# RotbotVision 视觉软件

使用手册



## VISIONGO



### 版本记录

版本	时间	描述
1.0	09/2017	初始版本
2.0	2020. 6. 1	

VISIONGC

a a b

## VISIONGO

## 第一章 RobotVision 简介

### **RobotVision**

主要特征有:

- unite; Suite; Suite;

- 6. 支持多标定;
- 7. 支持多跟随系统;
- 8. 支持脚本编程;
- 9. 面向工厂技术员开发,简便易上手。 人可学会;

(C) AL

### 1.1. 运行环境

- Windows 7/10 64bit
- Ubuntu16.04 64-bit

### 1.2. 关于唯一编号(Uid)

唯一编号是一个 5 位的整数,为 RobotVision 根据时间自动创建。RobotVision 在多个 位置使用了唯一编号,比如模板组、模板、DI/O通讯、串口通讯、网口通讯,以及检测函数 等等可能被其它引用的模块都分配了唯一编号。



### 全局参数设置

RobotVision 的体系结构可以分为全局/任务两个部分,全局参数可以在任何一个检测任务中使用,而任务中的参数只能在当前任务中使用。全局参数包括用于模板检测的模板, DI/O 通讯的设置,串口通讯的设置,网口通讯的设置,以及数据存储的设置。

### 1.3. 模板

	板编辑音			X
	0		0	
100				
序号 1	名称 11	Browse 强边缘强度	<ul> <li>№通道 ~</li> <li>60</li> <li>60</li> </ul>	Q Q ÷
序号 1	名称 11	Browse 强边缘强度 弱边缘强度	x通道 ~ 60 30 训修	₩ Q Q ÷ ÷
序号 1 添加	名称 11 删除	Browse 强边缘强度 弱边缘强度	<ul> <li>■ 路通道 ~</li> <li>60</li> <li>30</li> <li>30</li> <li>30</li> <li>30</li> </ul>	₩ Q Q ÷ ÷ *

图 2 模板设置主界面

● 关键词说明:

使用方法请参考 4.1.1 模版查找。

● 使用方法:

第一步:点击添加,会弹出一个对话框,要求输入组的名称,此时输入一个任意的名称 即可。

第二步:增加模板,点击 Browse 添加需要训练的模板图像。

第三步:按住鼠标左键把需要训练图像画出,右键可以擦除多余部分。

第四步:点击训练,等待几秒观察图像是否生成黑色轮廓点,点击保存即可。

注意:多个相机程序不能共用一个模板。

第五步:如果对自动生成的模板不满意或者未生成轮廓黑点,可以通过调节强边缘强度 和弱边缘强度进行调节。

第七步:点击确认按钮退出对话框。

### 1.4. DI/O 通讯

	COM11 ~ 新建	打开删除
Ħ	专有ID 类型 端	<u></u>
_ ≌输入输 出	96605 Comrelay CO	<u>1M1 1</u>
	通用输入输出	
6	□ 启用Ready信号	0
- 存图像	输入延迟时间(毫秒)	0
	输入端口的脉冲时间(ms)	0
	输出端口的脉冲时间(ms)	50
	输出延迟时间(奏秒)	0
存图像	输入延迟时间(臺秒) 输入端口的脉冲时间(ms) 输出端口的脉冲时间(ms) 输出端口的脉冲时间(os)	0 0 50 0

#### 图 3 DI/O 设置界面

● 使用方法:

第一步:选择卡的类型,目前只支持 Comrelay 卡,因此不需要选择;

第二步:在下拉列表中选择 COM 号,并点击新建按钮创建一个 DI/O 连接。在创建的过程中会自动连接,如果连接成功,背景会显示为绿色,否则为红色。<u>注意:下拉列表中的数</u> 据为软件自动获取,很多笔记本电脑上并没有串口,因此本列表中可能没有值。此时可以通过串口虚拟工具在本机上虚拟几个串口。

第三步:设置通用参数:

 允许从数字输入端控制任务切换:选中后表示可以通过 DI/0 控制任务切换,否则不能通过 DI/0 进行任务切换,<u>注意:如果选中,最后一个输入端口将用来作为任务切换的指示</u> 器,也就是说最后一个输入端口必须接线,否则还是起不到任务切换的作用。详细情况请查阅章节 7.2 任务管理.

② 启用 Ready 信号:所谓 Ready 信号,是指 IO 卡一打开就直接给出高电平信号,直到 IO 卡关闭。用户可以选中指定的输出口作为 Ready 信号口。<u>注意:因为 Ready 信号口将一直</u> 给出高电平,因此该端口不能作为正常的检测结果的输出口。

③ 输入端口的脉冲时间:预留参数,目前不起作用;

④ 输出端口的脉冲时间:是指输出信号的持续时间,单位为毫秒;输出脉冲时间默认为 50ms,根据具体项目随意调整,原则是让 PLC 等能准确获取到信号;

⑤ 输出延迟时间:是指程序检测完成到实际给出高电平的间隔时间,单位为毫秒,一般情况下设置为0,表示完成后立即给出高电平信号;

第四步:切换到输入页面和输出查看当前 IO 卡的输入和输出状态,在输出页面,通过 双击某一个端口,可设置输出状态;

