

EPSON RC+ 7.0 选项

*Vision Guide 7.0* Ver.7.5

*Software*

Rev.5

SCM231S5525F

翻译版

EPSON RC+ 7.0选项 Vision Guide 7.0 (Ver.7.5) Software Rev.5

EPSON RC+ 7.0 选件

# *Vision Guide 7.0 (Ver.7.5) Software*

Rev.5

©Seiko Epson Corporation 2012-2023

## 前言

感谢您购买我公司的机器人产品。本手册包含正确使用 EPSON RC+选件 Vision Guide 引导的须知信息。

使用软件之前，请仔细阅读本手册及其他相关手册。

请将本手册放在随手可及之处，以便随时查看。

本公司的产品均通过严格的测试和检查，以确保产品性能符合标准。但请注意，如果不在本手册中所规定的条件中使用，可能导致产品性能无法正常发挥。

本手册记述了我们可以预见的危险和问题。请务必遵守手册中所述的安全注意事项，已确保安全正确的使用我们的机器人系统。

## 软件许可证

紧凑型视觉系统用户在使用该选件之前请仔细阅读本软件许可证协议。

附录 A: 紧凑型视觉系统终端用户许可证协议

附录 B: 紧凑型视觉系统开源软件许可证

## 商标

Microsoft、Windows、Windows 商标、Visual Basic 和 Visual C++为美国 Microsoft Corporation 在美国及其它国家的注册商标或商标。

其它品牌与产品名称均为各公司的注册商标或商标。

## 关于标记

Microsoft® Windows® 8 operating system

Microsoft® Windows® 10 operating system

Microsoft® Windows® 11 operating system

在本手册中，Windows 8、Windows 10和Windows 11指的是上述各操作系统。在某些情况下，Windows一般是指Windows 8、Windows 10和Windows 11。

## 注意事项

禁止擅自复印或转载本手册的部分或全部内容。

本手册记载的内容将来可能会随时变更，恕不事先通告。

如您发现本手册的内容有误或需要改进之处，请不吝斧正。

## 制造商

**SEIKO EPSON CORPORATION**

## 联系方式

有关咨询处的详细内容，请参阅下记手册序言中的“销售商”。

机器人系统 安全手册 请首先阅读本手册

## 阅读本手册之前



本节介绍了您在阅读本手册之前应了解的事项。

### 安全注意事项

请由取得相关资格的人员对机器人及相关机器进行搬运和设置。另外，请务必遵守各国的相关法规与法令。

使用前请仔细阅读本手册及相关说明书，并正确使用本机器。阅读之后，请妥善保管，以便随时取阅。

### 正文中的符号的含义

 警告	该符号表示如果用户忽视该指示或处理不当，可能会导致死亡或重伤。
 注意	该符号表示如果用户忽略该指示或处理不当，可能会导致人生伤害或财产损失。

## 培训

使用Vision Guide 7.0之前，请参加本公司的“Vision Guide入门培训”。本公司定期或不定期的举办培训会，以帮助客户了解我们的产品。

参加正规培训，可以帮助您了解本公司产品，提高产品使用效率。有关培训的详细信息，请咨询当地经销商。



<b>1. 手册帮助</b>	<b>1</b>
1.1 Vision Guide 手册结构 .....	1
1.2 相关手册 .....	2
1.3 使用在线帮助 .....	2
<b>2. 安全</b>	<b>3</b>
2.1 符号 .....	3
2.2 安全注意事项 .....	3
2.3 机器人安全 .....	3
<b>3. 导言</b>	<b>4</b>
3.1 Vision Guide 7.0 概述 .....	4
3.1.1 Vision Guide 7.0 用途 .....	4
3.1.2 Vision Guide 的功能 .....	4
3.2 EPSON RC+ 7.0 必备的基本知识 .....	5
<b>4. Vision Guide 概述</b>	<b>6</b>
4.1 概述 .....	6
4.2 使用 Vision Guide 7.0 所需的基本概念 .....	6
4.2.1 视觉序列 .....	6
4.2.2 视觉对象 .....	6
4.2.3 属性 .....	7
4.2.4 结果 .....	7
4.2.5 运行时视觉命令 .....	7
4.2.6 EPSON RC+ 7.0 项目中的 Vision Guide 7.0 .....	7
4.3 坐标系 .....	8
4.3.1 图像坐标系 .....	8
4.3.2 相机坐标系 .....	8
4.3.3 机器人坐标系 .....	9
4.4 打开 Vision Guide 窗口 .....	9
4.5 Vision Guide 窗口结构 .....	10
4.5.1 标题栏 .....	11
4.5.2 工具栏 .....	11
4.5.3 图像显示区域 .....	15
4.5.4 Vision Guide 窗口选项卡 .....	15
4.5.5 执行面板 .....	16
4.5.6 流程图 .....	17
4.5.7 序列和校准目录树 .....	18

4.6 Vision Guide 窗口 .....	19
4.6.1 属性列表工具栏和按钮 .....	19
4.6.2 序列窗口 .....	20
4.6.3 对象窗口 .....	22
4.6.4 校准窗口 .....	24
4.6.5 步进选项卡 .....	25
4.6.6 机器人选项卡 .....	26
4.7 视觉序列和对象 .....	27
4.7.1 视觉序列概述 .....	27
4.7.2 视觉序列的定义和基础知识 .....	27
4.7.3 视觉对象概述 .....	28
4.7.4 视觉对象定义 .....	28
4.8 视觉序列的属性和结果 .....	29
4.8.1 视觉序列的属性 .....	29
4.8.2 视觉序列的结果 .....	31
4.9 校准 .....	32
4.9.1 校准的概述 .....	32
4.9.2 校准的定义 .....	32
4.9.3 用于图像失真校正的成像图案 .....	33
<b>5. 视觉序列 .....</b>	<b>34</b>
5.1 视觉序列窗口 .....	34
5.1.1 选择视觉序列 .....	35
5.1.2 序列窗口属性列表 .....	35
5.1.3 序列窗口结果列表 .....	35
5.1.4 序列流程图 .....	35
5.2 新建一个视觉序列 .....	36
5.2.1 概述 .....	36
5.2.2 快捷键 .....	36
5.2.3 对话框选项 .....	36
5.3 删除一个视觉序列 .....	37
5.3.1 概述 .....	37
5.3.2 快捷键 .....	37
5.3.3 对话框选项 .....	37
5.4 删除视觉对象 .....	38
5.5 改变序列顺序 .....	38
5.6 运行视觉序列 .....	39
5.6.1 视觉序列选择 .....	39
5.6.2 设置视觉序列的属性 .....	39



5.6.3	视觉序列的结果	39
5.6.4	序列流程图	39
5.6.5	运行视觉序列	39
5.6.6	多次(循环)运行序列	40
5.6.7	执行视觉序列过程中的图像显示区域	40
5.6.8	中止视觉序列循环	40
5.7	测试和调试视觉序列	41
5.7.1	检查视觉对象的合格颜色	41
5.7.2	检查单个视觉对象结果	41
5.7.3	单步执行视觉序列	42
5.7.4	使用视觉统计功能	42
5.7.5	在 RUN 窗口和 Vision Guide 窗口之间切换	42
5.8	通过 SPEL+运行视觉序列	43
5.9	图像采集	43
5.9.1	何时采集图像?	43
5.9.2	在多个视觉序列中使用相同的图像	43
5.9.3	使用图像缓冲区	44
5.9.4	使用外部触发器采集图像	45
5.9.5	使用彩色图像	46
<b>6.</b>	<b>视觉对象</b>	<b>47</b>
6.1	视觉对象的基本事项	47
6.1.1	搜索窗口	47
6.1.2	模型窗口	52
6.1.3	模型原点	53
6.2	使用视觉对象	54
6.2.1	ImageOp 对象	54
6.2.2	Geometric 对象	62
6.2.3	Correlation 对象	85
6.2.4	Blob 对象	112
6.2.5	Edge 对象	128
6.2.6	Polar 对象	136
6.2.7	OCR 对象	146
6.2.8	CodeReader 对象	153
6.2.9	ColorMatch 对象	158
6.2.10	LineFinder	164
6.2.11	LineInspector 对象	172
6.2.12	ArcFinder 对象	179
6.2.13	ArcInspector 对象	186

6.2.14 DefectFinder 对象 .....	194
6.2.15 Frame 对象.....	203
6.2.16 Line 对象 .....	208
6.2.17 Point 对象.....	214
6.2.18 BoxFinder 对象.....	223
6.2.19 CornerFinder 对象.....	232
6.2.20 Contour 对象 .....	239
6.2.21 Text 对象 .....	255
6.2.22 Decision 对象 .....	260
6.2.23 Coordinates 对象.....	263
6.2.24 从一个对象返回多个结果 .....	266
6.2.25 多对象自动搜索 .....	272
6.2.26 显示/隐藏所有视觉对象标签.....	277
6.2.27 打开所有视觉对象图形.....	277
6.2.28 仅显示当前对象.....	277
<b>7. 视觉校准 .....</b>	<b>279</b>
7.1 相机安装.....	281
7.2 镜头失真和相机倾斜校正 .....	282
7.3 基准点和相机点.....	283
7.3.1 可移动式相机基准点 .....	285
7.3.2 固定向下相机和独立相机的基准点.....	286
7.3.3 通过 TwoRefPoints 示教 .....	286
7.4 为校准创建视觉序列 .....	287
7.4.1 检测一个目标的视觉序列 .....	287
7.4.2 检测九个目标的视觉序列 .....	288
7.4.3 用于失真校正的视觉序列 .....	289
7.4.4 用于本地、工具和机械臂设定的视觉序列.....	289
7.5 校准图形用户界面 .....	290
7.5.1 创建一个新的校准 .....	290
7.5.2 删除校准.....	291
7.5.3 校准属性和结果.....	291
7.5.4 失真检测.....	294
7.5.5 点示教 .....	295
7.5.6 示教校准点 .....	297
7.5.7 校准完成对话框.....	297
7.6 校准程序.....	299
7.6.1 校准程序：可移动式相机 .....	299
7.6.2 校准程序：固定向下相机 .....	308

