击 / 可分别导入/导出模型文件。

说明

在导入模型时，会清空已有的颜色模型。

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

K 值

表示选取前 K 个颜色样本中所占数量最多的类作为最佳识别结果，K 值需小于所有标签类中的最小样本数。

KNN 距离

是指不同特征值之间的距离，包含欧式距离、曼哈顿距离、相交距离、偏移距离四种。推荐使用偏移距离。

欧氏距离

最常见的距离变量，是指多维空间里两个点之间的真实距离。

曼哈顿距离

在二维坐标系中两点之间的横坐标和纵坐标的距离总和。

相交距离

由于历史兼容性原因保留，但效果不如偏移距离好。

偏移距离

推荐选择此项，运行效果最优。

模块结果

*颜色识别*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

置信度

float 型，代表识别的字符的置信度。数组，其长度等于字符个数。

最佳匹配名称

string 型，代表添加颜色模型，设置每种颜色标签类型，用选取工具提取图像颜色并测量识别指定区域的颜色信息，调整参数设置，输出模板配置，建立完模板后加载图像并设定 ROI 限定目标区域，单次执行会输出每个模板对应的识别得分， 识别得分最高的为最佳匹配模型，该模型名称为最佳匹配名称。

最佳分数

float 型，代表添加颜色模型，设置每种颜色标签类型，用选取工具提取图像颜色并测量识别指定区域的颜色信息，调整参数设置，输出模板配置，建立完模板后加载图像并设定 ROI 限定目标区域，单次执行会输出每个模板对应的识别得分， 识别得分最高的为最佳分数。

类别名

string 型，代表创建颜色模型所需要输入的标签类名称，用来区别颜色类 别。

分数

float 型，代表匹配对象与模板的相似程度。当分数为 1 时，表示匹配对象与模板图像完全相同；当分数为 0 时，表示两者完全不一致。

模型 H 通道数据

float 型，代表将存储、传递与颜色相关的图像色调的模型通道数据。

模型 S 通道数据

float 型，代表将存储、传递与颜色相关的图像饱和度的模型通道数据。

模型 I 通道数据

float 型，代表将存储、传递与颜色相关的图像亮度的模型通道数据。

样本 H 通道数据

float 型，代表将存储、传递与颜色相关的图像色调的样本通道数据。

样本 S 通道数据

float 型，代表将存储、传递与颜色相关的图像饱和度的样本通道数据。

样本 I 通道数据

float 型，代表将存储、传递与颜色相关的图像亮度的样本通道数据。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

屏蔽区多边形

polygon 型，代表图像中需屏蔽的多边形区域，其内部的所有图像数据将被忽略。

多边形点数

int 型，代表屏蔽区多边形的顶点个数。

多边形点集

point 型，代表绘屏蔽区多边形各个顶点的集合。

多边形点 X

float 型，代表绘屏蔽区多边形各个顶点的 X 坐标。

多边形点 Y

float 型，代表绘屏蔽区多边形各个顶点的 Y 坐标。

13.11.5 颜色分割

*颜色分割*模块通过输入彩色图像，根据在指定颜色空间设置的三通道阈值将目标区域进行分割。该模块与*颜色抽取*模块的不同点在于：其输出的灰度图像，每个像素的灰度值标记了该像素所属的目标区域。不同目标区域使用不同的灰度值表示，非目标区域的灰度值为 0。

模块介绍

与*颜色抽取*模块相比，该模块增加如下功能。

- 同时抽取多个不同颜色的目标区域。
  每个颜色范围可设置一个类别名称，相同类别的颜色范围视为同一种颜色类别，抽取时取并集；而不同类别的颜色范围视为不同的颜色类别，不同类别的颜色范围之间可以存在交集，该模块将自动处理属于颜色交集范围内的像素。
  颜色分割时，每个像素最终仅可属于一种类别，在输出图像中使用该类别的灰度值表示。若像素不属于任何一个颜色类别，则在输出图像中的灰度值为 0。

  说明

  模块自动为每个类别赋予一个灰度值，可通过模块结果的**类别信息**查到每个类别所对应的灰度值。

  图像显示区域选择“输出渲染图”，以伪彩色的方式显示灰度图，有利于观察分割结果。
  举例：已分别设置深绿、浅绿和红色三个颜色范围。其中，深绿和浅绿颜色范围的类别名称设置为“绿色”而红色的类别名称设置为“红色”。那么在输出的灰度图像中，灰度值为 15 的区域表示绿色（包括深绿和浅绿），而灰度值为 31 的区域表示红色，灰度值为 0 的区域表示红色绿色以外的其它颜色。
- 可对输出的灰度图像进行孔洞填充和杂斑筛除的操作，从而减少物体颜色非均匀性和图像噪声对分割结果的影响。

  - 孔洞是被目标区域（即灰度值非 0 的区域）完全包围的非目标区域（即灰度值为 0 的 8 连通域）。设置孔洞面积时，面积小于设置数值的孔洞将被填充，与其周边的目标区域合并。
  - 杂斑是指同一种颜色所组成的面积较小的目标区域（灰度值非 0 的 8 连通域）。设置杂斑面积时，面积小于设置数值的杂斑将被删除，变为非目标区域或与其周边其它的目标区域合并。
- 增加色品图和明度轴，以及自动设定开关。
  - 在图像上通过**颜色区域选择**框选颜色后，色品图和明度轴中用红点表示所框选范围内的颜色，也可手动拖动游标调整颜色范围。
  - 打开**自动设定**参数时，每次框选颜色后，游标自动调整为模块推荐的颜色范围；关闭时，框选后游标位置不会自动调整。

使用方法

该模块通过判断三个通道的值是否在设定范围内进行筛选。一般用于分割目标区域和非目标区域色彩区分明显的场景，或分割多个颜色差异明显的目标。

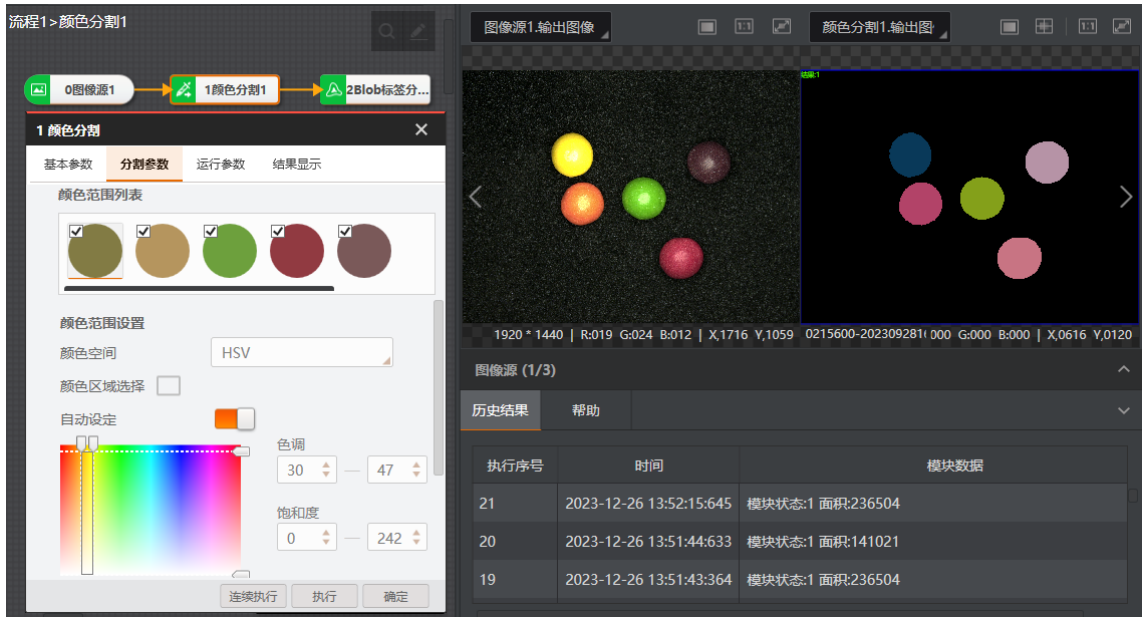


图 13-468 示例方案

该模块建议与 *Blob 标签分析* 模块配合使用。*Blob 标签分析*模块作为该模块的后序模块，可自动订阅该模块的类别名称和灰度值，计算各个分割区域的属性，并进行筛选和排序。

参数配置

此处仅对该模块的分割参数以及运行参数进行介绍，基本参数和结果显示各模块大同小异，此处不再赘述，具体参见 *基本参数* 和 *结果显示* 。

- 分割参数：

颜色范围列表

呈现已设置的颜色范围，单击 ＋ 可新增。支持 1 ~ 16 个颜色范围。

说明

选中每个颜色范围，可通过**颜色范围设置**设置参数。

颜色范围设置

可分别对每个颜色范围进行相关设置。

颜色空间

选择设置颜色范围的颜色空间。可选 RGB、HSV、HSI。

选择不同的颜色空间，需设置的参数有所差别。

RGB

通过 **R 通道**、**G 通道**、**B 通道**分别设置各通道的范围阈值。

HSV
HSI

通过**色调**、**饱和度**、**亮度**分别设置各通道的范围阈值，也可通过左侧示意图的滑块调整范围阈值。

自动设定

启用该参数时，色品图和明度轴的游标每次根据**颜色区域选择**框选的区域自动调整为算法推荐的颜色范围；未启用时，框选后游标位置不会自动调整。

颜色区域选择

在模块的输入图像框选指定目标区域，可快速生成对应的三通道范围阈值。

颜色反转

启用该参数时，则选择当前设置颜色范围外的颜色空间作为实际的颜色范围。

使用独立面积参数

未开启时，当前颜色范围使用运行参数中的**孔洞填充面积**和**杂斑筛除面积**参数；开启时，可单独设置自己的这两个参数。

孔洞填充面积

面积小于设置数值的孔洞将被填充，与其周边的目标区域合并。

杂斑筛除面积

面积小于设置数值的杂斑将被删除，变为非目标区域或与其周边其他的目标区域合并。

类别名称

可自定义设置对应颜色范围的类别名称。

说明

当多个颜色范围的类别名称相同时，模块会将颜色范围合并后作为一个类别输出。

- 运行参数：可设置该模块全局的**孔洞填充面积**和**杂斑筛除面积**。参数功能与开启**使用独立面积参数**时的参数功能相同，仅针对的颜色范围有所差别。

说明

此处的参数针对该模块的所有颜色范围，前面的参数针对选中的颜色范围。

模块结果

*颜色分割*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

总面积

float 型，代表所有 Blob 加起来的总面积。

类别信息

image 型，代表类别图信息。

类别名称

string 型，代表类别名称，保存在模型中。导入模型时，从模型文件中获取。

灰度值

int 型，代表每个类别的灰度值。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

输出渲染图

image 型，代表完成渲染后输出的图像。

输出渲染图

image 型，代表完成渲染后输出的图像。

输出渲染图宽度

int 型，代表输出渲染图的宽度大小。

输出渲染图高度

int 型，代表输出渲染图的高度大小。

输出渲染图像素

int 型，代表输出渲染图的像素格式。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

ROI 圆弧

ANNULUS 型，代表 ROI 圆弧。

检测圆弧中心点

point 型，代表检测圆弧的中心点坐标。

检测圆弧中心 X

float 型，代表检测圆弧中心点的 X 坐标。

检测圆弧中心 Y

float 型，代表检测圆弧中心点的 Y 坐标。

检测圆弧内径

float 型，代表检测圆弧的内径。

检测圆弧外径

float 型，代表检测圆弧的外径。

检测圆弧起始角度

float 型，代表检测圆弧的起始边与水平方向夹角。

检测圆弧角度范围

float 型，代表检测圆弧的起始边到终止边的角度范围。

13.11.6 彩图生成

*彩图生成*模块可将三张灰度图分别作为 R、G、B 通道的图像，合成一张彩色图像。

使用方法

该模块一般在相机配合三色光源，分别拍摄红光、绿光和蓝光灰度图的情况下使用，可合成彩色图。将三色光源分别采集的图像作为对应各通道的输入图像即可。如下图所示。

说明

订阅的 3 张图片，应确保只是灰度值的差异，其他需确保完全一致，例如相同的被测物、完全一致的摆放位置等。

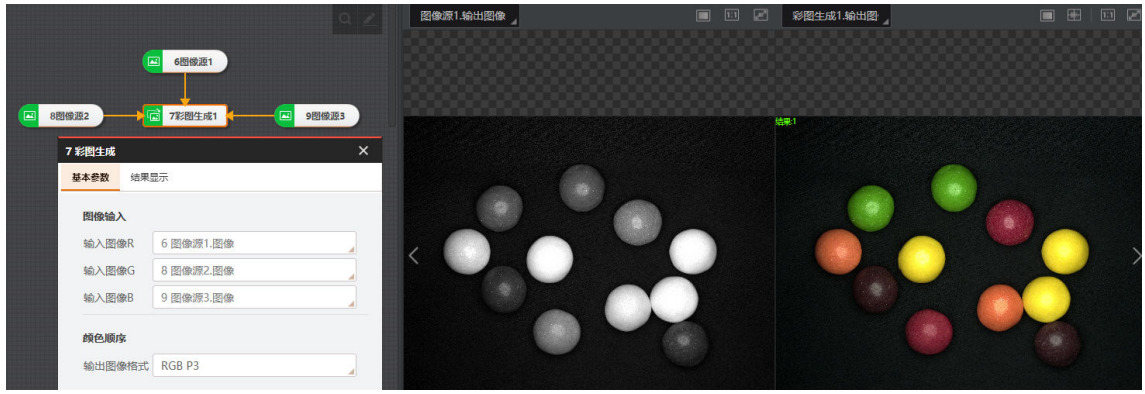


图 13-469 效果示意

该模块的前序模块一般为*图像源*等可输出灰度图的模块，后序模块一般为颜色处理等相关模块，进行相关检测。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见*结果显示*。

输入颜色 R
输入颜色 G
输入颜色 B

可分别选择作为图像 R、G、B 通道分量的灰度图。

输出图像格式

可设置输出图像的格式，可选 RGB P3 或 RGB C3。

模块结果

*彩图生成*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.12 拆分组合

拆分组合分类下的模块可对图像中的图形进行拆分、组合和筛选，包括划片拆分、二维阵列、多标签筛选、Box 融合、Box 重叠和 Box 过滤模块。

13.12.1 划片拆分

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

划片拆分模块可根据设置的运行参数，将输入图像的检测区域划分为多个划片区域，以供后续模块（如拷贝填充、Group 等）分别进行处理。
结合下图来看，L 代表整个检测区域水平方向的长度，S 代表每个划片水平方向的长度，M 代表水平方向的划片数量，F 代表水平方向的重叠率，则该模块遵循的计算公式为 S*M-(M-1)*F*L=L。其中，重叠率 F=相邻划片之间重叠的部分/检测区域长度 L。

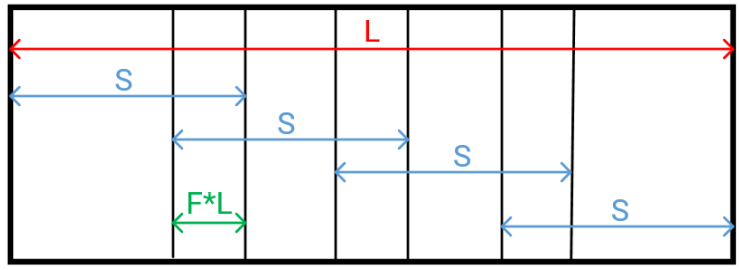


图 13-470 原理示意图

说明

- 水平方向划分后，图像最左侧和最右侧划片的重叠区域在靠近图像中心区域侧，靠边侧无重叠区域，如上图所示。
- 垂直方向的计算公式及原理与水平方向相同，不再赘述。

使用方法

当图像中存在循环阵列的相似图形需要对所以图形单独进行处理时，可用此模块来实现类似于划片后拆分处理的操作。如下图所示，将一张图片拆分成 3*3，即 9 个区域。



图 13-471 划片拆分示意图

该模块一般与拷贝填充、Group 等模块进行配合使用。例如，在 group 模块中可以先传入分片区域，再对这些区域逐个进行处理，执行后即可得到每个分片区域的检测结果，如下图所示。

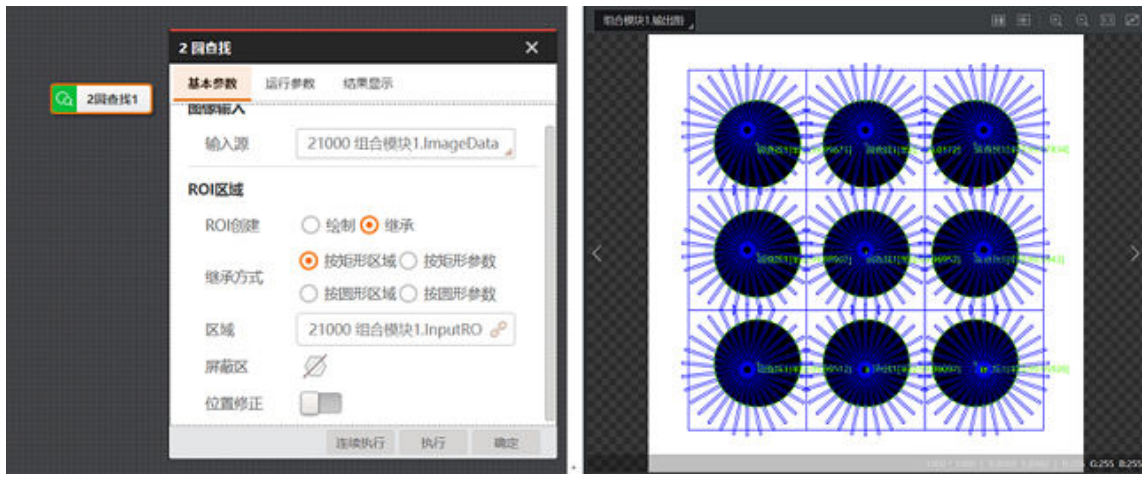


图 13-472 划片拆分检测结果

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

X/Y 方向划片数

即水平/垂直方向需要划分的划片数量。

X/Y 方向重叠率

即水平/垂直方向的相邻划片之间的重叠率；当重叠率设置为 0 时，则表示相邻划片之间无重叠部分。

模块结果

*划片拆分*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

检测区域

box 型，代表图像中需进行检测的区域。

检测区域中心点

point 型，代表检测区域的中心点。

检测区域中心点 X

float 型，代表检测区域中心点的 X 坐标。

检测区域中心点 Y

float 型，代表检测区域中心点的 Y 坐标。

检测区域宽度

float 型，代表检测区域的宽度。

检测区域高度

float 型，代表检测区域的高度。

检测区域角度

float 型，代表检测区域相对于水平方向（正准位姿）的旋转角度。顺时针旋转，角度为正；逆时针旋转，角度为负。若检测区域处于正准位姿，则为 0 度。

分片矩形

分片矩形中心

分片矩形中心 X

float 型，代表划片后每一个分片矩形的中心点 X 坐标。

分片矩形中心 Y

float 型，代表划片后每一个分片矩形的中心点 Y 坐标。

分片矩形宽度

float 型，代表划片后每一个分片矩形的宽度。

分片矩形高度

float 型，代表划片后每一个分片矩形的高度。

分片矩形角度

float 型，代表划片后每一个分片矩形的角度。

划片个数

int 型，代表总的划片个数。

13.12.2 二维阵列

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

二维阵列定位的目标需满足行列垂直和水平分布的特点。二维阵列示意图如下图所示，红色 box 代表缺失的目标，需通过二维阵列模块进行修正。

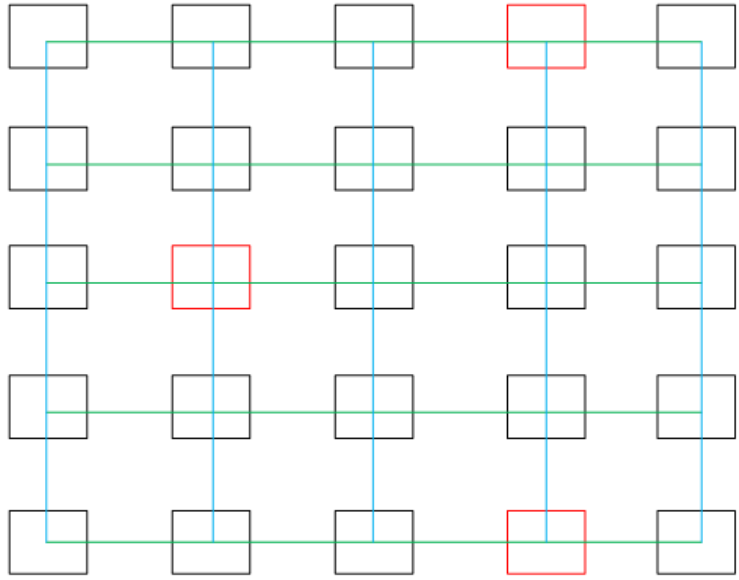


图 13-473 二维阵列示意图

二维阵列模块运行原理如下：

1. 将水平分布的目标中心点拟合成直线。
   拟合直线的最大条数即设置的**阵列行数**，每一条直线拟合时要求 box 中心点数至少是设定**阵列列数**的 0.5 倍。
2. 将垂直分布的目标中心点按照相同逻辑拟合成直线。
   拟合直线的最大条数即设置的**阵列列数**，每一条直线拟合时要求 box 中心点数至少是设定**阵列行数**的 0.5 倍。
3. 拟合得到水平和垂直直线后，拟合得到上图中的水平直线（绿色）和垂直直线（蓝色）。计算两两相交的交点，即为修正后的目标中心。原目标尺寸保持不变。
4. 获得修正后的目标中心后，同步计算每一行的最小包围 box。

使用方法

二维阵列工具可对二维阵列的初定位结果进行修改，包括补齐漏定位以及误定位。同时，还可计算阵列中的行最小包围 box（即同一行的 box 整合后的 box）。如下图，已知我们检测的对象的分布大致为阵列的形式，在第 1 行第 3 列的对象、第 2 行第 1 列，由于该对象与模板存在差异，导致匹配缺失，但我们仍要输出对应的 Box 区域，可以使用该模块进行补齐。

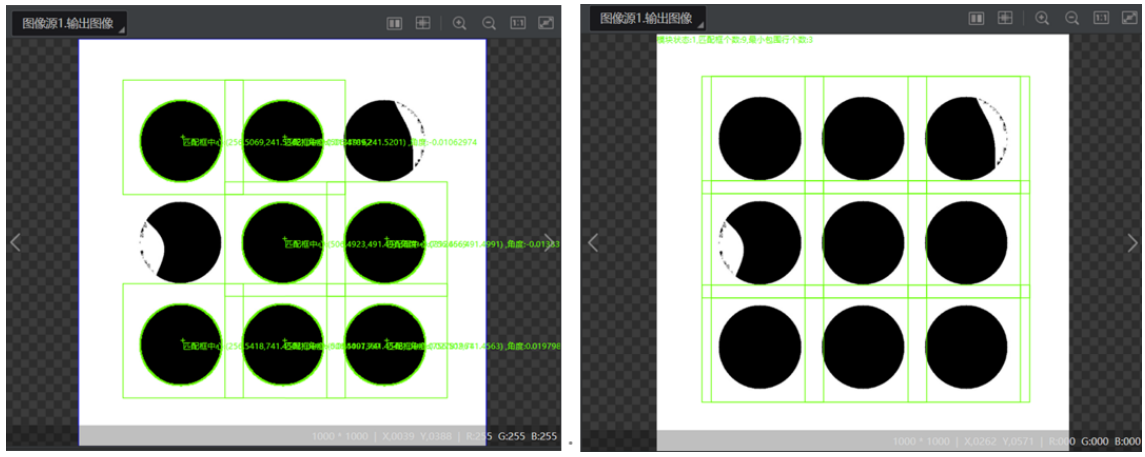


图 13-474 二维阵列示意图

该模块一般与模块匹配、目标检测等模块配合使用进行阵列化补齐。针对模块匹配的缺失情况，通过运行参数可设置二维阵列的阵列行数和阵列列数，补齐效果如下图所示。

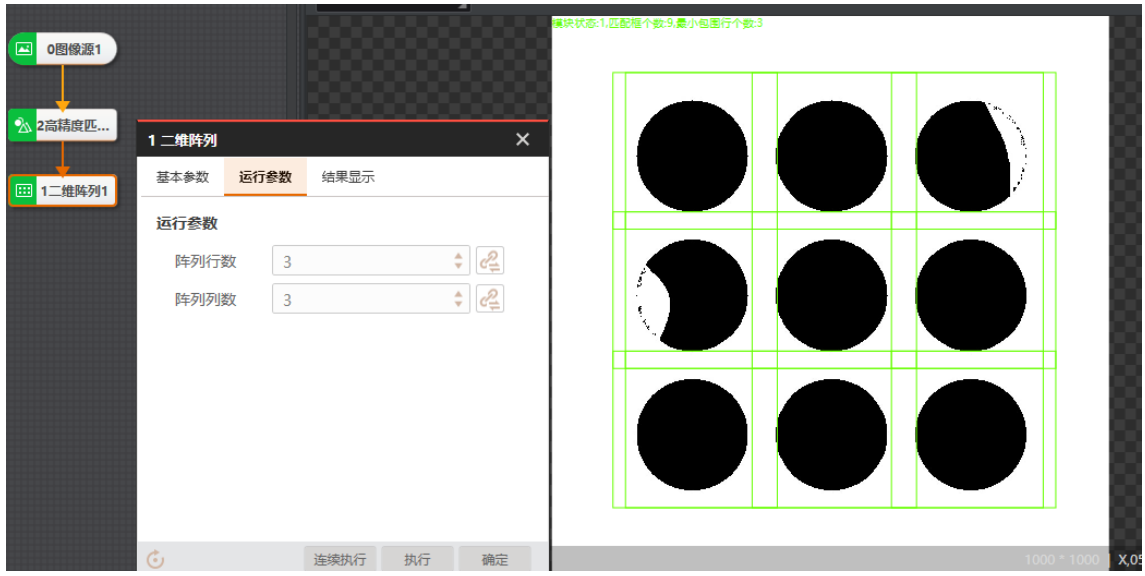


图 13-475 二维阵列执行结果

参数配置

以下仅介绍该模块的运行参数详情。通过配置运行参数，可定义当前模块如何处理输入的数据。

说明

- 基本参数详情，请参见 *基本参数* 。
- 结果显示参数详情，请参见 *结果显示* 。

阵列行数

二维阵列的行数，即拟合水平直线的最大条数。

阵列列数

二维阵列的列数，即拟合垂直直线的最大条数。

模块结果

*二维阵列*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

修正匹配框个数

int 型，代表修正后匹配框的个数。

修正匹配框

修正框中心

修正框中心 X

float 型，代表修正后匹配框的中心点 X，数组，其长度等于修正匹配框个数 。

修正框中心 Y

float 型，代表修正后匹配框的中心点 Y，数组，其长度等于修正匹配框个数 。

修正框宽度

float 型，代表修正后匹配框的宽度，数组，其长度等于修正匹配框个数。

修正框高度

float 型，代表修正后匹配框的高度，数组，其长度等于修正匹配框个数。

修正框角度

float 型，代表修正后匹配框的角度，数组，其长度等于修正匹配框个数。

最小包围行 Box 个数

int 型，代表行的最小包围的个数。

最小包围行矩形

最小包围行中心

最小包围行中心 X

float 型，代表行的最小包围的中心点 X，数组，其长度等于最小包围行 Box 个数。

最小包围行中心 Y

float 型，代表行的最小包围的中心点 Y，数组，其长度等于最小包围行 Box 个数。

最小包围行宽度

float 型，代表行的最小包围的宽度，数组，其长度等于最小包围行 Box 个数 。

最小包围行高度

float 型，代表行的最小包围的高度，数组，其长度等于最小包围行 Box 个数 。

最小包围行角度

float 型，代表行的最小包围的角度，数组，其长度等于最小包围行 Box 个数 。

13.12.3 多标签筛选

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

多标签筛选模块用于对实例分割掩膜图进行二值化处理，即将输入图像中指定像素列表的像素值设置为 255（显示白色），其余设置为 0（显示黑色）。
假如下图中各个数字代表像素值，当设置像素列表为【1, 5, 8】时，则对应位置的像素值均会变为 255，其余位置的像素值会变为 0。