### 1.5. 串口和网口

11	日定义	Modbus E	therC.	AT					
角	串口	~ 客户	端	新	ŧ		连接	修改	删除
_	承	专有ID	参数			类	型		
Ĵ	1	46049	[115:	200, 8, 1	N, 1]	串			
入输									
	~~~~		a	4. <del></del>		/+ m c <sup>-</sup> /a:	20		
Ť	通用	消息后缀	s Ng	<b>汉</b> 后锁	图数输出:	结果后缀	1		1.1.1.1.1.1.1
		整数部分	长度	4					-
		小数部分的	休度	3					-
				-					
		મ	加肥团						-
	读取撙	操作后的返回	回信息						
	医眼折	\$1F/后的X区增	비밀망	-					

确定

#### 图 4 串口和网口通讯设置界面

在 RobotVision 中,串口通讯和网口通讯被放在同一个界面,主要是因为很多通讯协议都同时支持串口和网口,把他们合并在一个页面显得更简洁。

● 使用方法

第一步:新建串口和网口连接,界面如下图,首先选择通讯协议,如果是 TCP/IP 连接的话,还需要选择是客户端还是服务器。

① 新建串口连接

自定义	Modb	us Ethe	rCAT				
串口	~	客户端	~	新建	连接	修改	删除

串口新建后不会自动分配端口号,此时需点击修改进行配置。

② 新建网口连接

自定义	Modb	us Ethe	rCAT				
Top/ip		客户端	~	新建	连接	修改	删除

如果建立的是 Modbus 连接,软件只能做客户端。

- 第二步:设置协议参数
- ① 设置自定义协议:

	函数输出结果后缀	如后缀	函数	消息后缀	通用
•		4	€度	整数部分十	
-		3	£度	小数部分的十	
			胞	心晶	
		-	恴	作后的返回的	读取操

- a. 整数部分长度/小数部分长度,设置浮点数的整数位长度和小数位长度,如果实际的整数 位长度大于设置的整数位长度,实际输出时还是原始数据。举例说明,对于 127.382 这 个数字,如果整数位长度和小数位长度设置位 4 和 2,输出的结果是 0127.38,如果设置 的整数位和小数位长度为 2 和 2,输出的结果是 127.38;
- b. 心跳包,如果设置了值,每隔3S钟,程序会把心跳包发送给下位机;
- c. 读取操作后的返回信息:RobotVision从下位机成功读取数据后反馈给下位机的数据。

- d. 合格图像的前缀:当前检测任务为合格时的消息前缀;
- e. 不合格图像的前缀:当前检测任务为不合格时的消息前缀;
- f. 函数间的分隔符:当检测任务中含有多个检测函数时,各个检测函数结果之间的分隔符;常用及特殊字符可通过下拉框选择;
- g. 消息后缀:整个消息的后缀,大部分情况下为\n,常用及特殊字符可通过下拉框选择;

的前缀	和物的前缀	31 - 2
		合格
的前缀	函数的前缀	不合格
的分隔符	11的分隔符	输出之
后缀	后缀	
NUL		

- h. 合格/不合格函数的前缀:与前面合格图像的前缀不同,合格图像表示的是整个检测任务,而合格函数只针对指定检测函数,一个检测任务包含多个检测函数。
- i. 输出之间的分隔符:一个函数含有多个输出,在此可设置输出之间的分隔符;
- j. 后缀:检测函数的后缀,常用及特殊字符可通过下拉框选择;

通用	消息后缀	函数后缀	函数输出结果后缀	
结构的	本内部元素的	分隔符		
数约	且内部元素的	分隔符		
		10		

- k. 结构体内部元素的分隔符:函数的某个输出为结构体时,结构体内部元素的分隔符,比如要输出一个点(123.45,234.56),假设分隔符为逗号,则输出结果为"123.45,234.56",
   234.56",假设分隔符时分号,则输出结果为"123.45;234.56"。
- 1. 数组内部元素的分割符:用来分隔数组中各个元素,常用为逗号。
  - ② 设置 Modbus 协议参数:

自定义	Modbus	Ethe	erCAT					
Tcp/ip	~	新建				连接	修改	删除
承	专有ID	参	数			类型		
1	87480	C	lient:127.0	), (), 1 ; 60	00	Tep/ip		
	从站号	1		-	放	大系数	1	
输入数	据格式	32位落	₽点数[DC	BA] 🗸	数据	諭出格式	32位;	浮点、
串	口模式	RTU/R	S232	~		RTS	无	
RTS <u>延</u>	时(ms)	1		-				
输入	输出							
	功能	输入	寄存器[3x]					~
任务切	J <mark>换</mark> 地址	0						•
软触	拔地址	0						*

a. 从站号:默认为1,表示第一个从站;

- b. 数据格式[Fromat]:可选项 DC BA、CD AB、BA DC、AB CD,其中 ABCD 表示高位在前,低位在后;数据格式[Int]:可选项 DC BA、CD AB、BA DC、AB CD;数据格式 [Short]:可选项 BA、AB、
- c. 串口模式:可选项 RTU/RS232、RTU/RS485,根据实际情况选择;

输入	输出		
心	跳包地址	0	
数	据首地址	1	
软触发	反馈地址	0	

- d. 心跳包地址[Heartbeat package address]:如果地址大于0,程序每隔3S会给从站发送
   心跳包,心跳包数据为1;
- e. 数据起始地址[Data start address]:程序自动发送数据的起始地址,如果地址为0,将不会发送;
- f. 软触发反馈地址[Software trigger feedback address]:当通过软触发命令触发相机时, 会在此地址写入 128 的数据。

### 1.6. 数据存储

<b>F85</b>							新建	<b>HHIR</b>
通用		* = 10				53	dynam	00161
	所写	专有加						
<b>_</b>	1	30520						-
字输入输出								
щ								
0								
山/网络	□诵社	TP/277						
STORE STORE		11 (木1子						
_	服务器II	?地址					21	4
	Ħ							
保存图像								
		密码					ì	接
		\ 12 <del></del> 미 65년		R. Lt	「米肉中/1	2 10/4 立		
	≥11未仔育	当怕产品的昏	調整	<b>P</b> · /1	17684/11/1	2. 19 <u></u>	Dro	owse
	☑保存7	下合格产品的	馏像	5			Bro	owse
	) <b>**</b> ** \ -		0.00000033	-				
	<b>谊</b> 11八章	5月名称						
	保存期限	Į 7					1	刪除
		33.L					(append)	Procession of the local data

o

#### 图 5 数据存储设置界面

数据存储包括检测数据存储和图像存储,其中图像还支持 FTP 存储, FTP 存储时需先建立 FTP 连接。

如上图所示,首先点击新建按钮并输入名称,然后在右边参数区设置可选参数。新建的 任何连接都支持数据存储,如果需要存储图像,需要在界面的右方选择是否保存合格图像和 不合格图像。

自动删除策略可自动删除一定时间以前的图像。自动删除操作只在程序启动的时候操作

### 第二章 标定

RobotVision 支持两中标定,一种为圆点标定,一种为九点标定。实际上还支持一种标定,即最简单的线性标定,通过设置 Scale 和 Offset 来实现,如下图所示

通用	Circle-Points	9-Points		
Manua 比列	l  值(mm/pixel)	Pixel source Robot Auto	e Calibration	~
1.000	000	Step X	0	*
ŕ		Step Y	5	4
0.000	000	Step A	0	•
		选择机器人		~
			Auto Calibrat	tion
			25	

线性标定通过单纯的设置比例值和偏移量,在此我们不过多说明,以下我们详细讲述圆 点标定和9点标定。

### 2.1. 圆点标定

圆点标定,主要是通过计算一组圆点的中心距的变化关系来实现标定的结果。在 RobotVision中,圆点标定的步骤如下:

第一步:标定板放置到相机视野内,拍一张照片。特别注意标定板要放置得非常平整,即图像中第一排得黑点离图像顶端得距离应该很接近。如下图6所示。

第二步:添加一个斑点检测函数,设置相应的参数找出所有的黑点,设置方法请参考 <u>4.1.2 斑点分析</u>。如下图 6 所示。



图 6 圆点标定图像界面

第三步: 创建标定任务并设置数据源

- a. 如下图所示,点击新建按钮创建一个标定任务;
- b. 选中标定任务,并在右侧的 Pixel source 中选择斑点分析的中的中心输出;

序号	名称	通用 Circle-Point:	s 9-Points
1	bd1	Manual	Pixel source [27435][03]Center ~
		比列值(mm/pixel)	Robot Auto Calibration
		1.00000	Step X 0
			Step Y 5
		0.00000	Step A 0
			选择机器人
			Auto Calibration

c. 切换到圆点标定界面并设置对应参数参数,如下图所示,

序号	名称	通	用 Circle-P	oints 9-Points	
,	bd1	X	由方向点的数量	18	4
		X	曲方向点的间距	1	
		Yş	由方向点的数量	4	
		Y	曲方向点的间距	0	
					开始
碇	冊條	View			

其中 X 轴方向点的数量表示斑点工具在 X 方向找的斑点数量,本例中为 18; Y 轴方向点的数量表示斑点工具在 Y 方向找到的斑点数量,本例中为 4; X 轴方向点的间距和 Y 方向点的间距是标定板的固有参数,表示标定板两个点之间的物理距离,单位为毫米。

0

d. 点击开始按钮完成标定,并查看标定结果。其中 Transform Matrix 即为最终的标定矩阵

结果	测试		
Cente	er of rotation	-1.000000	-1.000000
	PO	131.639651	440.132170
	Resolution	0.000000	0.000000
Tra	nsform Matrix		
0.01	11289	0.000001	-1.486018
-0.000000 0.000000		0.011291	-4.970065
		0.000000	1.000000
序号	Error value		
和	0.004987		
1	0.000748		
2	0.000437		
3	0.000412		
4	0.000646		
5	0.001042		
6	0.000708		
7	0.000066		
0	0.000540		
0			

### 2.2. 手眼标定

手(机械手)眼(相机)标定用于建立相机与机械手或其它自带坐标系统的平台间的坐标关系矩阵,以实现从相机坐标系到平台坐标系的转换。经过手眼标定后,软件可以直接输出特征原点在平台坐标系下的坐标或基于平台坐标系下某点的偏移量,坐标原点为平台的坐标原点。

RobotVision 软件的手眼标定针对不同的相机安装方法有不同的标定方式,相机固定朝下安装时选择相机朝下的标定方式,相机固定朝上安装时选择相机朝上的标定方式,相机安装在机械手上时选择眼随手动的标定方式。

### 2.2.1. 通用的标定过程

#### 2.2.1.1. 九点标定



图 7 九点标定手臂路线图

第一步:设定好工业手臂的起始位置(上图中的1号点),以及X方向和Y方向的步长,起 始位置坐标和步长请尽量为整数。起始位置一般为标准拍照位置。

第二步:按上图中的路径图跑9个位置,跑到任意一个位置时,需要保存当前位置的相机图片,同时记录当前点的手臂坐标。*注意:手臂在跑9个点的时候,手臂的角度绝对不能有变*化。

第三步:设计特征点查找程序,该步骤根据产品不同而不同。

第四步:打开标定界面,如下图所示,如果添加按钮为灰色,请先选择像素源。

找入 1	保存添加	Del			标
点标定	症转中心 #	标定点集			
	图像X	图像Y	机器人X	机器人Y	
	图像X	图像Y	机器人X	机器人Y	
					1

图 8 九点标定界面

手动	像素源 [60:	104][03]Center	×
比列值(mm/pixel)	机器人自动	标定	
1.00000	X步进值	0	mo
	Y步进值	5	
0. 00000	角度步进值	0	
	选择机器人		×
		自动 标定	

图 9 设置像素源

第五步:载入第二步中保存好第一个位置的图像,调整检测工具的位置和参数,使其能找到 我们需要的特征点,找到特征点后,在九点标定界面点击添加按钮,并输入第一个位置的机 器人坐标;

第六步:重复第五步,把9幅标定图像全部输入进来。

标宁					
标宁			9点标定	圆点标定	通用
TPALE			Del	、 保存 添加	载)
			标定点集	标定 旋转中心	9点
AY ^	机器人Y	机器人X	图像Y	图像X	
)0	88.000000	2864.000000	990. 320958	1184.824155	1
)0	88.000000	2538.000000	995.875106	1964. 354583	2
)00	408.000000	2538.000000	224.401558	1972. 474735	3
000	411.000000	2858.000000	219.512433	1192.563463	4
)000 )000	88.00 408.0	2538.000000 2538.000000 2858.000000	995. 875106 224. 401558 219. 512433	1964. 354583 1972. 474735 1192. 563463	2 3 4

点击保存按钮可以把标定数据保存到 TXT 文件,点击载入按钮可以载入事先保存好的标 定数据。

第七步:点击标定按钮,完成标定。

第八步:查看标定数据

	中心 -1.0000	000	-1.000000
	P0 1184.82	24155	990.320958
分辨	座壑 0.4133	39	0.415281
转扬	矩阵		
-0.4	13795	-0.001268	3354.473048
0.0	02120	-0.413045	495.867919
0.0	00000	0.000000	1.000000
序号	差异值		
和	22.530107		
1	1.701015		
2	2.468691		
3	0.638851		
4	4.253800		
5	7.391629		
2	1.491053		
6			
6 7	1.417238		
6 7 8	1.417238 0.833814		

图 10 标定数据

### 2.2.1.2. 旋转中心标定



#### 图 7 九点标定手臂路线图

第一步:设定好工业手臂的拍照位置(上图中的0号点,该位置一般和九点标定的起始位置 相同,即都为标准的拍照位置),以及角度步长,起始位置和步长请尽量为整数。<u>注意旋转</u> <u>中心标定实际上手臂的位置是不动的,只有角度在动,因此所有拍照位置的坐标都是一样,</u> 只是对应的手臂角度不同。其中1号位置代表的角度是标准拍照时手臂的角度。

第二步:按上图中的路径图旋转九个角度 9 个位置,跑到任意一个位置时,需要保存当前位 置的相机拍照图像。*注意:手臂在跑 9 个点的时候,手臂的 X,Y 不能有变化。* 第三步:设计特征点查找程序,该步骤根据产品不同而不同。

第四步:打开标定界面,如下图所示,如果添加按钮为灰色,请先选择像素源。

点标题	定 旋转中心标定点集		
	图像X	图像Y	
1			

图 8 旋转中心标定界面

手动	像素源 [601	04][03]Center	~
比列值(mm/pixel)	一机器人自动机	示定	
1.00000	X步进值	0	-
	¥步进值	5	
0.00000	角度步进值	0	-
	选择机器人		~
		自动 标定	

#### 图 9 设置像素源

第五步:载入第二步中保存好第一个位置的图像,调整检测工具的位置和参数,使其能找到 我们需要的特征点,找到特征点后,在九点标定界面点击添加按钮,像素点坐标会自动加到 表格中;

第六步:重复第五步,把9幅标定图像全部输入进来。

1. 标定	2. 检测 3. 脚本编程				
序号	名称	通用	3 圆点标定 9点标定		
1	w1	载)	入 保存 添加 Del		标定
		9点	标定 旋转中心标定点算	Ę	
			图像X	图像Y	^
		1	2689.029566	960.093015	
		2	2050. 590386	1139.260355	1
		3	1669.016309	1700.073242	
新建	删除 查看	4	1706.370084	2292. 319178	~

点击保存按钮可以把标定数据保存到 TXT 文件,点击载入按钮可以载入事先保存好的标 定数据。<u>注意:载入和保存会把九点标定的数据和旋转中心标定的数据同时载入和保存。</u> 第七步:点击标定按钮,完成标定。

第八步:查看标定数据

	迥	赋			
旋转中	中心	2239.34	2479	10	7.042328
	PO	1184.82	4155	99	0.320958
分辨	库率	0.41333	9	0.4	15281
转扬	短阵				
-0.4	137	95	-0.001268		3325.525391
0.0	0212	0	-0.413045		901.380754