7.6.3 校准程序：固定向上相机 .....	314
7.6.4 校准程序：独立相机 .....	321
7.7 使用相机的本地检测 .....	323
7.7.1 在工作平面上定义本地 .....	323
7.8 检测可移动式相机安装位置 .....	328
7.8.1 相机安装位置的工具设定 .....	328
7.8.2 相机安装位置的机械臂设定 .....	332
7.9 使用相机的工具设定 .....	335
7.10 使用相机的 3D 工具设置 .....	339
<b>8. 直方图工具 .....</b>	<b>344</b>
8.1 利用直方图 .....	344
8.2 直方图举例 .....	345
8.2.1 Correlation 对象使用的直方图 .....	346
8.2.2 Blob 对象使用的直方图 .....	346
<b>9. 使用 Vision Guide 统计 .....</b>	<b>349</b>
9.1 对话框选项/信息 .....	350
9.2 视觉对象和支持的统计 .....	351
9.3 支持的视觉对象结果 .....	351
9.4 通过 SPEL+ 可用的视觉对象统计 .....	352
<b>10. 教程 .....</b>	<b>356</b>
10.1 快速启动：Vision Guide 7.0 教程 .....	356
10.1.1 教程概述 .....	356
10.1.2 本教程所需的物品 .....	357
10.1.3 启动 EPSON RC+ 7.0 并创建一个新项目 .....	360
10.1.4 新建一个视觉序列 .....	360
10.1.5 教程的相机镜头配置 .....	360
10.1.6 利用 Blob 对象来寻找零件 .....	362
10.1.7 编写一个 SPEL+ 程序来与视觉序列联用 .....	367
10.1.8 校准机器人相机 .....	369
10.1.9 示教在视觉引导中使用的点 .....	374
10.1.10 使用视觉系统让机器人移向零件 .....	376
<b>11. 在 SPEL+ 语言中使用 Vision Guide 7.0 .....</b>	<b>378</b>
11.1 概述 .....	378
11.2 Vision Guide 7.0 的 SPEL+ 命令 .....	378

11.3 通过 SPEL+运行视觉序列: VRun.....	379
11.4 在 SPEL+中读取属性和结果: VGet、VSet .....	381
11.4.1 使用 VGet.....	382
11.4.2 使用 VSet .....	382
11.5 在序列和对象名称中使用变量.....	383
11.6 在 SPEL+中使用序列结果.....	384
11.7 通过 SPEL+语言读取多个结果 .....	384
11.8 视觉命令和多任务处理联用 .....	385
11.9 视觉系统和机器人联用.....	386
11.9.1 定位结果.....	386
11.9.2 定义一项工具 .....	388
11.9.3 机器人电路板的工具计算.....	390
11.9.4 定位相机以搜索托盘 .....	391

# 1. 手册帮助

## 1.1 Vision Guide 手册结构

### 导言

本节概述了整个手册的内容。为了帮助理解EPSON RC+7.0的基本特性，还介绍了在线帮助、安全特性以及参考案例。

### 如何安装：

本节介绍了Vision Guide 7.0必要的系统和产品配置，以及如何安装硬件和软件。

### 快速入门：初次应用Vision Guide 7.0

本节利用示例应用向用户介绍了第一次使用Vision Guide 7.0应如何操作。本节从新视觉对象的创建、Vision Guide 7.0 可移动式相机的校准、实际的机器人动作到Vision Guide 7.0检测到的零件，详细介绍了Vision Guide 7.0的使用。

### Vision Guide窗口

本节介绍了Vision Guide窗口的布局图并给出了使用说明。还包括Vision Guide工具栏、Image Display、Run Panel、Object、Sequence和Calibration选项卡信息。

### 视觉对象

本节介绍了Vision Guide 7.0可提供的不同类型的视觉工具，以及如何使用这些工具。

### 直方图和统计工具

本节介绍了对不同类型的视觉对象如何使用直方图，包括Blob、Correlation和Polar对象。还对Vision Guide统计工具进行了介绍，从具有统计对话框功能的Vision Guide窗口以及SPEL+语言到统计属性的获取。

### 视觉序列

本节介绍了什么是视觉序列以及如何使用并应用视觉序列，还阐述了Vision Guide序列的调试技术。

### 校准

本节介绍了如何使用不同类型的校准方式。

### 在SPEL+中使用Vision Guide 7.0

本节介绍了如何在SPEL+语言中运行视觉序列，以及如何获取视觉属性和结果。还解释了如何使用Vision Guide 7.0结果对机器人进行指导。

### 1.2 相关手册

使用Vision Guide 7.0时，请在参考Vision Guide 7.0手册的同时也参考下列相关手册。

#### *Vision Guide 7.0 Hardware & Setup*

该手册介绍了Vision Guide的正确使用、操作注意事项以及设置硬件时的警告。

#### *Vision Guide 7.0 Properties & Result Reference*

该手册包括视觉序列和视觉对象可使用的所有属性和结果的完整参考。还包括有关每种属性和结果的正确使用、注意事项以及警告信息。

#### *EPSON RC+7.0 User's Guide*

该手册包括有关使用EPSON RC+机器人控制系统的信息。

#### *SPEL<sup>+</sup> Language Reference Manual*

该手册完整地介绍了SPEL<sup>+</sup>语言的所有命令。

### 1.3 使用在线帮助

EPSON RC+ 7.0支持在线帮助系统。与通过手册进行查找的传统方法相比，帮助系统让查找信息变得更容易。

在EPSON RC+ 7.0中，有多种方式可以获取在线帮助：

- 可以随时按下F1功能键获取与上下文有关的帮助。将显示与您目前正在处理的项目有关的帮助。如果您需要得到关于屏幕中或对话框中某一项目的信息，这一功能是非常有用的。如果您在编辑一个程序，则将在光标的位置显示有关SPEL<sup>+</sup>关键词的帮助信息。您可以使用在线帮助查看语法信息，以便使用SPEL<sup>+</sup>语言。
- 如果可用，点击对话框中的<Help>按钮。
- 要查看目录或选择主题，可在帮助菜单中选择目录。点击用绿色突出显示的下划线文本可以选择主题。(进而跳转至您关注的主题。)
- 在Help菜单中选择目录，然后按下<S>按钮或点击<Search>按钮，搜索与特定主题的相关信息。

使用在线帮助时，您会发现有几项是以绿色突出显示的，并带有下划线。这些是超文本链接，如果您点击突出显示的文本，系统将跳转至帮助系统中与突出显示的文本相关的区域。您还会看到，有些文本是以绿色突出显示的，并且带有虚线下划线。点击这类文本后，会弹出一个窗口，窗口中将显示对突出显示文本更详细的介绍，也可能显示其他相关信息，您可以跳转至此类信息的相关页面。




本手册中的大部分信息也存在于Vision Guide 7.0帮助系统中，但是，为了提供正确的超文本链接并方便使用，其排列方式可能稍有不同。

## 2. 安全




在使用Vision Guide之前，请仔细阅读本手册。  
阅读后，请妥善保管本手册，以便日后随时取阅。

### 2.1 符号

在整本手册中，重要安全注意事项将用下列符号提示。  
请务必阅读各符号随附的说明。

 警告	<p>该符号表示如果不遵守这些相关的指示可能会导致重伤或死亡。</p>
 警告	<p>表示如果无视这些标识并错误地加以使用，则可能会因触电而导致人身伤害。</p>
 注意	<p>表示如果无视该标识并错误地加以使用，则可能会导致人身伤害或设备或设施受损。</p>

### 2.2 安全注意事项

 警告	<ul style="list-style-type: none"> <li>■ 不要将紧凑型视觉系统用于此目的，以确保安全。</li> <li>■ 必须在本手册所述条件下使用本产品。 在超出规定条件的环境中使用本产品不仅会缩短产品的使用寿命，还可能造成严重的安全问题。</li> </ul>
 注意	<ul style="list-style-type: none"> <li>■ 请从我们的供应商那里购买相机和相机电缆。 注意，其他制造商生产的相机或相机电缆不在保修范围之内。</li> </ul>
 注意	<ul style="list-style-type: none"> <li>■ 图像处理效果可能会受到光源改变、环境光线或周边电磁噪声的影响。 如果根据图像处理结果操作操纵器，则机器人系统在设计时应考虑到机器人操作可能会根据整个相机视野做出反应。</li> </ul>

### 2.3 机器人安全

无论是操作机器人还是操作其他自动化设备，务必始终将安全放在第一位。EPSON RC+ 7.0系统内置了多种安全功能，例如紧急停机、机器人暂停输入和安全保护输入。设计机器人单元时，应使用这些安全功能。

有关安全信息和指南，请参阅 *EPSON RC+ 7.0 User's Guide* 的安全章节。

# 3. 引言

## 3.1 Vision Guide 7.0 概述

### 3.1.1 Vision Guide 7.0 用途

Vision Guide 7.0主要致力于提供解决机器人指导应用所需的视觉工具，Vision Guide 7.0有能力提供多种不同类型的机器视觉应用。Vision Guide 7.0的常见应用包括定位和搬运零件、零件检查和测量应用。对于同时需要搬运和视觉的大多数应用，都可以通过Vision Guide 7.0和EPSON机器人来实现。

Vision Guide 7.0与其他视觉系统的主要差别之一是，Vision Guide 7.0形成了EPSON机器人系统的一个组成部分。因此，您会发现多种功能，例如Vision Guide 7.0系统内置的机器人-相机校准程序。最终结果大大缩短了完成您以视觉为基础的应用的时间。

### 3.1.2 Vision Guide 的功能

Vision Guide 7.0提供的一些主要功能和工具包括：

- 支持多个相机方向和校准的集成工具。校准比其他机器人视觉系统简单得多，而且更精确。
- 用户可以通过GUI操作轻松创建模型。在此创建的模型视觉系列可以直接在实际应用中使用。
- 完全融入EPSON RC+ 7.0编程和开发环境。
- 二进制对象分析工具可以测量物体的尺寸、形状和位置的。还有一些其他工具可以计算孔的数量并显示那些特征不明显的圆度。
- Geometric搜索工具，可以根据几何特征查找对象模型。
- Correlation相关搜索工具，利用先进的模板匹配定位技术，可以在各种光照条件下定位对象。
- 边缘检测工具，以亚像素精度定位特定的边缘。
- LineFinder工具和ArcFinder工具，以亚像素精度，轻松准确的检测直线和圆弧的边缘。
- Polar搜索工具是一种快速角度的搜索工具，可以快速测量复杂对象的旋转情况。这对于视觉引导的机器人中的拾取与放置应用非常有用。
- DefectFinder工具，可利用模板图像进行差异检查，以检测图像缺陷。
- Line、Point工具，有连接多个点以创建直线，并检测它的功能。
- Frame工具，可以根据参考框动态定位所有视觉工具。



- 对象参考机制，可以将某个视觉工具的定位结果，用于另一个视觉工具的定位基准。从而节省了程序调试时间。
- 直方图不仅能显示像素数据更多详细信息，还能设置所需工具的正常范围。是一款强大的程序。
- 内置统计计算，可计算平均值、标准偏差、范围、最小值和最大值，以及各种其他统计数据，可用作视觉工具设计时间或运行时间的参考。
- 配备了自动校正功能，可用于相机镜头、相机和机器人外围设备中的角度偏移。
- ColorMatch 工具和 ColorFilter 操作支持彩色相机和彩色图像。