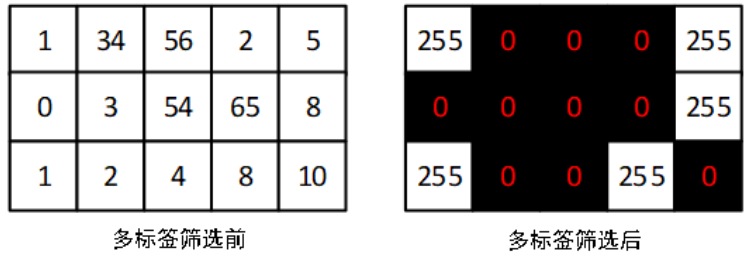


图 13-476 多标签筛选前后对比图

使用方法

该模块一般配合 DL 实例分割模块使用，类别编号订阅实例分割模块的目标类别。将实例分割模块输出的类别进行筛选，并输出符合要求的区域。

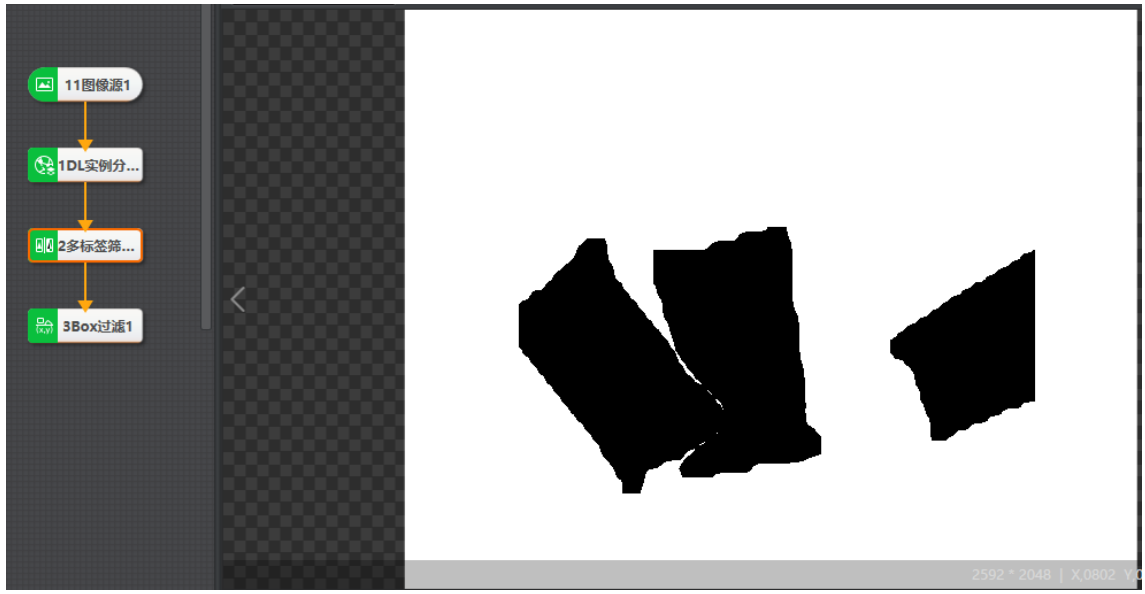


图 13-477 多标签筛选示例效果图

说明

由于实例分割识别的三个类别的灰度值分别为 1、2、3，其余区域的灰度值为 0；而多标签筛选模块的**类别编号**订阅的 DL 实例分割目标类别结果为 0，故在多标签筛选模块输出图像中，三个类别对应的灰度值被设为 0，其余背景对应的灰度值被设为 255。

参数配置

输入源

需订阅进行多标签筛选的图像源。

类别编号

需订阅像素列表，用于筛选对应类别的区域。比如：与 DL 实例分割模块配合使用时，**类别编号**可订阅 DL 实例分割模块的目标类别。



图 13-478 模块配置

模块结果

*多标签筛选*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出图像

输出图像数据

binary 型，代表输出图像的二进制数据。

输出图像宽度

int 型，代表输出图像的宽度。

输出图像高度

int 型，代表输出图像的高度。

输出图像像素格式

int 型，代表输出图像像素格式对应的枚举值。其中“17301505”对应 Mono 8 格式，“35127316”对应 RGB 24 格式。

13.12.4 Box 融合

Box 融合可将多个重叠率满足要求的 Box 或多个 Box 中心的距离满足要求的 Box 融合成一个 Box 并输出。

说明

Box 的含义：用于标记目标物体位置和大小的、带角度信息的矩形。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*Box 融合*模块的处理原理如下图所示，主要包含“计算融合 Box 列表”和“计算多 Box 的最小包围 Box 及标签”两个步骤。

1. 计算融合 Box 列表：将需要融合的 Box 编号记录在一个列表中，判断 Box 是否能够融合。判断条件有以下两种。
   - 判断两个 Box 的重叠率大于设定值。
   - 判断两个 Box 中心点的距离是否在设定值的范围内。
2. 计算多 Box 的最小包围 Box 及标签：
   - 计算多 Box 的最小包围 Box：先取多个 Box 的顶点，再计算点集的最小包围 Box 作为融合后的 Box。
   - 计算多 Box 的标签：根据设置的融合类型进行处理。

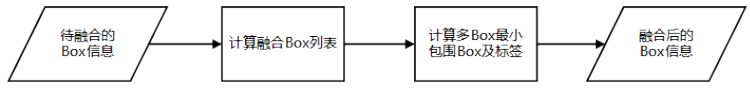


图 13-479 Box 融合原理

融合类型分为三种，具体如下：

- **无条件融合并按缺陷优先级输出标签**：对满足要求且标签在**标签优先级**中的 Box 进行融合，融合后的标签取优先级更高的标签。
- **按标签类别融合并输出标签**：对满足要求且标签相同的 Box 进行融合。
- **无条件融合并随机输出标签**：对满足要求且标签相同的 Box 进行融合，融合后的标签随机取融合前 Box 的标签。此时可不设置 **Box 标签**。

下图以 Box 重叠率作为判断条件进行说明。

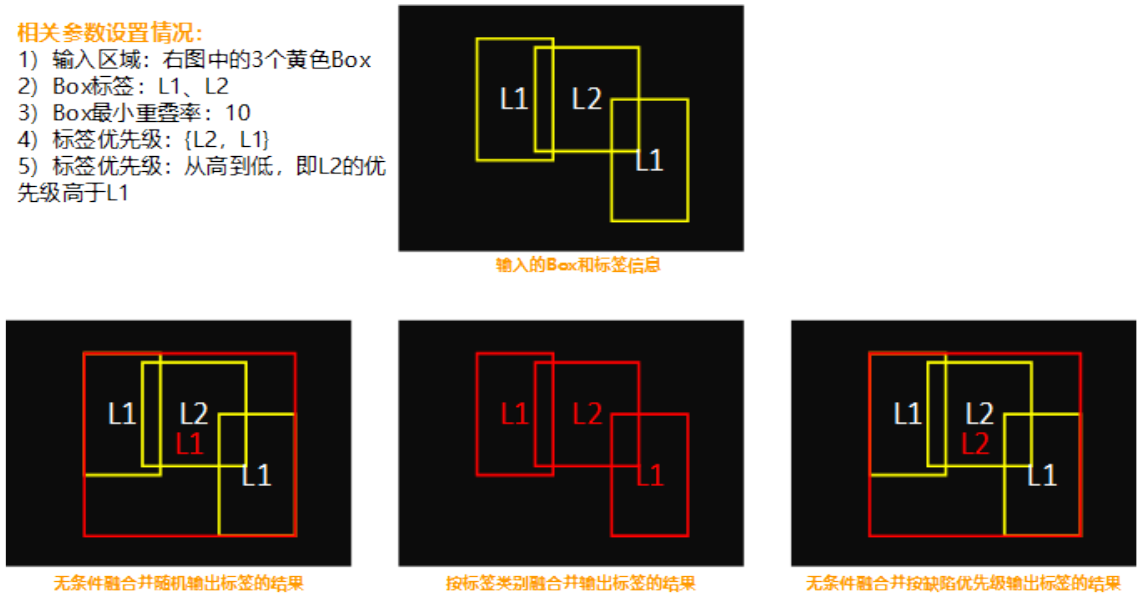


图 13-480 Box 融合效果图

使用方法

当多个检测对象存在重叠（例如图像分割、目标检测、模板匹配等工具的结果）时，若需通过重叠率或 Box 之间的距离、以及优先级等规则对重叠对象进行融合，则可使用 Box 融合模块将多个满足要求的 Box 融合成一个 Box 并输出，如下图所示的圆形、三角形和五角星。

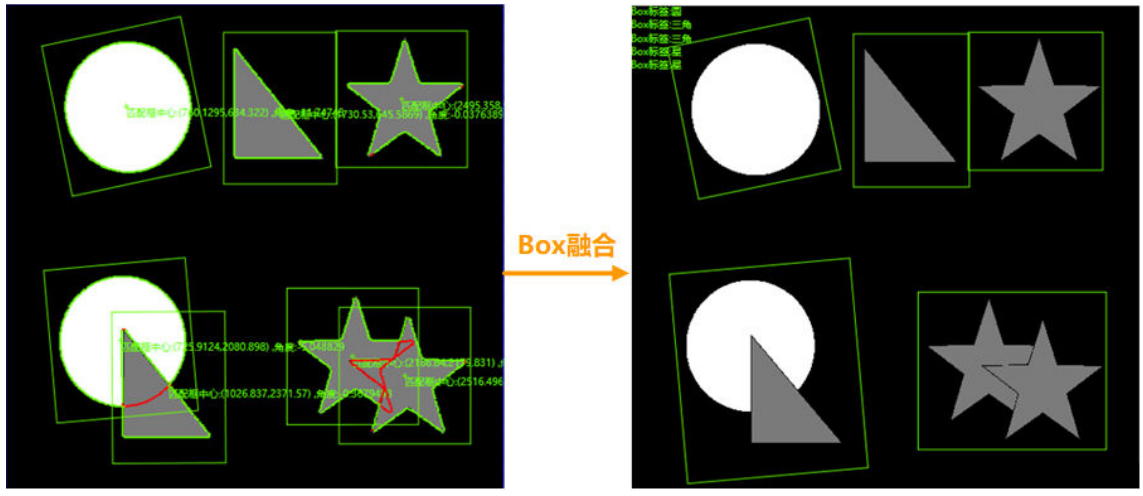


图 13-481 Box 融合示意图

- 该模块可与图像分割、目标检测模块进行配合使用。对于图像分割、目标检测等模块输出缺陷对象的最小外接矩形，根据融合规则进行图像的融合。
- 该模块还可与模板匹配等模块进行配合使用，标签可以使用对应的模板名称。
- 另外，若要对多个模块输出的 Box 进行汇总融合，可通过脚本的方式输出多个模块的 Box 以及对应的 string 类型名称数组，再通过 Box 融合模块进行融合。

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  按区域/按参数

  订阅需进行 Box 融合的区域。可选*按区域*或*按参数*。

  按区域

  通过**区域**参数订阅 Box 区域。

  按参数

  通过 **ROI 中心点 X/Y**、**ROI 宽度/高度**、**ROI 角度**参数订阅 Box 区域。

  Box 标签

  订阅 box 的标签信息。

- 运行参数：

  融合依据

  可设置多个 Box 的融合依据，可选重叠率或 Box 中心之间的距离。

  重叠率

  通过**重叠率**参数设置 Box 重合时的最小重叠率。

  说明

  两个 Box 之间的重叠部分占较小 Box 面积的比例。

  Box 中心之间的距离

  根据设置的 Box 中心点之间 X 方向和 Y 方向的距离判断是否融合。通过 **X/Y 方向的距离**设置最大距离。

  Box 融合类型

  可选三种融合类型。

  无条件融合并按缺陷优先级输出标签

  对支持的标签且满足融合依据的 Box 区域进行合并，合并后输出优先级较高的 Box 标签。此时需设置以下参数。

标签优先级

订阅标签的优先级，一般通过脚本、格式化或全局变量的方式订阅实现。

标签分割符

设置各个标签之间的分割符，下拉选择即可。

优先级排序

调整标签的优先级，可选升序、降序。

按标签类别融合并输出标签

对具有相同标签且满足融合依据的 Box 区域进行合并，合并后输出共同的标签。需订阅 **Box 标签**。

无条件融合并随机输出标签

对满足融合依据的 Box 区域进行合并，合并后随机输出一个标签。

模块结果

*Box 融合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

重合 Box 个数

int 型，代表经过融合后得到的 Box 的数量，其余 Box 由于重叠率未达到阈值所以两两之间不能被融合。

Box 标签

string 型，代表经过融合后得到的 Box 的标签，数组，其长度等于重合 Box 个数。

Box 融合

融合中心点

中心点 X

float 型，代表经过融合之后剩下 Box 的中心点 X，数组，其长度等于“重合 Box 个数”。

中心点 Y

float 型，代表经过融合之后剩下 Box 的中心点 Y，数组，其长度等于“重合 Box 个数”。

宽度

float 型，代表经过融合后得到的 Box 的宽度（Box 的长边），数组，其长度等于重合 Box 个数。

高度

float 型，代表经过融合后得到的 Box 的高度（Box 的短边），数组，其长度等于重合 Box 个数。

角度

float 型，代表经过融合后得到的 Box 的角度，是矩形框长边与图像坐标系 X 轴的夹角，图像坐标系如下图所示。数组，其长度等于重合 Box 个数。

13.12.5 Box 重叠率计算

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

*Box 重叠率计算*模块可计算并输出两个选定 Box 的重叠率。结果范围为 0~1，数值为 float 型。0 表示两个 Box 不重叠，1 表示两个 Box 完全重叠。
该模块原理为先计算两个 Box 交集的面积（分子），再计算最小 Box 的面积（分母），两个面积之比即两个 Box 的重叠率。以下图为例，橙色区域为交集，黄色和橙色区域为最小 Box。

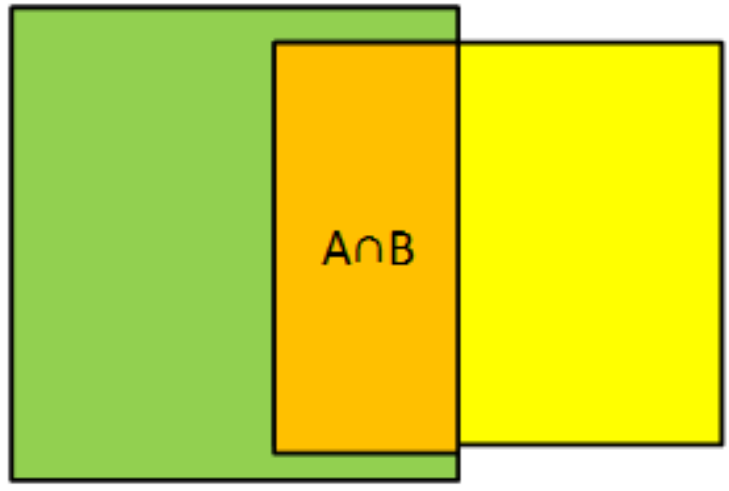


图 13-482 Box 重叠示意图

使用方法

当遇到多个检测对象重叠时，需要计算其重叠率，便于后续筛选或其他需求。输出的重叠率结果范围为 0~1，数值为 float 型。0 表示两个 Box 不重叠，1 表示两个 Box 完全重叠。
该模块无需与特定模块配合使用，仅需要输入一组 Box 即可，常用于目标检测、图像分割等模块计算重叠率时使用。该模块操作时，通过基本参数选定两个 Box 区域，且仅支持按区域参数方式订阅。Box 重叠模块执行结果如下所示。

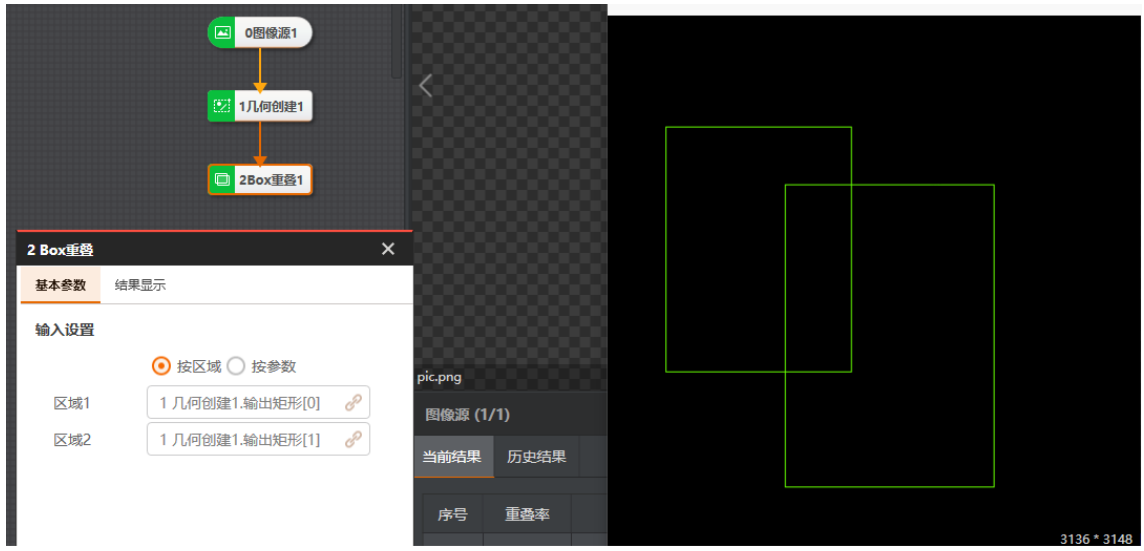


图 13-483 Box 重叠执行结果

参数配置

该模块操作时，通过基本参数选定两个 Box 区域即可，支持按区域按参数方式订阅。

说明

结果显示参数详情，请参见 *结果显示* 。

模块结果

*Box 重叠率计算*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

重叠率

float 型，范围为[0, 1]，0 代表两个 Box 不重叠，1 代表两个 Box 完全重叠 。

13.12.6 Box 过滤

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

Box 过滤模块基于输入的掩膜二值图，将目标重叠率低于设定值的 Box 区域过滤掉。目标重叠率即每个 Box 区域中像素为 255 的区域与整个 Box 区域的比值。
假设模块输入的图像为下图的二值图像，其中 3 种不同颜色的矩形框表示 3 个输入的 Box。模块先计算每个 Box 中像素值为 255（代表白色）的像素个数 N，若 N 与 Box 区域的比例小于设置的重叠率，则过滤；否则不过滤掉该 Box 区域。
若重叠率设置为 0.1，则绿色和蓝色的 Box 保留，红色的 Box 会被过滤。

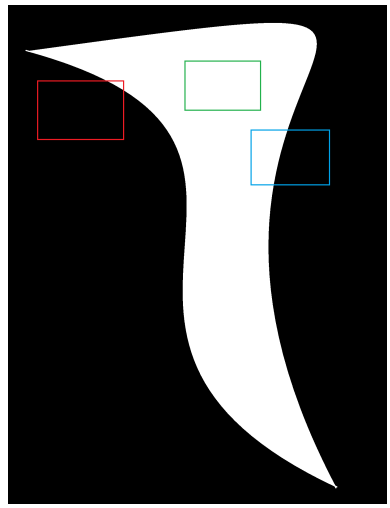


图 13-484 Box 过滤原理示意

使用方法

该模块推荐放在多标签筛选模块后使用。

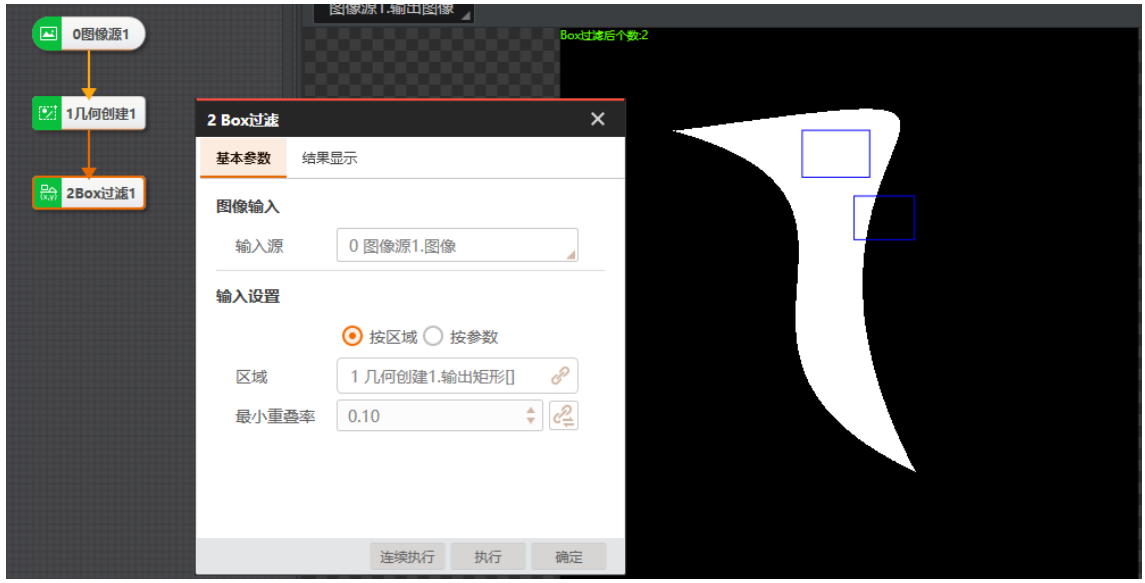


图 13-485 Box 过滤执行结果

参数配置

根据区域或参数的方式订阅 Box 区域，并设置最小重叠率参数即可。重叠率为二值图的 Box 区域内像素值为 255 的像素数 / 整个 Box 区域的像素数。

说明

结果显示参数详情，请参见 *结果显示* 。

模块结果

*Box 过滤*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

Box 索引

int 型，代表过滤后 Box 的索引。

Box 过滤后个数

int 型，代表过滤后 Box 的个数。

目标信息矩形

目标信息中心点

目标信息中心 X

float 型，代表每个定位框中心点的 X 坐标。

目标信息中心 Y

float 型，代表每个定位框中心点的 Y 坐标。

目标信息矩形宽度

float 型，代表每个定位框的宽度。

目标信息矩形高度

float 型，代表每个定位框的高度。

目标信息矩形角度

float 型，代表字符检测框中字符方向的底边与水平线的夹角。从水平线开始顺时针为正，逆时针为负数，范围是[-90，90]。

13.13 图形生成

图形生成分类下的模块可根据多个点拟合成圆、椭圆或直线，还可用于创建几何图像，包括圆拟合、椭圆拟合、直线拟合和几何创建模块。

13.13.1 圆拟合

圆拟合先检测有效特征点，形成点集后再拟合成圆。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

圆拟合用于将一组特征点拟合成一个圆，不能拟合成多个圆，往往用在圆查找提取特征点之后的阶段。使用圆拟合的前提是输入的特征点（至少为 3 个点）能大致组成一个圆。圆拟合可用于圆的测量。
如下图所示，紫色的点为输入的特征点，拟合这些边缘点后可得到一个绿色的圆。

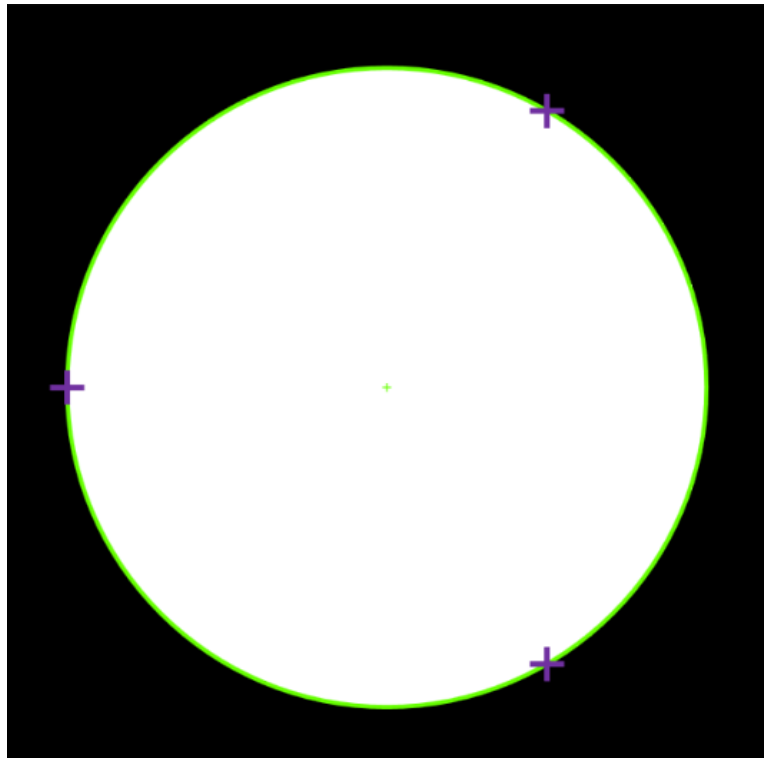


图 13-486 圆拟合示意图

圆拟合根据初始拟合方式得到初始圆，然后根据不同的权重函数迭代进行拟合，直到拟合结果满足要求。另外，在拟合的过程中会根据剔除的点数和剔除的距离舍弃偏离的点，所以圆拟合具有部分抗干扰点的能力。

使用方法

圆拟合可通过特征点状态情况拟合出一个圆，通过对该拟合圆的状态信息进行评估，检测该组特征点是否符合标准要求，常应用于“工件标准检测”、“车道检测”等。
以下示例通过 group 找到工件的所有顶点，并通过点集收集传给圆拟合，检测其生成的图形是否为标准的最小外接圆。

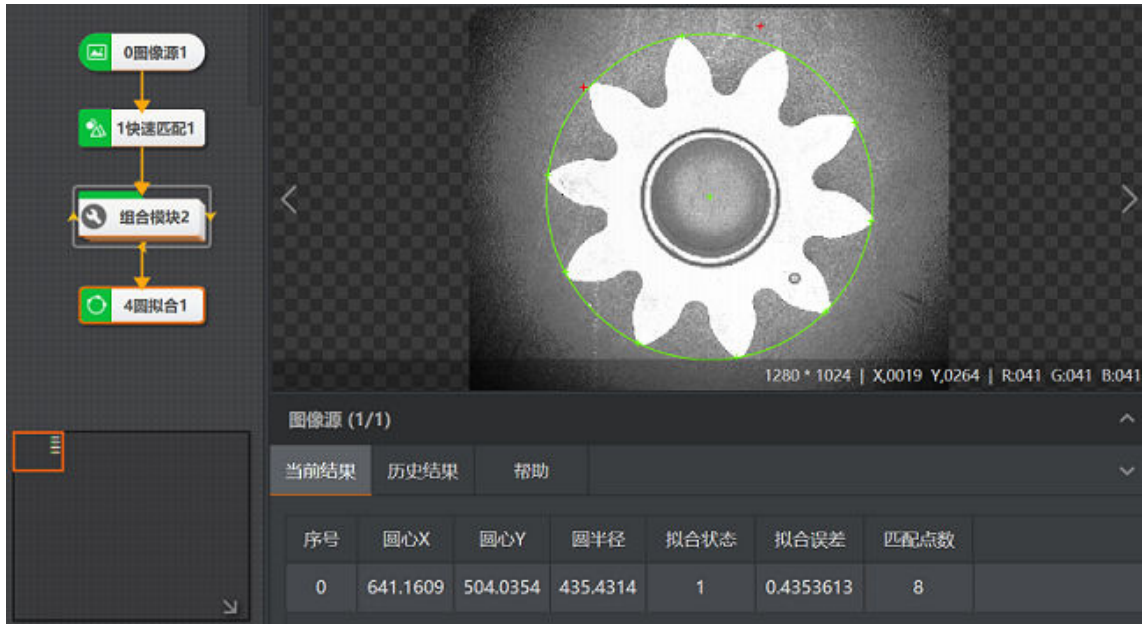


图 13-487 圆拟合使用示例

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  图像输入

  可设置需输入的图像源。

  输入源

  可选择前序模块的图像数据。

  说明

  该参数可不设置。

  拟合点

  选择流程中采集到的点集作为拟合来源。可通过**输入方式**参数选择按点或按坐标方式订阅点的数据。

- 运行参数：

  剔除点数

  误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，为获取更佳查找效果，建议与**剔除距离**结合使用。

  剔除距离

允许离群点到拟合直线的最大像素距离，值越小，排除点越多。

初始化类型

包括全局法和局部最优两种。

- 局部最优：按照局部的特征点拟合圆，如果局部特征更加准确反映圆所在位置，则采用局部最优，否则采用全局最优。
- 全局法：以查找到的全局特征点进行圆拟合。

权重函数

包括最小二乘、huber 和 tukey 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用最小二乘、huber、tukey。