0.0	0000	0	0.000000		1.000000
<b>齐号</b>	差	≩值 520107			
10	22.	530107			
1	1.701015				
2	2.4	68691			
2	2.4 0.6	68691 38851			
2 3 4	2.4 0.6 4.2	68691 38851 53800			
2 3 4 5	2.4 0.6 4.2 7.3	68691 38851 53800 91629			
2 3 4 5	2.4 0.6 4.2 7.3 1.4	68691 38851 53800 91629 91053			
2 3 4 5 7	2.4 0.6 4.2 7.3 1.4 1.4	68691 38851 53800 91629 91053 17238			
2 3 4 5 7 3	2.4 0.6 4.2 7.3 1.4 1.4 0.8	68691 38851 53800 91629 91053 17238 33814			

图 10 标定数据

#### 2.2.2. 眼随手动

#### **该安装方式既需要做九点标定**,也需要做旋转中心标定。 需要事先准备的物件:产品

九点标定步骤如下:

①放置产品到标准拍照位置,根据<u>章节 3.2.1.1 九点标定</u>中第一步的要求,预先规划并记录好九个拍照点的位置。

②按<u>章节 3.2.1.1 九点标定</u>中第二步的走点方式,手臂按顺序走到每个拍照位置,触发相机拍照并保存图像到硬盘。<u>注意:在此过程前需先进行相机参数设置,以使获取到清晰且合</u> 适的图片。

③按<u>章节 3.2.1.1 九点标定</u>的标定过程完成整个九点标定过程。

#### 旋转中心标定的步骤如下:

①放置产品到标准拍照位置,根据<u>章节 3.2.1.2 旋转中心标定</u>中第一步的要求,预先规划 并记录好九个拍照点的位置。

②按<u>章节 3.2.1.2 旋转中心标定</u>中第二步的走点方式,手臂按顺序走到每个拍照位置,触 发相机拍照并保存图像到硬盘。<u>注意:在此过程前需先进行相机参数设置,以使获取到清晰</u> 且合适的图片。

③按章节 3.2.1.2 旋转中心标定的第三步以后的标定过程完成整个旋转中心标定。

### 2.2.3. 手眼分离--眼(相机)朝下

#### 该安装方式只需要做九点标定,无需做旋转中心标定。

需要事先准备的物件:<u>标定针</u>和<u>标定板</u>。

九点标定步骤如下:

①把标定板放到相机视野下,**注意:标定板离镜头前端的距离应该跟产品到镜头前端的** 距离一样;

②用相机拍摄一张照片(照片至少包括标定板的3\*3个标定点);

③在照片上选9个点作为标定点, 需是3\*3的矩阵。

④用示教器手动把标定针按<u>章节 3.2.1.1 九点标定</u>中第二步的走点方式按顺序移动到九个 特征点的位置,并分别记录对应的手臂坐标。

⑤按<u>章节 3.2.1.1 九点标定</u>中第三步以后的标定过程完成的方式输入到表格中,然后点击标定完成整个标定过程。

#### 2.2.4. 手眼分离--眼(相机)朝上

该安装方式既需要做九点标定,也需要做旋转中心标定。

需要事先准备的物件: 产品

#### 九点标定步骤如下:

①手臂夹住或吸住产品,根据<u>章节3.2.1.1 九点标定</u>中第一步的要求,预先规划并记录好 九个拍照点的位置

②按<u>章节 3.2.1.1 九点标定</u>中第二步的走点方式,手臂带着产品按顺序走到每个拍照位置,触发相机拍照并保存好图像到硬盘。<u>注意:在此过程前需先进行相机参数设置,以使获取</u> <u>到清晰且合适的图片</u>。

③按章节 3.2.1.1 九点标定的第三步以后的标定过程完成整个九点标定。

#### 旋转中心标定的步骤如下:

①手臂夹住或吸住产品。根据<u>章节 3.2.1.2 旋转中心标定</u>中第一步的要求,预先规划并记 录好九个拍照点的位置。

②按<u>章节 3.2.1.2 旋转中心标定</u>中第二步的走点方式,手臂带着产品按顺序走到每个拍照 位置,触发相机拍照并保存图像到硬盘。<u>注意:在此过程前需先进行相机参数设置,以使获</u> <u>取到清晰且合适的图片</u>。 ③按章节 3.2.1.2 旋转中心标定的第三步以后的标定过程完成整个旋转中心标定。

### 第三章 检测模块详解

### 3.1. 坐标系统

坐标采用选择特征点跟随的方法,可以在图像上任何一个点生成坐标,除了特殊函数外,其他均支持坐标系统输入。



序号 名称

参数说明

		选择当前坐标系统函数之前的任意一个函数的点输出作为新建
		坐标系的基准点。
1	输入点	例子:[53276]Blob analysis: [3]Center
		解释:选取 Uid=53276 的斑点分析函数的第三个输出(中心点
		输出)作为当前坐标系的跟随点。
		选一个输入角度作为新建坐标系的角度
2	输入角度	例子:[53276Blob analysis: [9 ]Phi
		解释:选取 53276 斑点分析输出的角度值
		限定输出坐标系统的类型
		Default:默认,如果选择了输入角度,则为CT_XYA,否则为
		CT_XY ;
2	业标玄统米刑	CT_X:坐标系只在X方向跟随;
	王你示讥天王	CT_Y:坐标系只在Y方向跟随;
		CT_XY:坐标系在X、Y方向同时跟随,但是没有角度信息;
		CT_XYA:坐标系在 X、Y、A 方向同时跟随;
4	多坐标系统	预留参数,目前不支持

表 1 坐标系统函数输入参数

序号	名称	参数说明
1	坐标系统	新建的坐标系统 [1]1625.971358,2351.773308,0.000000(坐标系统所在图像中的 XY 原点位置 , 默认第一个值 X 第二个值 Y )

表 2 坐标系统函数输出参数

### 3.2. 模版查找



在使用模板查找的功能前,需先添加模板训练,点击上方工具栏模板。

步骤:1点击添加-2输入模板名称-3点击Browse选择图片-4鼠标左键涂抹需要识别的特征-5点击训练-6点击保存。

模板查找模块可实现在检测框范围查到多个模板,并以数组的形式输出查找到的模板的 相关信息,同时也可以输出多个坐标系,极大地简化了多模板和多坐标系的视觉应用。

序号	名称	参数说明
1	输入坐标系	选择坐标系

2	图像通道	如果输入为彩色图像,可选择颜色通道 取值范围: Default:默认通道,为CT_R。 CT_R:红色通道 CT_G:绿色通道 CT_B:蓝色通道
3	模板组专有 ID	选择训练的模板编号,具体请查看章节2.1 模板
4	查找工具框	调整检测框的位置、大小和方向。可通过鼠标在软件界面操作 ,也可以直接输入参数。(检测框的颜色为红色)
5	滤波方式	在二值化前对图像进行滤波,是抑制和防止干扰的一项重要措施 None:无 Mean:平均滤波法 Median:中位值滤波法 Gauss:高斯滤波法 Emphasize:高频强调滤波法
6	滤波模板大小	滤波模板大小
7	起始角度	查找模版轮廓的起始角度。一般为负数,但也可为正数。
8	查找模版轮廓的终止角度。一般为正数,但也可为负数,终           终止角度         角度须大于起始角度。比如起始角度设置-45 和终止角度设置           45,表示以模板角度为基准角度,在-45~45 的范围内进行:目标	
9	最小匹配分数	查找目标与模板的相似程度,取值范围(0,1],值越大,检测速度 越快,但是查找到目标的可能性越低,默认值为0.5。建议取值 在0.5以上。

10	贪心算法	取值范围(0,1],默认值0.9	
11	极性	设置查找模版轮廓时的极性。 Normal:无极性查找 Ignore global polarity:忽略全局明暗变化 Ignore local polarity:忽略局部明暗变化	
12	亚像素类型	选择亚像素的类型	
13	最多查找数量	最多查找数量 待匹配的目标轮廓的最大个数	
14	┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ ┃ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		
15	输出1的偏移量即对输出模块中序号为1的中心值进行补偿再输出		

#### 表 3 模板查找函数的输入参数

序号	名称	参数说明
1	数量	匹配到的模板的个数
2	中心值	匹配到模板的中心值
3	角度	匹配到模板的角度
4	分数值	匹配到模板的分数值
5	轮廓	匹配到模板的轮廓

#### 表 4 模板查找函数的输出参数

### 3.3. 斑点分析



斑点,在数学上也成为连通域。所谓斑点分析,一般是先通过连通域分析的相关方法把 图像上不相连的区域分割开来(如下图所示,每种颜色代表一个连通域),然后再通过连通 域的相关属性进行分类和刷选,并从中挑选出需要的连通域的过程。

连通域分析的方法主要有4连通和8连通两种方式,如下图。所谓4连通,即把与目标 点在上下左右等4个方向具有相连关系点才认为是一个连通域;而8连通把与目标点在8个 方向具有相连关系的点都认为属于同一个连通域。



4连



8连

从上图中很容易判断出来,4连通的算法要比8连通的算法的运算量小,因此4连通算法的运算速度也更快。显然8连通也有自己的优势,因为它更符合我们的习惯,比如以下场合,一般我们都会认为是一个连通域,但是如果采用4连通的算法,则会判定为3个连通域。



至于到底是用4连通还是8连通的办法,完全可以根据需要自由选择,在能满足现场要求的情况下,尽量使用4连通的方法,以提高运算速度。

序号	名称	参数说明
1	输入坐标系	选择坐标系
2	图像通道	如果输入为彩色图像,可选择颜色通道 取值范围: Default:默认通道,为CT_R。 CT_R:红色通道 CT_G:绿色通道 CT_B:蓝色通道
3	复合检测框	调整检测框的位置、大小和方向。可通过鼠标在软件界 面操作,也可以直接输入参数。 当前检测框的颜色为红色 可修改检测框的类型,如圆形、矩形、圆弧形。
4	滤波方式	图像滤波,即在尽量保留图像细节特征的条件下对目

		标图像的噪声进行抑制。 可选择的方式有均值滤波、中值滤波、高斯滤波、高频
		强调滤波。
5	滤波模板大小	滤波模板大小有 3、5、7、9、11
6	二值化参数	用此参数的上下限对图像进行二值化操作,把图像灰度 在上下限范围内的设置为前景,在二值化参数之外的设 置为背景。通过二值化操作后,把具有256个灰度级别 的灰度图像转化为只有2个灰度级别(0和255)的二 值化图像。
7	生态学处理	对二值化后的斑点进行形态学处理,包 括腐蚀、膨胀、闭运算(先膨胀后腐蚀)、开运算(先腐蚀后膨胀)。
8	算子大小	用来设置算子大小值
9	是否填充	选择是否对斑点进行填充操作
10	连通域分析	选择连通域算法4或8
11	通过面积筛选	选出在面积阈值内的斑点
12	通过中心点 × 选择	选出在中心点 x 阈值内的斑点
13	通过中心点 y 选择	选出在中心点 y 阈值内的斑点
14	通过正矩形宽选择斑点	选出在正矩形宽阈值内的斑点
15	通过正矩形高选择斑点	选出在正矩形高阈值内的斑点
16	通过斜矩形长边选择斑点	选出在斜矩形长边阈值内的斑点
17	通过斜矩形短边选择斑点	选出在斜矩形短边阈值内的斑点
18	通过斜矩形长宽比选择斑点	选出在斜矩形长宽比阈值内的斑点
19	通过角度选择斑点	选出在角度阈值内的斑点

20	通过序号选择斑点	选择序号在序号阈值内的斑点
21	排序	Ascending by Y Y 值升序排序 Descending by Y Y 值降序排序 Ascendingby X X 值升序排序 Descending by X X 值降序排序 Ascendingby Area 面积升序排序 Descending by Area 面积降序排序
22	填充2	选择是否对筛选后的斑点进行填充
23	联合	选择是否将筛选后的斑点联合成一个斑点
24	反转	选择是否进行反转凸显出产品轮廓
25	边缘孔洞	选择是否查找轮廓边缘孔洞
26	通过面积筛选	选出筛选后在面积阈值内的孔洞
27	通过斜矩形长宽比选择斑点	选出筛选后在斜矩形长宽比阈值内的孔洞
28	输出3的偏移量	位置偏移量即对斑点的中心点进行补偿再输出
29	输出 0 的公差	即对输出模块中序号为0的斑点数量是否在阈值内判定 产品 OK/NG
30	输出1的公差	即对输出模块中序号为1的斑点之间的距离是否在阈值 内判定产品 OK/NG
31	忽略中心点	忽略中心线默认为 true 则输出不显示中心线设为 false则输出显示中心线

#### 表 5 斑点分析函数的输入参数

序号	名称	参数说明
1	斑点数量	检测到的斑点的个数

序号	名称	参数说明	
2	斑点之间的距离	序号相邻的斑点的中心点间距	
3	面积	检测到的每个斑点的面积	
4	中心	检测到的每个斑点的中心坐标	
5	宽度	检测到的每个斑点的正矩形的宽度	
6	高度	检测到的每个斑点的正矩形的高度	
7	外接矩形半长	检测到的每个斑点的外接矩形半长	
8	外接矩形半宽	检测到的每个斑点的外接矩形半宽	
9	长宽比	检测到的每个斑点的斜矩形的长宽比	
10	角度	检测到的每个斑点的斜矩形的角度	
11	中心线	检测到的每个斑点的中心线	
表 6 斑点分析函数的输出参数			

### 3.4. 一维码检测器



序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道。

3	斜矩形检测框	调整检测框的位置、大小和方向。可通过鼠标在软件界面操作,也可以直接输入参数。(检测框的颜色为红色)
4	码的类型	选择待识别的一维码类型。
5	Background mode	默认的条码为白底黑码,如果要检测黑底白码的情况。下 拉选择
6	Expected quantity	预期识别码的数量

表 7 一维码检测器输入参数

序号	名称	参数说明		
1	条码	检测到的条码结果		
2	最小外接矩形框	最小外接矩形框的大小		
表 8 一维码检测器的输出参数				
### 3.5. 二维码检测器



序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道。
3	斜矩形检测框	调整检测框的位置、大小和方向。可通过鼠标在软件界面操作 ,也可以直接输入参数。(检测框的颜色为红色)

	Background	默认的条码为白底黑码,如果要检测黑底白码的情况。	下拉选
4	mode	择	

5	Expected quantity	预期识别码的数量
---	-------------------	----------

#### 表 7 二维码检测器输入参数

序号	名称	参数说明
1	条码	检测到的条码结果
2	最小外接矩形框	最小外接矩形框的大小

#### 表 8 二维码检测器输入参数

### 3.6. Datamatrix (矩阵式二维码)



序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道。
3	斜矩形检测框	调整检测框的位置、大小和方向。可通过鼠标在软件界面操作 ,也可以直接输入参数。(检测框的颜色为红色)
4	Background mode	默认的条码为白底黑码,如果要检测黑底白码的情况。下拉选择

5	Scan gap	
6	Edge threshold	
7	Time out (ms)	
8	Expected quantity	预期识别码的数量

序号	名称	参数说明
1	条码	检测到的条码结果
2	最小外接矩形框	最小外接矩形框的大小

### 3.7. 图像比对



采用图像相减的方法判断当前图像与模板图像之间的差异。对于彩色图像,可选择对某 个通道单独处理,默认为红色通道。因本工具的特殊性,暂不支持输入坐标系。

序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道 。
3	正矩形检测框	调整检测框的位置、大小。可通过鼠标在软件界面操作,也可以直接输入参数。(检测框的颜色为红色)
4	滤波方式	图像滤波,即在尽量保留图像细节特征的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	对比方法	选择不同的比较方式 Light:只计算当前图像比模板图像更亮的部分; Dart:只计算当前图像比模板图像更暗的部分; Both:以上两个情况同时计算
7	腐蚀系数	设置检测图像的腐蚀度
8	膨胀系数	设置检测图像的膨胀度

序号	名称	参数说明
9	最小灰度差	图像相减后灰度值小于二值化上限的定义为暗区域 , 大于二 值 化下限的定义为亮区域
10	通过面积筛选	选出筛选后在面积阈值内的图像
11	输出0的公差	即对输出模块中序号为0的数量是否在阈值内判定产 OK/NG

12	输出1的公差	即对输出模块中序号为1的最大面积是否在阈值内判定产品
		OK/NG

#### 表 9 图像比对的输入参数

序号	名称	参数说明
1	数量	检测到的图像数量
2	最大面积	检测到的最大面积
3	面积	检测到的图像面积
表 10 图像比对的输出参数		

### 3.8. 单极点测量



极点测量,如上图所示,可用来测量一个极高点到一条线段的垂直距离,其中线段和点的测量参数可以分别设置,该功能还可用于测量产品边缘是否有毛刺等,只要测量出极点的存在且与线段的距离超出一定的范围,即认为产品边缘存在毛刺。

序号	名称	参数说明
1	输入坐标系	选择输入坐标系

序号	名称	参数说明
2	图像通道	选择所需颜色通道的图像进行操作 取值范围: Default:默认通道,为CT_R。 CT_R:红色通道 CT_G:绿色通道 CT_B:蓝色通道
3	斜矩形检测框	调整检测框的位置、大小和方向。可通过鼠标在软件界面操 作,也可以直接输入参数。(检测框的颜色为红色)