## 3.2 EPSON RC+ 7.0 必备的基本知识

Vision Guide 7.0是一款以EPSON RC+ 7.0为核心环境的选件。要使用Vision Guide 7.0, 您需要具备EPSON RC+ 7.0的开发和EPSON机器人的知识。学习本手册之前, 请先熟悉以下各项知识。

- EPSON RC+ 7.0项目管理概念和使用方法。
- 使用EPSON RC+ 7.0创建并编辑SPEL+程序。
- 在Run窗口中运行SPEL程序。
- SPEL+的基本语言结构、功能和使用方法等。

初次使用EPSON RC+ 7.0的用户, 请先参加本公司的导入培训。

## 4. Vision Guide 概述

### 4.1 概述

在本章，我们将集中介绍一些概念和定义，以便您能对Vision Guide 7.0及其组件有一个全面的了解。本章包括以下内容。

- 使用Vision Guide 7.0需要了解的基本定义。
- 如何从EPSON RC+ 7.0中打开Vision Guide 7.0窗口。
- Vision Guide 7.0图像显示说明。
- 介绍所有视觉对象以及在工具栏上的操作按钮。
- 介绍属性和结果。
- 介绍序列窗口、对象窗口和校准窗口。
- 视觉对象和视觉序列的执行方法和使用案例。

### 4.2 使用Vision Guide 7.0所需的基本概念

下文是对一些基本概念的解释，帮助您更好地理解本章内容。

#### 4.2.1 视觉序列

视觉序列是指，按照特定的顺序对视觉对象进行分组，然后从通过Vision Guide窗口或SPEL+语言执行这些对象。

视觉序列有特定的属性，用于设置视觉序列执行。例如，Camera属性定义了要使用哪台相机为视觉序列捕获图像，RuntimeAcquire属性则定义了如何为视觉序列捕获图像。

可将视觉序列视作一种容器，里面存储了解决特定视觉过程或部分过程所需的视觉对象。总的来说，视觉序列是所有视觉处理的起点。

#### 4.2.2 视觉对象

视觉对象是一种视觉工具，可应用于相机捕获图像。

支持的视觉对象包括：Geometric Search、Correlation Search、Blob Analysis、Polar Search、Edge Detection、Line Search、Arch Search、Differential Test、Line Creation、Points和Frames。

所有的视觉对象(应用于当前图像)，在执行后都会返回结果，例如执行视觉对象需要的时间、位置信息、角度信息、是否查找到了检测对象以及结果是否通过等。

视觉对象有些属性用于定义视觉对象如何执行的特性。这些也会产生结果，即执行视觉对象之后返回的值。

### 4.2.3 属性

您可以将属性视为视觉对象或视觉序列上的参数集。

通过在Vision Guide窗口中进行点击操作来设置视觉属性，该窗口提供了一种快速创建并测试视觉应用的方法。

可通过SPEL+语言对视觉属性进行设置和检查。这满足了在运行时间对视觉对象进行动态修改的灵活性要求。

视觉序列和视觉对象属性非常强大。它们不仅易于修改，还能帮助理解和使用视觉对象。同时，对于更复杂的应用展现了足够的灵活性。

### 4.2.4 结果

结果是指，就是执行视觉对象或视觉序列以后返回的值。常用结果示例包括：

**Time**            返回执行视觉对象或视觉序列所需的时间。

**RobotXYU**    返回查找到的功能在机器人坐标系中的X、Y和U位置。

**Found**           回复是否查找到了视觉对象

**Passed**        返回视觉对象的测试结果合格与否。

可在Sequence和Object窗口中的Vision Guide窗口中查看视觉结果。SPEL+程序同样可以使用视觉结果。

### 4.2.5 运行时视觉命令

SPEL+机器人语言中已添加了一系列视觉命令，与机器人动作和视觉引导无缝结合。例如VRun命令，可以通过一个函数调用从SPEL+语言启动视觉序列。

VGet命令可以获得从视觉对象、序列和校准返回的结果。

视觉命令非常强大，因为视觉序列是在Vision Guide 7.0点击开发环境中进行创建、修改和维护，但是在Vision Guide 7.0中创建的所有东西也可以通过SPEL+语言进行访问。

所需视觉硬件

要使用Vision Guide 7.0，必须有下列硬件之一。

- 紧凑型视觉系统 CV1(固件版本 2.1.0.0 及以上)
- 紧凑型视觉系统 CV2(固件版本 2.3.0.0 及以上)
- PC 视觉系统 PV1

有关硬件详情，请参阅 *Vision Guide 7.0 Hardware & Setup* 手册。

### 4.2.6 EPSON RC+ 7.0 项目中的 Vision Guide 7.0

EPSON RC+ 7.0 是根据机器人的实际应用，创建一个项目，其中包括程序、示教点和机器人设置。

这种项目的结构，可以轻松实现一台机器人运行多个项目，并在不破坏之前的应用程序的基础上进行新环境应用的测试。

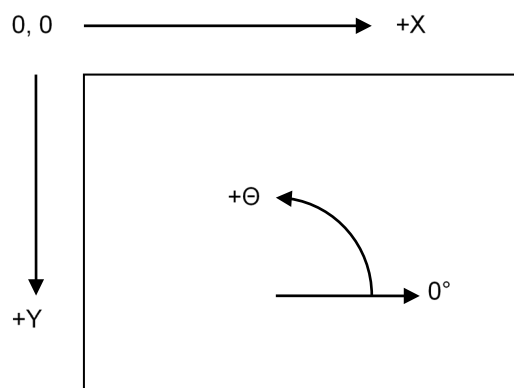
当您在创建一个包含Vision Guide 7.0的EPSON RC+ 7.0应用时，除项目中通常含有的其它各项之外，该应用所需的所有相关视觉序列和视觉对象也都存储在该项目中。这一点确保了您在打开一个已有项目时，您可以使用与该项目相关的所有事项。

## 4.3 坐标系

本节将介绍Vision Guide 7.0中使用的坐标系。

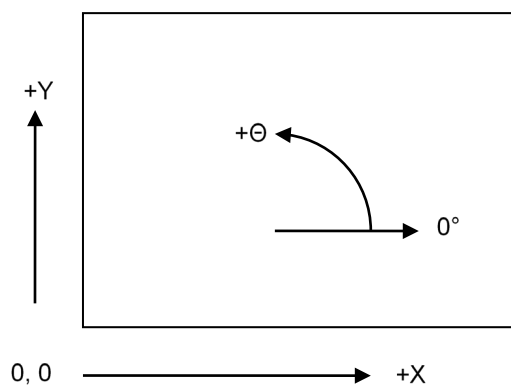
### 4.3.1 图像坐标系

图像坐标系用于显示相机图像和图片图像。单位为像素。



### 4.3.2 相机坐标系

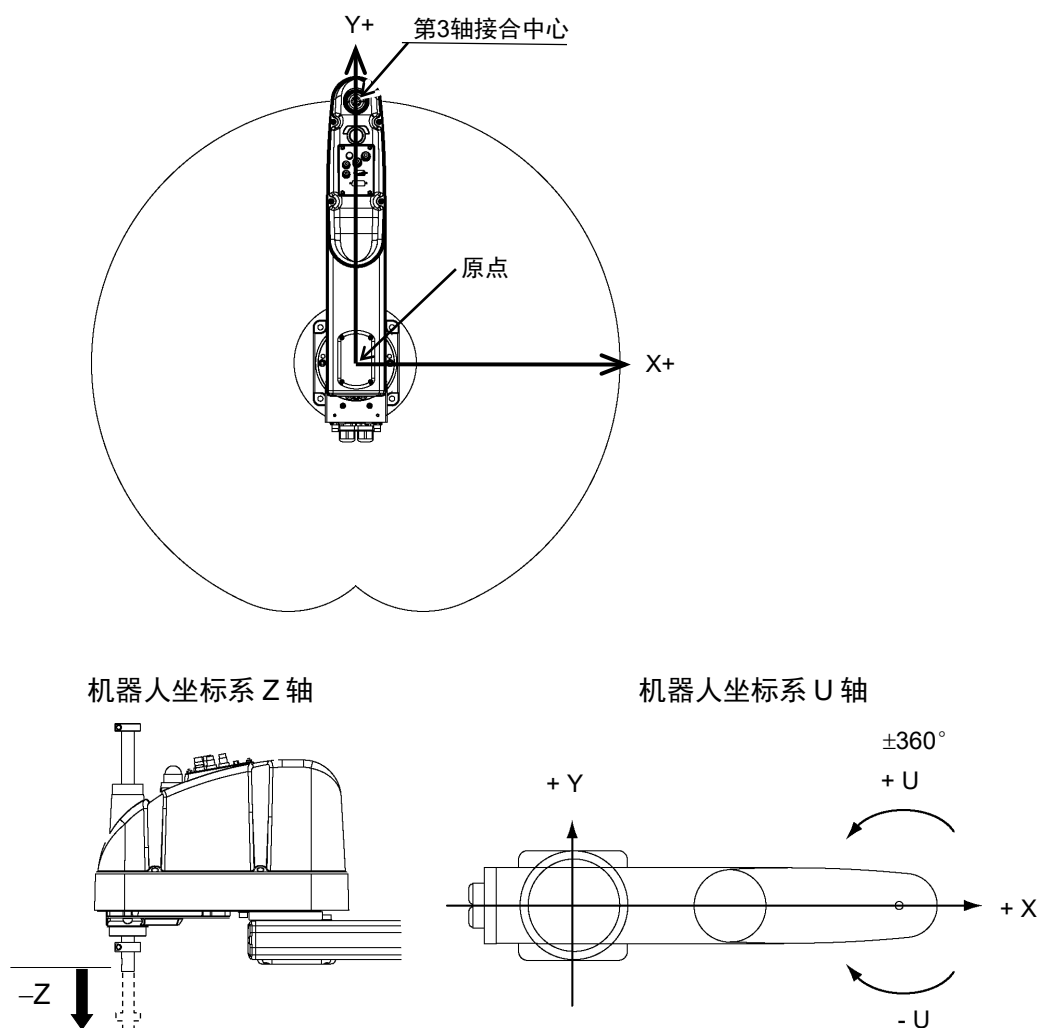
相机坐标系是相机视野内的坐标。单位为毫米。为了获得该系统中的坐标，必须利用相机方向对相机进行校准。