最大迭代次数

拟合算法的最大执行次数。

模块结果

*圆拟合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出圆环

ANNULUS 型，代表模块查找到的圆，记录了该圆的圆心、内径和角度等信息。

圆心

point 型，表示查找到的圆的圆心坐标。

圆心 X

float 型，代表圆心的 X 坐标。

圆心 Y

float 型，代表圆心的 Y 坐标。

内径

float 型，代表圆的内径。

圆半径

float 型，代表圆的半径。

圆起始角度

float 型，代表圆起始边与水平方向的夹角。

圆角度范围

float 型，代表圆起始边与终止边跨过的角度范围。

输出拟合点

point 型，代表每一个输出的拟合点。

输出拟合点 X

float 型，代表每一个用于拟合点的坐标 X。

输出拟合点 Y

float 型，代表每一个用于拟合点的坐标 Y。

拟合误差

float 型，用于描述拟合图形的准确性，越小越好。

拟合状态

int 型，代表输出拟合图形的状态。1 代表存在拟合图形，0 代表不存在拟合图形。

拟合点数

int 型，代表用于拟合的输入点数。

拟合距离

float 型，代表拟合点与生成拟合图形之间的最近距离。

匹配点

int 型，代表每一个输入点的匹配状态。0 代表不用于拟合，1 代表用于拟合。

匹配点数

int 型，代表**匹配点**为 1 的点个数。

13.13.2 椭圆拟合

椭圆拟合先检测有效特征点后，形成点集后再拟合成椭圆。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

椭圆拟合用于将一组特征点拟合成一个椭圆，不能拟合成多个椭圆，往往用在椭圆查找提取特征点之后的阶段。使用椭圆拟合的前提是输入的特征点（至少为 5 个点）能大致组成一个椭圆。椭圆拟合可以用于椭圆的测量。

如下图所示，图中绿色和红色点为输入的特征点，其中绿色为最终参与拟合的特征点，红色为被删除掉的特征点，绿色椭圆为椭圆拟合结果。

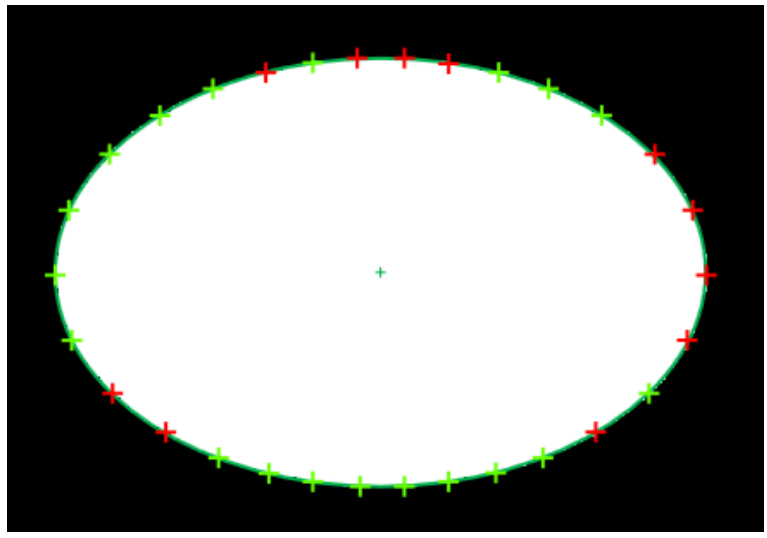


图 13-488 椭圆拟合示意图

使用方法

椭圆拟合可通过特征点状态情况拟合出一个标准的椭圆，通过对该拟合椭圆的状态信息进行评估，检测该组特征点是否符合标准要求，常见应用“工件标准检测”、“轨道检测”等。
椭圆拟合模块一般与点集模块进行配合使用。以下示例展示通过直线查找找到四个特征点，再通过几何创建建立中点，再将五个点共同输入点集，拟合成一个椭圆，由此来判断其十字中心是否符合要求。



图 13-489 椭圆拟合使用示例



图 13-490 椭圆拟合执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  拟合点

  选择流程中采集到的点集作为拟合来源。可通过**输入方式**参数选择按点或按坐标方式订阅点的数据。

- 运行参数：

  误差容忍度

  在进行椭圆拟合时会计算每个参与拟合的边缘点的拟合误差，该参数值越小时，拟合时删除掉的边缘点越多，拟合结果更加准确，但该值设置的过于小时，有可能造成过拟合。

模块结果

*椭圆拟合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出椭圆

圆心点

point 型，表示查找到的椭圆的圆心点。

中心 X

float 型，代表圆心点的 X 坐标。

中心 Y

float 型，代表圆心点的 Y 坐标。

椭圆长半轴

float 型，代表椭圆的长半轴长度。

椭圆短半轴

float 型，代表椭圆的短半轴长度。

椭圆角度

float 型，代表椭圆的角度。

输出拟合点

point 型，代表每一个输出的拟合点。

输出拟合点 X

float 型，代表每一个用于拟合点的坐标 X。

输出拟合点 Y

float 型，代表每一个用于拟合点的坐标 Y。

拟合误差

float 型，用于描述拟合图形的准确性，越小越好。

拟合状态

int 型，代表输出拟合图形的状态。1 代表存在拟合图形，0 代表不存在拟合图形。

拟合点数

int 型，代表用于拟合的输入点数。

匹配点

int 型，代表每一个输入点的匹配状态。0 代表不用于拟合，1 代表用于拟合。

匹配点数

int 型，代表**匹配点**为 1 的点个数。

13.13.3 直线拟合

直线拟合基于两个及以上的已知点拟合成直线，先检测顶点再拟合成直线。

本节内容包含：

- *模块原理*
- *使用方法*
- *参数配置*
- *模块结果*

模块原理

直线拟合用于将一组特征点拟合成一条直线，不能拟合成多条直线，往往用在直线查找提取特征点之后的阶段。使用直线拟合的前提是输入的特征点（至少为 2 个点）能大致组成一条直线。直线查找可用于直线的测量。
如下图所示，紫色的点为输入的特征点，拟合这些边缘点后可得到一条绿色的直线。



图 13-491 直线拟合示意图

直线拟合根据初始拟合方式得到初始直线，然后根据不同的权重函数迭代进行拟合，直到拟合结果满足要求。另外，在拟合的过程中会根据剔除的点数和剔除的距离舍弃偏离的点，所以直线拟合具有部分抗干扰点的能力。

使用方法

直线拟合可通过一组特征点状态情况拟合出一个标准的直线，通过对该拟合直线的信息进行评估，检测该组特征点是否符合标准要求，常应用于“宽度检测”、“车道检测”等。
直线拟合模块一般与点集模块进行配合使用，示例效果如下图所示。

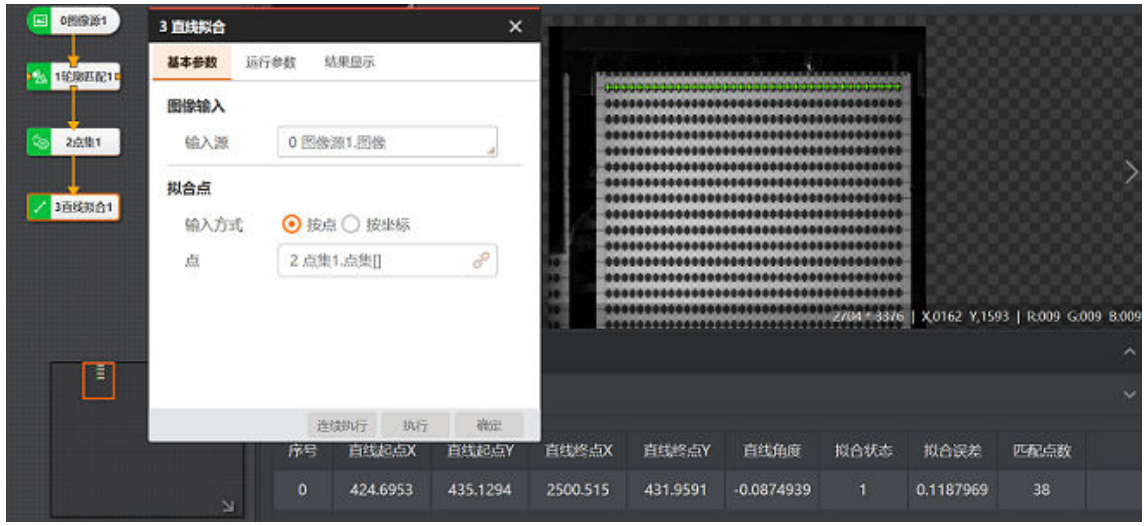


图 13-492 直线拟合执行结果

参数配置

此处仅对该模块的基本参数以及运行参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

- 基本参数：

  图像输入

  可设置需输入的图像源。

输入源

可选择前序模块的图像数据。

说明

该参数可不设置。

拟合点

选择流程中采集到的点集作为拟合来源。可通过**输入方式**参数选择按点或按坐标方式订阅点的数据。

- 运行参数：

剔除点数

误差过大而被排除不参与拟合的最小点数量。一般情况下，离群点越多，该值应设置越大，为获取更佳查找效果，建议与**剔除距离**结合使用。

剔除距离

允许离群点到拟合直线的最大像素距离，值越小，排除点越多。

初始化类型

包括全局法和局部最优两种。

- 局部最优：按照局部的特征点拟合直线，如果局部特征更加准确反映直线所在位置，则采用局部最优，否则采用全局最优。
- 全局法：以查找到的全局特征点进行圆拟合。

权重函数

包括最小二乘、huber 和 tukey 三种。三种拟合方式只是权重的计算方式有些差异。随着离群点数量增多以及离群距离增大，可逐次使用最小二乘、huber、tukey。

最大迭代次数

拟合算法的最大执行次数。

角度拟合约束

开启后，根据**期望直线角度**和**直线角度容忍范围**设置的数值判断直线的角度是否符合要求。

假设**期望直线角度**设为 10，**直线角度容忍范围**设为 20，则直线角度在-10° ~ 30°之间为符合要求。

模块结果

*直线拟合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

输出直线

直线起点

point 型，代表输出直线的起点坐标。

直线起点 X

float 型，代表输出直线起点的 X 坐标。

直线起点 Y

float 型，代表输出直线起点的 Y 坐标。

直线终点

point 型，代表输出直线的终点坐标。

直线终点 X

float 型，代表输出直线终点的 X 坐标。

直线终点 Y

float 型，代表输出直线终点的 Y 坐标。

直线角度

float 型，代表输出直线的角度。

输出拟合点

point 型，代表每一个输出的拟合点。

输出拟合点 X

float 型，代表每一个用于拟合点的坐标 X。

输出拟合点 Y

float 型，代表每一个用于拟合点的坐标 Y。

拟合误差

float 型，用于描述拟合图形的准确性，越小越好。

拟合状态

int 型，代表输出拟合图形的状态。1 代表存在拟合图形，0 代表不存在拟合图形。

拟合点数

int 型，代表用于拟合的输入点数。

拟合距离

float 型，代表拟合点与生成拟合图形之间的最近距离。

匹配点

int 型，代表每一个输入点的匹配状态。0 代表不用于拟合，1 代表用于拟合。

匹配点数

int 型，代表**匹配点**为 1 的点个数。

13.13.4 几何创建

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

几何创建可用于自由创建或继承相关的点、线、圆和矩形。该模块常应用于检测或测量应用中，作为基础标准配合检测。
当存在有些图形定位较难时，可通过鼠标移动或者修改 X、Y 坐标来改变生成图形的位置。如下图所示，圆查找不好定位的白色圆轮廓可以自行创建。



图 13-493 几何创建执行结果

参数配置

输入源

订阅前序模块输出的图像。

输入类型

选择绘制图像的输入类型，可选*绘制*或*订阅*。

绘制

需绘制形状，选择*形状*处的图标后在预览图像中进行绘制即可，可选形状如下。

- 矩形：使用鼠标在图中绘制，然后移动或者修改中心点的 X、Y 坐标、宽、高、和角度，可以配合位置修正使用。
- 点：使用鼠标在图中单击生成，然后移动或者修改 X、Y 坐标，可以配合位置修正使用。
- 直线：使用鼠标在图中拖动生成，然后移动端点或者修改端点 X、Y 坐标，可以配合位置修正使用。
- 圆：使用鼠标在图中单击生成，然后移动或者修改 X、Y 坐标，可以配合位置修正使用。

订阅

订阅前序模块中的图形或参数，选择对应*图形类型*和*输入方式*并进行订阅图形或参数，**图形类型**包括点、线、圆、矩形四种。

位置修正

使用该功能时，需确保前序模块中有位置修正模块。相关功能和原理参见 *位置修正* 章节。使用该功能时，需完成以下参数的设置。

选择方式

可选择位置修正信息的获取方式，可选**按信息**、**按点**和**按坐标**。无论选择哪种方式，均需订阅相关参数。

模块结果

*几何创建*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

点个数

int 型，代表创建点的个数。

直线个数

int 型，代表创建直线的个数。

圆个数

int 型，代表创建圆的个数。

矩形个数

int 型，代表创建矩形的个数。

输出点

point 型，代表输入点旋转后得到的点。

输出点 X

float 型，代表输入点旋转后得到点的 X 坐标。

说明

采用的图像坐标系为左手坐标系。

输出点 Y

float 型，代表输入点旋转后得到点的 Y 坐标。

输出直线

起点

起点 X

float 型，代表数组，创建直线的起点坐标 X。

起点 Y

float 型，代表数组，创建直线的起点坐标 Y。

终点

终点 X

float 型，代表数组，创建直线的终点坐标 X。

终点 Y

float 型，代表数组，创建直线的终点坐标 Y。

直线角度

float 型，代表输出直线的角度。

输出圆

圆心

point 型，表示查找到的圆的圆心坐标。

圆心 X

float 型，代表圆心的 X 坐标。

圆心 Y

float 型，代表圆心的 Y 坐标。

内径

float 型，代表圆的内径。

半径

float 型，代表数组，创建圆的半径。

起始角度

float 型，代表数组，创建圆的起始角度（默认为 0）。

角度范围

float 型，代表数组，创建圆的角度范围（默认为 360）。

输出矩形

矩形中心点

矩形中心点 X

float 型，代表矩形中心点的 X 轴坐标。

矩形中心点 Y

float 型，代表矩形中心点的 Y 轴坐标。

矩形宽度

float 型，代表最小外接矩形的宽度。

矩形高度

float 型，代表最小外接矩形的高度。

矩形角度

float 型，代表识别到码的矩形框角度，由码的点 0 和点 1 所连直线与水平线夹角构成，顺时针为正，逆时针为负。

说明

一般码有 4 个角点，按顺序分别为点 0、点 1、点 2、点 3，不同码制的顶点规则和标准有所差异，具体请以实际为准。

13.14 逻辑工具

逻辑工具分类下的模块可进行条件检测、逻辑判断等，包括条件分支、条件检测、逻辑等模块。

13.14.1 条件分支

条件分支模块结合条件检测和分支模块的功能，当订阅的条件符合要求时执行设定的模块。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

条件分支模块多用于工业生产对结果的判断、产品分类等方面的使用，根据判断结果确认当前检测类别以及是否进行后续操作等场景。
该模块可以配合逻辑工具的全系列模块进行使用，通过判断前序模块的输出结果，确认是否继续输入至后序模块执行，以及控制后续按照条件对应的分支进行执行。

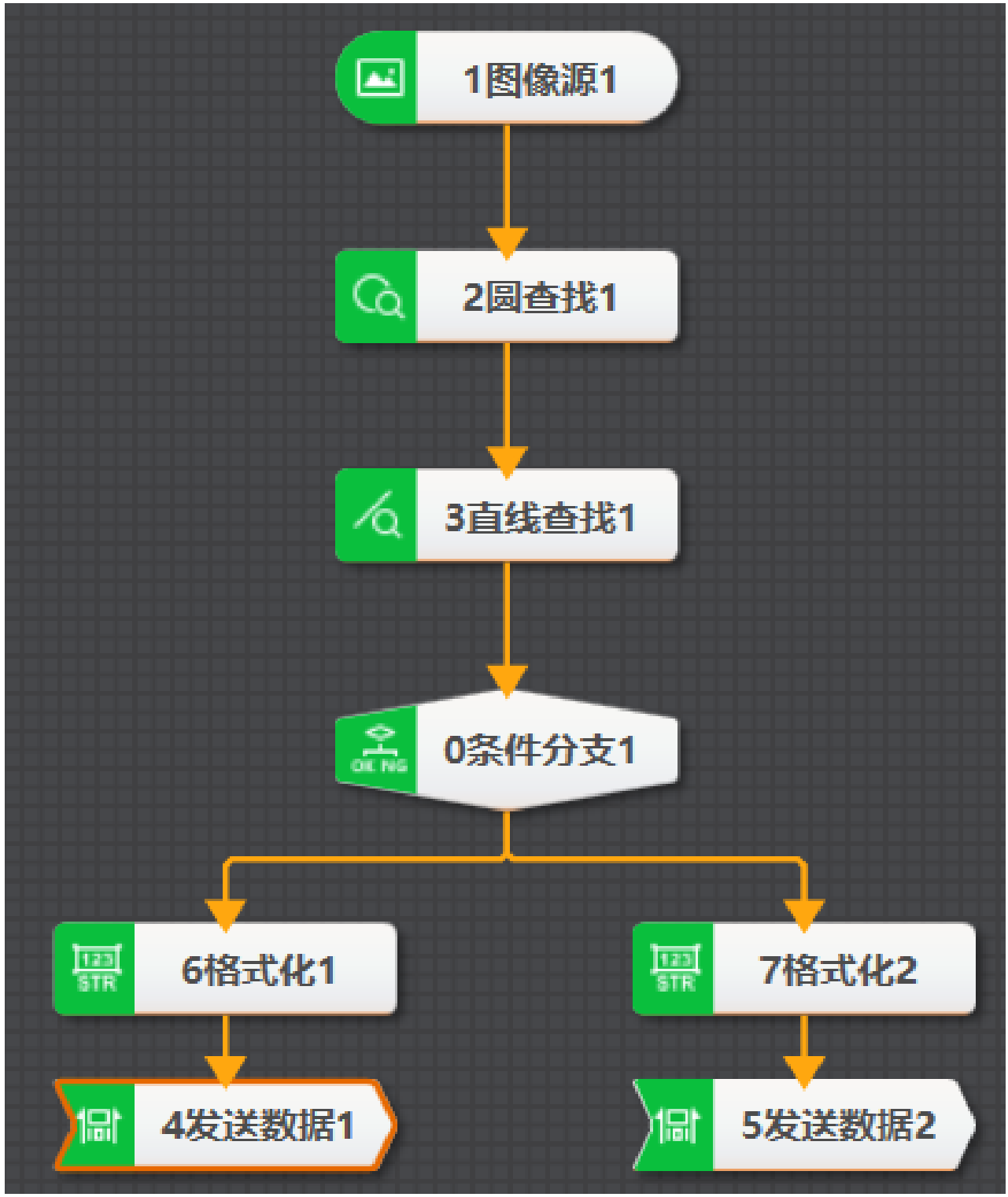


图 13-494 条件分支示例

该示例通过对图像工件中的圆和线进行定位，然后通过条件分支进行判断，确认查找结果。如果圆和线定位正常，则将正常结果（OK 信号）通过发送数据进行反馈；若定位异常，则不进行对应发送。



图 13-495 条件分支执行流程

参数配置

双击条件分支模块可进入参数配置窗口，配置步骤如下：

1. 根据实际需求选择需判断条件的类型，可选 int 或 float。
2. 单击 ⊞ 可添加一条数据。
3. 通过 🔗 订阅前序模块中需作为判断条件的数据来源。
4. 设置判断为**有效值范围**的区间。当条件数据在该范围内时为 OK，否则为 NG。
5. 设置执行模块的**判断**依据，选择 **OK** 还是 **NG** 的情况下执行后续模块。
6. 在**执行模块 ID** 处选择满足条件时执行的模块 ID，只能选择条件分支模块后面直接连接的模块。

模块结果

*条件分支*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

分支路线

string 型，代表满足条件可执行的分支模块 ID 列表。

13.14.2 条件检测

条件检测模块用于判断输入数据是否满足条件。若满足，显示 OK 字符；否则，显示 NG 字符。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

条件检测模块多用于工业生产对结果的判断，通过判断的结果确认当前结果是否符合条件、参数范围是否正常、检测结果是否符合要求等应用场景。

该模块可以配合逻辑工具的全系列模块进行使用，通过判断前序模块的输出结果，确认该结果是否符合条件检测模块中自定义的范围，从而判断是否符合现场使用要求，后续可以搭配分支模块对条件结果进行分支处理。

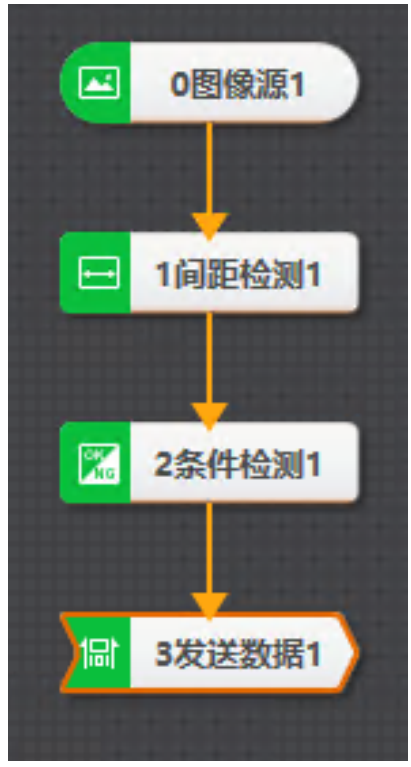


图 13-496 条件检测示例

该示例通过间距检测模块对固件的宽度进行检测，然后通过条件检测模块进行判断，当宽度处于 500~1000 且结果评分达到 0.5~1 时，判断该结果为 OK，否则为 NG，最后通过后续的发送数据模块将检测结果发出。

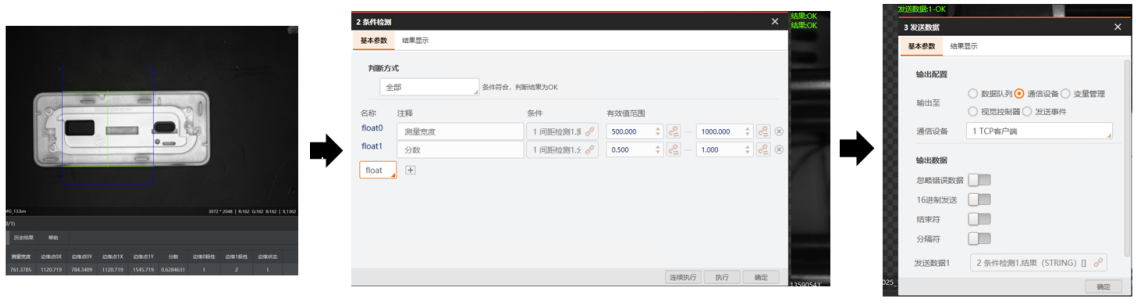


图 13-497 条件检测示例执行流程

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见*结果显示*。

判断方式

可选任意或全部。

任意

对判断条件中任意判断结果进行逻辑或运算。

全部

对判断条件中所有判断结果进行逻辑与运算。

判断条件

点击 ⊞ 可添加一条判断条件。每条判断条件由以下参数组成。

名称

可选条件的数据类型，分为 int 和 float 两种类型。

注释

可对该条数据通过注释说明。

条件

订阅需作为判断条件的数据来源，但数据类型需和**条件类型**对应。

有效值范围

符合该数据范围内的判定为 OK，否则判定为 NG。

说明

有效值范围处填写的数据，其小数点后的位数可通过 VM 安装路径下的 IfModuleAlgorithmTab.xml 进行配置——修改该 xml 文件中的 DecimalDigits 的数值即可，最多支持 3 位。以 64 位系统为例，该 xml 文件的相对路径为.\Applications\Module(sp)\x64\Logic\IfModule。

模块结果

*条件检测*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

结果（STRING）

string 型，所有条件项结果都为 OK 则为 OK，否则为 NG。

结果（INT）

int 型，代表所有条件项状态都为 1 则为 1，否则为 0，与结果（STRING）对应。

条件项结果

string 型，代表数组，每个条件项的结果（NG 或 OK）构成的数组。

条件项状态

int 型，代表数组，每个条件项的状态（0 或 1）构成的数组，与条件项结果对应。

13.14.3 分支模块

分支模块工具可以配置输入条件，并根据方案实际需求对不同的分支模块配置不同的条件输入值。当输入条件为该值时，则会执行该分支下的模块。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

分支模块多用于工业生产对结果的判断，通过判断结果确认当前结果是否符合条件、参数范围是否正常、检测结果是否符合要求等。
该模块可以配合逻辑工具的全系列模块进行使用，目前该模块输入值仅支持整数，不支持字符串。通过判断前序模块的输入值，确认该值是否有对应的分支条件。如果没有对应的分支条件，则不再继续往下执行；若有对应的分支条件，则执行对应分支下的模块。该模块可以实现单个流程中对不同条件结果分开处理的应用。

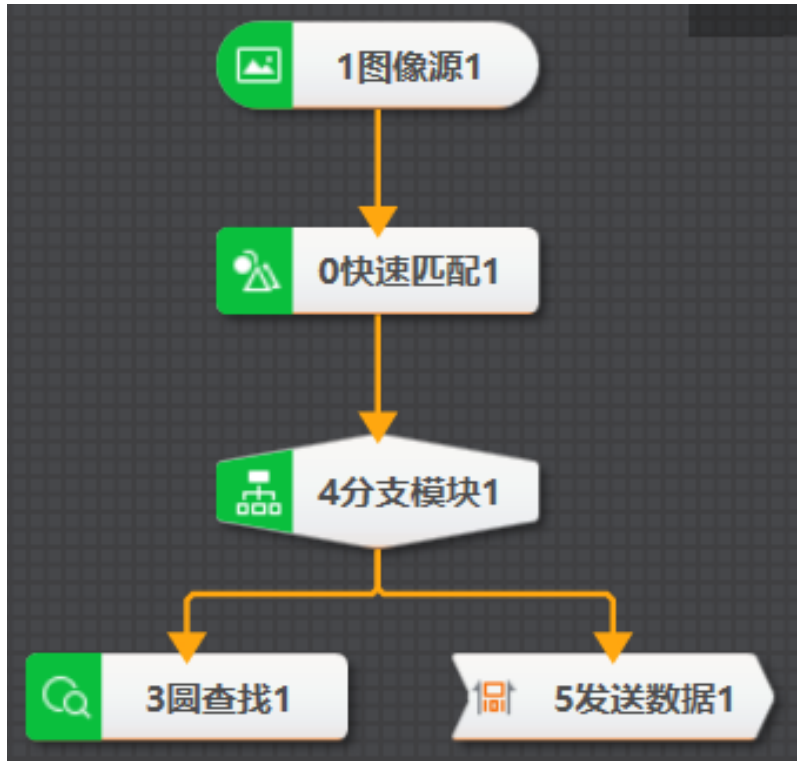


图 13-498 分支模块示例

该示例通过快速匹配模块对工件进行初步定位，再通过分支模块订阅匹配模块的状态，当匹配到对应工件时，执行圆查找模块对工件的圆孔进行测量。当未匹配到工件时，通过发送数据模块将 NG 结果发出，通过通讯告知工件未匹配到。



图 13-499 分支模块示例执行流程

参数配置

条件输入

订阅前序模块或者外部通信传输的 int 型数据。

说明

输入值仅支持整数，不支持字符串。若需要输入字符串格式，则需用分支字符或者字符识别模块配合分支模块使用。

分支参数

- 按值索引：根据条件输入传输进来的值与条件输入值对比决定分支走向，将配置界面的“条件输入值”与模块 ID 索引后的设置值比较，相同则该分路执行，不相同则该路不执行。
- 按位索引：将“条件输入值”在后台进行二进制序列转换，二进制序列与模块 ID 后位序相对应，该位为 1 时执行该模块（一次可执行多个），否则不执行。

模块结果

*分支模块*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

传输标志

int 型，代表传输标志。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

分支路线

string 型，代表满足条件可执行的分支模块 ID 列表。

13.14.4 分支字符

分支字符模块用于对输入的字符串进行检测。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

分支模块仅支持整数，不支持字符串。当对字符串类型输入值进行判断时，需要使用分支字符模块。分支字符模块对输入 string 类型字符进行检测，若检测通过，则进行数据传输。该模块与分支模块功能类似，该模块多用于工业生产对结果的判断，通过对前序模块输入的 string 类型字符的判断结果来确认当前结果是否符合条件、参数范围是否正常、检测结果是否符合要求等。

该模块主要配合输出 string 类型数据的模块进行使用，例如条件检测、格式化、脚本等。针对订阅的条件检测模块的 string 类型结果进行判断，当条件检测 OK 时则运行对应 OK 分支的