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找边的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	查找边的梯度	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出来
8	查找点的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
9	查找点的梯度	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出来

序号	名称	参数说明
10	输出0的公差	即对输出模块中序号为0的极值点到基线的距离是否在阈值 内判定产品 OK/NG

#### 表 11 单极点测量的输入参数

序号	名称	参数说明
1	极值点到基线的距离	极点到直线的距离
2	基线角度	检测到的基线角度
3	极值点	检测到的极点坐标
4	基准线	检测到的基准线坐标和长度、角度

表 12 单极点测量的输出参数

### 3.9. 极距测量



极距测量可用来测量两个极点之间的距离。在很多情况下我们要测量的不是两条边的距离,而是两个点之间的距离,这个时候用距离测量工具无法实现,因为距离测量工具是通过 拟合线然后求取线与线之间的距离。这个时候我们就可以用到极距测量工具。

序号	名称	参数说明
1	输入坐标系	选择输入坐标系
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三 个通道的图像各有差异,应选择图像质量最好的颜色通道。
3	组合矩形框	调整检测框的位置、大小、方向。可通过鼠标在软件界面操作,

		也可以直接输入参数。(检测框的颜色为红色)
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的 噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波 、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找点的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	查找点的梯度1	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出来
8	查找点的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
9	查找点的梯度 2	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出来
10	输出0的公差	即对输出模块中序号为0的点1和点2的距离是否在阈值内判定 产品OK/NG

#### 表 13 极距测量输入参数

序号	名称	参数说明
1	点1和点2的距离	2个极点间的距离
2	极值点1	检测到的极点1的坐标
3	极值点 2	检测到的极点 2 的坐标

#### 表 14 极距测量输出参数

### 3.10. 弧形测量



圆弧测量,可在圆弧范围内测量半径方向的边缘,如测量时钟的指针位置等 圆弧测量可同时测量两条边缘。

序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位

2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道。
3	圆环检测框	键入工具框的位姿
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	查找边的梯度	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出来
8	输出0的公差	即对输出模块中序号为0的角度是否在阈值内判定产品 OK/NG

#### 表 15 弧形测量的输入参数

序号	名称	参数说明
0	角度	两条直线的夹角
1	线1	直线1坐标、长度、角度
2	线2	直线2坐标、长度、角度

表 16 弧形测量的输入参数

### 3.11. 阵列测量



阵列测量广泛应用用于接插件测量当中,集合基准线,阵列测量可以很容易地输出诸如 共面度的指标。

序号	名称	参数说明
----	----	------

1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三 个通道的图像各有差异,应选择图像质量最好的颜色通道
3	矩形工具框	可设置键入工具框的位置大小、方向、间距、和数量与垂直排 列的方式
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的 噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤 波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	查找边的梯度	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出 来
8	输出的边缘模式	选择在工具框内查找的直线类型
9	水平基准线	选择相应的模式
10	起始点	根据前面建立好的函数来确定起始点
11	结束点	根据前面建立好的函数来确定结束点
12	垂直基准线	选择相应的模式
13	起始点	根据前面建立好的函数来确定起始点
14	结束点	根据前面建立好的函数来确定结束点

15	輸出0的公差	即对输出模块中序号为0的间距是否在阈值内判定产品 OK/NG
16	输出1的公差	输出1的公差
17	输出2的公差	即对输出模块中序号为 2 的到垂直基准线的距离是否在阈值内 判定产品 OK/NG

#### 表 16 阵列测量的输入参数

序号	名称	参数说明
0	间距	相邻针脚间距测量值集合
1	到水平基准线的距离	针尖上中心点到水平基线的距离集合
2	到垂直基准线的距离	针尖上中心点到垂直基线的距离集合
3	线段	每个 ROI 查找到的线的集合
4	水平基准线	输出水平基准线的坐标、长度、角度
5	垂直基准线	输出垂直基准线的坐标、长度、角度

表 17 阵列测量的输出参数

### 3.12. 圆心测量



通过圆环测量工具测量出圆周上的点并拟合出一个圆,圆心测量工具可用来对圆形物体进行 高精度半径测量,精度可达 0.02 像素。

序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位

让机器视觉进入每一家工厂

序号	名称	参数说明
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道 。
3	圆环检测框 r	调整检测框的位置、大小、方向、角度步进等。可通过鼠标 在软件界面操作,也可以直接输入参数。(检测框的颜色为 红色)
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找边的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	查找边的梯度	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会输出来
8	输出0的公差	即对输出模块中序号为0的直径是否在阈值内判定产品 OK/NG

### 表 18 圆心测量的输入参数

序号	名称	参数说明
0	直径	目标圆的直径
1	中心	目标圆的圆心坐标



序号	名称	参数说明
2	园	输出圆

表 19 圆心测量的输出参数

### 3.13. 圆心度测量



圆心度测量主要应用于测量两个圆的圆心距离。

序号 名称 参数说明
------------

1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道。
3	圆环检测框1	调整检测框的位置、大小、方向、角度步进等。可通过鼠 标在软件界面操作,也可以直接输入参数。(检测框的颜 色为红色)
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找模式1	设置检测框 1 的边缘模式。 Light to dark : 查找由亮到暗的边缘 Dark to light : 查找由暗到亮的边缘
7	边缘梯度1	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会 输出出来
8	圆环检测框 2	调整检测框的位置、大小、方向、角度步进等。可通过鼠 标在软件界面操作,也可以直接输入参数。(检测框的颜 色为红色)
9	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波
10	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11

	设置检测框1的边缘模式。
本代描書の	• Light to dark:查找由亮到暗的边缘
	• Dark to light:查找由暗到亮的边缘
计条样 庄 0	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会
过场休天 2	输出出来
<b>检山 0 的八</b> 关	即对输出模块中序号为0的圆心距离是否在阈值内判定产
1 11 0 11公左	品 OK/NG
	查找模式 2 边缘梯度 2 输出 0 的公差

#### 表 20 圆心度测量的输入参数

序号	名称	参数说明
0	圆心距离	两个输出圆的圆心距离
1	圆心连线	极两个输出圆的圆心连线
2	圆心1	输出圆1的圆心坐标
3	直径1	输出圆1的直径
4	圆 1	输出圆 1
5	圆心 2	输出圆 2 的圆心坐标
6	直径2	输出圆 2 的直径
7	圆 2	输出圆 2

表 21 圆心度测量的输出参数

### 3.14. 距离测量



距离测量,可用来测量两条直线的距离和角度等参数。

序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用哪个定位

2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道。
3	矩形工具框	调整检测框的位置、大小、方向。可通过鼠标在软件界面操作,也可以直接输入参数。(检测框的颜色为红色)
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标 图像的噪声进行抑制,可选择的方式有均值滤波、中值滤 波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找边的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	边缘梯度1	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会 输出来
8	边缘类型	设置边缘的类型 •Follow : 跟随 • Vertical : 垂直 • Horizital : 水平
9	边缘选择	可选择第一个、最后一个或者最后的边缘
10	矩形工具框	调整检测框的位置、大小、方向。可通过鼠标在软件界面操作,也可以直接输入参数。(检测框的颜色为红色)

11	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标图像的噪声进行抑制,可选择的方式有均值滤波、中值滤波、高斯滤波、高频强调滤波
12	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
13	查找边的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
14	边缘梯度2	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会 输出来
15	边缘类型	设置边缘的类型 •Follow : 跟随 • Vertical : 垂直 • Horizital : 水平
16	边缘选择	可选择第一个、最后一个或者最后的边缘
17	输出1的偏移量	即对输出模块中序号为1线1和线2的交点进行偏移量补充
18	输出1的公差	即对输出模块中序号为1线1和线2的距离是否在阈值内 判定产品 OK/NG
19	输出2的公差	即对输出模块中序号为1线1和线2的夹角是否在阈值内 判定产品 OK/NG

#### 表 22 距离测量的输入参数

序号	名称	参数说明
0	线1和线2的交点	输出2条线的交点坐标

序号	名称	参数说明
1	线1和线2的距离	输出2条线的距离
2	线1和线2的夹角	输出2条线的夹角
3	线1的角度	输出线1的角度
4	线2的角度	输出线2的角度
5	线1	输出线1的坐标、长度、角度
6	线2	输出线2的坐标、长度、角度

表 23 距离测量的输出参数

### 3.15. 位置测量



位置测量模块具有设置模板线段的功能,正常运行时可实时判断实际线段与模板线段的角度 和距离关系。因此可用于对产品进行位置校准。

序号	名称	参数说明
1	坐标系统	根据前面建立好的坐标系统来确定用

2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数 为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的 颜色通道。
3	矩形工具框	调整检测框的位置、大小、方向。可通过鼠标在软件界面操作,也可以直接输入参数。(检测框的颜色为红色)
4	滤波方式	图像滤波,即在尽量保留图像细节 <u>特征</u> 的条件下对目标 图像的噪声进行抑制,可选择的方式有均值滤波、中值滤 波、高斯滤波、高频强调滤波,可选择性使用可不用
5	滤波模板大小	滤波模板大小有 3*3、5*5、7*7、9*9、11*11
6	查找边的模式	设置检测框 1 的边缘模式。 • Light to dark : 查找由亮到暗的边缘 • Dark to light : 查找由暗到亮的边缘
7	查找边的梯度	待测边缘的梯度最小值,只有满足该梯度要求的边缘才会 输出来
8	边缘类型	设置边缘的类型 •Follow : 跟随 • Vertical : 垂直 • Horizital : 水平
9	边缘选择	可选择第一个、最后一个或者最后的边缘
10	输出0的公差	即对输出模块中序号为 0 关键点 X 坐标是否在阈值内判 定产品 OK/NG

11	输出1的公差	即对输出模块中序亏为 U 大键点 Y 坐标是否任阈值内判定 产品 OK/NG
12	输出2的公差	即对输出模块中序号为「线的头角是否在阈值内判定产品 OK/NG

表 24 位置测量的输入参数

序号	名称	参数说明
0	关键点X坐标	输出关键点 X 坐标
1	关键点 Y 坐标	输出关键点 Y 坐标
2	线的角度	查找到的线的角度
3	线段	输出线段

表 25 位置测量的输出参数

### 3.16. 斑点选择工具



斑点选择工具同理与斑点分析,是二次选择/筛选工具。当一个斑点分析检测函数找到多个相似的斑点,通过斑点选择工具可以选择我们自己想要的斑点。

序号	名称	参数说明
1	输入轮廓	选择一个斑点分析,进行二次筛选/选择。例如:选 择 55171
2	通过面积筛选	选出在面积阈值内的斑点

3	通过 X 点选中心择 选出在中心点 x 阈值内的斑点		
4	通过 Y 中心点选择	选出在中心点 y 阈值内的斑点	
5	通过正矩形宽选择斑点	选出在正矩形宽阈值内的斑点	
6	通过正短形高选择斑点	选出在正矩形高阈值内的斑点	
7	通过斜矩形长边选择斑点	选出在斜矩形长边阈值内的斑点	
8	通过斜矩形短边选择斑点	选出在斜矩形短边阈值内的斑点	
9	通过斜矩形长宽比选择斑点	选出在斜矩形长宽比阈值内的斑点	
10	通过角度选择斑点	选出在斜矩形角度阈值内的斑点	
11	排序	Ascending by Y Y 值升序排序 Descending by Y Y 值降序排序 Ascendingby X X 值升序排序 Descending by X X 值降序排序 Ascendingby Area 面积升序排序 Descending by Area 面积降序排序	
12	通过序号选择斑点	选择序号在序号阈值内的斑点	
13	输出3的偏移量	设置输出 3 中心 XY 的范围 如果超出设置范围 判定为NG	
14	输出0的公差	设置输出 0 斑点数量的范围 如果超出设置范围 判定 为 NG	
15	输出1的公差	设置输出1斑点距离的范围 如果超出设置范围 判定 为 NG	
16	忽略显示中心线	忽略显示中心线	

#### 表 26 斑点选择工具的输入参数

序号	名称	参数说明
1	斑点数量	显示在图像界面检测到的斑点的个数
2	斑点之间的距离	序号相邻的斑点的中心点间距

序号	名称	参数说明
3	面积	检测到的每个斑点的面积
4	中心	检测到的每个斑点的中心坐标
5	宽度	检测到的每个斑点的正矩形的宽度
6	高度	检测到的每个斑点的正矩形的高度
7	外接矩形半长	检测到的每个斑点的外接矩形的长边长
8	外接矩形半宽	检测到的每个斑点的外接矩形的短边长
9	长宽比	检测到的每个斑点的长宽比
10	角度	检测到的每个斑点的角度

表 27 斑点选择工具的输出参数

#### 让机器视觉进入每一家工厂

## **VISIONGO**

### 3.17. 灰度



通过判断图像某个区域内指定像素值区间的最大值、最小值、均值、像素点个数,以及像素 点个数占整个区域内的像素点的百分比是否在一个范围内来判断产品是否合格的方法。

序号	名称	参数说明
1	输入轮廓	根据前面建立好的函数有轮廓的来选择