### 4.3.3 机器人坐标系

机器人坐标系是机器人的坐标。单位为毫米。要获得此坐标系中的坐标。必须以机器人坐标校准相机。有关机器人坐标系详细信息，请参阅 *EPSON RC+ 7.0 用户指南* 手册。

SCARA机器人的机器人坐标系



## 4.4 打开 Vision Guide 窗口

在EPSON RC+ 7.0开发环境中打开Vision Guide窗口。

启动EPSON RC+ 7.0以后，有2种不同的方式可以打开Vision Guide窗口：

从主工具栏：在EPSON RC+ 7.0的主工具栏上您可以看见  <Vision>图标。点击 <Vision Guide>图标可以打开Vision Guide窗口。

从工具菜单：在工具菜单上选择[Vision]可以打开Vision Guide窗口。

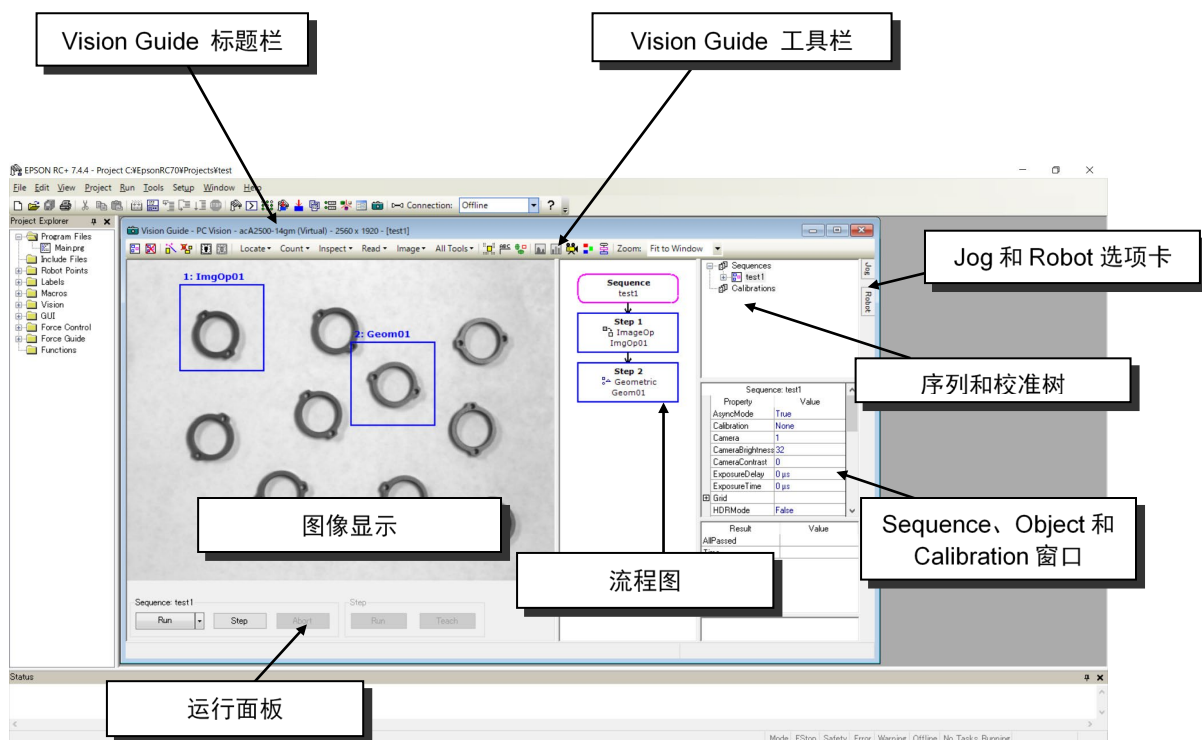
打开Vision Guide窗口以后，我们就可以开始使用Vision Guide 7.0了。随后几页内容介绍了Vision Guide窗口的基本构造。

### 4.5 Vision Guide 窗口结构

Vision Guide窗口是一个点击操作环境，您大部分的视觉开发都在这里进行。

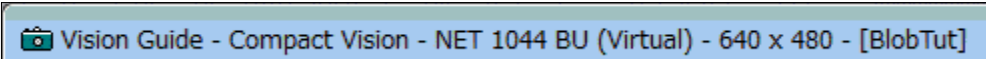
要使用Vision Guide 7.0，您需要做的第一件事就是了解该窗口及其主要部分。Vision Guide窗口可分为下列几大部分：

- 标题栏
- 工具栏
- 流程图
- 序列和校准树
- 图像显示
- Sequence、Object、Calibration窗口
- Jog和Robot选项卡
- 运行面板



### 4.5.1 标题栏

Vision Guide窗口上有一个标题栏，包括窗口标题、相机类型和分辨率，以及当前序列名称。视觉序列名称是用括号括起来的。示例标题栏如下。示例标题栏展示了正在用紧凑型视觉系统运行Vision Guide 7.0，当前使用的是“BlobTut”视觉序列。



NOTE  


请注意EPSON RC+ 7.0环境主标题栏和Vision Guide标题栏之间的区别。

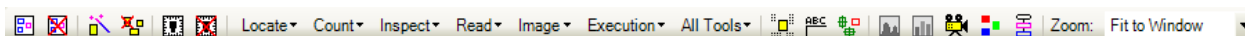
EPSON RC+ 7.0标题栏包括当前项目的名称，而Vision Guide窗口标题栏则包括当前序列的名称。

### 4.5.2 工具栏

一般工具栏都包含在 Microsoft Windows™ 应用中，它们使您可以快速访问 Microsoft Windows™ 产品的常用功能。

EPSON RC+ 7.0 和 Vision Guide 7.0 都使用了工具栏。

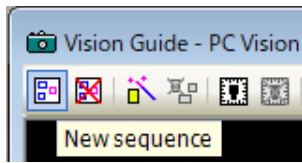
Vision Guide的工具栏位于Vision Guide窗口的顶部，就在标题栏下方，如下所示：



Vision Guide 窗口的 Vision Guide 的工具栏按钮分为了几个小组，以方便查找和使用。对工具栏各组按钮的简要说明如下：

- 最左边的两个按钮为一组，用于创建和删除视觉序列。
- 下面的两个工具栏按钮用于运行步骤向导和删除对象。
- 下面的两个工具栏按钮用于创建和删除校准。
- 下面的工具栏按钮组用于选择视觉对象类别(检测、计数、检查、读取、图像或所有工具)。
- 下面几个工具栏按钮分为一组，可视为Vision Guide 7.0环境的实用工具。它们包括仅显示当前对象、删除对象、关闭所有标签和显示图形的按钮。
- 下面两个工具栏按钮用于打开直方图和统计对话框。
- 最右边的两个按钮用于切换实时视频和固定视频，以及切换彩色和灰度图像。
- 最后面的工具栏按钮用于打开或关闭流程图显示。

Vision Guide工具栏上有对各工具栏按钮的工具提示，如下所示。











要查看某个工具栏按钮的说明，只需将鼠标指针移动到该工具栏按钮上，约2秒钟的时间您就可以看到该工具栏按钮的工具提示。

以下是对Vision Guide各工具栏按钮的一般说明。

按钮	说明
	<b>New sequence:</b> 创建一个视觉序列。将弹出一个对话框，要求用户输入新序列的名称。
	<b>Delete sequence:</b> 删除当前项目中的一个视觉序列。 如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。
	<b>New step:</b> 按下该按钮将出现步骤向导。 可使用步骤向导添加对象。 与菜单栏不同，在步骤向导中添加对象前，可以配置对象名及其在序列中的位置。
	<b>Delete object:</b> 删除当前的活动视觉对象。 要删除一个视觉对象，只需选择需要删除的视觉对象然后点击该按钮。 如果当前项目中没有视觉序列，或者如果当前视觉序列中没有视觉对象，则该按钮会变暗(无法选中)。
	<b>New calibration:</b> 打开Calibration对话框，新建校准。如果当前项目中没有视觉序列，则无法使用该按钮。
	<b>Delete calibration:</b> 打开Delete Calibration对话框，删除校准。 如果当前项目中没有视觉校准，则无法使用该按钮。
	<b>Show only current object:</b> 按下此按钮时，将只显示当前选择的对象。这样操作一个对象时不会干扰其他对象。
	<b>Force all labels off:</b> 按下该按钮会移除视觉对象上的选项卡。如果有很多视觉对象紧靠在一起，并且很难区分这些对象，这个功能就非常有用。如果当前项目没有视觉序列，则此按钮会变暗。
	<b>Force all graphics on:</b> 按下该按钮会显示视觉对象的所有图形(搜索窗口、模型原点、模型窗口、直线和标签)。 该按钮优先于各独立视觉对象的Graphics属性设置，让快速查看所有视觉对象变得容易，而不是修改各独立视觉对象的Graphics属性。
	<b>Histogram:</b> 点击该按钮会打开Histogram对话框。如果当前项目中没有视觉序列，或者如果当前视觉序列中没有视觉对象，则该按钮呈灰色。
	<b>Statistics:</b> 点击该按钮会打开Statistics对话框。如果当前项目中没有视觉序列，或者如果当前视觉序列中没有视觉对象，则该按钮呈灰色。
	<b>Freeze image:</b> 在播放图像和定格图像之间切换。
	<b>Color / Grayscale:</b> 在彩色图像和黑白图像之间切换。



按钮	说明
	Flow Chart: 启用或禁用流程图显示。
	Locate: 可选择检测类别的对象。(Geometric、Correlation、Blob、Edge、Polar、Arc Finder、Line Finder、Box Finder、Corner Finder、Frame、Line、Point、Contour和Coordinates)
	Count: 可选择计数类别的对象。(Blob、Correlation、Geometric)
	Inspect: 可选择检查类别的对象。(Blob、Defect Finder、Line、Line Inspector、Arc Inspector、Color Match)
	Read: 可选择读取类别的对象。(Code Reader、OCR)
	Image: 可选择图像类别的对象。(ImageOp、Text)
	Execution: 可选择执行类别的对象。(Decision)
	All Tools: 可选择全部对象。

#### 通过选择工具栏中的类别调用的按钮

按钮	工具提示：简单说明
	ImageOp(图像操作): 创建一个ImageOp对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	Geometric(几何形状): 创建一个Geometric对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	Correlation(相关搜索): 创建一个Correlation对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	Blob(二进制对象分析): 创建一个Blob对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	Edge(边缘检测): 创建一个Edge对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	Polar(极坐标搜索): 创建一个Polar对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	OCR: 创建一个OCR对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。 如果没有光学字符识别许可, 则此按钮会变暗(无法选中)。
	CodeReader: 创建一个CodeReader对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	ColorMatch: 创建一个ColorMatch对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	BoxFinder: 创建一个BoxFinder对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	CornerFinder: 创建一个CornerFinder对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。
	LineFinder: 创建一个LineFinder对象。 如果当前项目没有视觉序列, 则此按钮会变暗(无法选中)。