模块，当条件检测模块 NG 时则运行对应 NG 分支的模块。该模块可实现在单个流程中对不同条件结果分开处理的应用。

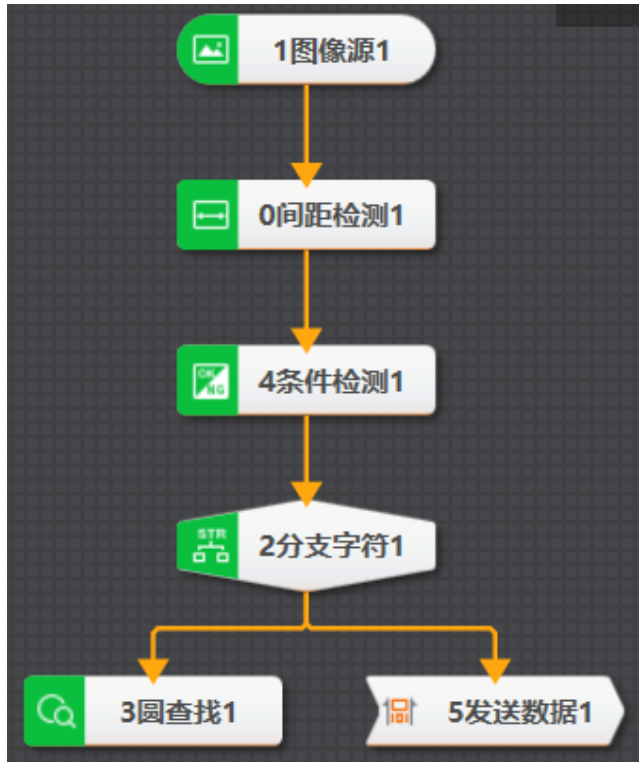


图 13-500 分支字符示例

该示例通过间距检测模块对工件宽度进行测量，再通过条件检测确认测量宽度是否符合要求。当宽度符合要求时运行 OK 分支下的圆查找模块；当宽度结果不符合要求时，通过发送数据模块发送 NG 结果进行反馈。



图 13-501 分支字符示例执行流程

参数配置

输入文本

订阅外部通信或格式化传输进来的数据。

条件输入值

输入 OK 或 NG，当条件检测 OK 时则运行对应 OK 的分支模块，当条件检测模块 NG 时则运行对应 NG 的分支模块。

调试模式

使能后，该分支不管是否匹配均会执行。通过箭头可调整前后执行顺序。

模块结果

*分支字符*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

分支路线

string 型，代表满足条件可执行的分支模块 ID 列表。

分支条件字符

string 型，代表订阅的条件字符串。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.5 文本保存

文件保存模块根据编辑的文本格式组成字符串并保存至本地文件。

使用方法

文本保存模块的输入变量只有 string 类型，最大长度为 4095 字节。开启运行后先生成缓存文件，缓存文件达到设置的文件容量后才产生文本，也可以打开实时存储，但会影响耗时。目前模块支持生成 TXT、CSV 文件，模块可以通过条件触发保存，可按日期生成文件目录。该模块主要配合输出字符串类型数据的模块进行使用，例如条件检测、格式化、脚本、读码模块等。通过订阅前序模块的字符串结果，将其保存到本地。该模块可以帮助完成数据的记录以及保存应用。

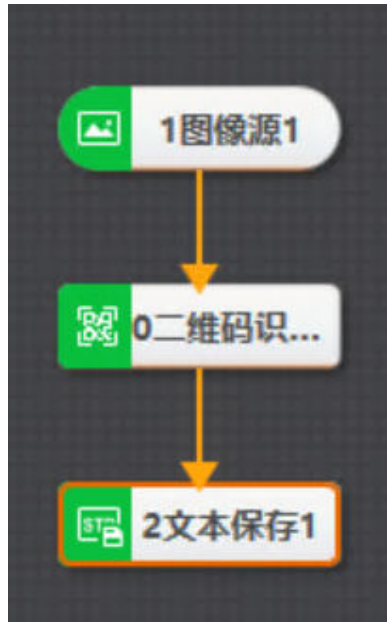


图 13-502 文本保存示例

该示例通过二维码识别模块识别二维码信息，再通过文本保存模块实现对编码信息进行收集并保存在本地。



图 13-503 文本保存示例执行流程

参数配置

文本输入参数如下：

数据源

- TXT：生成 TXT 格式文件。
- CSV：生成 CSV 格式文件，CSV 格式文件建议使用记事本打开，或者先另存新文档后用 EXCEL 打开。选择生成 CSV 文件时需要进行每一列生成数据的数据绑定。

输入文本

选择输入的文本，该文本内容会被存储到指定位置，通常选择格式化结果。

保存设置参数如下：

保存列名

当**数据源**选择 CSV 时可配置此参数。开启此开关可在生成的 CSV 文件中保存列名称；否则仅保存数据，不包含列名称。

异步保存

开启后，可异步保存文本到指定位置，以避免线程冲突。您可根据实际需求选择是否使用*异步保存*参数。

触发保存

启用后，当订阅的**触发变量**满足**保存条件**时，才可进行文件保存。

保存路径

自定义文本保存的位置。

文本保存数量

最大存储文件的数量。

文件容量(K)

每个文件的最大容量。

存储方式

设置达到最大存储数量或者所在磁盘空间不足时对文本处理的方式,可选择以下两种方式。

- 覆盖存储：覆盖之前的文本。
- 停止存储：停止存储当前文本。

时间戳设置

每次保存时在文本前追加的表示时间信息的字符串。

生成日期目录

使能后，会根据日期创建文件夹，文本文件保存在创建的文件夹中。此时还需设置如下参数：

文件保存天数

文件保存的时间期限，超出后会自动删除文件。

文件命名

设置文本文件的命名，支持常量输入及模块数据订阅。

实时存储

- 开启时，数据直接写在 txt 和 csv 文件中；
- 不开启时，数据先写入 cache 文件中，达到文件容量后转为 txt 或 csv 文件。

存储临时文件

点击该按钮可将 cache 文件以.txt 格式或者.csv 格式保存在固定路径下。

模块结果

*文本保存*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

保存路径

string 型，代表保存文本所在的文件夹路径。

文件名称

string 型，代表保存文本的文件名称。

13.14.6 逻辑

逻辑模块的逻辑运算功能可用于将多模块的输出结果进行综合判断，包含运算类型和运算数据。运算类型包括选择与、或、非、与非、或非。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

逻辑模块主要用于工业生产中，存在对多模块检测结果，需要对检测结果进行综合判断，确认当前检测结果是否符合要求的场景。
该模块可以配合大部分模块进行使用，通过订阅前序模块的模块状态、运行结果等，然后按照运算类型（与、或、非、与非、或非）进行综合运算，并将结果进行输出。

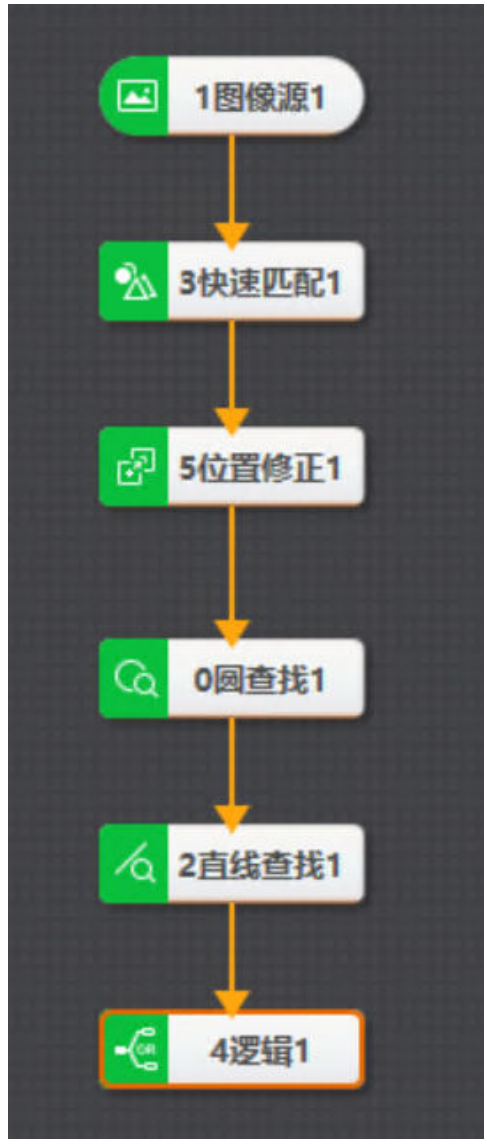


图 13-504 逻辑模块示例

该示例中需要对工件进行匹配定位、圆查找、直线查找，通过逻辑模块订阅前序模块的运行结果进行综合运算，判断逻辑为当三个模块结果均符合要求时认为该工件满足要求，后续通过发送数据模块将判断结果进行发送。



图 13-505 逻辑模块示例执行流程

参数配置

运算类型

- AND：与运算
- OR：或运算
- NOT：非运算
- NAND：与非运算
- NOR：或非运算

运算数据

选择用于逻辑运算的数据来源。

模块结果

*逻辑*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

运算结果（BOOL）

int 型，0 代表 NG，结果为非，1 代表 OK，结果为是。

运算结果（STRING）

string 型，NG 代表结果为非，OK 代表结果为是。

13.14.7 格式化

格式化模块主要用于将数据整合并格式化成字符串输出。该模块支持三种数据来源，分别为自定义输入的数据、前序模块的模块结果、自定义的数组。通常在需要对模块数据进行整合的时候使用。例如进行通信输出前，会使用该模块整理数据。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块可与较多模块配合使用，可以订阅前序模块输出的各类数据用于数据整合；也可将整合后的数据供后续模块订阅并使用。
下图展示使用快速匹配模块检测图像中的圆，并通过格式化模块将快速匹配模块检测后输出的模块状态、分数和匹配点的信息整合后输出的效果。



图 13-506 使用效果示意图

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见*结果显示*。
该模块基本参数如下图所示。



图 13-507 基本参数

该模块基本参数各区域的说明参见下表。

表 13-40 格式化模块基本参数介绍

序号	名称	描述
1	添加数据	点击 添加 可在配置数据规则添加一条新的数据。
2	数据类型	每条数据均支持插入自定义文本、订阅前序模块的模块结果、自定义数组。 • 插入文本：配置数据规则区域中选择某条数据，点击 T 后可自定义添加需显示的文本内容。 • 插入订阅：配置数据规则区域中选择某条数据，点击 后选择需订阅的数据即可添加到数据中。 • 插入数组：配置数据规则区域中选择某条数据，点击 后在插入数组页签中完成相关参数（分隔符、数组下标、数组列表）设置即可添加到数据中。 **说明** 数组输出成组数据。数组配置内包含多个数据时，按照多个数据的顺序组合输出。 ○ **分隔符**：可设置**数组列表**中每个数据之间的分隔符。 ○ **数组下标**：可设置输出多少个数组或第几个数组，可订阅 int 型数据或自定义输入。仅支持 int 型数据或*，该参数设置为*时，输出

序号	名称	描述
		全部数组；该参数设置为 int 型数据时，仅输出对应的那组数组。假设该参数设置为 0，则输出第一组数组数据，依次类推。 ○ **数组列表**：点击 ⊞ 可添加需要组装为数组的数据内容，通过**内容**订阅模块结果或自定义输入即可。 **说明** 假设**数组列表**由 2 个数据组成，分别为中心点 X 和中心点 Y，订阅的模块共输出 3 个中心点，**分隔符**为;，**数组下标**为*，**数组间分隔符**为()，则输出的数组数据为(X1;Y1)(X2;Y2)(X3;Y3)。
3	配置数据规则	可预览并配置每条数据的规则。通过左侧的序号选中某条数据，右键单击可进行删除或清空内容操作。 ● **删除**：删除整条数据。 ● **清空内容**：将整条数据的内容清空，但数据保留。
4	调整排序	可调整数据规则区域的每条数据以及数据之间的顺序。 ● 选择某条数据时，可点击 ↑ ↓ 调整上下顺序。 ● 选择数据中的某一部分时，可点击 ← → 调整前后顺序。
5	结束符&分隔符	可设置数据规则区域以及数组间的结束符或分隔符。 ● **输入结束符**：可设置配置数据规则区域中每条数据之间的结束符，可选\r（回车）、\n（换行）、\r\n（回车换行）。 ● **分隔符**：可设置配置数据规则区域中各个数据之间的分隔符。可下拉选择，也可自定义修改。 **说明** 该参数仅在数据未设置**输入结束符**时有效。 ● **数组间分隔符**：可设置每个数组之间的分隔符。可下拉选择，也可自定义修改。
6	删除	可删除配置数据规则区域中已设置的所有内容。

模块结果

*格式化*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

格式化结果

string 型，代表返回每条参数格式化后的结果。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.8 字符比较

字符比较模块可以根据输入的字符和设置的字符进行比较，若一致则输出对应的索引值，否则不输出。该模块输出的是文本列表中第一个与输入文本完全匹配的索引项。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

字符比较模块主要用于工业生产字符识别、读码类项目，确认识别信息与设置需要的信息是否一致的场景。
该模块可以配合识别类模块进行使用，例如字符识别和读码模块。使用字符比较模块的输入文本参数订阅符合标准的字符结果作为基准，然后将需要与输入文本进行比较的其他识别模块输出的字符结果设置为索引项，完成字符比较后输出完全匹配的结果。使用时需注意，该模块只会输出文本列表中第一个与输入文本完全匹配的索引项。

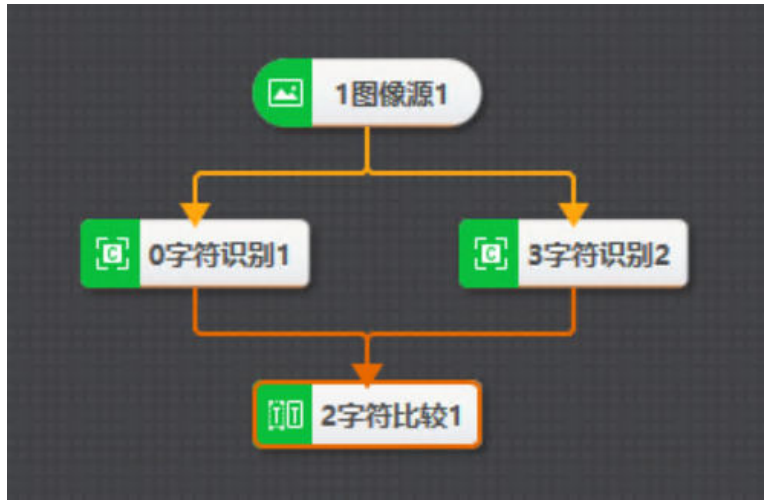


图 13-508 字符比较示例

该示例中通过将字符识别 1 模块的识别结果设置为输入文本，将字符识别 2 模块的识别结果作为索引，实现比较图中两行字符内容结果是否相同的应用。



图 13-509 字符比较示例执行流程

参数配置

输入文本

输入的字符信息。

索引

索引值支持手动修改。

文本

手动输入或订阅需要与输入字符进行比较的字符结果。

说明

对于手动输入的文本内容，目前仅支持带小写字母、数字和符号“.”，其他特殊字符例如[]、()、|、，等不能输入，同时也不支持类似“1.1.1”这种格式。

模块结果

*字符比较*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

分支条件字符

string 型，代表目标匹配字符串：支持订阅 string 类型模块输出参数/手动输入字符。

索引

int 型，代表匹配成功后，模块结果输出索引值。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.9 脚本

可在流程中调用*脚本*模块，并通过自定义脚本代码控制流程数据处理。

使用限制

- *脚本*模块仅支持使用标准 C#语言（Windows 版本）进行编写。
- *脚本*模块只能对流程内的数据进行处理。如需对方案下所有流程的批量执行逻辑进行控制，可通过 *全局脚本* 实现。

界面概览

将*脚本*模块拖入流程编辑区后，双击该模块打开脚本编辑窗口。

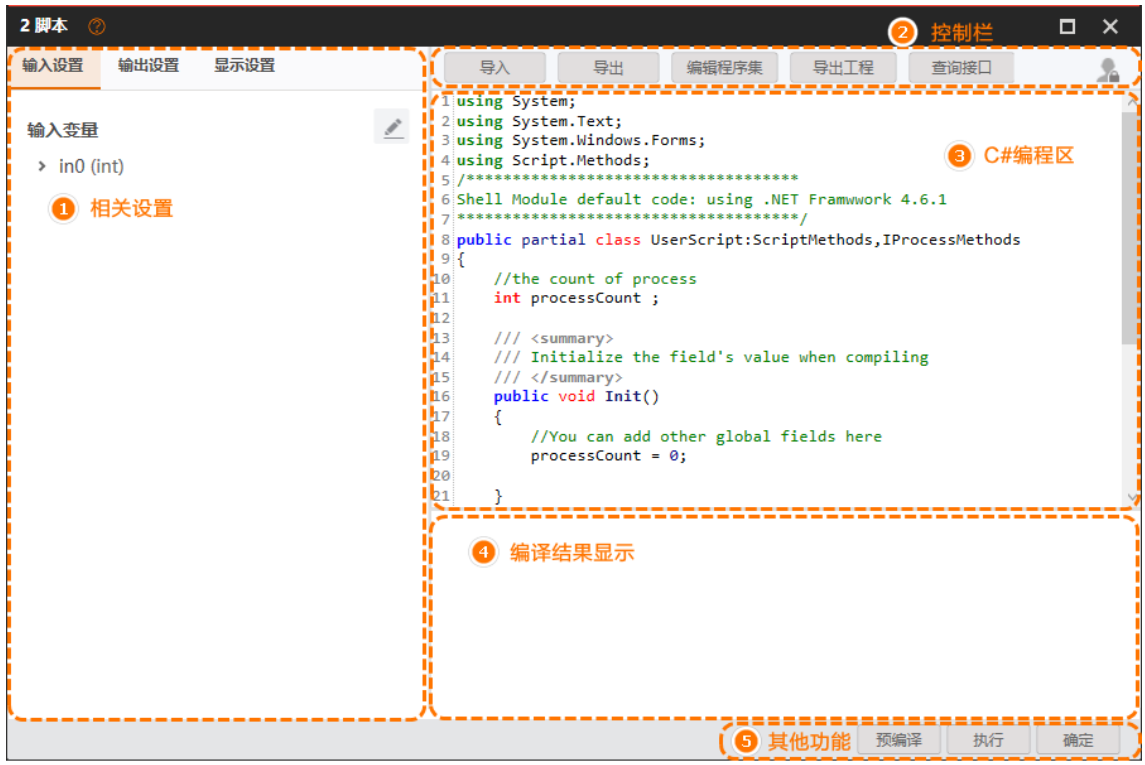


图 13-510 脚本编辑窗口

该窗口各区域说明如下：

表 13-41 脚本编辑窗口介绍

窗口区域	描述
相关设置	对设置输入输出的变量，可自定义变量名称。变量支持多种数据类型，主要为 int（整型）、int[]、float（浮点型）、float[]、string（字符串）、string[]、byte（字节）、image（图像）、ROIBOX（ROI 内的识别框）、ROIBOX[]、ANNULUS（圆环）、POLYGON（多边形）、POINT（点）、LINE（线）、FIXTURE（修正信息）、RECT（矩形）、ELLIPSE（椭圆）、pointset（点集）。 **说明** 不带“[]”的变量为一个值，带“[]”的变量为数组。 • **输入设置**：可通过**初始值**绑定*脚本*前序模块的结果或全局/局部变量。 • **输出设置**：输出变量可作为*脚本*需显示的内容或后序模块的输入。部分类型（请以实际为准）的输出变量勾选后，将在**显示设置**中显示，作为该模块的信息输出，并可在该模块的图像上显示。 • **显示设置**：显示**输出设置**中勾选的输出变量。其中**参数名称**可自定义修改，除 image 类型外的数据还可单击 进行显示相关设置。
控制栏	• *导入*：导入脚本程序（格式：.cs）。 • *导出*：导出脚本程序（格式：.cs）。 • *编辑程序集*：添加程序集，与全局脚本添加程序集相同，详情参见 *程序集添加* 。 • *导出工程*：将脚本程序导出，导出后可使用 Visual Studio 进行调试。 • *查询接口*：查询设置或获取不同数据类型参数的接口以及其他常用接口。 • ：单击后可设置密码以启用加密权限。启用加密后，实时生效，未验证密码只能修改输入变量的值或订阅，以及执行*预编译*、*执行*和*确定*功能；验证密码成功后可进行查看、编辑代码等其他操作；再次单击 并输入密码可取消加密。
C#编程区	可在此处通过调用脚本接口等方式自定义开发脚本。 该区域提供脚本的默认代码，默认代码的简要解读参见下文的 *默认代码导读* 。 **说明** 在 C#编程区自定义脚本代码之前，需先在输入/输出变量编辑区定义输入变量（例如 int 型的 in0）和输出变量（例如 int 型的 out0）。完成定义后，可在

窗口区域	描述

	C#编程区中直接编写代码，获取或设置**数组类型以外**类型的变量的取值。示例如下：
	`public bool Process()`
	`{`
	`  int a = in0;`
	`  out0 = 32;`
	`  return true;`
	`}`

编译结果显示	显示该模块编译后的结果信息。
其他功能	• *预编译*：对脚本程序进行预编译，单击该按钮即调用 Init 方法（该方法详情见下文的 *使用方法*）。 • *执行*：单击该按钮即调用 Process 方法（该方法详情见下文的 *使用方法*）。 • *确定*：保存修改后的代码并退出脚本编辑界面。

使用方法

- **调用脚本模块**
  在流程中调用*脚本*模块时，无针对前序/后序模块的特定要求。只要脚本逻辑与流程逻辑匹配，*脚本*模块可在流程中的任意环节调用。
- **自定义脚本代码**
  在流程中调用*脚本*模块后，可在 C#编程区调用 *脚本开放接口* 自定义脚本代码。其中的核心接口为 和 。
  - 可在 方法中实现变量初始化和句柄创建等初始化逻辑，相关工作会在加载方案时完成。
  - 可在 方法中实现变量计算和逻辑处理等具体的功能，具体的功能在流程执行时执行。

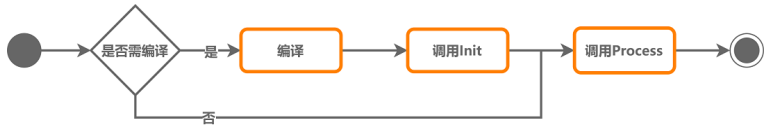


图 13-511 Init 与 Process 方法的执行顺序

说明

*脚本*模块还支持调用 C#程序集和非托管库，具体调用方法与全局脚本的第三方库调用相同，详情参见 *第三方库调用方法* 。

应用示例

脚本的具体应用示例，请参见 *应用示例：通信数据收发控制* 。

默认代码导读

*脚本*模块在 C#编程区提供的默认代码如下。其中 Init 方法只有在加载方案或者修改代码后预编译时才会运行，Process 方法在流程执行的时候运行。
可在 public partial class UserScript:ScriptMethods,IProcessMethods 中自行实现代码逻辑。

```
using System;
using System.Text;
using System.Windows.Forms;
using Script.Methods;
/************************************
Shell Module default code: using .NET Framwwork 4.6.1
************************************/
public partial class UserScript:ScriptMethods,IProcessMethods
{
    //the count of process
    //执行次数计数
    int processCount ;

    /// <summary>
    /// Initialize the field's value when compiling
    /// 预编译时变量初始化
    /// </summary>
    public void Init()
    {
        //You can add other global fields here
    //变量初始化，其余变量可在该函数中添加
        processCount = 0;

    }

    /// <summary>
    /// Enter the process function when running code once
    /// 流程执行一次进入 Process 函数
    /// </summary>
    /// <returns></returns>
    public bool Process()
    {
        //You can add your codes here, for realizing your desired function
    //每次执行将进入该函数，此处添加所需的逻辑流程处理

        return true;
    }
}
```

脚本 Demo 源码

以下为脚本 Demo 文件 ScriptDemo.cs 的源码。该文件可从 VM 安装路径下获取，相对路径为：.\Applications\Module(sp)\x64\Logic\ShellModule。

```
using System;
using System.Text;
using System.Windows.Forms;
using Script.Methods;
public partial class UserScript:ScriptMethods,IProcessMethods
{
    //the count of process
    //执行次数计数
    int processCount ;

    /// <summary>
    /// Initialize the field's value when compiling
    /// 预编译时变量初始化
    /// </summary>
    public void Init()
    {
        //You can add other global fields here
    //变量初始化，其余变量可在该函数中添加
        processCount = 0;

    }

    /// <summary>
    /// Enter the process function when running code once
    /// 流程执行一次进入Process函数
    /// </summary>
    /// <returns></returns>
    public bool Process()
    {
        //You can add your codes here, for realizing your desired function
    //每次执行将进入该函数，此处添加所需的逻辑流程处理

    //获取设置int
        int a = 0;
        GetIntValue("in0",ref a);
        SetIntValue("out0", a);

        //获取设置float
        float f=0f;
        GetFloatValue("in1",ref f);
        SetFloatValue("out1", f);

        //获取设置string
        string s="";
        GetStringValue("in2",ref s);
        SetStringValue("out2", s);

        //获取设置二进制数据
        byte[] by = new byte[]{};
        GetBytesValue("in8",ref by);
        SetBytesValue("out8", by);
```

```
        int count=0;
        //获取设置 int 数组
        int[] narry = new int[3];
        GetIntArrayValue("in3",ref narry,out count);
        SetIntArrayValue("out3",narry,0,narry.Length);

        //获取设置 float 数组
        float[] farry = new float[3];
        GetFloatArrayValue("in4",ref farry,out count);
        SetFloatArrayValue("out4",farry,0,farry.Length);

        //获取设置 strig 数组
        string[] sarry = new string[3];
        GetStringArrayValue("in5",ref sarry,out count);
        SetStringArrayValue("out5",sarry,0,sarry.Length);

    //设置/获取 16 进制数据
        byte[] tempBytes = new byte[] { };
        GetBytesValue("in1", ref tempBytes);
        SetBytesValue("out1", new byte[]{0x00,0x02});

        //设置/获取图像数据
        ImageData imagedata = new ImageData();
        GetImageValue("in6", ref imagedata);
        SetImageValue("out6", imagedata);

    //设置/获取 ROIBOX 数据
    RoiboxData roidata = new RoiboxData();
        GetRoiboxValue("in7", ref roidata);
        SetRoiboxValue("out7", roidata);

        //设置全局变量
        GlobalVariableModule.SetValue("var1", "323");
        //2.获取全局变量
        object paramValue = GlobalVariableModule.GetValue("var1");

        //获取模块结果数据
        //GetModule 传入模块的名称，如果存在 group 中，需要加上 group 的名称
        //GetValue() 传入的是模块的输出参数名称
        object result = CurrentProcess.GetModule("图像源 1").GetValue("Height");
        object result1 = CurrentProcess.GetModule("组合模块 1.图像源 1").GetValue("Height");

        //设置模块运行参数
        //如果该模块在 group 内部，需要加上 group 模块的参数名称
        //SetValue(),key 值为具体的参数名称，
        CurrentProcess.GetModule("BLOB 分析 1").SetValue("FindNum", "4");
        CurrentProcess.GetModule("组合模块 1.BLOB 分析 1").SetValue("FindNum", "4");

        //5.通信发送数据
        //GetDevice(int index), index 为通信设备的 ID
        //tcp,udp,串口发送数据调用函数
        GlobalCommunicateModule.GetDevice(1).SendData("msg",DataType.StringType);
```

```
        //plc 设备发送数据
        //GetAddress(int index),index 为 address 的 ID
        GlobalCommunicateModule.GetDevice(2).GetAddress(1).SendData("100",DataType.IntType);
        GlobalCommunicateModule.GetDevice(2).GetAddress(1).SendData("100",DataType.StringType);
        GlobalCommunicateModule.GetDevice(2).GetAddress(1).SendData("100",DataType.FloatType);

        //modbus 发送设备数据
        GlobalCommunicateModule.GetDevice(3).GetAddress(1).SendData("100",DataType.IntType);
        GlobalCommunicateModule.GetDevice(3).GetAddress(1).SendData("100",DataType.StringType);
        GlobalCommunicateModule.GetDevice(3).GetAddress(1).SendData("100",DataType.FloatType);

        return true;
    }
}
```

模块结果

*脚本*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

out*

类型由输出变量设置时决定，显示输出变量的信息。

VS 调试

可使用 Visual Studio（以下简称为 VS）调试脚本代码，确保其能正常运行。

前提条件

已安装 2013 版或更新版本的 Visual Studio。

操作步骤

1. 在*脚本*模块编辑窗口的控制栏，单击*导出工程*。
2. 在打开的脚本工程文件夹中，使用 VS 打开脚本程序（.sln 文件）。
3. 在 VS 菜单栏选择 *生成 → Build*，编译脚本程序。
4. 将脚本程序附加至 VM 主进程。
   1) 在 VS 菜单栏选择 *调试 → 附加到进程* 。
   2) 在*附加到进程*对话框进程列表中，选中 VM 主进程。
   3) 单击*附加*。

   附加后，出现弹窗报错“无法附加到进程，已附加了一个调试器”。
5. 按照以下两种方式中的任意一种解决该报错。

- 打开任务管理器，结束进程 vServerApp.exe，再到 VS 中重新附加至主进程的 exe 文件。
- 关闭 VM，将 VM 安装路径下 vServerApp.exe.config 文件中的 DumpEnable 的值改为 false（如下图所示），保存后重新打开 VM。