2	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三个通道的图像各有差异,应选择图像质量最好的颜色通道
3	感兴趣范围	用此参数的上下限对图像进行二值化操作,把图像灰度在上下

		限范 围内的设置为前景,在二值化参数之外的设置为背景。
		通过二值化操作后,把具有256个z度级别的灰度图像转化
		为只有2个灰度级别(0和255)的二值化图像只需设置二
		值化上限值和下限值
4	输出0的公差	即对输出模块中序号为0的均值是否在阈值内判定产品 OK/NG
5	输出1的公差	即对输出模块中序号为1的最大值是否在阈值内判定产品 OK/NG
6	输出2的公差	即对输出模块中序号为2的最小值是否在阈值内判定产品 OK/NG
7	输出3的公差	即对输出模块中序号为 3 的像素数量是否在阈值内判定产品 OK/NG
8	输出4的公差	即对输出模块中序号为4的像素占比是否在阈值内判定产品 OK/NG

#### 表 28 灰度的输入参数

序号	名称	参数说明
0	均值	输出均值
1	最大值	输出最大值
2	最小值	输出最大值
3	像素数量	输出像素数量
4	像素占比	输出像素占比

表 29 灰度的输出参数

### 3.18. 颜色分类器



通过颜色的判断来区分待检测产品属于哪一类产品。本模块支持九个颜色模板,即可实现在 九个颜色模板范围内找出与当前产品颜色最接近的模板,模块可直接输出最接近的颜色模板 编号,以及颜色值。例如把斑点分析找到的轮廓加入到输入轮廓中,就会得出输出结果。用 户需手动设置颜色模板,并把值抄入到对应的颜色模板中,如上图最接近的颜色模板为

序号	名称	参数说明
1	输入坐标系	选择输入坐标系

2	复合检测框	调整检测框的位置、大小和方向。可通过鼠标在软件界面操 作,也可以直接输入参数。(检测框的颜色为红色) 可修改检测框的类型,如圆形、矩形、圆弧形。
3	颜色模版1	设置颜色模板的 R、G、B 参数
4	颜色模版 2	设置颜色模板的 R、G、B 参数
5	颜色模版 3	设置颜色模板的 R、G、B 参数
6	颜色模版 4	设置颜色模板的 R、G、B 参数
7	颜色模版 5	设置颜色模板的 R、G、B 参数
8	颜色模版 6	设置颜色模板的 R、G、B 参数
9	颜色模版7	设置颜色模板的 R、G、B 参数
10	颜色模版 8	设置颜色模板的 R、G、B 参数
11	颜色模版 9	设置颜色模板的 R、G、B 参数
12	输出4的公差	即对输出模块中序号为4的色差值是否在阈值内判定产品 OK/NG
13	颜色模版 11	设置颜色模板的 R、G、B 参数

#### 表 30 颜色分类器的输入参数

序号	名称	参数说明
1	Red value	输出红色值
2	Green value	输出绿色值
3	Blue value	输出蓝色值
4	颜色模板	输出与手动输入的颜色模板最接近的序号
5	色差	输出色差

#### 表 31 颜色分类器的输出参数

### 3.19. 对比度



序号	名称	参数说明
2	输入轮廓	根据前面建立好的函数有轮廓的来选择
3	图像通道	选择图像颜色通道。黑白图像无需设置该参数,默认参数为 default,即为选用 R 通道图像;当图像为彩色时,因 R\G\B 三 个通道的图像各有差异,应选择图像质量最好的颜色通道。




4 输出 0 的公差 即对输出模块中序号为 0 对比度是否在阈值内判定产品 OK/NG

表 31 对比对的输入参数

序号	名称	参数说明
0	对比度	输出对比度的值

#### 表 32 对比对的输入参数

### 3.20. 点到点



可求出任意 2 点之间距离、角度。

序号	名称	参数说明	
1	输入点1	根据前面建立好的函数有输出点的来根据前面建立好的函	
2	输入点 2	数有输出点的来选择	
3	输出0的公差	即对输出模块中序号为0距离值是否在阈值内判定产品 OK/NG	
4	输出1的公差	即对输出模块中序号为1角度是否在阈值内判定产品 OK/NG	

#### 表 33 点到点的输入参数

序号	名称	参数说明			
0	距离值	输出 2 点间的距离值			
1	角度	输出2点间的角度			

表 34 点到点的输出参数



### 3.21. 点到线



#### 可求出任意点到线之间的距离值、线段。

序号	名称	参数说明
1	输入点1	根据前面建立好的函数有输出点的来选择
2	线段	根据前面建立好的函数有输出线段的来选择
3	输出0的公差	即对输出模块中序号为0距离值是否在阈值内判定产品OK/NG

表 35 点到线的输入参数



序号	名称	参数说明			
1	距离值	两个中心点的距离值			
2	线段	左到右分别是 X 位置,Y 位置,长度,角度			

表 36 点到线的输出参数

### 3.22. 新建点



通过选择一个函数中心点,进行偏移得到一个新的点

序号 名称

参数说明

1	输入点	选择一个函数,作为输入点
2	偏移量	X:X 的偏移量 Y:Y 的偏移量

表 37 新建点的输入参数

序号	名称	参数说明		
1	1     点集     当前新建点的 XY 坐标			
表 38 新建点的输出参数				

#### 3.23. 新建线



通过两个函数的中心点,连接成一条新的线。

#### 表 11 新建线的输入参数

序号	名称	参数说明
1	输入点1	起始点选择
2	输入点 2	终点选择
3	输出0的公差	设定两个中心点的距离值范围 超出则判定为 NG

#### 表 39 新建线的输出参数

序号	名称	参数说明	
1	距离值	两个中心点的距离值	
2	线段	左到右分别是 X 位置,Y 位置,长度,角度	

## 第四章 脚本(Script)编程

### 4.1. LUA 介绍

RobotVision 采用 Lua 作为脚本语言, Lua 是一个小巧的脚本语言。是巴西里约热内卢 天主教大学(Pontifical Catholic University of Rio de Janeiro)里的一个研究小组,由 Roberto Ierusalimschy、Waldemar Celes 和 Luiz Henrique de Figueiredo 所组成并于 1993 年开发。 其设计目的是为了嵌入应用程序中,从而为应用程序提供灵活的扩展和定制功能。 Lua 由标准 C 编写而成,代码简洁优美,几乎在所有操作系统和平台上都可以编译,运行。 Lua 脚本可以很容易的被 C/C++ 代码调用,也可以反过来调用 C/C++的函数,这使得 Lua 在应用程序中可以被广泛应用。在目前所有脚本引擎中,Lua 的速度是最快的。

#### 4.2. 基本的 LUA 函数

#### 4.2.1. LUA 的数学函数库(math)

序号	函数名称	函数功能	示例	结果
1	abs	取绝对值	math.abs(-15)	15
2	acos	反余弦函数	math.acos(0.5)	1.04719755
3	asin	反正弦函数	math.asin(0.5)	0.52359877
4	atan2	x/y的反正切值	math.atan2(90.0, 45.0)	1.10714871
5	atan	反正切函数	math.atan(0.5)	0.463647609
6	ceil	不小于 x 的最大整 数	math.ceil(5.8)	6
7	cosh	双曲线余弦函数	math.cosh(0.5)	1.127625965

8	COS	余弦函数	math.cos(0.5)	0.87758256
9	deg	弧度转角度	math.deg(math.pi)	180
10	exp	计算以 e 为底 x 次 方值	math.exp(2)	7.3890561
11	floor	不大于 x 的最大整 数	math.floor(5.6)	5
12	fmod (mod)	取模运算	math.fmod(14, 5) math.mod(14, 5)	4
13	frexp	把双精度数 val 分 解为数字部分(尾 数)和以2为底的指 数 n,即 val=x*2n	math.frexp(10.0)	0.625 4
14	ldexp	计算 value*2 的 n 次 方 , 即 val=x*(2^n)	math.ldexp(10.0, 3)	80
15	log10	计算以 10 为基数 的对数	math.log10(100)	2
16	log	计算一个数字的自 然对数	math.log(2.71)	0.9969
17	max	取得参数中最大值	math.max(2.71,100, -98, 23)	100
18	min	取得参数中最小值	math.max(2.71,100, -98, 23)	-98
19	modf	把数分为整数和小 数	math.modf(15.98)	15 0.98
20	pow	得到 x 的 y 次方	math.pow(2, 5)	32
22	rad	角度转弧度	math.rad(180)	3.1415927
22	random	获取随机数	math.random(1,100) math.random(100)	1~99
23	randomseed	设置随机数种子	math.randomseed(os.time())	用 math.random

				函数之前必须使用
				此函数设置随机数
				种子
24	sinh	双曲线正弦函数	math.sinh(0.5)	0.5210953
25	sin	正弦函数	math.sin(math.rad(30))	0.5
26	sqrt	开平方函数	math.sqrt(16)	4
27	tanh	双曲线正切函数	math.tanh(0.5)	0.46211715
28	tan	正切函数	math.tan(0.5)	0.5463024

#### 4.2.2. LUA 的字符串函数库(string)

Lua 解释器对字符串的支持很有限。一个程序可以创建字符串并连接字符串,但不能截取子串,检查字符串的大小,检测字符串的内容。在Lua 中操纵字符串的功能基本来自于 string 库。

cc 字符串库中的一些函数是非常简单的:

1. string.len(s) : 返回字符串 s 的长度。

2. string.rep(s, n): 返回重复 n 次字符串 s 的串。

你使用 string.rep("a", 2^20)可以创建一个 1Mbytes 的字符串(比如,为了测试需要)

3. string.lower(s)函数和 string.upper(s)函数用来将字符在大小写之间转换。

如果你想不关心大小写对一个数组进行排序的话,你可以这样: table.sort(a, function (a, b) return string.lower(a) < string.lower(b) end)

string.upper 和 string.lower 都依赖于本地环境变量。所以,如果你在 European Latin-1 环境下,表达式:string.upper("a??o")-->A??O"

4. string.sub(s,i,j) : 截取字符串 s 的从第 i 个字符到第 j 个字符之间的串。

Lua 中,字符串的第一个字符索引从1开始。你也可以使用负索引,负索引从字符串的结尾向前计数:-1指向最后一个字符,-2指向倒数第二个,以此类推。所以, string.sub(s, 1, j)返回字符串 s 的长度为 j 的前缀 ; string.sub(s, j, -1) 返回从第 j 个字符开始的后缀。如果

不提供第 3 个参数, 默认为-1, 因此我们将最后一个调用写为 string.sub(s, j); string.sub(s, 2, -2)返回去除第一个和最后一个字符后的子串。

s = "[in brackets]"

print(string.sub(s, 2, -2)) --> in brackets

记住:Lua 中的字符串是恒定不变的。string.sub 函数以及 Lua 中其他的字符串操作函数都不会改变字符串的值,而是返回一个新的字符串。一个常见的错

误是认为 string.sub(s, 2, -2)这个函数会改变字符串 s 的值。如果你想修改一个字符串变量的值, 你必须将变量赋给一个新的字符串: s = string.sub(s, 2, -2)。

5. string.char 函数和 string.byte 函数用来将字符在字符和数字之间转换。

string.char 获取 0 个或多个整数,将每一个数字转换成字符,然后返回一个所有这些字符连接起来的字符串。string.byte(s, i)将字符串 s 的第 i 个字符的转换 成整数;第二个参数 是可选的,缺省情况下 i=1。下面的例子中,我们假定字符用 ASCII 码表示:

print(string.char(97)) --> a

i = 99; print(string.char(i, i+1, i+2)) --> cde

print(string.byte("abc")) --> 97

print(string.byte("abc", 2)) --> 98

print(string.byte("abc", -1)) --> 99

上面最后一行,我们使用负数索引访问字符串的最后一个字符。

6. string.format()函数来生成具有特定格式的字符串。

函数的第一个参数是格式(formatstring), 之后是对应格式中每个代号的各种数据. 由于格式字符串的存在, 使得产生的长字符串可读性大大提高了. 这个函数的格式很像 C 语言中的 printf().函数 string.format 在用来对字符串进行格式化的时候, 特别是字符串输出, 是功能强大的工具。这个函数有两个参数, 你完全可以照 C 语言的 printf 来使用这个函数。第一个参数为格式化串:由指示符和控制格式的字符组成。指示符后的控制格式的字符可以为: +进制'd'; +六进制 'x'; 八进制'o'; 浮点数'f'; 字符串's'。在指示符'%'和控制格式字符之间还可以有其他的选项:用来控制更详细的格式, 比如一个浮点数的小数的位数:

格式字符串可能包含以下的转义码:

%с	接受一个数字,并将其转化为 ASCII 码表中对应的字符
%d, %i	接受一个数字并将其转化为有符号的整数格式
%0	接受一个数字并将其转化为八进制数格式
%u	接受一个数字并将其转化为无符号整数格式
%х	接受一个数字并将其转化为十六进制数格式,使用小写字母
%X	接受一个数字并将其转化为十六进制数格式,使用大写字母
%е	接受一个数字并将其转化为科学记数法格式,使用小写字母 e

%E 接受一个数字并将其转化为科学记数法格式,使用大写字母 E

%f 接受一个数字并将其转化为浮点数格式

%g(%G) 接受一个数字并将其转化为%e(%E,对应%G)及%f 中较短的一种格式

%q 接受一个字符串并将其转化为可安全被 Lua 编译器读入的格式

%s 接受一个字符串并按照给定的参数格式化该字符串

为进一步细化格式,可以在%号后添加参数.参数将以如下的顺序读入:

(1) 符号: 一个+号表示其后的数字转义符将让正数显示正号. 默认情况下只有负数显示符号。

(2) 占位符: 一个 0, 在后面指定了字串宽度时占位用. 不填时的默认占位符是空格。

(3) 对齐标识: 在指定了字串宽度时, 默认为右对齐, 增加-号可以改为左对齐。

(4) 宽度数值。

(5) 小数位数/字串裁切: 在宽度数值后增加的小数部分 n, 若后接 f(浮点数转义符, 如%6.3f)则设定该浮点数的小数只保留 n 位, 若后接 s(字符串转义符, 如%5.3s)则设定该字符串 只显示前 n 位。

在这些参数的后面则是上述所列的转义码类型(c, d, i, f, ...).

print(string.format("pi = %.4f", Pl)) -> pi = 3.1416

d = 5; m = 11; y = 1990

print(string.format("%02d/%02d/%04d", d, m, y)) ->05/11/1990

tag, title = "h1", "a title"

print(string.format("<%s>%s</%s>", tag, title, tag)) -> <h1>a title</h1>

第一个例子,%.4f 代表小数点后面有4位小数的浮点数。第二个例子%02d 代表以固定的两位显示十进制数,不足的前面补0。而%2d 前面没有指定0,不足两位时会以空白补足。对于格式串部分指示符得详细描述清参考 lua 手册,或者参考C 手册,因为 Lua 调用标准C 的 printf 函数来实现最终的功能。

以下走一些例于.	
string.format("%c", 83)	输出:S
string.format("%+d", 17.0)	输出:+17
string.format("%05d", 17)	输出:00017
string.format("%o", 17)	输出:21
string.format("%u", 3.14)	输出:3
string.format("%x", 13)	输出:d
string.format("%X", 13)	输出:D
string.format("%e", 1000)	输出:1.000000e+03
string.format("%E", 1000)	输出:1.000000E+03

string.format("%6.3f", 13)
string.format("%q", "One\nTwo")
string.format("%s", "monkey")
string.format("%10s", "monkey")
string.format("%5.3s", "monkey")

输出:13.000 输出:"One\Two" 输出:monkey 输出:monkey

### 4.2.3. LUA 的条件函数(If)