### 按钮 工具提示：简单说明

---



**LineInspector:** 创建一个LineInspector对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**ArcFinder:** 创建一个ArcFinder对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**ArcInspector:** 创建一个ArcInspector对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**DefectFinder:** 创建一个DefectFinder对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**Frame:** 创建一个Frame对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**Line:** 创建一个新的Line对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**Point:** 创建一个新的Point对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**Contour:** 创建一个Contour对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**Text:** 创建一个Text对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。



**Decision:** 创建一个Decision对象。  
如果当前项目没有视觉对象，或者序列中没有对象，则此按钮会变暗(无法选中)。



**Coordinates:** 创建一个Coordinates对象。  
如果当前项目没有视觉序列，则此按钮会变暗(无法选中)。

### 4.5.3 图像显示区域

图像显示区域位于 Vision Guide 窗口中，Vision Guide 工具栏的正下方，用于显示相机中看到的图像。

图像显示区域用于显示图像处理的每个图形。例如，在图像显示区域中，可以看到在不同位置显示的多个矩形和十字线。这些图形帮助简化使用视觉对象工具的开发工作。

Vision Guide 窗口和图像显示区域在 *4.5 Vision Guide 窗口的组成部分* 中进行了描述。

如果扩大 Vision Guide GUI 窗口的大小，图像显示区域也会随之变大。同样，Run 对话框的图像显示，也会随着窗口大小的变化而改变。图像显示区域和选项卡组之间设有一个分隔栏，可以调节图像显示的大小。即使更改图像显示的尺寸，也可以保持图像的纵横比。

### 4.5.4 Vision Guide 窗口选项卡

选项卡的主要作用是可以快速访问相关数据和分组数据。与使用附加菜单项或多窗口相比，选项卡更便于访问数据。

Vision Guide 窗口使用选项卡提供单一的窗口视觉开发环境，让学习和记忆系统变得容易。

位于 Vision Guide 窗口右侧的是由 2 个选项卡组成的 1 组选项卡，分别为 Jog 和 Robot。这些选项卡位于序列目录树旁的固定位置，可在图像显示区域之外使用。

这两个选项卡在 Vision Guide 7.0 开发环境下使用，让您可以轻松访问 Vision Guide 7.0 使用相关的知识和帮助。本章稍后会介绍各选项卡的详细信息。

### 4.5.5 执行面板

执行面板就位于图像显示区域的下方。执行面板的作用是在Vision Guide 7.0开发环境中执行并调试序列。执行面板如下所示：



#### 执行序列

创建了视觉序列并添加了视觉对象以后，点击位于执行面板左侧的<Run>按钮即可运行视觉序列。

#### 循环

可在通过点击<Run>按钮右侧的▼按钮显示的<Cycles>中指定循环次数。此框中的数字是几，视觉序列就会执行几次。

#### 中止

如需暂停视觉序列，请点击执行面板上的<Abort>按钮。<Abort>按钮仅在序列实际运行时才可以使用。<Abort>按钮也可用于中止正在等待闪光触发的序列。

#### 分步运行

点击<Step>按钮，可以运行视觉序列中的一个步骤。在每一个步骤中执行一个视觉对象。

如需执行下一步，请再次点击<Step>按钮。第一次点击<Step>按钮时，视觉对象会被蓝色虚线包围，并且步进模式处于非活动状态。

再次点击<Step>按钮时，会执行序列中的第一个对象，然后是第二个对象，以此类推，直到执行了序列中的最后一个视觉对象。如果要返回步骤执行的开始，点击执行面板左侧的<Run>按钮并运行整个视觉序列。

用户可以在运行视觉对象时，查看特定对象的结果值。

使用流程图或选择树形图来选择要确认的对象。选中视觉对象以后，点击执行面板左侧的<Run>按钮，就会开始执行视觉序列。

从流程图或序列目录树选择视觉对象时，您可以查看您选择的特定视觉对象的结果。



执行面板左侧的<Run>按钮可以快速试验并调试整个视觉序列，而无需从 SPEL+运行视觉序列。您还可以修改和测试视觉序列，如果不需要该结果，也可以返回到之前保存的视觉序列。因此，执行面板左侧的<Run>按钮不会在每次运行时，自动保存视觉序列。但是，在 Run 窗口中运行视觉序列时，如果启用了自动保存，那么视觉序列就会项目的其余部分一起自动保存。

### 执行对象

创建视觉序列并添加视觉对象后，点击<Run>按钮(执行面板右侧的第二个按钮)执行视觉对象。

### 示教

如果创建的视觉对象需要示教，请点击<Teach>按钮，在示教窗口进行示教。

如果对象不需要示教，则无法点击<Teach>按钮。

### 校准

从序列目录树中选择校准对象时，执行面板上仅显示<Teach Points>和<Calibrate>按钮。

点击<Teach Points>按钮显示点示教窗口以执行示教。

点示教完成后，可通过点击<Calibrate>按钮执行校准。如果未执行点示教，则无法点击<Calibrate>按钮。

## 4.5.6 流程图

流程图显示在图像显示区域的右侧。流程图展示所选序列中对象的处理流程。第一个流程代表当前所选序列，第二及之后的流程则为序列中包含的对象。对象按照执行顺序排列。

流程框通常以蓝色显示。点击了想选择的对象流程时，框的颜色变为粉色。同时，图像显示区域的对象窗口也变为粉色。但即使点击序列流程，其颜色也不会变为粉色。

选择了序列或对象时，所选序列或对象的属性和结果显示在流程图右侧的属性和结果列表中。相应节点的背景颜色变为灰色。

序列中包含的所有对象均成功执行后，序列流程的框颜色变为绿色，如果其中任意一个失败则为红色。

对象流程成功执行后，其框颜色变为绿色，如果失败则为红色。

右键点击流程可调用各种操作。

右键点击序列流程时可进行以下操作。

<b>New sequence</b>	创建一个视觉序列。
<b>Delete sequence</b>	删除当前项目的视觉序列。
<b>Change step order...</b>	修改序列的步骤顺序。
<b>Run sequence</b>	运行整个视觉序列。
<b>Add new step</b>	显示步骤向导。使用步骤向导添加对象。

右键点击对象流程，可以对当前活动的视觉对象，进行以下操作。

<b>Run object</b>	运行视觉对象。 如果Enabled属性设置为“False”则无法选择该选项。
<b>Copy</b>	复制视觉对象。
<b>Cut</b>	剪切视觉对象。
<b>Delete</b>	删除视觉对象。
<b>Teach model</b>	当视觉对象需要进行示教时，请示教对象的模型窗口部分。
<b>Show model</b>	显示完成示教的模型。
<b>Edit Window</b>	激活窗口编辑模式，用户可以在其中设置搜索窗口的不要紧的像素。
<b>Disable object</b>	指定当Enabled属性设置更改为“False”时，不运行视觉对象。
<b>Enable object</b>	指定当Enabled属性设置更改为“True”时，运行视觉对象。

### 4.5.7 序列和校准目录树

序列和校准目录树显示在流程图的右上部。所有序列和校准项目显示在目录树中。序列目录树中的序列和对象节点可使用与流程图中的流程相同的方式操作。

点击校准节点将执行面板更改为显示校准用面板。

## 4.6 Vision Guide 窗口

### 4.6.1 属性列表工具栏和按钮

工具栏和按钮显示在属性列表的上方。您可以选择属性列表和结果列表的显示模式。(默认为普通模式)

显示模式如下。

**按钮**    **描述**



**普通模式:**

显示当前所选的视觉序列、视觉对象和校准的常用属性和结果。如果您是 Vision Guide 的初学者，或只是想简单的修改属性值并测试时，请选择普通模式。

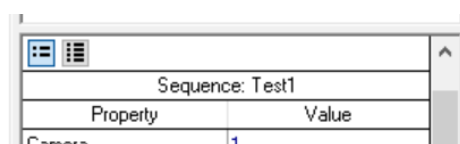


**高级模式:**

显示当前所选的视觉序列、视觉对象和校准的所有可用属性和结果。当您需要进行高级设定，并使用到所有属性和结果调整视觉序列时，请选择高级模式。

在此页面选择的模式会应用于序列、对象和校准中，当您重新启动EPSON RC+7.0时，也会保留相同设置。

当您选择高级模式后，会显示当前序列、对象和校准的所有可用属性和结果。在您关闭并重新启动EPSON RC+7.0时，仍会显示为高级模式。

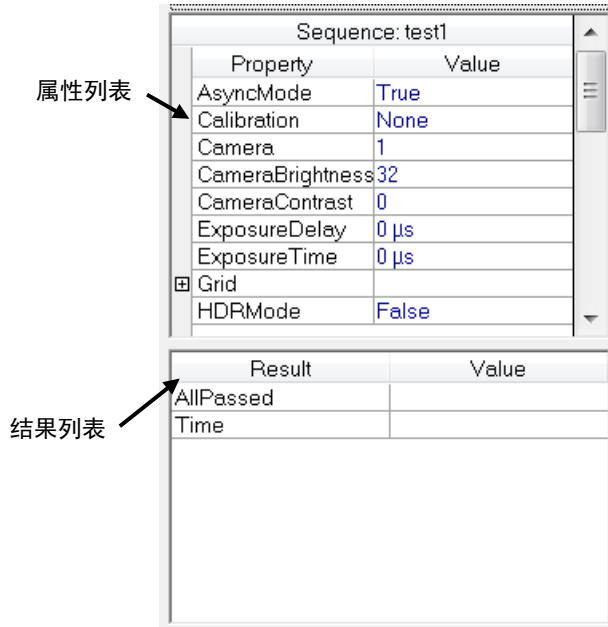


### 4.6.2 序列窗口

可通过点击流程图中的视觉序列或在序列目录树中选择序列，序列窗口将在窗口右侧显示。

序列窗口用于：

- 设置视觉序列的属性。
- 检验整个视觉序列的结果。



#### 序列窗口：属性列表

属性列表中显示了视觉序列的属性，看起来像一个小型的电子数据表。属性名称位于左侧，值位于右侧。

首先点击某一属性的值字段，然后输入一个值或从显示的列表选择一个值，就可以设置视觉属性了。

其中一些视觉序列属性包括：

- Camera**            指定该视觉序列要使用哪台相机。
- RuntimeAcquire**   确定该序列的采集方式  
(例：strobbed, none, stationary)
- Calibration**       确定该视觉序列要使用什么校准。
- ImageBuffer**       规定视觉序列中要使用的缓冲区编号  
0: 每台相机所独有的缓冲区编号(默认)  
1到10: 视觉序列中公用的缓冲区
- ImageSize**        规定视觉序列中要使用的相机分辨率  
默认: 相机分辨率