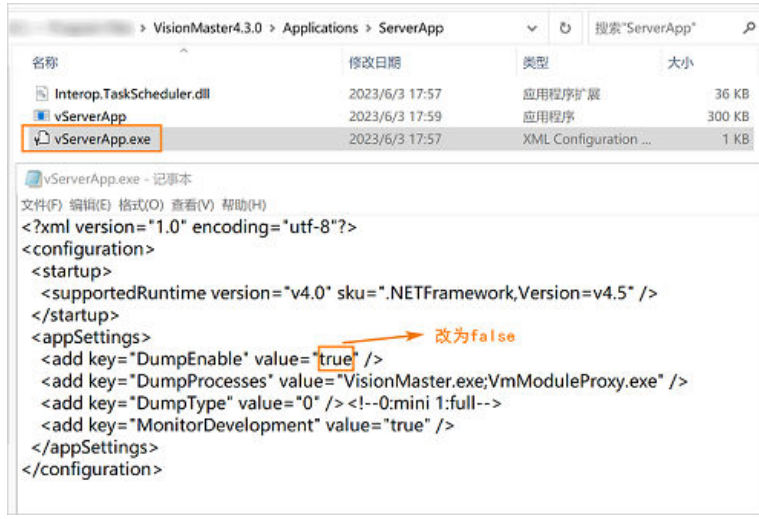


图 13-512 配置 DumpEnable

6. 单击 VM 快捷工具条的 ⊙ 单次执行流程，同时查看 VS 中是否进入断点。

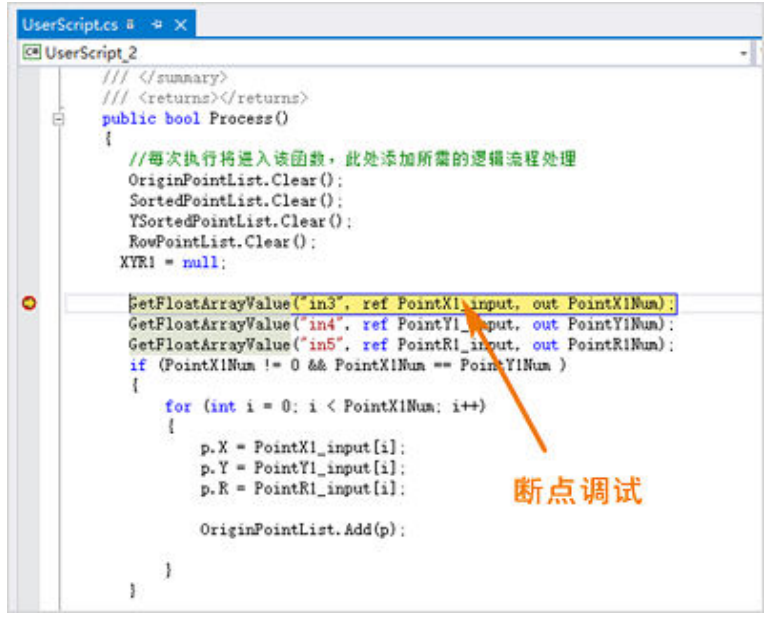


图 13-513 断点调试

说明

如需继续调试，需在 VS 中重新选择附加到进程。请勿直接点击启动。

脚本开放接口

脚本提供开放接口，方便您通过几行代码快速实现流程控制和通信控制。

脚本开放接口概览

表 13-42 脚本全量接口

实现类型	方法	描述
初始化	*Init*	初始化脚本
流程逻辑处理	*Process*	定义单个流程的执行逻辑
全局变量处理	*GlobalVariableModule.SetValue*	设置全局变量的值
	GlobalVariableModule.GetValue	获取全局变量的值
处理模块结果与参数	*CurrentProcess.GetModule*	获取模块结果数据
	CurrentProcess.GetModule.SetValue	设置模块运行参数的值
	SetModuleParam	设置模块运行参数
	GetModuleParam	获取模块运行参数的值
	BytesToPointset	将点集二进制数据转换为轮廓点数组数据
	PointsetToBytes	将轮廓点数组数据转换为点集二进制数据
发送通信数据	*SendData*	指定通信设备发送特定类型的数据
获取数据	• *GetIntValue* • *GetFloatValue* • *GetStringValue* • *GetBytesValue* • *GetIMAGEValue* • *GetRoiboxValue* • *GetIntArrayValue* • *GetFloatArrayValue* • *GetStringArrayValue*	推荐您在脚本中使用变量名称方式获取输入数据，假设您要获取输入变量名称为 **in0** 的 POINT 类型数据，代码示例如下。 **说明** • 使用 GetIntValue、GetRoiboxValue 等接口为上一代数据获取方式，出于方案兼容

实现类型	方法	描述

		考虑，保留了这些接口，但不推荐使用。
		• 实际使用中，请根据您的变量名称和变量类型修改代码片段，其中不同变量类型在脚本中对应不同的数组类型，命名规则为**<变量类型>Data[]**（适用变量类型为非数组类型），例如变量为 POINT 类型，则对应脚本中的 PointData[]类型。变量为数组类型的在脚本中直接使用。
		• 变量名称需要保证唯一性，即多个变量不要使用同一个名称。
		`// 获取整个输入数组数据`
		`PointData[] point1 = in0;`
		`// 分别获取输入数组的每个数据`
		`PointData[] point2 = new PointData[in0.Length];`
		`for (int i=0;i<in0.Length;i++)`
		`{`
		`point2[i] = in0[i];`
		`}`

输出数据	• *SetIntValue* • *SetIntArrayValue* • *SetFloatValue* • *SetFloatArrayValue* • *SetStringValue* • *SetStringArrayValue* • *SetBytesValue* • *SetImageValue* • *SetRoiboxValue* • *SetStringValueByIndex* • *SetIntValueByIndex* • *SetFloatValueByIndex*	推荐您在脚本中使用变量名称方式设置数据输出，假设您要设置输出变量名称为 **out0** 的 POINT 类型数据输出，代码示例如下。 **说明** • 使用 SetIntValue、SetRoiboxValue 等接口为上一代输出数据设置方式，出于方案兼容考虑，保留了这些接口，但不再推荐使用。 • 实际使用中，请根据您的变量名称和变量类型修改代码片段，其中不同变量类型在脚本中对应不

实现类型	方法	描述

		同的数组类型，命名规则为**<变量类型>Data[]**（适用变量类型为非数组类型），例如变量为 POINT 类型，则对应脚本中的 PointData[]类型。变量为数组类型的在脚本中直接使用。
		• 变量名称需要保证唯一性，即多个变量不要使用同一个名称。
		`// 输入直接赋值给输出`
		`out0=in0;`
		`// 自定义数据赋值给输出`
		`PointData[] pt = new PointData[in0.Length];`
		`for (int i=0;i<in0.Length;i++)`
		`{`
		`pt[i] = new PointData();`
		`pt[i].PointX=i;`
		`pt[i].PointY=i;`
		`}`
		`out0 = pt;`

渲染相关	*ShowImage*	在渲染控件上显示图像
	DrawShape	在渲染控件上绘制图形
调试	*ConsoleWrite*	将信息打印至 DebugView
	ShowMessageBox	将错误信息通过弹窗提示

Init

接口原型

public void Init(){}

功能描述

初始化脚本。可在此方法中实现初始化相关操作 。该方法在加载方案或预编译全局脚本时执行。

Process

接口原型

public bool Process(){}

功能描述

定义*脚本*模块所在流程的执行逻辑。

GlobalVariableModule.SetValue

接口原型

GlobalVariableModule.SetValue(string paramName,string paramValue)

功能描述

设置全局变量。

输入参数

paramName：string 类型，全局变量名称

输出参数

paramValue：string 类型，全局变量的值

返回值

0：调用成功

非 0 返回值：调用失败

GlobalVariableModule.GetValue

接口原型

object GlobalVariableModule.GetValue (string paramName)

功能描述

获取全局变量的值。

输入参数

paramName：string 类型，变量名称

返回值

如果调用成功，返回全局变量的值。

如果调用异常，返回 null。

说明

返回值为 object 类型，如需转成其他类型，请将 object 转成 string 再转至其他类型。

CurrentProcess.GetModule

接口原型

CurrentProcess.GetModule(string paramModuleName).GetValue(string paramValueName)

功能描述

获取指定模块某个结果参数的值。

输入参数

paramModuleName：string 类型，模块名称。请从流程中查找模块名称。

paramValueName：string 类型，模块结果中某个参数的名称。请从 SDK 手册中查找参数名称。该 SDK 手册可从 VM 安装路径中获取：*..\Development\V4.x\Documentations*。

说明

如果对应的模块在流程的 *Group* 中，传入 paramModuleName 的模块名称需附带 Group 名称，例如：GetModule("Group1.图像源 1")。

返回值

如果调用成功，返回结果参数的值。

如果调用异常，返回 null。

CurrentProcess.GetModule.SetValue

接口原型

CurrentProcess.GetModule(string paramModuleName).SetValue(string paramValueName，string paramValue)

功能描述

设置模块运行参数的值。

输入参数

paramModuleName：string 类型，模块名称。请从流程中查找模块名称。

paramValueName：string 类型，参数名称。请从 SDK 手册中查找参数名称。该 SDK 手册可从 VM 安装路径中获取：*..\Development\V4.x\Documentations*。

paramValue：string 类型，参数值

说明

如果对应的模块在流程的 *Group* 中，传入 paramModuleName 的模块名称需附带 Group 名称，例如：GetModule("Group1.图像源 1")。

返回值

0：调用成功。

非 0 返回值：调用异常。

SendData

表 13-43 具体方法

接口原型	说明
GlobalCommunicateModule.GetDevice(int deviceID).GetAddress(int addressID).SendData(string data,DataType dataType)	指定某个 PLC/Modbus 设备发送 Int、float 或 string 类型数据。 • 输入参数： ◦ deviceID：int 类型， *通信管理* 中设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。 ◦ addressID：int 类型， *通信管理* 中设备的地址 ID。地址 ID 的示例见下文的 *图 8-11*。 ◦ data：string 类型，待发送的数据，如果发送多个，请用;隔开。 ◦ dataType：DataType 类型，待发送数据的类型，包含 int、float 和 string 三种。 • 返回值： ◦ 0：调用成功。 ◦ 非 0 返回值：调用异常。
GlobalCommunicateModule.GetDevice(int deviceID).GetAddress(int addressID).SendData(byte[] bytedata,DataType.ByteType)	指定某个 PLC/Modbus 设备发送十六进制数据。 • 输入参数： ◦ deviceID：int 类型， *通信管理* 中设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。 ◦ addressID：int 类型， *通信管理* 中设备的地址 ID。地址 ID 的示例见下文的 *图 8-11*。 ◦ bytedata：byte[]类型，待发送的十六进制数据。 • 返回值： ◦ 0：调用成功。 ◦ 非 0 返回值：调用异常。
GlobalCommunicateModule.GetDevice(int deviceID).SendData(string data)	指定某个 TCP、UDP 或串口发送 string 类型的数据。 • 输入参数： ◦ deviceID：int 类型， *通信管理* 中设备的设备 ID。设备 ID 的示例见下文的 *图 8-11*。 ◦ data：string 类型，待发送的数据。 • 返回值：

接口原型	说明
	○ 0：调用成功。 ○ 非 0 返回值：调用异常。
GlobalCommunicateModule.GetDevice(int deviceID).SendData(byte[] bytedata)	指定某个 TCP、UDP 或串口发送十六进制的数据。 • 输入参数： ○ deviceID：int 类型，*通信管理*中设备的设备 ID。设备 ID 的示例见下文的*图 8-11*。 ○ bytedata：byte[]类型，待发送的十六进制数据。 • 返回值： ○ 0：调用成功。 ○ 非 0 返回值：调用异常。



上图的示例中，TCP 客户端和三菱 MC 的设备 ID 分别为 1 和 2，三菱 MC 的地址 ID 为 1。

图 13-514 设备 ID 与地址 ID 示例

SetModuleParam

接口原型

public int SetModuleParam(uint nModuleID, string paramKey, string paramValue)

功能描述

设置模块运行参数。

输入参数

nModuleID：uint 类型，模块 ID。

paramKey：string 类型，模块运行参数。

paramValue：string 类型，模块运行参数的值。

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetModuleParam

接口原型

public int GetModuleParam(uint nModuleID, string paramKey, ref string paramValue)

功能描述

获取模块运行参数。

输入参数

nModuleID：uint 类型，模块 ID。

输出参数

paramKey：string 类型，模块运行参数

paramValue：ref string 类型，模块运行参数的值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

BytesToPointset

接口原型

int BytesToPointset(byte[] inVariant, ref ContourPointData[] contourPointArray)

功能描述

将点集二进制数据转换为轮廓点数组数据。

输入参数

inVariant：byte[]类型，待转换的点集二进制数据。

输出参数

contourPointArray：ContourPointData[]类型，轮廓点数组数据。

说明

ContourPointData[]为脚本模块定义的轮廓点数据结构。

返回值

- 0：转换成功。
- 非 0 返回值：转换失败。

PointsetToBytes

接口原型

byte[] PointsetToBytes(ContourPointData[] contourPointArray)

功能描述

将轮廓点数组数据转换为点集二进制数据。

输入参数

contourPointArray：ContourPointData[]类型，待转换的轮廓点数组数据。

> **说明**
> ContourPointData[]为脚本模块定义的轮廓点数据结构。

返回值

- 非 null 返回值：转换成功，为点集二进制数据。
- null：转换失败。

GetIntValue

接口原型

int GetIntValue(string paramName, ref int paramValue)

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：int 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetFloatValue

接口原型

int GetFloatValue (string paramName, ref float paramValue)

功能描述

获取 float 类型变量的值。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：float 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetStringValue

接口原型

int GetStringValue (string paramName, ref string paramValue)

功能描述

获取 string 类型变量的值。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：string 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetBytesValue

接口原型

int GetBytesValue (string paramName,ref byte[] paramValue)

功能描述

获取 byte 数组类型变量的值。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：byte[]类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetIMAGEValue

接口原型

int GetIMAGEValue (string paramName, ref Image paramValue)

功能描述

获取图像数据。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：Image 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetRoiboxValue

接口原型

int GetRoiboxValue(string paramName, ref RoiboxData roiboxData)

功能描述

获取 ROI 的 BOX 数据（识别框等）。

输入参数

paramName：string 类型，变量名称

输出参数

roiboxData：RoiboxData类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetIntArrayValue

接口原型

int GetIntArrayValue(string paramName, ref int[] paramValue，out int arrayCount)

功能描述

获取 int 数组变量。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：int[]类型，变量值

arrayCount：int 类型，数组个数

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetFloatArrayValue

接口原型

int GetFloatArrayValue(string paramName, ref float[] paramValue，out int arrayCount)

功能描述

获取 float 型数组变量。

输入参数

paramName：string 类型，变量值

输出参数

paramValue：float[]类型，变量值

arrayCount：int 类型，数组个数

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

GetStringArrayValue

接口原型

int GetStringArrayValue(string paramName, ref string[] paramValue，out int arrrayCount)

功能描述

获取 string 类型数组变量的值。

输入参数

paramName：string 类型，变量名称

输出参数

paramValue：string[]类型，变量值

arrayCount：int 类型，数组个数

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetIntValue

接口原型

int SetIntValue(string key, int value)

功能描述

设置 int 型变量的值。

输入参数

key：string 类型，变量名称

value：int 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetIntArrayValue

接口原型

SetIntArrayValue(string key, int[] valueArray, int index, int len)

功能描述

设置 int 数组变量的值。

输入参数

key：string 型，key 值。

valueArray：string[]，数组

index：int 型，数组的索引

len：int 型，数组的长度

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetFloatValue

接口原型

int SetFloatValue (string key, float value)

功能描述

设置 float 型变量值。

输入参数

key：string 类型，变量名称

value：float 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetFloatArrayValue

接口原型

SetFloatArrayValue(string key, float[] valueArray, int index, int len)

功能描述

设置 float 数组变量的值。

输入参数

key：string 型，key 值。

valueArray：string[]，数组

index：int 型，数组的索引

len：int 型，数组的长度

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetStringValue

接口原型

int SetStringValue (string key, string value)

功能描述

设置 string 型变量的值。

输入参数

key：string 类型，变量名称

value：string 类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetStringArrayValue

接口原型

SetStringArrayValue(string key, string[] valueArray, int index, int len)

功能描述

设置 string 数组变量的值。

输入参数

key：string 型，key 值。

valueArray：string[]，数组

index：int 型，数组的索引

len：int 型，数组的长度

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetBytesValue

接口原型

int SetBytesValue (string key, byte[] value)

功能描述

设置十六进制数据。

输入参数

key：string 类型，变量名称

value：byte[]类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetImageValue

接口原型

int SetImageValue (string key, ImageData value)

功能描述

设置图像数据。

输入参数

key：string 类型，变量名称

value：ImageData类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetStringValueByIndex

接口原型

int SetStringValueByIndex(string key, string value, int index, int total)

功能描述

按照索引设置 string 型数组内某个元素的值。

输入参数

key：string 类型，变量名称

value：string 类型，变量值

index：int 类型，数组的索引

total：int 类型，数组元素个数

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetIntValueByIndex

接口原型

int SetIntValueByIndex(string key, int value, int index, int total)

功能描述

按照索引设置 int 型数组内某个元素的值。

输入参数

key：string 类型，变量名称

value：int 类型，变量值

index：int 类型，数组的索引

total：int 类型，数组元素个数

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetFloatValueByIndex

接口原型

int SetFloatValueByIndex (string key, float value, int index, int total)

功能描述

按照索引设置 float 型数组内某个元素的值。

输入参数

key：string 类型，变量名称

value：float 类型，变量值

index：int 类型，数组的索引

total：int 类型，数组元素个数

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

SetRoiboxValue

接口原型

int SetRoiboxValue(string paramName, RoiboxData roiboxData)

功能描述

设置 ROI 的 BOX 数据（识别框等）。

输入参数

paramName：string 类型，变量名称

roiboxData：RoiboxData类型，变量值

返回值

- 0：调用成功。
- 非 0 返回值：调用异常。

ShowImage

接口原型

public void ShowImage(ImageData imageData)

功能描述

在渲染控件上显示图像。

输入参数

imageData：ImageData 类型，图像数据对象

返回值

无

备注

该接口仅型号中带 6210 或 7120 的加密狗支持使用。

DrawShape

接口原型

public void DrawShape(object shapeData, ShapeConfig shapeConfig = null)

功能描述

在渲染控件上绘制图形。

输入参数

shapeData：object 类型，图形数据对象

shapeConfig：ShapeConfig 类型，默认参数，图形属性对象

返回值

无

备注

该接口仅型号中带 6210 或 7120 的加密狗支持使用。

ConsoleWrite

接口原型

void ConsoleWrite(string content)

功能描述

将异常信息打印至 DebugView 中。

输入参数

Content：string 类型，待打印的内容

返回值

无

ShowMessageBox

接口原型

void ShowMessageBox(string msg)

功能描述

脚本运行异常时，通过弹窗提示。

输入参数

msg：string 类型，弹窗内容

返回值

无

应用示例：通信数据收发控制

本章节以“通过脚本控制通信数据收发”为例，介绍*脚本*模块如何在方案中应用，以及方案最终实现效果。

本节内容包含：

- *方案需求*
- *方案思路*
- *方案搭建*
  - *步骤 1：配置 TCP 客户端*
  - *步骤 2：搭建流程*
  - *步骤 3：开发脚本代码*
  - *步骤 4：配置全局触发*
- *方案效果测试*

方案需求

假设某视觉业务场景需要满足如下需求：

表 13-44 方案需求

需求	描述
需求 1	VM 接收到 TCP 客户端发送的任意字符串（string）型数据时，触发 VM 执行流程。
需求 2	流程执行完成时，VM 输出 int、float、string、IMAGE、byte 和 ROIBOX 这 6 种类型的数据。
需求 3	需求 2 中提及的各类型数据，需输出回至 TCP 客户端。

方案思路

思路框架

基于上述需求，方案搭建的思路框架如下：

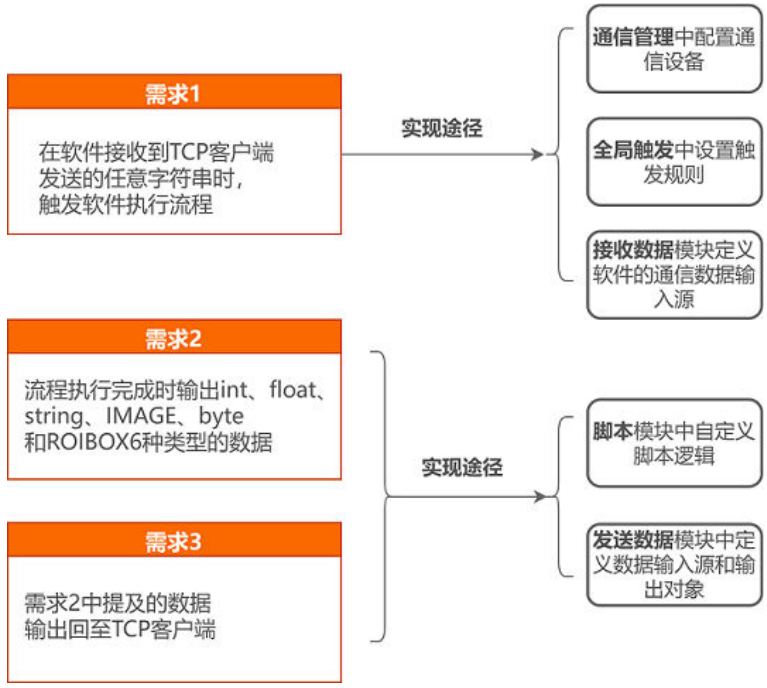


图 13-515 方案思路框架

思路详解

- 针对**需求 1**：
  - 可在 VM 的 ***通信管理*** 中将 TCP 客户端作为通信设备启用，并在流程中调用 ***接收数据*** 模块，接收 TCP 客户端发送的数据。
  - 可将 ***全局触发*** 的触发模式设置为**字符串触发**，且将**匹配模式**设置为**不匹配**。**不匹配**表示 VM 不对接收到的字符串进行匹配，默认接收到字符串就触发执行流程。
- 针对**需求 2** 和**需求 3**：
  - 可在流程中调用 ***脚本*** 模块，并自行开发脚本逻辑，定义上述 6 种类型数据的收发。
  - 可将 ***发送数据*** 模块作为脚本模块的后序模块，用于接收***脚本***模块输出的数据。
  - 可将***发送数据***模块的输出对象配置为 TCP 客户端，并定义具体输出的数据。

方案搭建

基于上述方案思路，可按以下四大步骤搭建该方案。

步骤 1：配置 TCP 客户端

为实现接收 TCP 客户端发送的数据，需在 VM 中启用 TCP 客户端并完成其 IP 地址和端口号等配置。

前提条件

已启用 TCP 客户端。

说明

本示例中假设目标 TCP 客户端的 IP 地址为 *127.0.0.1*，端口号为 *7920*。

操作步骤

1. 在软件主界面单击快捷工具条上的 打开*通信管理*窗口。
2. 配置 TCP 的 IP 地址和端口号，并开启*数据上传*和*自动重连*，同时确保设备列表中的 *TCP 客户端*已开启。

   目标 IP

   TCP 客户端的 IP 地址。

   目标端口

   TCP 客户端的端口号。

   数据上传

   开启后，VM 软件界面上的数据会根据通信模块接收的数据实时更新。

   自动重连

   开启后，VM 与 TCP 客户端断开连接时将自动重连。

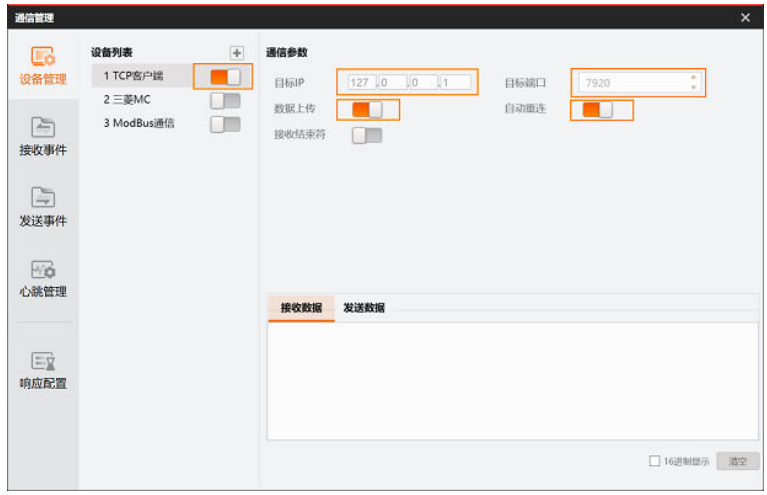


图 13-516 TCP 客户端通信配置

步骤 2：搭建流程

操作步骤

1. 将 ***接收数据*** 模块作为流程的首个模块，并配置该模块的如下参数。

   该模块用于接收 TCP 客户端发送的数据。

   数据源

   选择**通信设备**（如下图所示）。

   通信设备

   选择 TCP 客户端（如下图所示）。



图 13-517 接收 TCP 客户端数据的配置

2. 在***接收数据***模块后调用 ***图像源*** 模块，并完成该模块**图像源**参数的配置。
3. 在***图像源***模块后，按需调用多个模块。

   此处调用的多个模块，可任意搭配。整体可输出 int、float、string、IMAGE、byte 和 ROIBOX 这 6 种类型数据的模块即可。
4. 调用 ***脚本*** 模块，并完成如下配置。

*脚本*模块在流程中的作用为，接收流程上游的特定数据，并控制最终输出的数据。

1) 定义*脚本*模块的输入变量（如下图所示）。
2) 定义*脚本*模块的输出变量（如下图所示）。

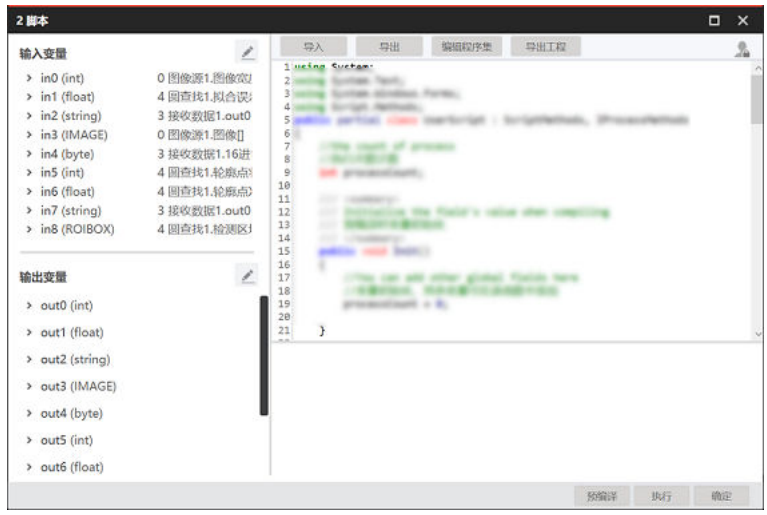


图 13-518 脚本输入输出配置

5. 调用 *发送数据* 模块，并对该模块做如下配置。
   1) 将输出目标设置为 TCP 客户端。



图 13-519 设置输出目标

   2) 将该模块发送的数据设置为*脚本*模块输出的 int、float、string、IMAGE、byte 和 ROIBOX 这 6 种类型的数据，并可启用数据的分隔符。



图 13-520 配置输出数据

结果说明

至此，流程搭建完成。

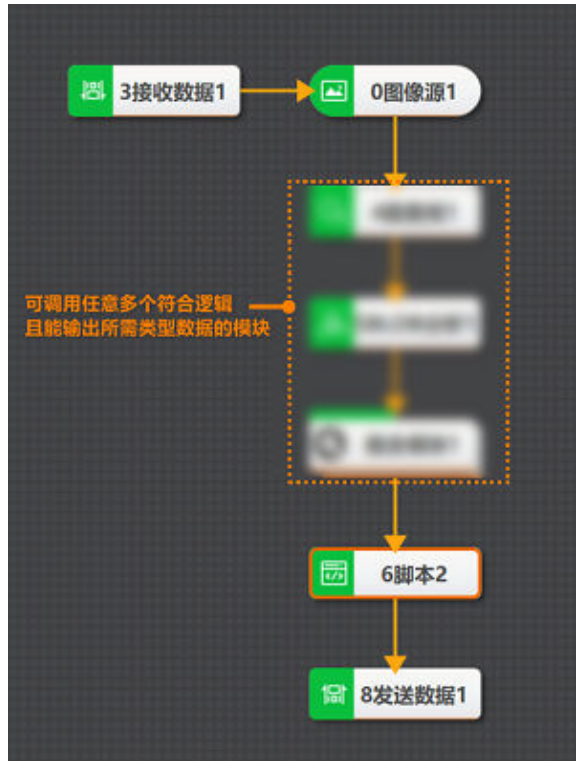


图 13-521 流程搭建结果

步骤 3：开发脚本代码

前提条件

已在流程中调用*脚本*模块，且已在该模块定义其输入/输出变量。

操作步骤

1. 双击*脚本*模块打开脚本配置窗口。
2. 在该窗口的 C#编程区调用 *Init* 方法，在该方法中实现变量初始化。
3. 调用 *Process* 方法，并在该方法中调用如下获取和输出数据的方法，完成对各类型变量输入和变量值的定义。