```
1. if...else 语句
```

if(boolean\_expression)

then

```
--[ statement(s) will execute if the boolean expression is true --]
```

else

--[ statement(s) will execute if the boolean expression is false --]

end

如果布尔表达式的值为 true, 那么 if 代码块将被执行, 否则 else 代码块将被执行。

Lua 程序设计语言假定条件表达式为布尔 true 为 true,以及它是否是布尔假,则假定为

```
false 值。但应当注意的是,在Lua 中零值被视为 true。
```

```
--[local variable definition --]
a = 100;
--[check the boolean condition --]
if(a<20)
then
--[if condition is true then print the following --]
print("a is less than 20")
else
--[if condition is false then print the following --]
print("a is not less than 20")
end
print("value of a is :%d", a)
c 当建立和运行上面的代码,它会产生以下结果。
a is not less than 20
value of a is : 100
```

2. if...else if...else 语句

if 语句后面可以跟一个可选的 else if...else 语句,这是非常有用的组合语句,以测试各种条件单个 if...else if 语句。

当使用 if , else if , else 语句有几点要记住使用:

- if 可以有零或一个 else , 但必须在 elseif 之后。
- if 之后可以有零到很多 else if 在 else 之前。
- 一旦一个 else if 成功,其它的 elseif 和 else 将不会被测试。

if(boolean\_expression 1)

then

--[ Executes when the boolean expression 1 is true --]

else if( boolean\_expression 2)

--[ Executes when the boolean expression 2 is true --]

CCCelse if( boolean\_expression 3)

--[Executes when the boolean expression 3 is true --]

else

--[ executes when the none of the above condition is true --]

### 4.2.4. LUA 的循环函数(for/while)

1. for 循环

for 循环是一个循环控制结构,可以有效地编写需要执行的特定次数的循环。 Lua for 语句语法如下:

for init, max/min, increment

do

statement(s)

end

下面是控制在一个循环的流程:

• 初始化步骤首先被执行,并且仅一次。这个步骤可让您声明和初始化任何循环控制变量。

• 接着是 max/min,这是最大或最小值,直到该循环继续执行。它在内部创建了一个条件检查的初值和最大值/最小值之间进行比较。

• for 循环体执行后,控制流跳回至递增/递减声明。这个语句可以更新任何循环控制变量。

• 条件现在 新计算评估。如果这为真则循环执行,并 复这个过程(循环体,然后增加 一步,然后再条件)。如果条件为假,则循环终止。

#### 2. while 循环语句

在 Lua 编程语言中的 while 循环语句,只要给定的条件为真时将 复执行的目标语句。 Lua while 语句语法如下:

while(condition)

do

statement(s)

end

在这里,声明(S)可以是单一语句或语句块。该条件可以是任何表达式,并且真正是任意 非零值。循环迭代当条件是 true。

当条件为假,则程序控制进到紧接在循环之后的一行。

C3. break 语句

当循环中遇到 break 语句,循环立即终止,程序控制继续下一个循环语句后面。

如果您正在使用嵌套循环(即一个循环里面另一个循环), break 语句将停止最内层循环 的执行并开始执行的下一行代码的程序后段。

Lua break 语句语法如下:

Break

#### 4.3. 自定义的关键词及函数

#### 4.3.1. 可直接使用的嵌入式函数

关键字	支持的函数	说明及示例
	Sloop	说明:延时函数,单位毫秒
	Sieeb	vxBase.Sleep(1000)
WRaco	Savelmage	说明:图像保存函数,输入文件的名称,无需路径。默认
VXDase		保存在程序目录下的 images 文件夹内
		vxBase.SaveImage("t.jpg")
	Log	说明:记录 log

		vxBase.Log("message")
G	atCounter	说明:获取检测总数和合格数
	letCounter	totalFrames,passFrames= vxBase.GetCounter()
		说明:获取指定相机的共享内存,注意此函数每次调用后
Ge	etSharedMemory	会把共享内存清零
		SharedMemory = vxBase.GetSharedMemory(1)
50	SatSharadMamory	说明:设置共享内存
		vxBase.SetSharedMemory("123.45")

### 4.3.2. 项目及函数操作

关键字	支持的函数	说明及示例
wTack	GetResult	说明:获取当前检测任务的结果,返回值为 true 或者 false
		res=vxTask.GetResult()
VAIUSK	SetResult	说明:强制设置当前检测任务的结果
		vxTask.SetResult(true)
	GetResult	说明:获取指定函数的检测结果,返回值为 true 或者 false
		res = vxFxn.GetResult(Uid)
	SetResult	说明:强制设置指定函数的检测结果
WEVD		vxFxn.SetResult(Uid,false)
	GetInput	说明:获取指定函数指定输入端口的数据
		x,y= vxFxn.GetInput(Uid,1)
	GetOutput	说明:获取指定函数指定输出端口的数据
		x,y= vxFxn.GetOutput(Uid,1)

### 4.3.3. 变量操作

关键词	函数说明及示例		
	1. 新建一个点 , 点的坐标为 ( 123.456,147.852 )		
	rpt = vxRealPoint.New(123.456,147.852)		
	2. 获取点的元素		
	x,y=vxRealPoint.Get(rpt)		
vxRealPoint	3. 计算点到点的距离		
	d = vxRealPoint.P2PDist(x1,y1,x2,y2)		
	4. 计算点 (X1,Y1) 绕点(centerx,centery)顺时针旋转指定角度后的新的点坐		
	标		
	x,y=vxRealPoint.Rotate(x1,y1,centerx,centery,angle)		
	1. 通过位置、长度和角度创建一条线段		
	NewLine=vxLine.New(x,y,l,a)		
	2. 通过2个点创建一条线段		
vyline	NewLine=vxLine.NewBy2Point(x1,y1,x2,y2)		
VALINE	3. 复制一条线段		
	NewLine=vxLine.NewByLine(Line)		
	4. 获取线段的元素		
	x,y,l,a=vxLine.Get(NewLine)		
	1. 通过圆心和半径创建一个圆		
	NewCircle=vxCircle.New(x,y,r)		
WCirclo	2. 获取圆的元素		
VXCIICIE	x,y,r=vxCircle.Get(NewCircle)		
	3. 通过一个点数组拟合一个圆		
	x,y,r=vxCircle.Fit(ptArray)		
	1. 通过点和角度创建一个向量		
	NewXYA=vxXYA.New(x,y,a)		

	2. 获取向量的元素
	x,y,a=vxXYA.Get(NewXYA)
	1. 创建一个大小为 10 的点数组,数组元素初始化为(0,0)
	NewArray=vxArray.New("array1", 10, vxRealPoint.New(0,0))
	2. 获取点数组的第一个元素
101 Arrow	X,y=vxArray.Get(NewArray, 0)
VXAITAy	3. 给点数据的第一个元素赋值
	vxArray.Set(NewArray, 0, vxRealPoint(12.4,77.3))
	4. 获取数组大小
	szArray = vxArray.Size(NewArray)

### 4.3.4. DI/O 操作(vxDio)

关键字	支持的函数	说明及示例
	GetDIStatus	获取 I/O 卡指定输入的状态值
	Gerbistatus	status=vxDio.GetDIStatus(uniqueId, 0)
vxDio	GetDOStatus	获取 I/O 卡指定输出的状态值
		status=vxDio.GetDOStatus(uniqueId,0)
		设置指定输出的状态值
		status=vxDio.SetDOStatus(uniqueId,0,true)

### 4.3.5. 串口操作(vxUart)

关键字	支持的函数	说明及示例
		串口读取函数
vxUart	Read	vxUart.Read(18704,10,"1","3")
		示例

		1. 自定义协议,在自定义协议中,后面两个参数属于保留参数,不
		会用到,因此直接设置设置为0就好了。以下整条语句的意思是从
		Uid=18704的串口设备上读取10个字符;
		vxUart.Read(18704,10,"0","0")
		2. Modbus 协议,在 modbus 协议中,第二输入参数属于保留参数
		,不会用到。以下整条语句的意思是通过 Uid=18704 的串口,读取
		Slave=1的 Modbus 设备的第三个保持寄存器的值。
		vxUart.Read(18704,10,"0","0")
		串口写入函数
	Write	vxUart.Write(18704,"10",0,"1","3")
		1. 自定义协议,在自定义协议中,后面两个参数属于保留参数,不
		会用到,因此直接设置设置为0就好了。以下整条语句的意思是网
		Uid=18704 的串口设备上写入字符串"abc",字符串的长度为 3;
		vxUart.Write(18704, "abc", 3, "0", "0")
		2. Modbus 协议,在 modbus 协议中,第三输入参数属于保留参数
		,不会用到。以下整条语句的意思是通过 Uid=18704 的串口,在
		Slave=1的 Modbus 设备的第3个保持寄存器上写入10。
		vxUart.Write(18704, "10", 0, "1", "3")

### 4.3.6. 网口操作(vxTcp)

关键字	支持的函数	说明及示例
		网口读取函数
		vxTcp.Read(18704,10,"1","3")
		示例
vxTcp	Read	3. 自定义协议,在自定义协议中,后面两个参数属于保留参
		数,不会用到,因此直接设置设置为0就好了。以下整条语
		句的意思是从 Uid=18704 的网口设备上读取 10 个字符 ;
		vxTcp.Read(18704,10,"0","0")

	4. Modbus 协议,在 modbus 协议中,第二输入参数属于保
	留参数 , 不会用到。以下整条语句的意思是通过 Uid=18704
	的网口,读取 Slave=1的 Modbus 设备的第三个保持寄存器
	的值。
	vxTcp.Read(18704,10,"0","0")
	网口写入函数
	vxTcp.Write(18704,"10",0,"1","3")
	3. 自定义协议,在自定义协议中,后面两个参数属于保留参
	数,不会用到,因此直接设置设置为0就好了。以下整条语
	句的意思是网 Uid=18704 的网口设备上写入字符串"abc",
	字符串的长度为3;
	vxTcp.Write(18704, "abc", 3, "0", "0")
	如果当前连接为服务器,也可以通过指定客户端的 IP 和端口
	号进行点对点发送,比如如下第一条语句表示把"abc"发给送
Write	IP 为 192.168.3.71, 端口号不限的客户端, 第二条语句表示把
	"abc"发送到 IP 为 192.168.3.71 , 端口号为 7181 的客户端
	vxTcp.Write(18704, "abc", 3, "192.168.3.71", "0")
	vxTcp.Write(18704, "abc", 3, "192.168.3.71", "7181")
	4. Modbus 协议,在 modbus 协议中,第三输入参数属于保
	留参数 , 不会用到。以下整条语句的意思是通过 Uid=18704
	的网口,在 Slave=1的 Modbus 设备的第3个保持寄存器上
	写入10。
	vxTcp.Write(18704, "10", 0, "1", "3")

### 4.3.7. 脚本输入输出操作(vxScript)

脚本输入输出 首创建输入计数 COUNT 和输出 如下图 每执行一次脚本就会加一操作 然后获取结果输出到检测框上

关键字	支持的函数	说明及示例			
vxScript	GetInput()	input0=vxScript.GetInput(0)			
	SetInput()	vxScript.SetInput(0,123.456)			
	SetResult()	vxScript.SetResult(true)			
	SetOutput()	vxScript.SetOutput(0,123.456)			

#### 4.4. 常见脚本示例

#### 4.4.1. 相机之间通讯

在 RobotVision 内,相机之间通讯一般采用共享内存的方式,在 RobotVision 的脚本中 定义了两个函数 vxBase.SetSharedMemory()和 vxBase.GetSharedMemory()来进行共享内存 的存取。