有关视觉序列属性的详细信息，请参阅 5. 视觉序列 或 *Vision Guide 7.0 Property and Results Reference Manual*。



**序列窗口：结果列表**

结果列表中显示了视觉序列结果，就位于属性列表下方。结果列表只显示执行视觉序列以后 Value 字段中的值。在执行之前，结果字段不显示任何值。

结果列中显示了以下结果：

**AllPassed**      表示是否所有视觉对象的测试结果都是Pass。

**Time**            返回处理视觉序列所需的时间。

### 4.6.3 对象窗口

可通过点击流程图中的视觉对象或在序列目录树中选择对象，对象窗口将显示在窗口右侧。

对象窗口用于：

- 设置视觉对象的属性值。
- 执行视觉对象或视觉序列以后查看结果。



#### 对象窗口：属性列表

属性列表中显示了视觉对象的属性，看起来像一个小型的电子数据表。属性名称位于左侧，值位于右侧。

首先点击某一属性的 Value 字段，然后输入一个值或从显示的列表中选择一个值，就可以设置视觉对象属性了。

视觉对象的属性随视觉对象的类型变化而变化，一些比较常用的视觉对象属性如下：

**AbortSeqOnFail** 当设为True时，如果此对象在执行过程中不合格，则整个视觉序列将中止。

**Accept** 用于与Score结果比较的阈值。  
如果Score大于Accept属性值，则认为已查找到该对象。

**Frame** 确定将哪个视觉框应用于定位该对象。

**PassType** 可定义检测结果良品/不良品的判断标准。

有关视觉对象属性的详细信息，请参阅本手册 6. 视觉对象 以及 *Vision Guide 7.0 Properties and Results Reference Manual*。

**对象窗口：结果列表**

结果列表中显示了视觉对象结果，就位于属性列表下方。结果列表显示执行了该视觉对象或整个视觉序列以后 Value 字段中的值。在执行之前，某一具体视觉对象的结果列表可能不会显示任何值，或显示上次运行视觉对象时产生的值。

在运行整个视觉序列时，这对于显示 Object 窗口很有用。允许您在运行该序列时查看某个对象的结果。

Vision Guide 7.0 使您可以运行一个视觉序列然后在视觉对象之间切换，查看每个独立的视觉对象的结果。您必须使用视觉对象下拉列表在视觉对象之间选择。如果在图像显示中点击选择了一个视觉对象，则会清除对象的结果。

Vision Guide 7.0 使这些结果在视觉对象之间切换时也保持可用。

结果随视觉对象类型不同而不同。一般视觉对象结果如下所示：

<b>Found</b>	显示是否查找到了视觉对象。
<b>Passed</b>	返回测试结果合格与否。。
<b>Time</b>	返回处理视觉对象所需的时间。

右键点击结果列表，可以对当前活动的视觉对象，进行以下操作。

<b>Copy all results</b>	将所有结果复制到剪切板。
<b>Export all results</b>	将所有结果导出到CSV文件中。 可以以表格格式保存，在处理返回多结果的视觉对象时很实用。

有关示教视觉对象的详细信息，请参阅 6. 视觉对象。

执行面板中对象的<Run>按钮用于运行当前选择的视觉对象。通过操作对象的<Run>按钮，当前视觉对象会检查自身属性的设置是否正确。

如果视觉对象与其他视觉对象关联(例如，中心位置由 Correlation 对象结果定义的 Polar 对象)，则将先执行定义的视觉对象，然后再执行当前视觉对象。

例如，假设您想运行 Polar 对象，而该 Polar 对象的 CenterPointObject 属性由 Correlation 对象的结果定义。在这种情况下，从流程图将中选择 Polar 对象选为当前对象。然后，点击<Run Object>按钮，Correlation 对象就会先执行，紧接着才是 Polar 对象。

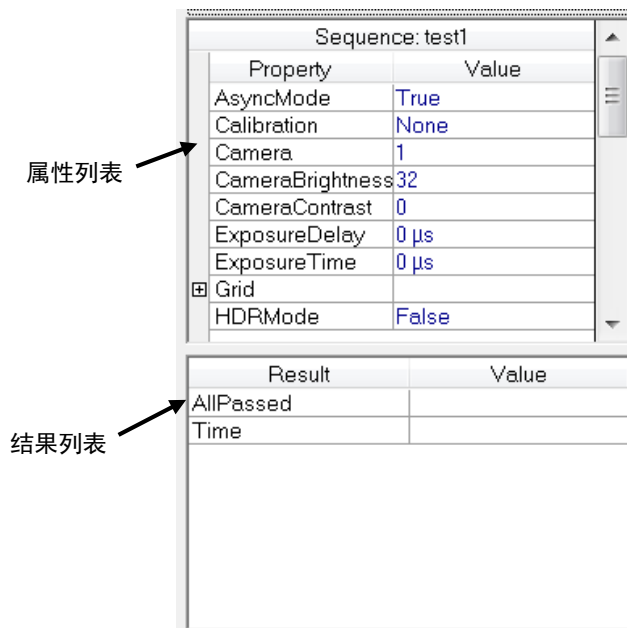
**NOTE**

请充分理解执行视觉对象和执行视觉序列之间的区别。要运行一个视觉对象，只需点击执行面板中对象的<Run>按钮。

运行前，务必确保在流程图或序列目录树中选中了正确的视觉对象。运行视觉对象，只会执行该视觉对象以及该对象所需的视觉对象(例如，定义Line对象StartingPoint属性的其他对象，或定义Polar对象CenterPointObject属性的其他对象等)。而在执行视觉序列时，将执行该序列中的所有视觉对象。

## 4.6.4 校准窗口

可通过在序列目录树中选择校准，将校准窗口显示在窗口右侧。  
校准窗口用于设置校准，查看校准结果。



## 校准窗口：属性列表

属性列表中显示了视觉 Calibration 的属性，看起来像一个小型的电子数据表。属性名称位于左侧，值位于右侧。首先点击某一属性的 Value 字段，然后输入一个值或从显示的列表中选择一个值，就可以设置属性了。

有关 Calibration 属性的信息，请参阅 4.9 校准或 *Vision Guide 7.0 Property and Results Reference Manual*。

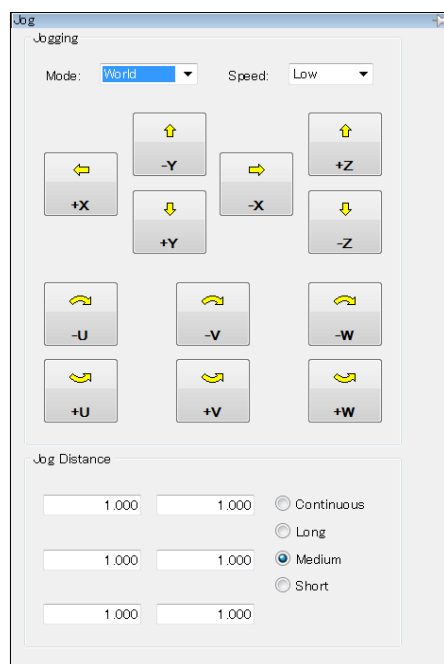
## 校准窗口：结果列表

结果列表中显示了视觉校准结果，就位于属性列表下方。

有关相机校准的详细信息请参阅 4.9 校准。

### 4.6.5 步进选项卡

步进选项卡用于在校准时或显示实时图像时候控制机器人步进。



可通过点击序列和校准目录树右侧的<Jog>按钮显示步进选项卡。步进选项卡也是可以放置于任意位置的弹出式面板。

选择[Jog]选项卡开始与控制器通信。如果通信失败，则显示屏将返回上级选择的选项卡。

开始步进之前，必须打开机器人电机。您可以从机器人管理器或者命令窗口打开电机。

#### 步进选项卡：设置步进模式

从[Mode]选项框中设置步进模式。可选择的有World、Tool、Local、Joint和ECP(如果激活了ECP)。您可以在机器人管理器中选择当前Tool、Local和ECP。

#### 步进选项卡：设置步进速度

从[Speed]选项框中设置步进速度。

#### 步进选项卡：设置步进距离

点击[Jog Distance]组中的选项按钮设置步进距离。选择了Long、Medium、Short距离以后，您可以改变距离，只需在步进距离文本框里键入。

#### Jog选项卡：控制机器人步进

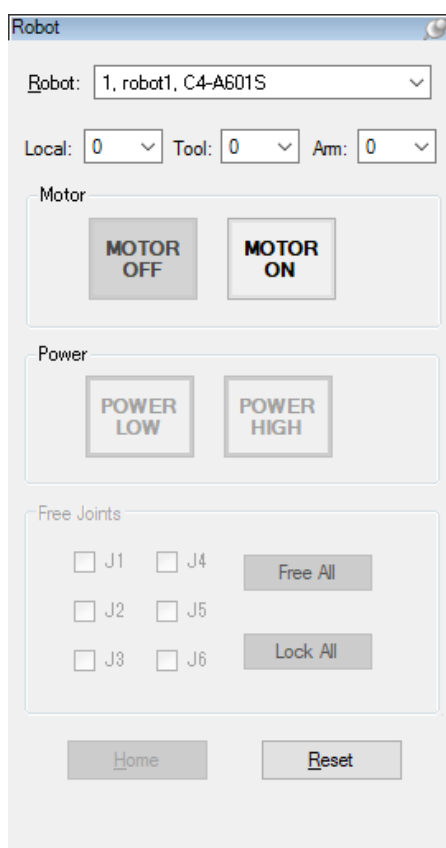
设置完步进模式、步进速度和步进距离以后，点击步进按钮让机器人开始步进。如果将步进距离设置为Continuous，按下按钮期间机器人会一直运动。对于其他距离，机器人将根据规定的步进距离进行运动。如果连续点击按钮，则机器人会一直保持运动。

### 4.6.6 机器人选项卡

可通过点击序列和校准目录树右侧 **Jog** 选项卡下方的 **<Robot>**按钮显示 **Robot** 选项卡。**Robot** 选项卡也是可以放置于任意位置的弹出式面板。