表 13-45 获取/输出数据的方法

获取数据	输出数据
• *GetIntValue* • *GetFloatValue* • *GetStringValue*	• *SetIntValue* • *SetFloatValue* • *SetStringValue*

获取数据	输出数据
• *GetBytesValue* • *GetIMAGEValue* • *GetRoiboxValue*	• *SetBytesValue* • *SetImageValue* • *SetRoiboxValue*

以其中的 GetIntValue 和 SetIntValue 方法为例，上述方法的调用形式如下：

```
//1.int 数据测试
int na = 0, ncount = 0;
GetIntValue("in0", ref na);
SetIntValue("out0", na);
```

该示例中的 in0 和 out0，即*脚本*模块输入/输出变量中所配置的 int0 和 out0，如下图所示。

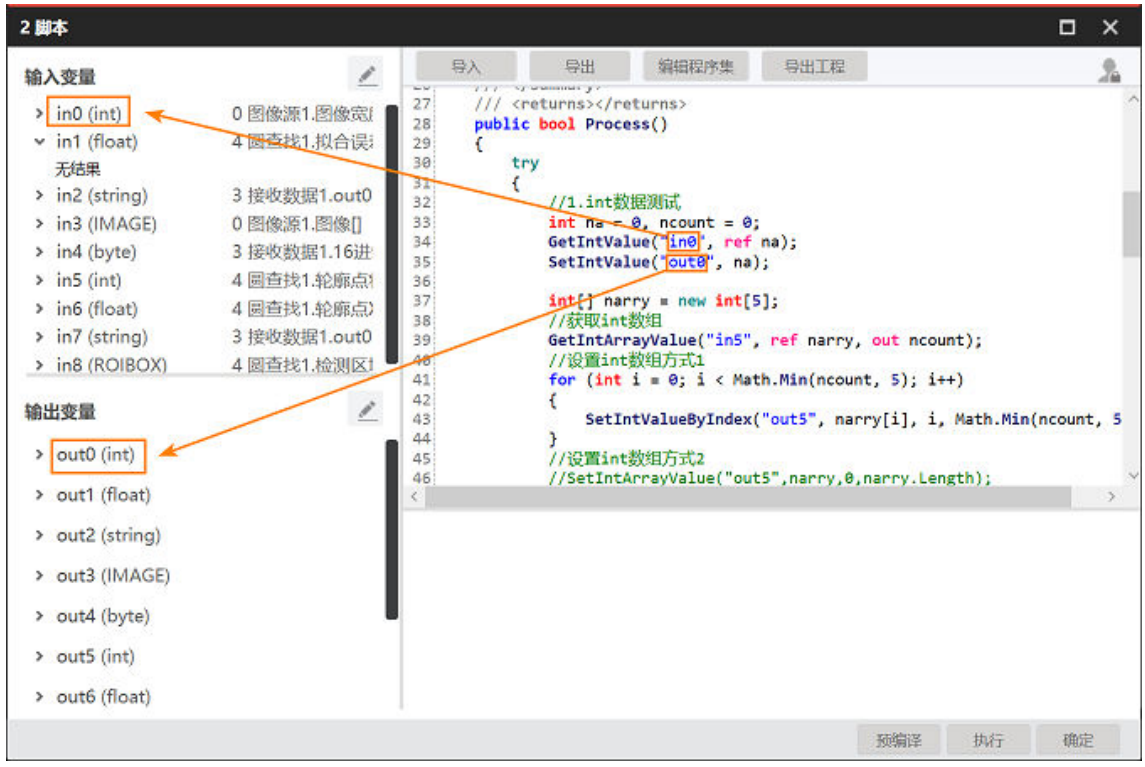


图 13-522 脚本输入/输出变量中的 int0 和 out0

4. 调用 *ShowMessageBox* 方法，实现在脚本代码运行异常时弹窗提示。

示例

以上步骤的示例代码如下：

```
using System;
using System.Text;
using System.Windows.Forms;
using Script.Methods;
public partial class UserScript : ScriptMethods, IProcessMethods
```

```
{
    //the count of process
    //执行次数计数
    int processCount;

    /// <summary>
    /// Initialize the field's value when compiling
    /// 预编译时变量初始化
    /// </summary>
    public void Init()
    {
        //You can add other global fields here
        //变量初始化，其余变量可在该函数中添加
        processCount = 0;

    }

    /// <summary>
    /// Enter the process function when running code once
    /// 流程执行一次进入 Process 函数
    /// </summary>
    /// <returns></returns>
    public bool Process()
    {
        try
        {
            //1.int 数据测试
            int na = 0, ncount = 0;
            GetIntValue("in0", ref na);
            SetIntValue("out0", na);

            int[] narry = new int[5];
            //获取 int 数组
            GetIntArrayValue("in5", ref narry, out ncount);
            //设置 int 数组方式 1
            for (int i = 0; i < Math.Min(ncount, 5); i++)
            {
                SetIntValueByIndex("out5", narry[i], i, Math.Min(ncount, 5));
            }
            //设置 int 数组方式 2
            //SetIntArrayValue("out5",narry,0,narry.Length);

            //2.float 数据测试
            float fa = 0;
            int fcount = 0;
            GetFloatValue("in1", ref fa);
            SetFloatValue("out1", fa);

            float[] farry = new float[5];
            //获取 float 数组
            GetFloatArrayValue("in6", ref farry, out fcount);
            //设置 float 数组方式 1
```

```
            for (int i = 0; i < Math.Min(fcount, 5); i++)
            {
                SetFloatValueByIndex("out6", farry[i], i, Math.Min(fcount, 5));
            }
            //设置 float 数组方式 2
            //SetFloatArrayValue("out6",farry,0,farry.Length);


            //3.string 数据测试
            string stra = ""; int strcount = 0;
            GetStringValue("in2", ref stra);
            SetStringValue("out2", stra);

            string[] strarry = new string[5];
            //获取 string 数组
            GetStringArrayValue("in7", ref strarry, out strcount);
            //设置 string 数组方式 1
            for (int i = 0; i < Math.Min(strcount, 5); i++)
            {
                SetStringValueByIndex("out7", strarry[i], i, Math.Min(strcount, 5));
            }
            //设置 string 数组方式 2
            //SetStringArrayValue("out7",strarry,0,strarry.Length);

            //6.设置/获取 16 进制数据
            byte[] tempBytes = new byte[] { };
            GetBytesValue("in4", ref tempBytes);
            SetBytesValue("out4", tempBytes);

            //7.设置/获取图像数据
            ImageData imagedata = new ImageData();
            int n1 = GetImageValue("in3", ref imagedata);
            int n = SetImageValue("out3", imagedata);

            //8.获取 roibox 数据
            RoiboxData roibox = new RoiboxData();
            GetRoiboxValue("in8", ref roibox);
            SetRoiboxValue("out8",roibox);
            //获取 roi 数组
            RoiboxData[] roiboxArray = new RoiboxData[100];
            int boxcount = 0;
            GetRoiBoxArrayValue("in8", ref roiboxArray, out boxcount);
            SetRoiBoxArrayValue("out9", roiboxArray, 0, boxcount);

        }
        catch (Exception ex)
        {
            ShowMessageBox(ex.ToString());
        }
        return true;
    }
}
```

步骤 4：配置全局触发

为了实现在 VM 接收到 TCP 客户端发送的字符串时触发流程执行，需配置全局触发。全局触发是 VM 的配置工具，用于配置针对方案的全局触发规则。

前提条件

已完成流程搭建。

操作步骤

1. 在软件主界面的快捷工具条单击 ，打开*全局触发*窗口。
2. 选择*字符串触发*页签，并单击 添加字符串触发规则，完成规则配置。

   匹配模式

   设置为**不匹配**。**不匹配**表示 VM 不对接收到的字符串进行匹配，默认接收到字符串就触发执行流程。

   触发命令类型

   设置为**执行流程**，即在符合该触发规则时触发流程的执行。

   触发配置

   选择已搭建的流程。

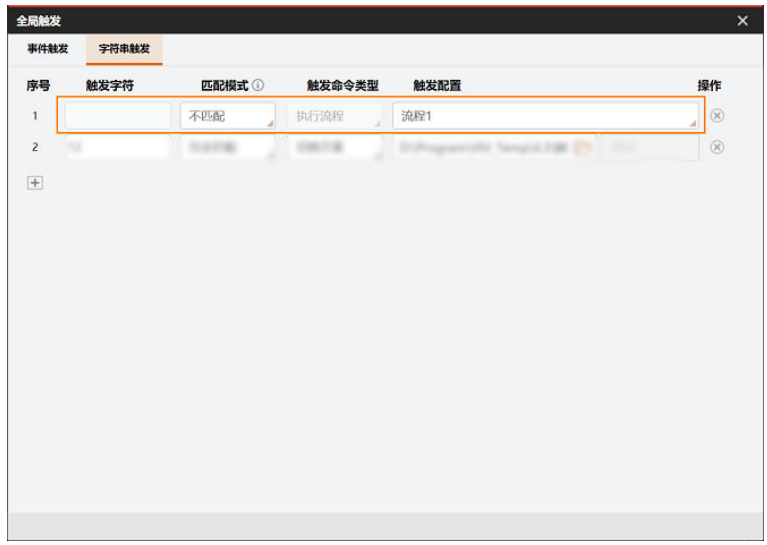


图 13-523 字符串触发配置

至此，方案已搭建完成。

方案效果测试

完成方案搭建后，可使用市面上的网络调试工具对方案效果进行测试。

从以下两张截图可见，VM 的流程最终输出的 int、float、string、IMAGE、byte 和 ROIBOX 这 6 种类型的数据，发送到了 TCP 客户端。该方案最终实现了业务需求：“VM 接收到 TCP 客户端发送的任意字符串时，触发流程执行，并将流程输出的 6 中类型的数据发送至 TCP 客户端”。



图 13-524 流程最终发出的数据

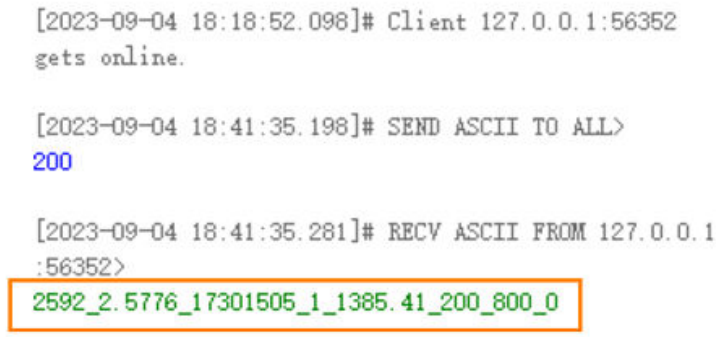


图 13-525 TCP 客户端的数据收发（截取自某网络调试工具）

13.14.10 Python 脚本

*Python 脚本*模块功能和*脚本*模块大同小异，差别主要为编程语言的不同。*Python 脚本*基于 Python 开发语言，*脚本*基于 C#开发语言。

说明

*Python 脚本*模块只能对流程内的数据进行处理。如需对方案下所有流程的批量执行逻辑进行控制，可通过 *全局脚本* 实现。

本节内容包含：

- *界面概览*
- *编程指引*
- *脚本接口*
- *结构体*
- *模块结果*

界面概览

将 *Python 脚本*模块拖入流程编辑区后，双击该模块打开 Python 脚本编辑窗口。



图 13-526 Python 脚本

该窗口各区域说明如下：

表 13-46 脚本编辑窗口介绍

窗口区域	描述
输入/输出变量编辑区	设置输入输出的变量，可自定义变量名称。变量支持多种数据类型，主要为 int（整型）、float（浮点型）、string（字符串）、byte（字节）、image（图像）、ROIBOX（ROI 内的识别框）、ANNULUS（圆环）、POLYGON（多边形）、POINT（点）、LINE（线）、FIXTURE（修正信息）、RECT（矩形）、ELLIPSE（椭圆）、pointset（点集）。 **说明** 不带“[]”的变量为一个值，带“[]”的变量为数组。 • **输入设置**：可通过**初始值**订阅 *Python 脚本*前序模块的结果或全局/局部变量。 • **输出设置**：可作为 *Python 脚本*后序模块的输入。
控制栏	• *导入*：导入脚本程序（格式：.py）。 • *导出*：导出脚本程序（格式：.py）。 • *导出工程*：将脚本程序导出，导出后可使用 Visual Studio 进行调试。 • ：单击后可设置密码以启用加密权限。启用加密后，实时生效，未验证密码只能修改输入变量的值或订阅，以及执行*预编译*、*执行*和*确定*功能；验证密码成功后可进行查看、编辑代码等其他操作；再次单击 并输入密码可取消加密。
Python 编程区	可在此处通过调用脚本接口等方式自定义开发脚本。 该区域提供脚本的默认代码及相关说明。 **说明** 在 C#编程区自定义脚本代码之前，需先在输入/输出变量编辑区定义输入变量（例如 int 型的 in0）和输出变量（例如 int 型的 out0）。完成定义后，可在 C#编程区中直接编写代码，获取或设置变量的值。
结果显示	显示该模块编译后的打印和保存信息。
其他功能	• *预编译*：可对脚本程序进行预编译，单击该按钮即调用 Init 方法。 • *执行*：单击该按钮即调用 Process 方法。 • *确定*： 保存修改后的代码并退出脚本编辑界面。

编程指引

可在 Python 编程区可自定义编写脚本代码，其核心接口 Process。

可在 Process 方法中实现变量计算和逻辑处理等自定义逻辑，相关功能在流程执行时生效。在 Process 方法内自定义代码逻辑时，成功返回 0，否则返回其他值。

注意

- 自定义代码逻辑需写入 try...except...中，方便定位异常，同时避免内存泄漏。
- Python 代码严格要求按照 4 个空格缩进，不支持 tab 缩进。

进入 Process 方法后，需先初始化变量，否则无法访问输入/输出变量、全局/局部变量。具体代码如下：

```
moduleVar = IoHelper(data, INIT_MODULE_VAR)
globalVar = IoHelper(data, INIT_GLOBAL_VAR)
localVar = IoHelper(data, INIT_LOCAL_VAR)
```

说明

- 其中 moduleVar 表示该模块中设置的输入和输出变量，globalVar 表示方案中的全局变量，localVar 表示模块所处流程或 *Group* 模块中的局部变量。
- 所有变量都需严格根据变量类型使用。若需转换类型转换，应在用户代码内通过显示转换代码进行类型转换。
- 默认代码中已有初始化变量的相关逻辑，不需要手动额外添加。

读写输入输出变量的方式如下：

- **输入变量**为只读，无法修改。假设输入变量名为 in0，类型为 int，可通过 tmp = moduleVar.in0 读取该变量的值。
- **输出变量**为只写，无法读取。 假设输出变量名为 out0，类型为 int，可通过 moduleVar.out = 9 设置该变量的值。

说明

全局变量和局部变量支持读取和修改，操作方法类似，但需将 moduleVar 对应更换为 globalVar 或 localVar。

使用方法

Python 脚本模块支持自定义代码逻辑处理输入数据，实现基本算法模块无法实现的数据处理逻辑，帮助您提高开发效率和灵活适应各种应用场景。

使用场景

*Python 脚本*模块适用于在实现一些复杂或基本模块无法实现的逻辑时使用，通过自定义代码实现这些数据处理逻辑。

调用模块

*Python 脚本*模块支持绝大部分数据类型的处理，无针对前置或后置模块的特定要求，在流程中可搭配各类模块使用，根据脚本自定义的代码逻辑处理输入数据，将结果数据输出给后置模块。即只要脚本逻辑与流程逻辑匹配，模块可在流程中的任意环节调用。
在流程中调用 *Python 脚本*模块后，可在 Process()方法中实现变量计算和逻辑处理等具体的功能。

应用示例

假设在某视觉业务场景中需要将定位目标匹配点的 X、Y 坐标按照 X 坐标值从小到大排列输出。基于此场景，方案流程如下。

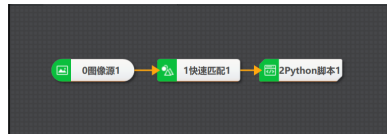


图 13-527 方案流程

该示例通过*图像源*模块输入目标图像；通过*快速匹配*模块对目标进行定位；再通过 *Python 脚本*模块对匹配点的 X、Y 坐标进行排序输出。下文只对 *Python 脚本*模块配置进行介绍，*图像源*和*快速匹配*模块详细说明请参见 *图像源* 和 *快速匹配* 。

1. 双击 *Python 脚本*模块。
2. 在 *Python 脚本*对话框，配置输入和输出变量。
   输入变量订阅*快速匹配*模块的 float 型数组；输出变量也设置为 float 型数组供后续使用，如下图所示。

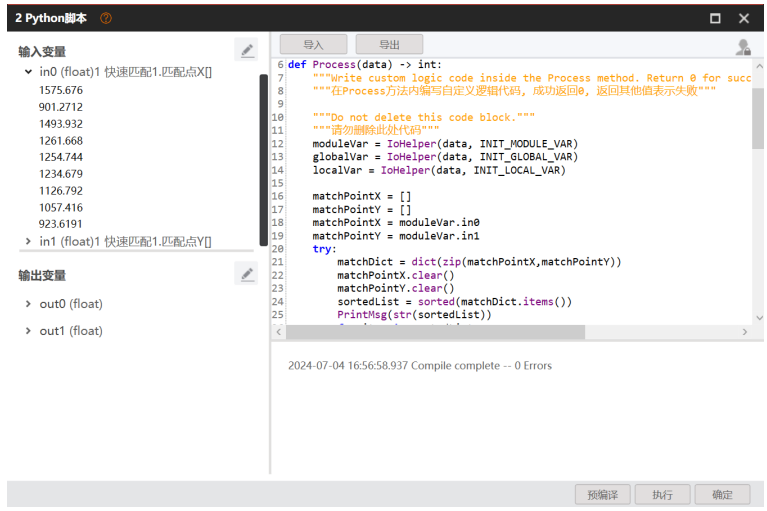


图 13-528 配置变量

3. 在 Python 编程区，输入如下示例代码。

说明

示例代码仅供参考，使用过程中请根据实际需求编写代码。

```
# coding: utf-8
import sys
from ioHelper import *


def Process(data) -> int:
    """Write custom logic code inside the Process method. Return 0 for success or any other value for failure."""
    """在 Process 方法内编写自定义逻辑代码，成功返回 0，返回其他值表示失败"""

    """Do not delete this code block."""
    """请勿删除此处代码"""
    moduleVar = IoHelper(data, INIT_MODULE_VAR)
    globalVar = IoHelper(data, INIT_GLOBAL_VAR)
    localVar = IoHelper(data, INIT_LOCAL_VAR)

    #创建空列表
    matchPointX = []
    matchPointY = []
    #输入变量赋值到列表
    matchPointX = moduleVar.in0
    matchPointY = moduleVar.in1
    try:
        #组装为字典
        matchDict = dict(zip(matchPointX,matchPointY))
        matchPointX.clear()
        matchPointY.clear()
```

```
        #按字典键（X 坐标）值进行排序
        sortedList = sorted(matchDict.items())
        PrintMsg(str(sortedList))
        for item in sortedList:
            #字典拆分
            matchPointX.append(item[0])
            matchPointY.append(item[1])
        #赋值到输出变量
        moduleVar.out0 = matchPointX
        moduleVar.out1 = matchPointY
    except BaseException as e:
        PrintMsg(e)
    return 0
```

4. 进行*预编译*和*执行*操作。

代码执行完毕后可对比查看*快速匹配*模块图像和 *Python 脚本*模块输出变量，验证执行结果，如下图所示。

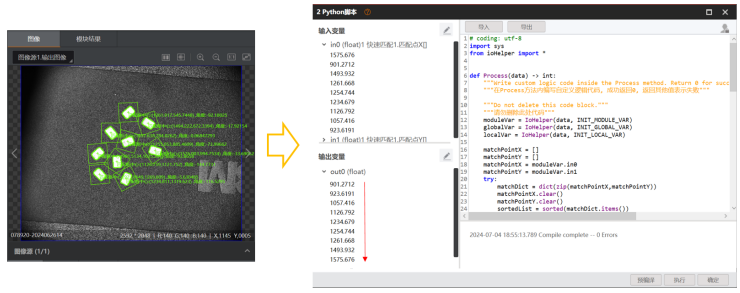


图 13-529 验证执行结果

脚本接口

*Python 脚本*提供个别开放接口，方便您通过代码控制变量。

- *globalVar.GetValue* ：获取全局变量
- *globalVar.SetValue* ：设置全局变量
- *localVar.GetValue* ：获取局部变量
- *localVar.SetValue* ：设置局部变量

globalVar.GetValue

接口原型

globalVar.GetValue(paramName)

功能描述

获取全局变量

输入参数

paramName：string 类型，全局变量名称。

返回值

成功返回全局变量的值，失败返回 None。

globalVar.SetValue

接口原型

globalVar.SetValue(paramName，paramValue)

功能描述

设置全局变量

输入参数

paramName：string 类型，全局变量名称。

paramValue：全局变量的值。

返回值

成功返回 0，失败返回非 0。

localVar.GetValue

接口原型

localVar.GetValue(paramName)

功能描述

获取局部变量

输入参数

paramName：string 类型，局部变量名称

返回值

成功返回全局变量的值，失败返回 None。

localVar.SetValue

接口原型

localVar.SetValue(paramName，paramValue)

功能描述

设置局部变量

输入参数

paramName：string 类型，局部变量名称

paramValue：局部变量的值

返回值

成功返回 0，失败返回非 0。

结构体

使用 *Python 脚本*编写代码时，会使用到一些结构体。

具体包含如下结构体：

- *RoiAnnulus*
- *Circle*
- *ELLIPSE*
- *Fixture*
- *ImageData*
- *Line*
- *Point*
- *PointSet*
- *Rect*
- *RoiBox*
- *RoiPolygon*
- *Annulus*

RoiAnnulus

对应 ROI 圆环。相关参数如下：

center_x

ROI 圆环中心点 X

center_y

ROI 圆环中心点 Y

inner_radius

ROI 圆环内径

outer_radius

ROI 圆环外径

start_angle

ROI 圆环起始角度

angle_extend

ROI 圆环展开角度

Circle

对应圆。相关参数如下：

radius

半径

center_x

圆心 X

center_y

圆心 Y

ELLIPSE

对应椭圆。相关参数如下：

center_x

椭圆中心点 X

center_y

椭圆中心点 Y

major_radius

椭圆长轴

minor_radius

椭圆短轴

angle

椭圆角度

Fixture

对应位置修正信息。相关参数如下：

init_point_x

基准点 X

init_point_y

基准点 Y

init_angle

基准角度

init_scale_x

基准尺度 X

init_scale_y

基准尺度 Y

run_point_x

运行点 X

run_point_y

运行点 Y

run_angle

运行角度

run_scale_x

运行尺度 X

run_scale_y

运行尺度 Y

ImageData

对应图像数据。相关参数如下：

width

图像宽度

height

图像高度

pixel_format

像素格式

buffer

图像数据

dataLen

数据长度

Line

对应直线。相关参数如下：

start_point_x

直线起点 X

start_point_y

直线起点 Y

end_point_x

直线终点 X

end_point_y

直线终点 Y

Point

对应点。相关参数如下：

point_x

点 X

point_y

点 Y

PointSet

对应点集。相关参数如下：

buffer

点集数据

dataLen

数据长度

Rect

对应矩形。相关参数如下：

rect_x

矩形起点 X

rect_y

矩形起点 Y

width

矩形宽度

height

矩形高度

RoiBox

对应 Box。相关参数如下：

center_x

Box 中心点 X

center_y

Box 中心点 Y

width

Box 宽度

height

Box 高度

angle

Box 角度

RoiPolygon

对应多边形。相关参数如下：

point_num

多边形点数

point_x

多边形顶点 X

point_y

多边形顶点 Y

Annulus

对应圆环。相关参数如下：

center_x

圆环中心点 X

center_y

圆环中心点 Y

inner_radius

圆环内径

outer_radius

圆环外径

start_angle

圆环起始角度

angle_extend

圆环展开角度

模块结果

*Python 脚本*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.11 Group

在复杂方案中，模块过多可能导致查看或修改方案时不够直观，此时可使用 Group 进行模块整合，同时 Group 也兼容循环的功能。

本节内容包含：

- *使用方法*
- *局部变量*
- *参数配置*
- *模块结果*

使用方法

对于 Group 模块，通过双击“组合模块”即可进入 Group 内部，此时会单独弹出一个组合模块的流程窗口。在该窗口界面中，可以直接拖动相关模块进行连接，搭建完成后单击 可返回至组合模块外面。



图 13-530 Group 示例

Group 模块多用于“多目标检测场景”、“多目标精定位”等场景。前序模块一般搭配特征匹配、位置修正、Blob 分析等定位模块输入定位信息，Group 模块内部一般搭配数据集合、点集、图形收集等模块进行数据汇总，后序模块选择较为广泛，可搭配逻辑模块、运算模块等。对于 Group 模块，右键单击还可执行以下操作：

- *导出 Group*：可将该模块以 gro 文件形式导出到本机。

  说明

  选中流程编辑区域的空白处，通过*导入 Group* 可导入本机保存的 gro 文件。

- *启用加密*：可设置该模块的密码。每次双击该模块需输入密码方可进入。
- *添加到自定义工具*：该模块会自动添加至左侧工具箱中的自定义组工具中呈现，便于快速拖动使用。

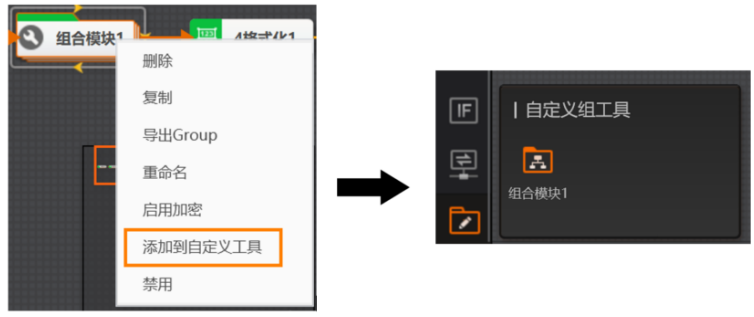


图 13-531 添加到自定义工具

局部变量

单击流程栏的 var，可设置 Group 模块的局部变量。此处设置的局部变量，仅供该 Group 中的模块使用。详细介绍参见 *全局/局部变量*。

参数配置

单击组合模块左侧或流程栏的 可进行输入设置、输出设置、运行参数、显示设置以及循环设置，并支持图像显示和查看输出结果。其中，输入设置、输出设置、运行参数、显示设置、图像显示、输出结果与流程配置功能基本一致，仅针对的对象有所差别，此处不再赘述，具体参见 *步骤三：定义各流程的输入输出*。

说明

- Group 支持累积输出功能，在*输出设置*页签为参数打开*累积输出*开关后，可累积此输出参数结果，即输出元素数量为 Group 模块循环次数的结果数组。使用此功能可以代替*数据集合*模块功能。
- Group 支持累积显示功能，在*显示设置*页签为参数打开*累积显示*开关后，可累积显示参数结果。
- Group 的结果显示只有当**输出设置**完成配置后才会输出“模块状态:1”。若未完成配置，即使 Group 模块里的模块运行状态均为 1，也会显示模块状态为 0。同时历史结果中只显示模块输出设置的数据类型，未配置的数据类型不会输出。

循环设置为 Group 模块特有功能，可设置 Group 所包含模块如何循环执行。在进行循环设置前，需在 Group 内部完成模块的搭建，并完成输入/输出等设置。

说明

Group 循环最多可以显示 10000 次渲染结果。

循环使能

开启**循环使能**后，在 Group 内订阅参数时，您可嵌套订阅**循环索引**，用于表示 Group 循环执行的次数。

循环起始值

自定义循环计数起始值，一般设置为 0。

循环结束值

需订阅相关参数值，循环结束值与循环起始值之差即为循环次数。

循环间隔(ms)

单次循环的间隔时间。

中断循环

开启后，需完成相关参数设置。当满足条件时，Group 内循环终止。

数据类型

选择中断循环的数据类型。

源比较值

可自定义或绑定源比较值。

目的比较值

目的比较值和源比较值之间进行比对，满足条件时即达到循环中断要求。

模块结果

Group 模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.12 点集

点集工具可以将其他模块输出的相关数据组合为点的集合，便于后续模块获取及传输。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

点集模块支持点或者坐标输入，可开启循环使能放入循环模块使用，搭配 Group 模块做点的汇集。后续模块可搭配图形拟合工具，通过点集输出一组特征点，图形拟合根据其拟合直线、圆、椭圆等形状。
该示例通过输入两个圆查找模块的圆心点，再通过点集模块输出两个圆心点的集合。