示例:4个相机,每个相机检测矩形产品的一个角点,相机2,相机3,相机4计算出角 点信息后把数据存到各自的共享内存中,相机1读取其它相机的共享内存数据,汇总后计算 产品的长度,宽度和对角线长度,以下脚本为相机的部分脚本:

--定义函数的 Uid, 方便后续使用

const fxnId=59006

--定义特征点的基准位置

const basex, const basey = -10.585576,11.936240

--获取函数的检测结果

res=vxFxn.GetResult(const\_fxnId)

if (res) then

--获取结果信息

centerArray = vxFxn.GetOutput(const fxnId,0)

x,y = vxArray.Get(centerArray,1)

--计算偏移量

cam1\_offsetx=const\_basex-x

cam1\_offsety=const\_basey-y

--格式化偏移量数据到字符串

const\_str = string.format("%07.3f,%07.3f",cam1\_offsetx,cam1\_offsety)

--设置格式化的数据到共享内存

vxBase.SetSharedMemory(const\_str)

End

--延时 100ms , 等待相机检测完成

vxBase.Sleep(100)

--获取其它相机的共享内存

sharedMemory2 = vxBase.GetSharedMemory(2)

sharedMemory3 = vxBase.GetSharedMemory(3)

sharedMemory4 = vxBase.GetSharedMemory(4)

--判断共享内存是否获取成功,如果成功的话,共享内存数据的长度应该大于0

if (#sharedMemory1>0 and #sharedMemory2>0 and #sharedMemory3>0 and #sharedMemory4 >0) then

--把相机 2 共享数据中的前 1~7 位转化为浮点数 ,并设置到 cam2\_offsetx

cam2\_offsetx = tonumber(string.sub(sharedMemory2, 1, 7))

--把相机 2 共享数据中的前 9~15 位转化为浮点数 ,并设置到 cam2\_offsety

cam2\_offsety = tonumber(string.sub(sharedMemory2, 9, 15))

cam3\_offsetx = tonumber(string.sub(sharedMemory3, 1, 7))

cam3\_offsety = tonumber(string.sub(sharedMemory3, 9, 15))

cam4\_offsetx = tonumber(string.sub(sharedMemory4, 1, 7))

cam4\_offsety = tonumber(string.sub(sharedMemory4, 9, 15))

••

end

# 4.4.2. 通过 TCP/IP Custom 协议发送数据

第一步:获取 Uid=65040 的斑点分析工具的第三个输出(斑点的中心)

centerArray = vxFxn.GetOutput(65040,3)

第二步:取出数组的第一个元素,即把第一个斑点的中心坐标分别赋值给 X 和 Y

x,y = vxArray.Get(centerArray ,1)

第三步:格式化数据到字符串

const\_str = string.format("%f;%f", x,y)

第四步:通过TCP/IP的方式把格式化后的字符串发送出去。

其中 18704 表示的是 Uid=18704 的 TCP/IP 设备, #const\_str 表示获取 const\_str 的长度,

127.0.0.1 和 9978 表示接收当前信息的 TCP 设备的 IP 地址和端口号。

vxTcp.Write(18704, const\_str, #const\_str, "127.0.0.1", "9978")

### 第五章 创建首个检测任务

以瓶盖检测为例辅助说明:项目要求检测瓶盖有无,瓶盖是否歪斜。通讯:数字通讯。 思路:打光思路:打白产品和背景形成鲜明对比,软件思路:使用斑点分析工具检测。



表 12 瓶盖打光图

#### 5.1. 新建任务

软件主界面点击[新建]按钮,打开如下对话框,在 A 区有一个编辑框可供输入当前任务的说明,用户可输入任何字符。一般都会输入主要的检测内容,以方便记忆该检测任务的功能。

如果连上了可识别的相机,在 B 区会显示已经连上的相机,用户可选择一个相机以跟当前任务绑定。

✔ 新建				×
	<b>日</b> 系列 机械手 产品计	] 智能相 抓取定位及引 数 / 高精度测	<b>り</b> 导 / 读码 量	00
CAM1 文件名	▼ 123  称 CAM1_TSK006_1	A X 23. vxproj		刷新相机
序号	相机型号	相机序列号	相机IP地址	
1	MV-CA050-10GC	00D43148962 B 🔀	192. 168. 0. 5	
				取消

注意:RobotVision 采用 GenlCam 协议采集图像,要让 RobotVision 识别到相机相机,该相机必须支持 GenlCam 协议。

### 5.1.1. 项目名称的命名格式

项目文件名称的命名格式 CAM?\_TSK???\_\*.vxproj (其中?代表一个数字,\*代表任意个字符 或者数字)。

序号	符号	说明
1	?	1~4 的数字,代表的是相机的序号,最多支持同时打开 4 个相机
2	าาา	001~999 的数字,代表任务编号。自动切换任务时需要通过任务编号找
2	:::	到对应的项目文件并切换。
3	*	自定义的备注信息可以是数字或者字符,支持中文

#### 5.2. 获取最好的图像

在启动任何项目之前,请注意首先一定要获取最好的图像,那么什么是最好的图像。我 们人眼认为的最好图像对电脑来说可能并不是最好。对于人眼来说,我们认为颜色很丰富的 图片是令人赏心悦目的图片,但是对于机器视觉来说,我们大部分都采用黑白相机,黑白图 像对计算机来说是更容易识别的图像。因此我们这里所说的最好图像,可以认为是对计算机 来说最容易识别的图像。

那么什么图像对计算机来说是最好识别的图像,一般我认为对比高的图像,计算机更容易识别,对计算机来说是更好的图像。

#### 5.2.1. 设置曝光



如上图所示,找到曝光时间选项,设置为5000(具体值根据实际项目确定,目的是让产品亮

度适中),所得图像如右图所示。

### 5.2.2. 设置图像 AOI

+	Analog Controls		
±	Image Format Controls AOI Controls		
	Width	2448	=
	Height	2048	
	X Offset	8	
	Y Offset	4	
	Center X	False	
	Center Y	False	
	Binning Horizontal	1	
	Binning Vertical	1	
	Image Quality Control		
	Acquisition Controls		-
		Le das	

找到 AOI 的对应参数, AOI 的对应参数一般有 4 个, 分别是 Width、Height、X Offset和 Y Offset。

下面以图的形式解释 4 个参数的意义:

假设蓝色大矩形表示图像采集芯片,橙色小矩形表示我们需要设置的 AOI,其中 X Offset 和 Y Offset 表示 AOI 矩形框到蓝色矩形框左上角的距离。Width 和 Height 分别表示 橙色矩形框的宽度和高度。



宽度和高度使用默认值时的图像 宽度和高度都设置为 500 时的图像



设置图像 AOI 在某些芯片上可以可以提高采集速度,然后 AOI 最主要的目的是选择我们 最关心的图像区域,可以有助于图像处理速度和效率的提升。

#### 5.2.3. 设置触发方式

触发方式一般由 3 个参数组成,分别是 Trigger Mode, Trigger Source(触发源)和 Trigger Activation(触发极性)。Trigger Mode 一般只有两个选择,分别是 On 和 Off,表 示是否需要开始触发模式。Trigger Source 一般可选择 Software(软触发)和 Line1(硬触发), Line2(硬触发)等。Trigger Activation 一般可选择 Rising Edge(上升沿)和 Falling Edge(下降沿)。

在使用硬触发时,请特别注意相机支持的硬触发的信号电压,如果接入的触发信号电压 高于相机的核定触发电压,可能会烧坏相机。

下图设置的是上升沿硬触发,触发源为Line1,即从Line1上输入触发信号。



#### 5.2.4. 设置包大小和巨型帧

包大小一般是配合巨型帧一起设置的,巨型帧的设置方法请参考<u>章节 7.1 TCP/IP 包大小与巨型帧</u>。包大小对应的相机参数是 Transport Layer 下的 Packet Size。Packet Size 的大小根据 巨型帧的大小来设置,默认的一般是 1500,常见的有 9000 和 4000。巨型帧和包大小的设置 最明显的作用是传输效率提高了,对于大分辨率相机不会丢帧了。

Transport Layer		
Payload Size	5013504	
Packet Size	8164	
Auto Functions		

#### 5.2.5. 相机参数保存

RobotVision 不自己保持相机配置,所有的相机配置都是直接调用 GENICAM 协议直接把参数保存在相机内部。用户可以直接用相机自带的 SDK 进行参数设置和保存,也可以在 RobotVision 内直接操作。

相机参数保存如下图所示。

Configuration Set Selector 选择 UserSet1, 然后直接双击 User Set Save 即可完成保存 操作。

需要特别注意的是 Default Startup Set (默认启动配置)也必须选择 UserSet1,这样才能确保相机在上电后会载入我们需要的相机相机配置文件。

一般相机都有3套用户设置文件,我们只用任意一套就可以了。

Ξ.	Configuration Sets	
	Configuration Set Selector	VserSet1
	Vser Set Load	Command
	Vser Set Save	Command
	Default Startup Set	VserSet1
÷	Transport Layer	
	Auto Functions	

图 11 相机参数的读取和保存

### 5.3. 标定

详细标定过程请参照标定第三章 标定

#### 5.4. 添加函数模块

第一步

在[检测模块]页面找到函数列表 框,并点击按钮[新建]



第二步

在弹出的对话中选择要添加的函 数模块。例:新建斑点分析。

吳吠	」「馬性」		
序号	名称	描述	
1	坐标系统		
2	模板查找		
3	斑点分析		
4	一维码检测器		
5	二维码检测器		
6	图像比对		
7	单极点测量		
8	极距测量		
9	弧形测量		
10	阵列测量		
11	圆心测量		
12	同心度测量		
13	距离测量		
14	位置测量		
15	斑点选择工具		
•	m		•



#### 第三步

图像界面调整检测框的位置,使 检测框包含产品可以出现的所有 位置

在输入界面调整输入参数。找到 想要的特征。参数的具体说明和 设置请参考<u>章节:第四章 检测</u> 模块详解。

輸入	Show and Save 数字输入输出 串口	
[1	1][SRC] 坐标系统	
[2	2][E] 图像通道	Default
Ð [3	3][COP] 复合检测框	Rect; 308; 247; 279; 230; 0
[4	4][E] 滤波方式	None
[5	5][E] 滤波模板大小	3
3 [6	5][TOR] 二值化参数	255; 255
	下限值	255
	上限值	255
[7	7][E] 生态学处理	None
[8	3][E] 算子大小	3
[9	9][B] 填充1	False
[]	[0][E] 连通域分析	8
E] [1	11][TOR] 通过面积筛选	5000; 30000
	下限值	5000
	上限值	30000
E] [1	12][TOR] 通过中心点X选择	0; 0
E [1	13][TOR] 通过中心点Y选择	0; 0
± [1	14][TOR] 通过正矩形宽选择斑点	0; 0
E] [1	[5][TOR] 通过正矩形高选择斑点	0; 0
± [1	16][TOR] 通过斜矩形长边选择斑点	0; 0
E] [	[7][TOR] 通过斜矩形短边选择斑点	0: 0
<b>∄</b> [1	18][TOR] 通过斜矩形长宽比选择斑点	0; 0

#### 第四步

查看函数的输出

如果产品合格会在图像区域显示 OK的字样,并在图像四周会有 绿色边框;否则显示NG字样和 红色边框

ł	机设置 输出	
	[0][I] 斑点数量	1
	[1][ARR] 斑点之间的距离	
-	[2][ARR] 面积	
	[1]	11210
Ð	[3][ARR] 中心	
	[1]	297.441570, 182.119715
-	[4][ARR] 宽度	
	[1]	120
Ē	[5][ARR] 高度	
	[1]	120
3	[6][ARR] 外接矩形半长	
	[1]	59.7034
Ð	[7][ARR] 外接矩形半宽	
	[1]	58.5854
-	[8][ARR] 长宽比	
	[1]	1.01908
Ē	[9][ARR] 角度	
	[1]	63.6495
	[10][ARR] 中心线	

#### 5.4.1. 调整函数的执行顺序

在 RobotVision 内,所有的检测函数是按顺序执行的,排名在前的更早执行,排名靠后的执行更后一点,脚本也认为是一个函数,脚本的执行是在其它所有检测函数执行后之后的。要调整函数的执行顺序,可通过以下2个按钮。



图 12 调整函数的执行顺序

当函数 B 需要用到函数 A 中的输出时,函数 A 必须排名在函数 B 之前,否则函数 B 无法引用函数 A 的输出,如下图所示,当坐标系统函数排名在斑点分析函数之前时,坐标系统的输入点参数获取不到斑点分析的输出,如果把执行顺序调整一下,把斑点分析调到坐标系统函数之前,坐标系统的输入点参数就可以获取到斑点分析的中心输出。

序号	名称	软触发指令	标定	描.	输入	Show and Save	数字输入输出	串口