**Robot** 选项卡用于设置各种功能，让机器人步进时可能用到这些功能。您可以进行以下操作：

- 选择当前机器人
- 打开或关闭电机电源
- 更改电源
- 设置关节
- 执行 Home 或 MCal
- 执行 Reset



## 4.7 视觉序列和对象

### 4.7.1 视觉序列概述

本节介绍视觉序列相关的基础知识和使用方法。

- 视觉序列定义和基础知识
- 序列窗口
- 创建和删除视觉序列
- 视觉序列的属性和结果
- 从 Vision Guide 窗口运行视觉序列
- 视觉序列的测试和调试
- 如何通过 SPEL+语言运行视觉序列
- 图像采集(从视觉序列捕获图像的方法)

### 4.7.2 视觉序列的定义和基础知识

#### 视觉序列的概念

视觉序列是按照特定的顺序对视觉对象进行分组，并可以从 Vision Guide 窗口或 SPEL+语言中执行。

和视觉对象一样，视觉序列有特定的属性，用于设置序列执行。例如，Camera 属性定义了要使用哪台相机采集该视觉序列的图像，RuntimeAcquire 属性则定义了如何从相机采集图像。

虽然视觉对象可以单独运行和测试，但大多数视觉应用都需要一系列视觉对象，按顺序运行并计算最终结果。

视觉序列用于按照特定顺序运行视觉对象。创建并测试单个视觉对象后，测试视觉序列。

#### 视觉序列的工作原理

视觉序列包含一个或多个视觉对象。视觉序列的主要目的是按照规定的顺序，依次执行这些视觉对象。

以下步骤解释了运行视觉序列的结果。

1. 要设置视觉序列的执行，请先设置其属性。这些属性包括使用哪台相机采集图像，以及用什么方法采集图像(Strobed、Stationary或No acquire at all)。
2. 图像采集后，储存在帧缓冲区中，以供视觉对象使用。(如果将RuntimeAcquire 属性设置为“None”时，则使用先前存储在帧缓冲器中的图像执行该序列。这可以将同一图像用于多个序列。
3. 使用步骤2中采集的图像，按顺序依次执行视觉对象。执行完所有视觉对象以后，就可以在序列窗口和对象窗口中显示序列和对象结果，也可以通过SPEL+语言显示。

### 视觉序列和EPSON RC+ 7.0 项目

EPSON RC+ 7.0 是基于项目管理的理念设计的软件。

每个 EPSON RC+ 7.0 项目都包含特定的应用时所需的程序、示教点和其他机器人配置参数。

如果在 EPSON RC+ 7.0 项目中使用 Vision Guide 7.0，则在该项目中创建的所有视觉序列，都将被保存到项目中。

这使得用户在下一次打开某个项目时，该项目中之前创建的所有视觉序列都可以直接使用或进行修改。

### 4.7.3 视觉对象概述

视觉对象是所有工作的核心，并且似乎Vision Guide 7.0的主要组成部分。

本章介绍了应用Vision Guide 7.0所需的基本知识和详细信息，并介绍了如何Vision Guide 7.0实现视觉应用。

- 视觉对象的定义以和功能。
- 解释不同类型的视觉对象的形状。
- 如何定位视觉对象并确定大小。
- 视觉对象的用途。
- 如何使用视觉对象。
- 视觉对象的属性和结果。
- 视觉对象的具体说明。
- 视觉对象的实用工具  
(直方图、统计、视觉对象图形处理)。

### 4.7.4 视觉对象定义

#### 什么是视觉对象？

视觉对象是可以应用于图形图像的视觉工具。

第一次看见视觉对象时很多视觉开发者会认为，视觉对象只不过是支持 Vision Guide 7.0 的算法的别称。但是，使用过 Vision Guide 7.0 以后，您会发现视觉算法只是视觉对象定义的一部分。

视觉对象就像容器，里面包含了使用具体视觉算法所需的信息。但是，它是一个非常强大的工具，通过独特的方式(属性和结果)，用户仅需对数据进行简单的处理和测试，即可完成视觉应用的开发，相较于其他视觉系统来说效率更高。

视觉对象支持的工具有：Correlation Search、Blob Analysis、Polar Search、Edge Detection、Line Creation、Points 和 Frames 等。所有视觉对象执行后都会返回结果，例如执行视觉对象所需的时间、位置信息、角度信息，是否检测到视觉对象等。

更多详细信息，请参阅每个视觉对象的描述。



## 4.8 视觉序列的属性和结果

视觉序列与视觉对象一样，也有属性和结果。主要区别在于，视觉序列的属性和结果适用于整个视觉序列，而视觉对象的属性设置只适用于一个视觉对象。

### 4.8.1 视觉序列的属性

视觉序列的属性通常用于设置视觉序列执行，设置正确的图像采集值，确保视觉序列中的所有视觉对象都可以正常运作。

视觉序列的所有属性都适用于整个视觉序列。例如，**Calibration** 属性设置在视觉序列级别而不是视觉对象级别。

这样，视觉序列中的所有视觉对象都使用 **Calibration** 属性所规定的校准，可以保证结果都是根据同一个校准值得出的。

下面的列表是对视觉序列属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
<b>Calibration</b>	指定视觉序列中使用的相机校准。 必须指定本属性，以返回机器人坐标系和相机坐标系中的结果。 默认: none
<b>Camera</b>	指定要在视觉序列中使用的相机。 默认: 1
<b>CameraBrightness</b>	指定亮度。 默认: 128
<b>CameraContrast</b>	指定对比度值。 默认: 128
<b>Description</b>	设定用户指定的描述 默认: 空白
<b>ExposureDelay</b>	指定曝光前的延迟时间。 (单位: 微秒) 默认: 0
<b>ExposureTime</b>	指定在异步复位模式中使用时的曝光时间。(单位: 微秒) 默认: 0
<b>ImageBuffer</b>	指定采集的图像存放在哪个缓冲区内。 默认: 0
<b>ImageColor</b>	指定要获取的色相。 默认: 1 - All
<b>ImageFile</b>	指定文件名，文件中包含磁盘上您想要处理的图像。 默认: None(没有图像文件)

属性	说明
ImageFileScale	指定图像文件的放大倍数。 默认：0(适应图像尺寸)
ImageSize	指定采集图像的分辨率。 默认：当前相机的最大分辨率。
ImageSource	指定图像源。 默认：1 - Camera
Index	显示序列的索引。
HDRMode	将拍摄的图像显示为HDR图像。 默认：False
Name	视觉序列的名称。
RuntimeAcquire	定义如何采集视觉序列中使用的图像。 默认：1 - Stationary
RuntimeFreeze	让视觉序列定格图像显示。 默认：True
SaveImage	显示一个对话框，将当前图像保存至磁盘。
ShowProcessing	规定是否显示图像处理。 默认：True
StrobeBlackVideo	指定在序列开始之后，是否在受到触发接收图像之前，将图像恢复为黑色。 默认：True(将图像恢复为黑色)
StrobeDelay	指定闪光灯闪光之前的延迟时间。(单位：微秒) 默认：0
StrobeTime	指定闪光灯的闪光时间。(单位：微秒) 默认：0
TriggerMode	指定电子快门的触发模式。 默认：Leading Edge
GridColor	指定图像显示区域中显示的网格的颜色。
GridPitchX	指定图像显示区域中显示的网格X间距。
GridPitchY	指定图像显示区域中显示的网格Y间距。
GridShow	指定是否在图像显示区域中显示网格。 默认：False
GridType	指定图像显示区域中显示的网格类型。 1 - Cross Hair 2 - Rectangle
GridUnits	指定图像显示区域中显示的网格间距单位。 1 - Pixel 2 - MM 选择“2 - MM”时，如果未执行校准，网格不显示。

### 4.8.2 视觉序列的结果

视觉序列结果适用于整个视觉序列。

这些结果很有用，因为这些结果可以显示有关整个视觉序列的信息。例如，**AllPassed** 结果返回是否查找到视觉序列里的所有视觉对象。

下表是对视觉序列结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
<b>AllPassed</b>	表示是否所有视觉对象的测试结果都是Pass。
<b>Time</b>	返回执行视觉序列所需的总执行时间。 包括执行所有视觉对象的累计时间，以及采集图像所需的时间。

## 4.9 校准

### 4.9.1 校准的概述

校准是Vision Guide 7.0系统的重要组成部分。  
要在机器人坐标系中定位各零件，或进行物理测量，您必须对系统进行校准。  
Vision Guide 7.0支持移动相机，固定向上相机，固定向下相机和移动相机的校准。

### 4.9.2 校准的定义

Vision Guide 7.0支持每个项目的多个校准。任何校准都可以在同一项目中的多个视觉序列中使用。

每个校准都包括下列数据：

校准名称	视觉序列参考的校准名称。(最多16个字母数字字符)
相机编号	校准的相机编号。
相机方向	相机安装方法
校准目标序列名称	用于校准相机的视觉序列名称。 (可以是当前项目中的任何序列。)
机器人校准速度和加速度 (可选项)	校准过程中使用的速度和加速度。
校准过程中使用的机器人 手臂编号和工具编号	校准过程中可以使用的机械臂和工具。在示教和校准 点位之前，定义机械臂和工具。
用于基准点和相机位置的 机器人点位	每次校准后将保存这些点位。
光输出	这些是可选标准输出，可用于在校准过程中控制亮 度。
图像失真校正	指定是否执行图像失真(相机倾斜和镜头失真)校正。
图像失真校正目标序列名	用于图像失真校正的视觉序列名称。 (可使用当前项目中的任意序列)