图 13-532 点集示例

参数配置

点输入

选择**按点**或者**按坐标**输入其他模块的点数据，比如圆心点、顶点。最多仅支持添加 16 项点数据。

循环使能

开启后，会将每次循环得到的点集添加到输出点集中。

模块结果

*点集*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

点集

点 X

float 型，代表点集 X 坐标。

点 Y

float 型，代表点集 Y 坐标。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.13 耗时统计

*耗时统计*主要用于统计流程间指定模块间的耗时总和，常用于检测节拍、检测部分模块耗时、验证方案耗时是否稳定。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

*耗时统计*用于统计流程中两个模块之间的耗时，放在需统计模块的后面即可。

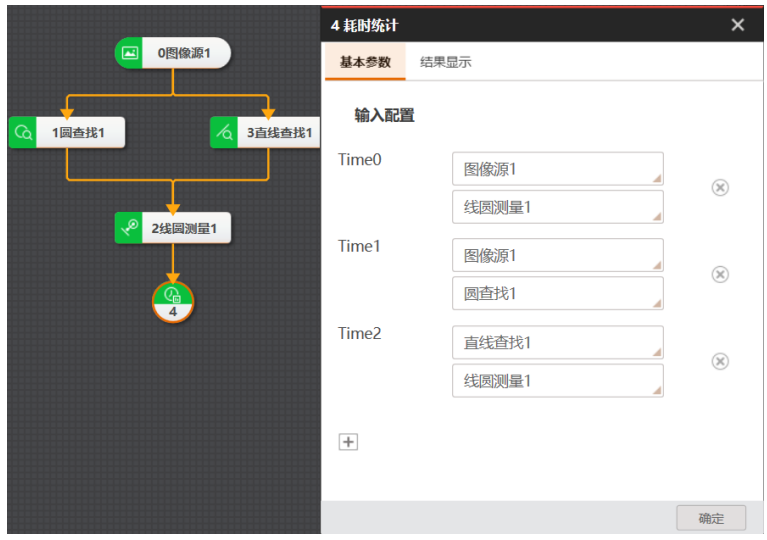


图 13-533 耗时统计示例

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见*结果显示*。

每点击 ⊞ 一次，可添加 1 个需统计耗时的数据 Time*。最多可添加 64 个。在每个 **Time***的第一个下拉框选择需统计耗时数据的开始模块，第二个下拉框选择结束模块即可。

说明

第二个下拉框只能选择第一个下拉框选择模块的后序模块。

模块结果

*耗时统计*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

Time*

float 型，代表运行时该模块基本参数中设置的各个 Time*之间耗费的时间。

13.14.14 数据集合

当 Group 循环内部循环执行，生成多个结果数据时，可通过数据集合模块进行整合输出，通过清空信号设置可控制输出所有循环结果或输出最后一次数据结果。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

数据集合模块一般在 Group 模块内使用，常应用于 Group 循环内部数据整合的场景。在循环外部使用格式化模块绑定 Group 输出，可以输出完整数组。

说明

Group 模块支持*累积输出*开关，开启后可将输出参数结果累积，推荐您使用此方式整合输出多个结果数据。当您在 *Group* 模块中添加使用*数据集合*模块时，同时开启*累积输出*开关可能导致结果错误。

该示例是当循环内部的快速特征匹配执行多次时，数据集合绑定快速特征匹配的匹配点 X，同时将清空信号设置成空，输出结果如下图所示。



图 13-534 数据集合示例

参数配置

清空信号

- 空：完整输出循环数据结果，同时在下一次循环开始前清空数据。
- 0：完整输出循环数据结果，但是在下一次循环开始前不进行数据清空。
- 非空非 0：输出最后一次循环生成的数据结果。

名称

自定义数据名称，数据要输出时需要在 Group 输出配置里将其配置成输出数据。

类型

需要整合的数据类型，包括 int、float、string 类型。

数据源

订阅需要整合的前序模块数据。

模块结果

*数据集合*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.15 触发模块

触发模块可通过流程 ID 选择方案中除该模块所在流程以外的其他流程，使得触发模块执行时触发其他流程执行。

使用方法

触发模块多用于节拍紧凑、流程逻辑性较强的应用。
该示例的触发模块处于流程 1 中，**流程 ID** 可选择流程 1 以外的其他流程，如流程 2 或流程 3，如下图所示。在 Group 模块输出结果时，使用触发模块设置**流程 ID** 为流程 2，则会触发流程 2 开始运行。

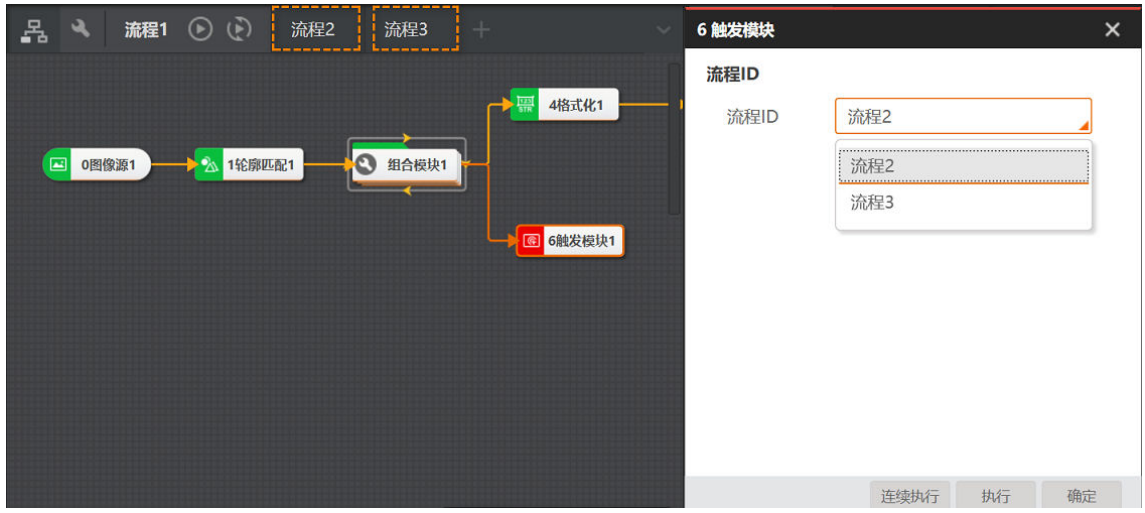


图 13-535 触发模块示例

参数配置

流程 ID

可选方案中触发模块所在流程之外的其他流程 ID。

模块结果

*触发模块*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.16 数据记录

*数据记录*模块主要用于整合并存储前置模块输出的各种数据（int 或 float 型数据）。

本节内容包含：

- *使用场景*
- *使用方法*
- *参数设置*
- *模块结果*
- *应用示例*

使用场景

*数据记录*模块整合并存储前置模块输出的 int 或 float 型数据，并将储存的数据提供给运行界面进行图形绘制时使用。当需要在运行界面呈现数据占比或趋势时，*数据记录*模块可以被图表类控件的数据源订阅，可以更方便快捷地展示当前缓存数据的趋势情况。

使用方法

*数据记录*模块可与较多模块配合使用，前置模块只要输出 int 或 float 型数据即可；后续由运行界面数据展示类控件订阅使用该模块数据，用于展示缓存数据波动趋势。例如运行界面的*饼图*订阅该模块数据后可查看数值分布情况，*线图阵列*或*多数据线图*订阅该模块数据后可查看数据趋势。

参数设置

该模块参数主要分为以下几类：

清空信号

可设置是否清空模块中已存储的数据。可自定义或订阅 int 型数据。

空值或为 0 时，不清空数据；其他数值时，则清空数据。

集合数据

每单击一次 ⊞ 可添加一个数据。每个数据由以下参数组成。

名称

可自定义设置数据的名称。

类型

设置数据的类型，可选 int 或 float。

数据源

可从前序模块输出的数据中订阅同类型的数据，也可自定义设置。

模块结果

*数据记录*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

此外，还可输出**集合数据**中添加的参数。

应用示例

假设在某视觉业务场景中需要查看灰度匹配定位图像中匹配小球个数变换情况。基于此场景，方案流程示例如下。



图 13-536 方案流程

该示例通过*图像源*模块输入待检测小球图像；通过*灰度匹配*模块匹配定位图像中小球个数；通过*数据记录*模块记录匹配个数并进行缓存；再通过运行界面*线图阵列*控件展示个数变化情况。下文只对*数据记录*模块配置进行介绍，*图像源*、*灰度匹配*模块和*线图阵列*控件详细说明请参见 *图像源* 、 *灰度匹配* 和 *线图阵列* 。

1. 双击*数据记录*模块。
2. 在*数据记录*对话框，添加 int 型集合数据，*数据源*订阅*灰度匹配*模块的匹配个数。



图 13-537 添加集合数据

3. 编辑运行界面添加*线图阵列*控件，*数据源*选择*数据记录*模块的集合数据。
4. 在*数据记录*对话框，单击*连续执行*。

执行完成后，进入运行界面查看*线图阵列*趋势图，与*灰度匹配*模块的预览图对比，验证结果。

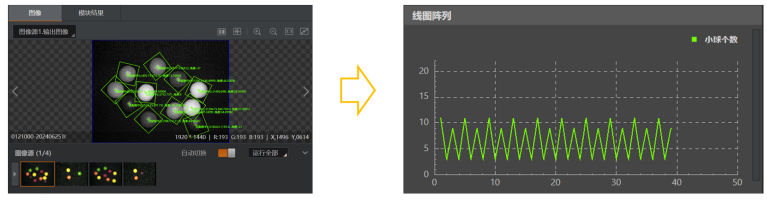


图 13-538 结果验证

13.14.17 图形收集

图形收集模块可根据实际需求订阅其他模块的数据，收集不同类型的图形（如点、直线、矩形框、扇环形、文本和椭圆），并综合显示在图像上。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

图形收集模块多用于界面效果显示以及输出渲染图像等。
该模块无固定搭配模块，可以在 Group 中使用图形收集模块。

说明

Group 模块支持*累积输出*开关，开启后同时在*输出设置*页签勾选显示框，可将图形叠加显示，推荐您使用此方式整合显示多个结果图形。



图 13-539 图形收集示例

该示例使用 Group 模块做间距测量，条件检测模块中将满足需求判为 OK 且渲染信息为绿色，将不满足判为 NG 且渲染信息为红色，在 Group 中使用图形收集将匹配框、条件检测结果状态及间距值按照对应颜色显示在图像上，最后在 Group 外使用输出图像模块进行渲染图像输出。

参数配置

点击 ⊞ 可添加一个需显示的图形。支持的图形为点、直线、矩形框、扇环形、文本和椭圆。每个图形通过以下参数完成设置。

参数名称

选择图像类型后会显示对应名称，若图形类型重复添加，则会在名称上加后缀作以区分，如矩形框 0。此外，也可手动修改参数名称。

类型

选择图形类型并点击 ⊞ 后，会自动生成，不可修改。

关联关系

订阅对应图形区域，选择*按区域*时需订阅**区域**和**目标状态**，选择*按参数*时需订阅图形相关参数信息。

说明

- 支持通过订阅脚本模块输出自定义图形，图形数据通过脚本实现。
- 若设置的图形订阅的**目标状态**为空或未订阅时，直接输出订阅的**区域**。

模块结果

*图形收集*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.14.18 数据筛选

*数据筛选*模块可将订阅模块的数据按照设置的条件筛选，并输出筛选后的结果。

本节内容包含：

- *使用场景*
- *使用方法*
- *参数配置*
- *模块结果*
- *应用示例*

使用场景

*数据筛选*模块用于将前置模块输入的数据按特定条件进行筛选，输入数据支持多种数据类型，筛选条件按照“或”逻辑执行，筛选条件内容可设置 Int 或 Float 型判断范围对数据进行筛选。

使用方法

*数据筛选*模块可与较多模块配合使用，前置模块只要输出此模块支持的数据类型数据即可；筛选后输出数据根据实际需求供后续模块进行使用。

参数配置

该模块参数分为基本参数和筛选设置。需先完成基本参数设置，再进行筛选设置。

- 基本参数：

  输入源

可选择该模块的前序模块，作为下面数据的输入源。

数据

选择数据类型后，单击 ⊞ 即可添加数据。每个数据由以下参数组成。

说明

最多支持添加 10 条数据。

参数名称

可自定义设置数据的名称。

类型

显示数据的类型。添加时可选择数据类型，添加后无法修改。

可选数据类型为：多选、int（整型）、 float（浮点型）、 BOX（目标区域）、 POINT（点）、 LINE（线）、 ANNULUS（圆环）、 RECT（矩形）、POLYGON（多边形）、ELLIPSE（椭圆）。

说明

类型选择多选时，可批量添加多个多种类型的数据。

数据源

可从**输入源**选择的模块中订阅同类型的数据，也可自定义设置。

- 筛选设置操作方法如下：
  1. 单击**条件列表（或）**右侧的 ⊞ 添加筛选条件。

     说明

     可添加多个筛选条件。不同条件之间是“或”的关系，即其中一个满足，则模块输出 OK；均不满足时，模块输出 NG。
  2. 右侧选择数据类型（可选 Int 或 Float）并单击 ⊞ 添加数据判断条件。

     说明

     每个筛选条件可添加多个数据判断条件。不同判断条件之间是“且”的关系，即所有都满足，则该筛选条件为 OK，否则为 NG。
  3. **内容**处订阅需判断的参数。仅支持订阅该模块**基本参数**中添加的参数。
  4. 通过**判断条件**设置数据判断的条件，在范围内为 OK，否则为 NG。

模块结果

*数据筛选*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

数据个数

int 型，代表动态输出的输出个数。

此外，还可输出*基本参数*中添加的参数。

应用示例

假设在某视觉业务场景中需要筛选出经过 *Blob 分析*模块定位、分析方块后，满足方块个数为 1、像素面积在 50000 和 100000 之间的方块数据。基于此场景，方案流程示例如下。

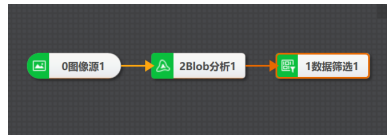


图 13-540 方案流程

该示例通过*图像源*模块输入待检测方块图像；通过 *Blob 分析*模块定位、分析图像中方块个数和像素面积；通过*数据筛选*模块筛选复合要求的数据。下文只对*数据筛选*模块配置进行介绍，*图像源*和 *Blob 分析*模块详细说明请参见 *图像源* 和 *Blob 分析* 。

1. 双击*数据筛选*模块。
2. 在*数据筛选*对话框的*基本参数*页签，**输入源**选择 Blob 分析；添加 int 型参数并订阅 *Blob 分析*模块的 **Blob 个数**；添加 float 型参数并订阅 *Blob 分析*模块的**总面积**。



图 13-541 基本参数

3. 在*筛选设置*页签，增加筛选条件，筛选条件中增加 Int 和 Float 型判断范围的逻辑内容，*内容*分别订阅新增的参数，*判断范围*分别为 1~1 和 50000.00~100000.00。



图 13-542 筛选设置

4. 在*数据记录*对话框，单击*连续执行*。

执行完成后，在*数据筛选*模块的模块结果处查看筛选数据，与 *Blob 分析*模块的预览图对比，验证结果。



图 13-543 结果验证

13.14.19 数据分类

*数据分类*模块可将数据按照各个分类中的筛选条件进行分类。

本节内容包含：

- *使用场景*
- *使用方法*
- *参数配置*
- *模块结果*
- *应用示例*

使用场景

*数据分类*模块用于将前置模块输入的数据按特定条件进行分类，输入数据支持多种数据类型。分类默认按照“且”逻辑执行，当打开*开启互斥*开关后，数据默认加入第一个满足条件的分类，在分类列表中排在上面的分类优先级越高。分类的满足条件间按照“或”逻辑执行，满足条件的逻辑内容按照“且”逻辑执行，满足条件逻辑内容可设置 Int 或 Float 型判断范围对数据进行筛选。

使用方法

*数据分类*模块可与较多模块配合使用，前置模块只要输出此模块支持的数据类型数据即可；分类后输出数据根据实际需求供后续模块进行使用。

参数配置

该模块参数分为基本参数和分类设置。需先完成基本参数设置，再进行分类设置。

- 基本参数：

  输入源

  可选择该模块的前序模块，作为下面数据的输入源。

  数据

选择数据类型后，单击 ⊞ 即可添加数据。每个数据由以下参数组成。

说明

最多支持添加 10 条数据。

参数名称

可自定义设置数据的名称。

类型

显示数据的类型。添加时可选择数据类型，添加后无法修改。

可选数据类型为：多选、int（整型）、 float（浮点型）、 BOX（目标区域）、 POINT（点）、 LINE（线）、 ANNULUS（圆环）、 RECT（矩形）、POLYGON（多边形）、ELLIPSE（椭圆）。

说明

类型选择多选时，可批量添加多个多种类型的数据。

数据源

可从**输入源**选择的模块中订阅同类型的数据，也可自定义设置。

- 分类设置操作方法如下：
  1. 单击**分类**右侧的 ⊞ 添加分类。

     说明

     可添加多个分类。每个分类之间可能存在重叠的条件，此时数据会同时存在两个分类中。此时可启用**开启互斥**功能，则数据被高优先级的分类选中，其他分类中不会出现。**分类**列表中从上往下，优先级依次降低。
  2. 单击**条件列表（或）**右侧的 ⊞ 添加筛选条件。

     说明

     可添加多个筛选条件。不同条件之间是“或”的关系，即其中一个满足，则模块输出 OK；均不满足时，模块输出 NG。
  3. 右侧选择数据类型（可选 Int 或 Float）并单击 ⊞ 添加数据判断条件。

     说明

     每个筛选条件可添加多个数据判断条件。不同判断条件之间是“且”的关系，即所有都满足，则该筛选条件为 OK，否则为 NG。
  4. **内容**处订阅需判断的参数。仅支持订阅该模块**基本参数**中添加的参数。
  5. 通过**判断条件**设置数据判断的条件，在范围内为 OK，否则为 NG。

模块结果

*数据分类*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

此外，还可输出**分类设置**中添加的分类相关参数。

应用示例

假设在某视觉业务场景中需要根据个数和面积分类经过 *Blob 分析*模块定位、分析的方块。基于此场景，方案流程示例如下。



图 13-544 方案流程

该示例通过*图像源*模块输入待检测方块图像；通过 *Blob 分析*模块定位、分析图像中方块个数和像素面积；通过*数据分类*模块按设置的个数、面积范围对当前目标图像进行分类。下文只对*数据分类*模块配置进行介绍，*图像源*和 *Blob 分析*模块详细说明请参见 *图像源* 和 *Blob 分析* 。

1. 双击*数据分类*模块。
2. 在*数据分类*对话框的*基本参数*页签，*输入源*选择 Blob 分析；添加 int 型参数并订阅 *Blob 分析*模块的 **Blob 个数**；添加 float 型参数并订阅 *Blob 分析*模块的**总面积**。

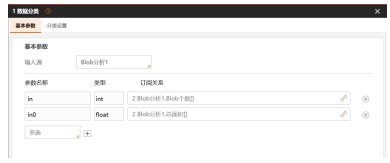


图 13-545 基本参数

3. 在*分类设置*页签，增加分类 0 和分类 1，**Blob 个数**为 1 属于分类 0；总面积在 50000.00~200000.00 范围内属于分类 1。



图 13-546 分类设置

4. 在*数据分类*对话框，单击*连续执行*。

执行完成后，在*数据筛选*模块的模块结果处查看筛选数据，与 *Blob 分析*模块的预览图对比，验证结果。



图 13-547 结果验证

13.14.20 数据排序

*数据排序*模块可对前序模块输出的结果根据要求进行排序。

本节内容包含：

- *使用场景*
- *使用方法*
- *参数配置*
- *模块结果*
- *应用示例*

使用场景

*数据排序*模块用于将前置模块输入的数据按特定条件进行排序，输入数据支持多种数据类型，排序时进行组合排序，当前一规则无法区分大小时，会按顺序使用后面的规则。主要应用在对数据输出有顺序要求的场景中。

使用方法

*数据排序*模块可与较多模块配合使用，前置模块只要输出此模块支持的数据类型数据即可；排序后输出数据根据实际需求供后续模块进行使用。

参数配置

该模块参数分为基本参数和筛选设置。需先完成基本参数设置，再进行筛选设置。

- 基本参数：

  输入源

  可选择该模块的前序模块，作为下面数据的输入源。

  数据

选择数据类型后，单击 ⊞ 即可添加数据。每个数据由以下参数组成。

说明

最多支持添加 10 条数据。

参数名称

可自定义设置数据的名称。

类型

显示数据的类型。添加时可选择数据类型，添加后无法修改。

可选数据类型为：多选、int（整型）、 float（浮点型）、 BOX（目标区域）、 POINT（点）、 LINE（线）、 ANNULUS（圆环）、 RECT（矩形）、POLYGON（多边形）、ELLIPSE（椭圆）。

说明

类型选择多选时，可批量添加多个多种类型的数据。

订阅关系

可从**输入源**选择的模块中订阅同类型的数据，也可自定义设置。

- 排序设置操作方法如下：
  1. 选择数据类型（可选 Int、Float 或 Point），并单击 ⊞ 添加排序规则。

     说明

     可添加多个筛选条件。排序时进行组合排序，当前规则无法区分大小时，会按顺序使用后面的规则。
  2. **参数**处订阅需排序的参数。仅支持订阅该模块**基本参数**中添加的参数。
  3. 设置**排序规则**。Int 或 Float 类型数据可选顺序或倒序，Point 类型数据可选顺时针或逆时针。

模块结果

*数据排序*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

此外，还可输出*排序设置*中添加的参数。

应用示例

假设在某视觉业务场景中需要按照匹配点 X 坐标顺序，排序输出灰度匹配模块定位小球的匹配点、匹配框数据。基于此场景，方案流程示例如下。



图 13-548 方案流程

该示例通过*图像源*模块输入待检测小球图像；通过*灰度匹配*模块匹配定位图像中小球个数；通过*数据排序*模块按照匹配点 X 坐标顺序，排序输出灰度匹配模块定位小球的匹配点、匹配框。下文只对*数据排序*模块配置进行介绍，*图像源*、*灰度匹配*模块详细说明请参见 *图像源* 和 *灰度匹配* 。

1. 双击*数据排序*模块。
2. 在*数据排序*对话框的*基本参数*页签，*输入源*选择灰度匹配；添加 POINT 型参数并订阅*灰度匹配*模块的**匹配点**；添加 BOX 型参数并订阅*灰度匹配*模块的**匹配框**。

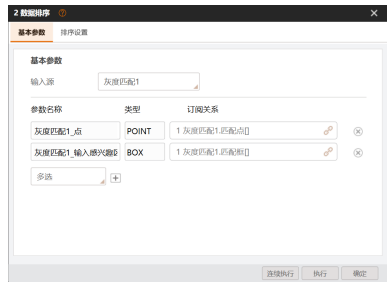


图 13-549 基本参数

3. 在*排序设置*页签，增加 float 型参数，订阅匹配点的 X 坐标，*排序规则*选择顺序。

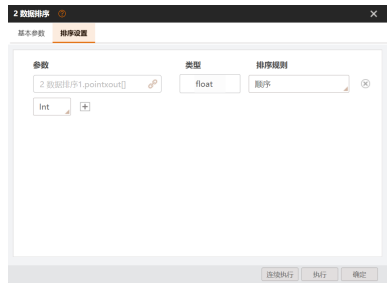


图 13-550 排序设置

4. 在*数据排序*对话框，单击*连续执行*。

执行完成后，在*数据排序*模块的模块结果处查看排序数据，与*灰度匹配*模块的预览图对比，验证结果。

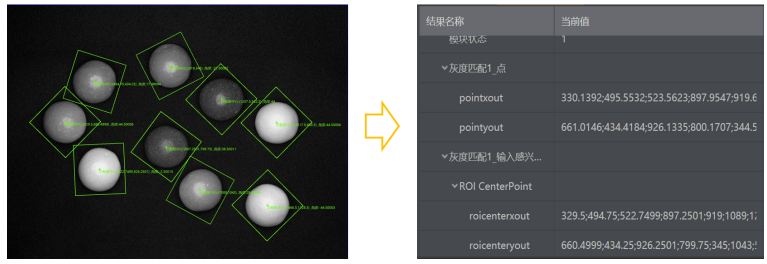


图 13-551 结果验证

13.15 通信

通信分类下的模块主要实现 VM 与外部设备（例如通信设备、相机等）通讯的功能。

13.15.1 接收数据

接收数据模块主要用于获取外部通信设备、全局变量和数据队列的数据，并将这部分数据在流程中使用。当需要从外部上位机获取数据到流程内部时，需使用该模块。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块一般与 *协议解析* 模块配合使用。协议解析模块作为后序模块，对该模块接收到的数据根据对应的解析规则转换为最终需要的数据。

说明

该模块接收的数据一般为 PLC 设备发送的 16 进制组装结果。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示*。

数据源

可选 *数据队列*、*通信设备* 和 *变量管理* 这 3 种。选择不同的数据源，设置有所差别。

数据队列

可从 *数据队列* 的参数队列中获取数据。

数据队列

可选择方案中已创建的数据队列。

获取行数

可设置模块执行一次时，从数据队列中获取几行数据。

输入数据

设置从数据队列中获取的数据，可新增多个。点击 ⊞ 后，自定义**名称**，**数据**从当前选择的数据队列中选择需要接收的队列即可。

通信设备

可从选择的 *通信设备* 中获取数据。

通信设备

可选择方案中已添加的通信设备。

输入数据

选择不同的设备，参数设置有所差别。但都只能接收一个数据。

- 通信设备选择 PLC 设备时，自定义**名称**，**数据**从当前选择的通信设备中选择有效地址即可；
- 通信设备选择非 PLC 设备时，自定义名称即可。

说明

TCP 客户端、TCP 服务端、UDP 和串口属于非 PLC 设备，其余都属于 PLC 设备。

变量管理

可从选择的 *全局/局部变量* 中获取数据。

点击 ⊞ 后，自定义**名称**，**数据**从方案的全局/局部变量中选择需要接收的变量即可。可新增多个。

模块结果

*接收数据*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

16 进制组装结果

byte 型，代表接收的数据转换为 16 进制后的结果。

13.15.2 发送数据

发送数据模块主要用于将流程中组装的数据发送给数据队列、通信设备、全局/局部变量和视觉控制器。当需要将流程中数据发送给通信设备或视觉控制器的 IO 输出时，需使用该模块。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块用于给外部通信设备等发送数据时，一般与 *协议组装* 模块配合使用。协议组装模块作为前序模块，对流程中需要输出的数据按照需求进行组装。
该模块用于控制视觉控制器的 IO 输出时，一般与 *条件检测* 模块配合使用。条件检测模块作为前序模块，对设置的条件判断是 OK 还是 NG，并将 OK 或 NG 作为结果输出。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示* 。

输出至

可选 *数据队列* 、 *通信设备* 、 *变量管理* 、 *视觉控制器* 和 *发送事件* 这 5 种。选择输出到不同的设备，设置有所差别。

数据队列

可将订阅的数据发送给 *数据队列* 的参数队列。

数据队列

可选择方案中已创建的数据队列。

忽略错误数据

开启该功能时，可在多数据发送时忽略部分数据错误继续发送。

输出数据

设置发送给数据队列的数据，可新增多个。点击 ⊞ 后，选择**选择队列**，**选择数据**订阅前序模块的模块结果即可。

说明

选择数据只能订阅和**选择队列**同一类型的数据。

通信设备

选择时，可将订阅的数据发送给外部 *通信设备* 。

通信设备

可选择方案中已添加的通信设备。

忽略错误数据

开启该功能时，可在多数据发送时忽略部分数据错误继续发送。

输出数据

选择不同的设备，参数设置有所差别。

通信设备选择 PLC 设备（即除 TCP 客户端、TCP 服务端、UDP 和串口的设备）

输出数据处先选择寄存器地址，并订阅需输出的数据。点击 ⊞ 可添加多条数据。

通信设备选择非 PLC 设备（即 TCP 客户端、TCP 服务端、UDP 和串口设备）

输出数据处除订阅发送数据外，还可设置其他参数。

16 进制发送

启用该功能时，数据以 16 进制格式发送；未启用时，以 ASCII 字符格式发送。

说明

此处的设置需与 TCP 服务端的接收设置保持一致，

结束符

启用该功能后，需选择具体的结束符，可选\r（回车）、\n（换行）、\r\n（回车换行）。每次发送的数据以选择的结束符收尾。

分隔符

启用该功能后，需设置具体的分隔符，可下拉选择或自定义输入分隔符。各发送数据间使用分隔符隔开。

说明

启用 **16 进制发送**参数时，无需设置该参数。

发送数据*

可自定义或订阅需输出的数据。点击 ⊞ 可添加多条数据。

变量管理

可将订阅的数据发送给 *全局/局部变量* 。

忽略错误数据

开启该功能时，可在多数据发送时忽略部分数据错误继续发送。

输出数据

点击 ⊞ 后，**选择变量**处选择需接收数据的变量管理，**选择数据**订阅前序模块的模块结果即可。可新增多个。

说明

选择数据只能订阅和**选择变量**同一类型的数据。

视觉控制器

选择 *视觉控制器* 时，可根据 **IO*输出条件**订阅的字符串信息（一般为 OK 或 NG）结合控制器管理中**输出类型**的设置，决定 OK 还是 NG 时 IO 输出信号进行电平反转。