1	[29177]坐标系统				[2	][SRC] 输入点		
2	[68303]斑点分析				[3	][SRC] 输入角度	1	
					[4	][E] 坐标系类型	CT_XYA	
					[5	][B] 多坐标系统	False	

序号	名称	软触发指令	标定	描	输入 Show and Save 数字输入输出 串口 网口
1	[68303]斑点分析				[2][SRC] 输入点
2	[29177]坐标系统				[3][SRC] 输入角度
					[4][E] 坐标系类型 [68303]Blob analysis:[3]Center
					[5][B] 多坐标系统 False

#### 5.4.2. 设置软触发命令和标定

任务中的每个函数都支持独立的软件触发命令和标定。多软触发命令的作用可以减少任 务切换的频率。该功能在单手臂多吸嘴的情况下尤其有用。

RobotVision 的这个功能可实现在不切换任务情况,不适用逻辑函数的情况下,有效控制函数执行与否。比如当前任务中有2个检测检测函数,分别是函数A和函数B,假设函数A的软触发命令为SGSTA,函数B的软触发命令为SGSTB,在正常执行时,当RobotVision接收到SGSTA时,函数A会执行,函数B不会执行,当RobotVision接收到STB时,函数A不执行,函数B会执行。

另外每个软触发命令都可以携带自己的曝光时间,如下图所示,当接收到 STA 的触发命 令时,会适用 5000ms 的曝光时间进行拍照,当接收到 STB 的触发命令时,会使用 3000ms 的曝光时间进行拍照。

注意:触发指令只能以SGST开头



### 5.5. 设置可让用户输入的数据

	网口	User Inputs 保存 数字输入输出 串口	酸輸入				
^	Default	[2][E] 图像通道					
	128; 255	🖃 [6][TOR] 二值化参数	□ [6][TOR] 二值化参数				
	128	下限值					
	255	上限值     255       [9][B]填充1     Fals       [11771]斑点分析     Fals					
	False						
		[73508]Script					
	899.6	[0][R] 左边长度[AB]_下限值					
	900	[1][R] 左边长度[AB]_上限值	编辑				

#### 图 13 用户输入参数设置界面

如上图,点击左侧编辑按钮,弹出以下对话框。勾选需要在用户界面显示的参数,然后 点击确定按钮即可。

=	[60104]斑点分析		^
	[2]图像通道	~	
	[4]滤波方式		
	[5]滤波模板大小		
	[6]二值化参数	~	
	[7]生态学处理		
	[8]算子大小		
	[9]填充1	~	
	[10]连通域分析		
	[11]通过面积筛选		
	[12]通过中心点X泷		
	[13]通过中心点W道		
	[14]通过正矩形宽:		
	[15]通过正矩形高;		
	[16]通过斜矩形长;		
	[17]通过斜矩形短:		
	[18]通过斜矩形长		
	[19]通过角度选择		~

注:试用版本无法进入用户界面

### 5.6. 显示和保存

通用	皮是	去有TD			Ĩ.
	1	30520			-
Ħ	-	30320			
w zsaâλ saâ					
出					
-					
		9 8635			
山/阿弥	□通过	FTP保存			
_	服务器I	P地址		21	*
	E	<b>山山</b> 山			
存图像	5	нл-а 		8	
		密码		连接	ŧ
	☑保存	合格产品的图像	E:\打光图片\12.19生产	Brows	se
	☑俾友	不全格立品的图像		Brow	te l
				51 011	
	请输入	专有名称			
	保存期	浪 7		1	删除
	1000000000000	4361		Accessed I have	
					确定
					NUVE

函数输入	数字输入输	出串口	网口	保存			
插入列	删除列				☑ 保存数据	☑ 保存图像	30520[本地]] ~
		分数值					
		@83918:03,	/0				

图 14 数据和图像保存界面

### 5.6.1. 数据显示和保存

RobotVision 可对任意数据进行保存,保存的数据可以是当前任务中生成的数据,也可以是用户指定的其它常量数据。

如果要保存数据,首先必须勾选保存数据的多选框。数据保存时以上图中表格的格式进行保存,数据保存为 csv 格式,可直接用 excel 打开。

以上图为示例,大致设置步骤如下:

1. 根据需要插入列,此例中插入7列;

 逐一选中各个数据格,并设置每个数据格要显示的数据。行名,列名和内容可以单独 设置,内容也直接选择。

#### 5.6.2. 图像保存

如上图所示橙色部分为图像保存区域,如果需要保存图像,请先勾选多选按钮,然后在 下拉框中选择保存协议即可。

#### 5.6.3. 图像显示

上方工具栏上,设置,点击后弹出如下对话框,根据需要进行勾选



#### 5.7. 设置通讯

#### 5.7.1. 添加数字输入输出(DI/O)

选择数字输入输出页面,如果已经添加了 DI/O 设备的话,在下图的下拉框中会显示已 经添加的 DI/O 设备,如果这里没有显示,请参考<u>章节 2.2 DI/O 通讯</u>添加设备。

如下图所示,我们选择一个 DI/O 设备,并选择输出 2 进行 IO 设置。如下图所示的设置,表示的是当第一个函数(D1)和第二个函数(D2)的检测结果都为不合格时,给输出 2 这个端口上给出高电平信号。
输入	Show and Save	数字输入输出	串口	网口			
数字输	入输出 97838[Ca	omrelay,COM1]			~	输出2	~
🖃 D02				D1&&D2			
	[60104]斑点分析			D1			
	[11771]斑点分析			D2			$\sim$
	[73508]Script						

每个检测函数有 2 种状态, D 表示不合格输出, !D 表示合格输出。然后函数结果之间又有 2 种关系,分别是与(&&)和或(||)。

D1 & & D2: 函数 1 和函数 2 都为不合格时,输出高电平;

D1 || D2: 函数1和函数2只要有一个不合格, 即输出高电平;

D1 & & !D2: 函数 1 不合格,并且函数 2 合格时,输出高电平;

### 5.7.2. 添加串口网口输出

因为串口和网口的同学输出设置完全一模一样,我们这里把串口和网口的通讯输出设置 合在一起来讲。

o

选择串口或者网口页面,如果已经添加了串口或者网口设备的话,在下图的下拉框中会显示已经添加的串口或网口设备,如果这里没有显示,请参考<u>章节2.3 串口和网口</u>添加设备

输入	Show and Save	数字输入输出	串口	网口		
串口号	34810[COM1, Cus	tom]				~ 编辑
逐 1	函数名称 [60104]斑点分	輸[3	)出口名称 3]中心	<u>ت</u>		
		✔ 请选择			×	

=	[60104]斑点分析		^	
	[0]斑点数量			
	[1]斑点之间的距离			
	[2]面积			
	[3]中心	2		
	[4]宽度			4
	[5]高度			-
	[6]外接矩形半长			
	[7]外接矩形半宽			
	[8]长宽比			
	[9]角度			
	[11771]斑点分析			
	[0]斑点数里			
	[1]斑点之间的距离			
	[2]面积			
	[3]中心			
	[4]宽度			
	[5]高度		~	
_	F 311151		<u> </u>	

点击编辑按钮,弹出设置对话框。我们只要在设置对话框中选中我们需要从该串口或者 网口设备发送出去的数据就可以了。软件根据选择,在正常检测过程中自动把所有选中的数 据发送出去。发送的格式请参考对应的设备页面。

### 5.8. 切换到用户界面

设置模式	新建	打开	保存	设置	模板	工具箱	开始	
CAM1_TSKD10vxproj 🗸 🗸	×							
L Q Q [] 显示 巴 [	83918]Shape det	ector 🗸					[P]25.3	4, 351.

点击工具栏第一个按钮可在用户界面和设置界面进行相互切换。当从用户界面切换到设置界面时,需要输入密码,默认密码为:RV2020。

### 5.9. 查看日志

RobotVision 会自动保存日志,日志存储路径在程序目录下的 logs 文件夹,日志会每天自动创建。

## 第六章 FAQ

### 6.1. TCP/IP 巨型帧

巨型帧(Jumbo Frame)是帧长大于 1522 字节的以太网帧。这是一种厂商标准的超长帧 格式,专门为千兆以太网而设计。巨型帧的长度各厂商有所不同,从 9000 字节~64000 字 节不等。采用巨型帧能够令千兆以太网性能充分发挥,使数据传输效率提高 50%~100%。

巨型帧需要在相互通讯的 2 个通讯端口端上同时支持,而且与以前的以太网产品不兼容,因此主要会应用于千兆主干的端口之间以及服务器端口接入到网络主干的链路。交换机把 巨型帧格式的数据转发向不兼容巨型帧的端口时应进行帧格式的转换,即把巨型帧帧格式的 数据转换成标准以太网的帧格式,从而保证其正常工作。相反,从不兼容巨型帧的端口向支 持巨型帧的端口转发数据时,交换机可以把多个标准以太网帧合并成超长巨型帧帧,从而提 高传输效率。应该说巨型帧在一些领域里是非常有用的,它是有意设计为加速大文件传输服 务的。以太网标准定义的最大帧长度为 1518 字节,这样一个大的文件就需要被切碎成为若干 块,放到多个以太网帧中。而每个数据块传输的时候都会引入帧头和尾的开销。倘若能够用 一个大的帧完成文件的传输,则会减少很多帧的开销,提高网络的利用率和传输速率。

### 6.1.1. 巨型帧设置(Windows)

第一步:打开网络连接,找到对应的以太网设备,如下图所示,鼠标右键选择属性,打开属 性对话框,在属性对话框中点击配置按钮,打开网卡设置。



图 15 本地网络连接

第二步:在网卡设置对话框中选择高级页面,在属性列表中找到巨型帧或者 Jumbo Frame 字样的属性,把右边的值设置为最大。

聲它的值。 〔②: w Control	值(⊻):	
E(P):	值(V):	
w Control		
w condion	4KB MTU	•
errupt Moderation		_
4 Checksum Offload		
ge Send Offload (IPv4)		
twork Address		
ority & VLAN		
ceive Buffers		
ed & Duplex P Checksum Offload (IPv4)		
insmit Buffers		
P Checksum Offload (IPv4)		
ike on Magic Packet		
ike on pattern match		

### 6.2. 工具箱的使用

### 6.2.1. 镜头选型工具

	步骤1:选择相机芯片光学		
祝大	步骤2:以下A、B、C三组参数	牧任意填写2组:	
	A. 焦距(mm)	0	•
	B. 相机工作距离(mm)	0	▲ ▼
	C. 视野范围		
	目标区域的宽度(mm)	0	<b>•</b>
	目标区域的高度(mm)	0	•
	步骤3: 点击"计算"按钮获取组	課。计算	

选择镜头可以通过工具箱进行计算。打开工具箱如上图,首先在步骤1里面选择相机芯片 光学步骤2可以按照现场合适的工作距离,以及所需要的视野范围输入到B、C后的框内, 最后再把鼠标点在A焦距框,计算按钮才可以点,点击计算得出所需要的结果。

当然在 A、B、C 中有两组的值可以确定就能算出另一组的结果。

#### 目录

RotbotVision 视觉软件	1
版本记录	2
第一章 RobotVision 简介	3
1.1. 运行环境	
1.2. 关于唯一编号(Uid)	
第二章 全局参数设置	
2.1. 模板	6
2.2. DI/O 通讯	
2.3. 串口和网口	
2.4. 数据存储	
第三章 标定	
3.1. 圆点标定	
3.2. 手眼标定	
3.2.1 通用的标定过程	17
9.2.1. 週7日17小足足住	
5.2.2. 眼视于幼 2.2.2 手明八离 明(扫扣)胡玉	
5.2.5.	
5.2.4. 于眽汀芮眽(相机)朝上	
4.1. 坐你杀统	
4.2.	
4.3. 堆只分析	
4.4. 一维码检测器	
4.5	
4.6. 图像比灯	
4./. 甲攸只测重	
4.8. 攸距测重	
4.9. 弧形测量	
4.10. 阵列测量	
4.11. 回心测重	
4.12. 圆心度测量	
4.13. 距离测量	
4.14. 位直测量	
4.15. 圾点选择上具	65
4.16. 灰度	
4.17. 颜色分奀器	
4.18. 对比度	
4.19. 点到点	
4.20. 点到线	
4.21. 新建点	
4.22. 新建线	
第五章 脚本(Script)编程	
5.1. LUA 介绍	
5.2. 基本的 LUA 函数	
5.2.1. LUA 的数学函数库(math)	
5.2.2. LUA 的子符串函数库(string)	
5.2.5. LUA 的条件函数(II)	
5.2.4. LUA 的循环函数(for/while)	
5.3. 目定义的关键词及函数	
5.3.1. 可直接使用的嵌入式函数	
5.3.2. 项目及函数操作	
5.3.3. 变量操作	
5.3.4. DI/O 操作(vxDio)	89

	5.3.5. 串口操作(vxUart)	
	5.3.6. 网口操作(vxTcp)	
	5.3.7. 脚本输入输出操作(vxScript)	
5.4.	常见脚本示例	
	5.4.1. 相机之间通讯	
	5.4.2. 通过 TCP/IP Custom 协议发送数据	94
第六章 仓	J建首个检测任务	
6.1.	新建任务	95
	6.1.1. 项目名称的命名格式	
6.2.	获取最好的图像	
	6.2.1. 设置曝光	
	6.2.2. 设置图像 AOI	
	6.2.3. 设置触发方式	
	6.2.4. 设置包大小和巨型帧	
	6.2.5. 相机参数保存	
6.3.	标定	
6.4.	添加函数模块	101
	6.4.1. 调整函数的执行顺序	
	6.4.2. 设置软触发命令和标定	
6.5.	设置可让用尸输入的数据	
6.6.	显示和保仔	
	6.6.1. 数据显示和保存	
	6.6.2. 图像保存	107
	6.6.3. 图像显示	
6.7.	设置通讯	
	6.7.1. 添加数字输入输出(DI/O)	
	6.7.2. 添加串口网口输出	
6.8.	切换到用户界面	111
6.9.	查看日志	
第七章 F	AQ	
7.1.	TCP/IP 巨型帧	112
	7.1.1. 巨型帧设置(Windows)	
7.2.	工具箱的使用	114
	7.2.1. 镜头选型工具	