视觉控制器

可选择方案中已添加的 IO 相关的控制器。点击参数右侧的 ⋯ 可打开控制器管理进行相关操作。

控制器型号

显示已选择**视觉控制器**对应的品牌。

忽略持续时间

若启用该参数，当各路 IO 满足输出条件并发生电平变化后，将锁定当前电平状态。此时在 *控制器管理* 中配置的**持续时间**参数无效。

IO*输出条件

可订阅 string 类型结果，推荐订阅结果内容时 OK 或 NG 的模块结果。

说明

若订阅的字符串内容不是 OK 或 NG，或未满足设置的条件时，IO 输出的电平信号不发生变化。

发送事件

可将订阅的数据通过 *发送事件* 发送给外部通信设备。

发送事件

可选择方案中已添加的发送事件。

忽略错误数据

开启该功能时，可在多数据发送时忽略部分数据错误继续发送。

输出数据

对加载的当前发送事件中的参数订阅数据源，可订阅前序模块的模块结果或变量管理。

模块结果

*发送数据*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

发送数据

hex 型，代表通过模块发送给的 16 进制数据。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.15.3 相机 IO 通信

相机 IO 通信模块主要用于根据设定的**输出类型**，结合订阅数据时 OK 还是 NG 判断是否输出 IO 信号，进行电平反转。当需要 VM 根据判断结果控制相机输出信号时，可使用该模块。

使用方法

该模块一般与 *条件检测* 模块配合使用。条件检测模块作为前序模块，校验判断条件后输出 OK 或 NG 的结果。该模块作为后续模块，根据订阅的结果判断是否符合条件。若符合，则 IO 输出信号输出电平反转信号。

参数配置

此处仅对该模块的基本参数进行介绍，结果显示各模块大同小异，此处不再赘述，具体参见 *结果显示*。

IO*输出条件

可订阅或自定义相机每个 IO 输出信号的输出条件。一般订阅 OK 或 NG 的字符信息，例如条件检测模块的结果。自定义输入时，仅支持输入 OK 或 NG。

说明

选择不同的**相机类型**，可设置的 IO 输出信号数量有所差异。普通相机（即标准面阵相机）支持 2 个 IO 输出信号，线阵相机支持 4 个 IO 输出信号，智能相机支持 3 个 IO 输出信号。

关联相机

可选择方案中已添加的全局相机。点击参数右侧的 可打开 *相机管理* 进行相关操作。

说明

仅支持选择相机管理中的*全局相机*。

相机类型

可设置关联相机的类型，可选普通相机（即标准面阵相机）、线阵相机和智能相机。选择普通相机或线阵相机时，还需设置持续时间相关参数。

说明

智能相机仅适用于将 VM 安装在 X86 智能相机和 X86 开放平台设备中并使用的情况。

持续时间使能

启用该功能后，还需设置**持续时间**参数，单位为 ms。用于控制 IO 输出信号的电平持续时长。若未启用该功能，则直接进行电平反转，知道下次满足条件再反转回来。

输出类型

可设置 IO 输出信号的条件，可选 OK 时输出或 NG 时输出。

模块结果

*相机 IO 通信*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

结果（STRING）

string 型，所有条件项结果都为 OK 则为 OK，否则为 NG。

13.15.4 协议解析

协议解析模块主要用于将订阅的数据按照规则分割成需要的数据。当需要对 PLC 等通信设备发送的数据或其他数据进行拆分时，可使用该模块。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块一般与 *接收数据* 模块配合使用。接收数据模块作为前序模块，用于从通信设备接收数据。

参数配置

方式选择

模块支持多种解析方式，可选文本解析、脚本解析和字节解析。

文本解析

根据**输出列表**制定的规则以及**分隔符**进行解析。都满足时，方可正常解析并输出结果；否则，模块解析失败，不输出结果。

解析内容

订阅需解析的数据，可以自定义输入需解析的数据。

分隔符

设置解析内容中各个数据间的分隔符，可下拉选择也可自定义修改。

输出列表

设置需解析的数据，可新增多个。点击 后，自定义**名称**，**类型**（支持 int、float 和 string）根据需求选择即可。

脚本解析

可根据 python 脚本中设置的规则对 16 进制数据、字符串进行解析。解析后，均以字符串方式输出。

解析内容

订阅需解析的数据，可以自定义输入需解析的数据。

路径选择

点击 选择需加载的脚本文件。

VM 安装后自带该模块的脚本示例 Receive.py，可到软件安装路径下获取，相对路径为：.\Applications\Module(sp)\x64\Communication\DataAnalysisModule。

说明

- 示例脚本仅供参考，可根据需求自行修改或全新开发。
- 此处使用的脚本为通信相关脚本，使用 python 开发。与 VM 中的全局脚本、脚本模块存在差别，不可混用。

输出列表

显示脚本文件中的参数名称及类型。

由 getOutputParam()函数定义，仅支持 string、int、float 三种类型。

说明

模块结果中参数的内容由 handleMessage(info)处理。

字节解析

可对 16 进制数据、字符串根据设置的规则进行解析。解析后，输出的数据类型以 **ASCII 数据**参数的设置为准。

解析内容

订阅需解析的数据，可以自定义输入需解析的数据。

ASCII 数据

未开启该功能时，以 16 进制方式进行解析；开启时，以 ASCII 字符方式进行解析。

输出列表

点击 ⊞ 可添加一个参数，每个参数需根据需求进行设置。支持添加多个参数。

名称

解析后数据的名称，默认为 out*，可自定义修改。

类型

可设置此条解析数据的类型，可选 int、float 和 string。

起止位置

选择需解析字节的起始和终止位。第一位为起始位，第二位为终止位。模块对选择的起始和终止位间的数据（包含起始和终止位）进行解析。

顺序

可设置数据解析后的排列顺序，可选 ABCD、BADC、DCBA、CDAB。

模块结果

*协议解析*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

13.15.5 协议组装

协议组装模块主要用于将订阅的数据按照设置的规则进行数据组装，输出 string 数据供后续模块使用。

本节内容包含：

- *使用方法*
- *参数配置*
- *模块结果*

使用方法

该模块一般与 *发送数据* 模块配合使用。该模块作为前序模块，将需要输出的数据进行组装，供发送数据模块订阅并发送给其他设备或数据队列、全局变量等。

参数配置

方式选择

模块支持两种组装方式，分别为文本组装和脚本组装。

文本组装

根据**分隔符**将**组装列表**的数据按顺序进行组装。

分隔符

可设置组装列表各个数据之间的分隔符，可下拉选择也可自定义修改。

脚本组装

根据 python 脚本中设置的规则将**组装列表**的数据按顺序进行组装。组装规则由 packMessage(info)函数定义。组装后的数据以 16 进制数据的方式输出。

路径选择

点击 选择需加载的脚本文件。

VM 安装后自带该模块的脚本示例 Send.py, 可到软件安装路径下获取, 相对路径为:.\Applications\Module(sp)\x64\Communication\DataAssembleModule。

说明

- 示例脚本仅供参考，可根据需求自行修改或全新开发。
- 此处使用的脚本为通信相关脚本，使用 python 开发。与 VM 中的全局脚本、脚本模块存在差别，不可混用。

组装列表

点击 后添加需组装的数据，可添加多条数据。每条数据自定义或订阅需组装的数据即可。

模块结果

*协议组装*模块的模块结果具体如下：

耗时（ms）

float 型，代表运行该模块所耗费的时间。

模块状态

int 型，0 代表 NG，此时模块呈现红色；1 代表 OK，此时模块呈现绿色。

组装结果

string 型，代表组装后的具体数据。

16 进制组装结果

byte 型，代表接收的数据转换为 16 进制后的结果。

第 14 章 常见问题

本章节介绍软件的一些常见问题。

14.1 单台工控机是否可以安装多套 VM 软件？

是的。您可以在单台工控机上安装多个版本的软件。
但是，当您运行软件时，默认运行最近一次安装的版本。如果需要使用其他版本的软件，您需要通过版本切换工具（4.2 及以上版本提供）切换版本。

说明

版本切换工具的使用说明，请参见 VM 用户手册中的对应章节。

14.2 是否支持加密方案？

是的。您可以在软件中加密您的方案。在保存方案时，单击*加密选项*设置方案密码。设置方案密码后，当您重新打开该方案时，需先输入密码，才可访问和编辑该方案。

注意

请妥善保存该密码。该密码无法找回。

14.3 获取软件使用授权的方式有哪些？

软件为收费软件，安装后无法直接使用。您可以通过如下 3 种方式获取软件使用授权：本机加密狗、软加密和远程加密狗，详情请参见 *获取软件使用授权* 章节。

14.4 安装时是否可选择多种授权方式？

不可以。软件安装时，您仅可选择 1 种**授权方式**。

说明

如需切换**授权方式**，您需要重新安装软件，并选择需要的**授权方式**。

14.5 能否在单台工控机上使用多个加密狗？

不可以。若您在安装时选择**本机加密狗**，在使用软件时，您仅可接入一个加密狗设备。若在单台工控机上接入多个加密狗设备，可能引起加密狗冲突，导致您无法正常使用软件。

14.6 能否在单台工控机上使用多套授权码？

可以。您可以在**激活工具**中输入多套授权码。若授权码提供的能力集有差别，软件支持多个授权码包含的全部能力集。

14.7 是否对相机出图有分辨率限制？

没有限制。在使用时，*图像源*模块对相机出图尺寸并无限制。

14.8 如何获取 SDK 和算子开发包？

VM 完成安装后，包含 SDK 及算子开发包。您可通过如下路径，获取对应文件。

- SDK：位于 VM 安装路径下，相对路径为.\Development
- 算子开发包：C:\Program Files (x86)\MVDAlgorithmSDK

说明

VM 安装路径下还包含：

- 软件：.\Applications
- 软件所需驱动：.\Drivers

14.9 软件都需要哪些驱动？

VM 完成安装后，您可在安装路径下的“Driver”文件夹找到支持 VM 正常运行的各类驱动。此处介绍部分驱动及其作用。

加密狗硬件驱动

在您将加密狗硬件接入工控机后，若加密狗指示灯未正常亮起，您可尝试重新安装加密狗驱动。

- 新版本加密狗指示灯为蓝色。您可进入.\SenseShield 文件夹并双击运行“install.bat”安装新版加密狗驱动。
- 旧版本加密狗指示灯为红色。您可进入.\EliteIV 文件夹并双击运行“install.bat”安装旧版加密狗驱动。

相机驱动

相机驱动为“MVS_SDK_Setup.exe”，当相机调用和加载等出现问题时，您可手动运行该 exe 程序，安装相机驱动。

算子包驱动

算子包驱动包含名称以“MVDAlgorithmSDK”的 exe 程序。当算子版本不匹配 VM 版本时，您可手动运行算子 SDK 的 exe 程序，安装版本适配的算子包。

运行库驱动

运行库支撑 VM 的正常运行，包含 VC 和.Net 运行库，其驱动文件名分别为以“vcredist”和“.NET”开头。若软件使用出现问题，您可尝试重新安装运行库。

第 15 章 问题排查

若在软件安装、启动及使用过程中遇到问题，您可通过此章节查看是否有相关解决方法。

15.1 安装问题排查

本节介绍如何解决安装软件时遇到的部分异常现象或报错。

15.1.1 安装过程中提示文件在 GAC 目录下不存在

在软件安装过程中，存在二次开发程序占用，导致 GAC 注册异常。需清除当前环境后，重新注册 GAC。

现象

如下图所示，软件提示“文件在 GAC 目录下不存在”。



图 15-1 安装报错提示

原因

GAC 用于储存多个应用使用的公共程序集。该问题是由于二次开发程序占用了对应路径，导致了安装时，GAC 注册异常。

解决方法

您可以按照如下步骤，清除当前 GAC 环境并重新注册。

1. 打开 *VersionSwitchingAssistant.exe* 工具。您可从安装路径下/Applications/Tools/VersionSwitchAssistant 获取该工具。
2. 单击*清除当前*，清除当前版本环境配置。
3. 清除环境配置后，在工具中单击当前安装的版本。

15.1.2 安装过程中提示环境变量配置异常

若软件安装过程中提示环境变量配置异常，进入系统属性设置，手动添加所需环境变量。

现象

如下图所示，软件提示“环境变量配置异常”。

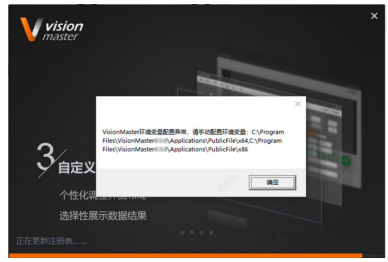


图 15-2 安装报错提示

解决方法

您可以按照如下步骤，手动配置环境变量。

1. 右击计算机图标，在右键菜单中选择*属性*。
2. 在*属性*中，依次选择 *高级系统设置* → *高级* → *环境变量*，打开**环境变量**。
3. 单击系统变量中的 *Path* 变量，再单击*编辑*，打开*编辑环境窗口*。
4. 单击*新建*，根据报错弹窗中的信息，手动配置环境变量。

说明

此处配置的环境变量路径和您在安装 VM 时选择的安装路径相关。您需分别配置：

- “VM 安装路径”+“\VisionMaster4.4.0\Applications\PublicFile\x64”。
- “VM 安装路径”+“\VisionMaster4.4.0\Applications\PublicFile\x86”。

假设您在安装时选择将 VM 安装在 D 盘的 Program Files 文件夹，则此处您需分别输入：

- D:\Program Files\VisionMaster4.4.0\Applications\PublicFile\x64
- D:\Program Files\VisionMaster4.4.0\Applications\PublicFile\x86

5. 配置完成后，单击*确定*并继续安装。

15.1.3 双击安装包或补丁包无反应，且无进程状态显示

该问题可能由工控机的系统安全等级过高导致，需解除对应安装包的锁定。

现象

双击运行安装包或补丁包时，任务管理器中未显示安装相关进程。
相关进程指如下内容：

- 4.3.0 以下版本：VmModuleProxy.exe、主进程服务的 exe 文件（名称中带 Server）、主进程的 exe 文件。
- 4.3.0 及以上版本：主进程的 exe 文件。

原因

该问题可能由工控机的系统安全等级过高引起。

解决方法

您可以按照以下步骤，解除系统对安装包和补丁包的限制。

1. 右键单击安装包/补丁包，单击*属性*。
2. 在**常规**中，单击**安全**中的*解除锁定*，取消勾选。
3. 单击*确定*或*应用*。
4. 再次安装该安装包/补丁包。



图 15-3 解除锁定

说明

若其他安装包也存在同样问题，可使用以上方式处理。

15.1.4 安装过程中提示 MVS 正在运行中

若软件安装过程中报错，提示 MVS 软件正在运行，需先关闭 MVS 软件。

现象

如下图所示，软件提示 *MVS 正在运行*。

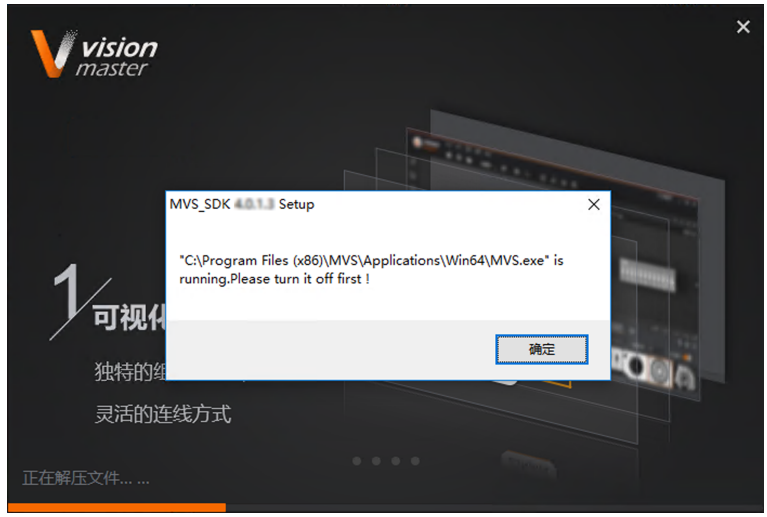


图 15-4 安装报错提示

解决方法

关闭 MVS 软件，并继续安装即可。

15.2 启动问题排查

本节介绍如何解决启动软件时遇到的部分异常现象或报错。

15.2.1 启动软件时提示找不到方法

安装依赖库不完整，可能会导致启动软件时，软件提示找不到方法。

现象

如下图所示，软件提示*找不到方法*。

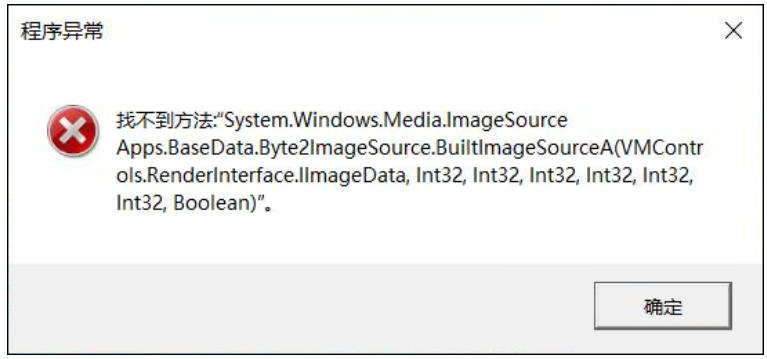


图 15-5 程序异常提示

原因

软件依赖库安装不完整。

解决方法

您可以关闭防火墙及其他杀毒软件后，重新安装软件。

说明

若您安装过其他版本的软件，您可尝试清除当前 GAC 环境并重新注册。具体操作如下：

1. 运行 ClearGAC.exe 程序，清除当前 GAC 环境。该工具可从 VM 安装路径下获取。相对路径：.\ Development\V4.x\ComControls\Assembly\Tool。
2. 在相同路径下，双击运行 GAC.bat，重新注册 GAC。

15.2.2 启动软件时提示加密狗未检测到或检测异常

加密狗异常可能由多种原因导致，您可以参照本节内容，检查加密狗状态。

现象

在软件安装时，选择通过**本机加密狗**授权。



图 15-6 加密狗异常

解决方法

您可以参照本节内容，逐步检查加密狗硬件情况并尝试解决该报错问题。

检查加密狗指示灯

在硬件加密狗设备接入工控机的情况下，检查加密狗硬件的指示灯是否正常亮起。

- 旧型号加密狗的指示灯为红色。
- 新型号加密狗的指示灯为蓝色。

检查加密狗接入数量

检查环境中是否存在其他加密硬件。例如，向含加密信息的工控机接入硬件加密狗，可能导致加密狗冲突。若存在该现象，您需拔掉硬件加密狗。

检查加密狗硬件识别

打开工控机系统的设备管理器后，尝试插拔加密狗。若设备管理器刷新，则工控机可以识别该硬件；若无，更换其他接口后，重新尝试。

若您在不同 USB 接口或工控机插拔加密狗过程中，设备管理器仍未刷新，则加密狗硬件可能存在问题，您可联系技术人员寻求帮助。

检查驱动

在设备管理器中，右键单击加密狗设备并选择*属性*，即可在*设备状态*中查看设备的状态。若此处提示错误，您可尝试重新安装驱动。

- 针对 iMVS-VM-7200(pro)、iMVS-VM-6200(pro)型号的新版加密狗，可进入 VM 安装路径获取加密狗新驱动，相对路径：.\Drivers\SenseShield。

说明

iMVS-VM-7200(pro)、iMVS-VM-6200(pro)型号仅支持 4.x.0 及以上版本软件。

- 针对 iMVS-VM-*1**型号的旧版加密狗，可进入 VM 安装路径查看加密狗旧驱动，旧驱动相对路径：.\Drivers\EliteIV。

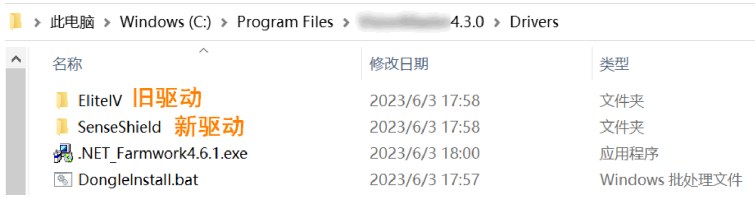


图 15-7 加密狗驱动位置

15.2.3 软件安装后启动异常

启动异常可能为多种原因导致，您可以检查软件安装环境，排查该问题。

1. 检查防火墙、白名单、软件限制以及系统权限。
2. 右键点击计算机图标，依次选择 *属性 → 高级系统设置 → 环境变量 → 系统变量 → Path 环境变量*，检查 VM 环境变量是否正确写入，并将 VM 环境变量置顶，确保 VM 环境变量在相机和算子环境变量的上方。

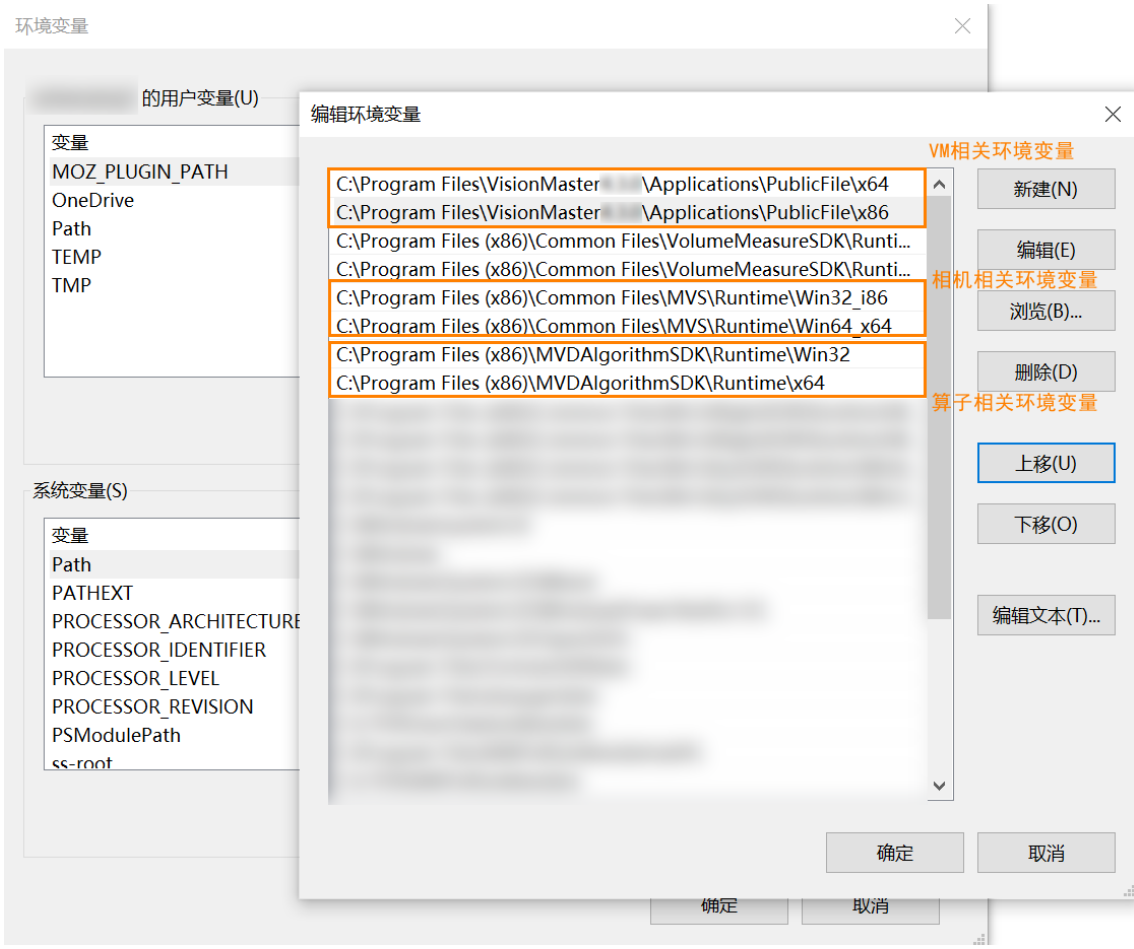


图 15-8 查看环境变量

3. 检查是否有名为“PYTHONHOME”的环境变量，该环境变量需删除，否则会影响软件正常启动。
4. 工控机上有多个版本 VM 时，检查版本与算子是否匹配；若不匹配，可使用 VM 安装路径下的版本切换工具进行版本切换，该工具相对路径为：.\Applications\Tools\VersionSwitchAssistant。
5. 若安装过 VM 3D 或二次开发环境，需使用 VM 安装路径下的 ClearGAC.exe 清除 GAC，该程序的相对应路径为.\Development\V4.x\ComControls\Assembly\Tool\ClearGAC.exe。清除后，还需执行 GAC.bat。
6. 检查环境中运行时库是否正常，运行库异常时，需手动执行 VM 安装路径下名称以 vcredist 开头的 4 个库，如下图所示。该 4 个库的相对路径：.\Drivers。

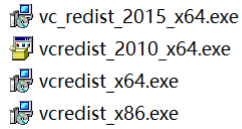


图 15-9 运行时库异常需执行文件

7. 确认是否在管理员权限下安装软件；若非在管理员权限下安装，您需重新安装 VM。

15.3 使用问题排查

本节介绍如何解决使用软件时遇到的部分异常现象或报错。

15.3.1 全局相机硬触发时无法触发其他流程

在软件中，若多个流程绑定同一相机时，图像优先传递至最靠前的可用流程。

现象

多个流程绑定同一全局相机，且通过相机硬触发执行流程时，仅能触发单个流程。

原因

在软件中，硬触发可以触发图像源模块绑定该全局相机的流程。
当多个流程同时绑定同一相机时，软件按照流程的顺序，优先将图像传递给可用的流程，用于处理图像。
如下图所示，相机以固定速率出图时，若流程 1 能够在出图间隔内运行完流程，则图像不会输入至流程 2。

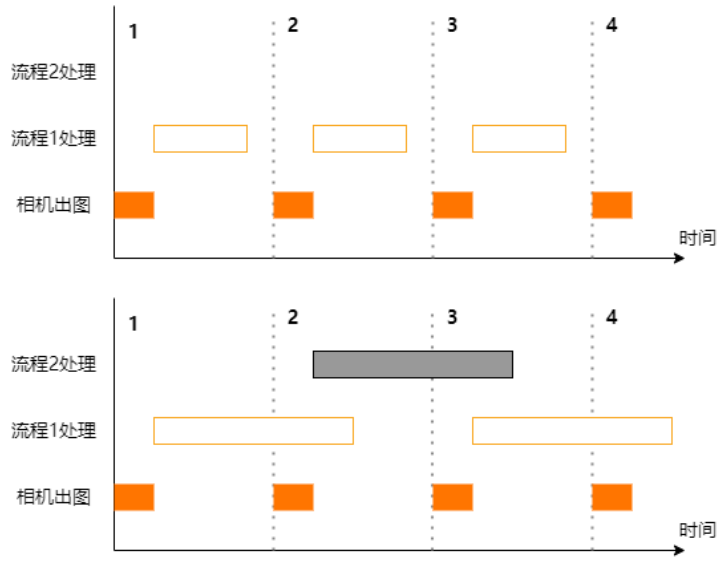


图 15-10 多流程处理

解决方法

如果想要同时触发若干流程，您可以将单个相机绑定单个流程，并通过标志信息控制图像输入哪一个流程。

15.3.2 添加全局相机时提示动态库加载失败

相机 SDK 安装失败或版本不一致时，软件加载动态库可能失败。

现象

在*相机管理*中添加全局相机时，报错提示*动态库加载失败*。

原因

相机 SDK 安装失败，或当前安装的相机 SDK 版本与 VM 安装路径下 MVS_SDK_Setup.exe 的版本不一致。

解决方法

重新安装 MVS_SDK_Setup.exe，该程序相对路径：.\Drivers\MVS_SDK_Setup.exe。

15.3.3 拖拽模块提示动态加载库失败

算子开发包安装失败，或环境变量未生效，可能导致拖拽模块时提示动态加载库失败。

现象

运行 VM 后，当从模块箱拖拽某一模块到流程区时，报错提示“动态库加载失败。”

原因

该问题可能由以下原因引起：

- 安装时，算子开发包安装失败。
- 环境变量未生效。

解决方法

针对如上原因，您可以：

- 重新安装算子开发包。算子开发包的安装程序位于 VM 安装路径下，相对路径为：.\Drivers\MVDAlgorithmSDK_STD.exe。
- 重启工控机。若重新安装算子开发包后仍出现该报错，则问题可能是环境变量未生效。您可以尝试重启工控机。

HIKROBOT

让机器更智能，让智能更普惠

扫一扫，欢迎关注

“HIKROBOT”官方微信！

杭州海康机器人股份有限公司

电话：400-989-7998

网站：www.hikrobotics.com

UD12345B