所有校准都使用九个点，覆盖了校准相机的视野。

通过覆盖整个视野，可以在校正系统的相机倾斜和镜头失真时，获得最佳结果。

在整个视野执行距离测量时，执行图像失真校正。通过执行图像失真校正，可校正  
相机倾斜(工作平面和相机光轴的角度偏差)和镜头失真。

#### 何时应该进行校准？

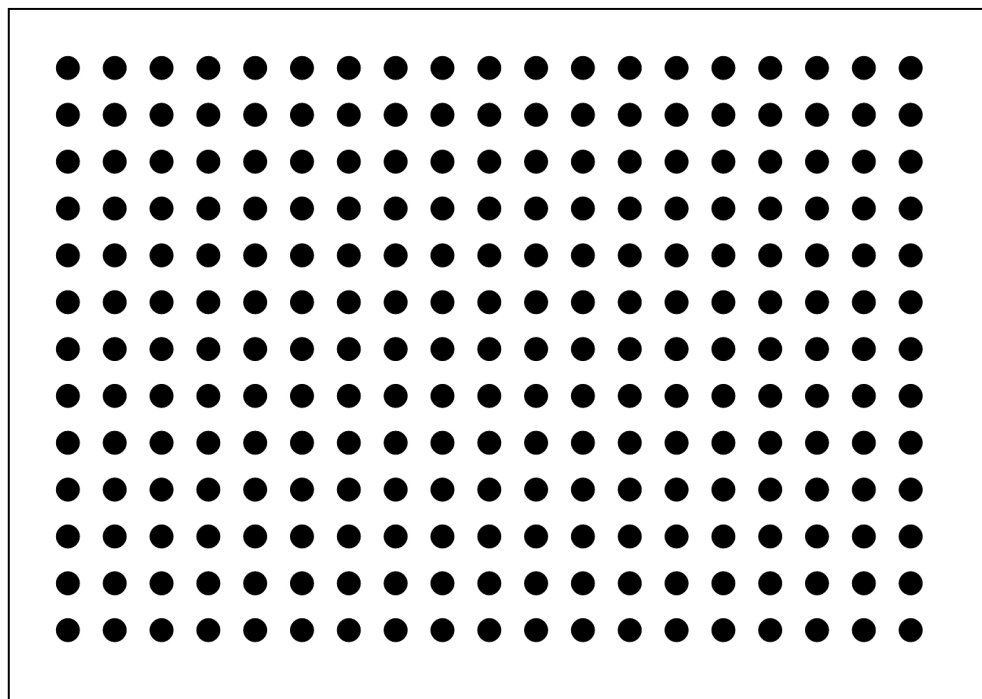
当需要进行物理测量或者确定机器人坐标系的位置时，都需要进行校准。

视觉对象中返回的部分结果也需要校准。例如，CameraX、CameraY、RobotX、  
RobotY和RobotU结果，都需要校准。

### 4.9.3 用于图像失真校正的成像图案

以下为图像失真校正使用的图像示例。

要执行图像失真校正，网格在水平和垂直线上必须具有相同的间距，且在整个相机视野中必须能看到多于100个的网格。

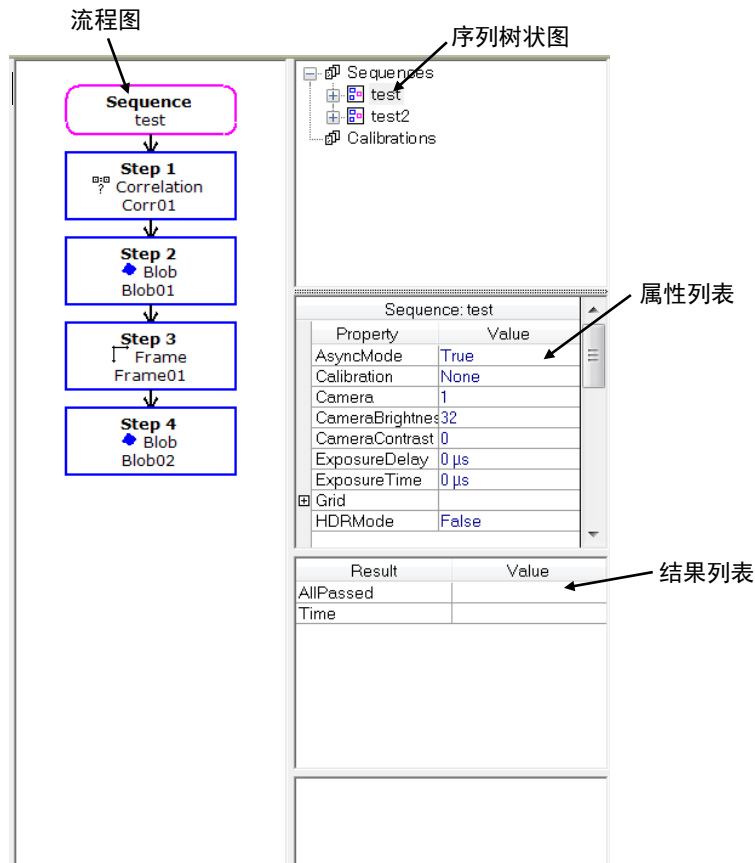


## 5. 视觉序列

### 5.1 视觉序列窗口

视觉窗口用于：

- 设置视觉序列的属性
- 检验整个视觉序列执行的结果



### 5.1.1 选择视觉序列

点击序列树状图左侧的“+”键，显示该项目所有视觉序列的列表。您可以点击视觉序列中的任意一个，选择您想要操作的序列。

### 5.1.2 序列窗口属性列表

属性列表中显示了视觉序列的属性，看起来像一个小型的电子数据表。属性名称位于左侧，设置值位于右侧。

点击特定属性的设置字段并在字段中输入值，或从自带的列表选择一个值，来设置视觉属性。

有关视觉序列属性的详细信息，请参阅 [4.8.1 视觉序列的属性](#)。

### 5.1.3 序列窗口结果列表

视觉序列的结果显示在，属性列表正下方的结果列表中。

在结果列表中，除非运行视觉序列，否则结果值得字段不会显示任何值。

有关视觉序列结果的详细信息，请参阅 [4.8.2 视觉序列的结果](#)。

### 5.1.4 序列流程图


序列流程图显示再图像显示区域得右侧。该流程图显示了视觉序列中已定义的所有视觉对象。

流程图中显示得顺序，即为执行视觉对象的顺序。

有关详细信息，请参阅 [5.5 改变序列顺序](#)。


## 5.2 新建一个视觉序列

### 5.2.1 概述

创建视觉序列。点击 Vision Guide 工具栏  <New Sequence>按钮。也可以通过右键点击流程图中的序列流程，或序列目录树中的序列节点后选择<New Sequence>，打开对话框。键入一个不超过 16 个字母的名称然后点<OK>。

在新建序列时，还可以在已有序列下拉列表中，复制一个序列来创建。

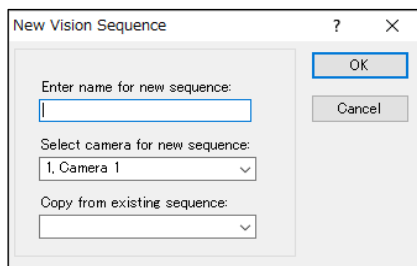
### 5.2.2 快捷键

工具栏 : 

键 : None

### 5.2.3 对话框选项


选项	说明
OK	使用屏幕上输入的序列名称，创建视觉序列。
Cancel	取消创建视觉序列的操作。
Help	打开New Sequence命令的帮助说明。





## 5.3 删除一个视觉序列

### 5.3.1 概述


点击 Vision Guide 工具栏- <Delete Sequence>按钮，通过右键点击流程图中的序列流程或序列目录树中的序列节点后选择<Delete Sequence>，则可以打开对话框，并显示出当前项目中已创建的所有视觉序列。

用鼠标或方向键选择要删除的视觉序列。

将被删除的视觉序列的名称会被突出显示，请点击 Delete Vision Sequence 对话框中的<Delete>按钮。会弹出一个确认消息对话框。

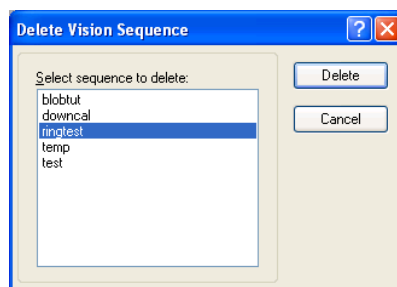
注意，一旦您删除了一个视觉序列并保存了当前项目，则无法恢复序列。如果您误删了一个序列，您可以执行[File]-[Restore]将整个视觉项目恢复为上一次保存的状态。如果您觉得以后您还想使用序列，请务必保存序列。即使将序列保存在项目中，并不代表您必须要使用该序列。

### 5.3.2 快捷键

工具栏 :   
键 : None


### 5.3.3 对话框选项

选项	说明
Selection List	选择要删除的序列。
Delete	删除突出显示的视觉序列。
Cancel	取消删除视觉序列的操作。
Help	打开Delete Sequence 命令的帮助说明。



### 5.4 删除视觉对象

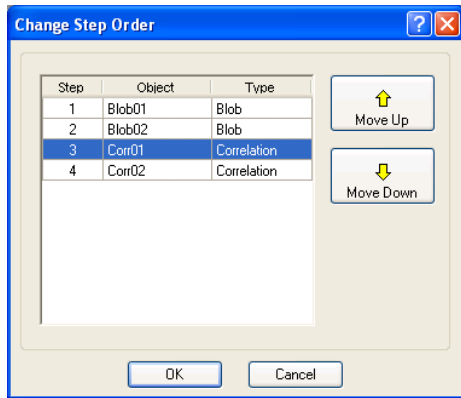
要删除一个视觉对象，首先须选择您想删除的视觉对象。

选中要删除的视觉对象以后，点击 Vision Guide 工具栏-  <Delete Object>按钮。将出现一个确认对话框。点击<Yes>删除对象。

如果当前项目中没有视觉序列，或者如果当前视觉序列中没有视觉对象，则<Delete Object>工具栏按钮呈灰色。

### 5.5 改变序列顺序

要更改序列步骤的顺序，右键点击流程图中的序列流程，或序列目录树中的序列节点后，点击<Change Order>按钮。点击以后会出现一个[Change Step Order]的对话框。要改变对象的步骤号，首先在列表中选中该对象，然后点击Move Up或Move Down按钮。点击<OK>确认更改。



在修改视觉对象的执行顺序时，应特别注意有关联的那些视觉对象。例如Line对象的StartingPoint为“Corr01”，EndPoint为“Blob01”。如果将Line对象的步骤移动到“Corr01”对象之前，即使仍在“Blob01”之后，也将会出现一个警告。提示Line对象的StartingPoint属性将修改为“Screen”。视觉对象之间可能有各种组合或关联，因此处理那些需要与其他视觉对象的结果相关联的对象时，需小心。

## 5.6 运行视觉序列

### 5.6.1 视觉序列选择

运行视觉序列之前，您需要在序列目录树中选择您想要运行的视觉序列。通过序列目录树可以轻松地在项目内切换视觉序列。

### 5.6.2 设置视觉序列的属性

选择了运行视觉序列以后，可能需要您设置序列的一些参数。

视觉序列参数可以在序列窗口的属性列表中设置。在运行视觉序列之前，必须先正确设置属性。例如，如果如需运行序列，该序列使用相机 2 采集的图像，那么就必须将序列的 Camera 属性值设置为 2。

新用户最常犯的错误就是，将 Camera 属性设置为错误的值，并运行视觉序列。

使用多台相机时的重要注意事项：

Camera 属性用于选择视觉序列要使用的相机。系统会自动检测到已启动的相机。如果 Vision Guide 7.0 正在显示某相机的视频，并且打开了该相机的电源，则必须先从序列 Camera 属性中重新选中该相机，然后才能观看视频。

### 5.6.3 视觉序列的结果

您运行了视觉序列以后，可能想查看序列结果。可通过序列窗口上的结果列表查看视觉序列结果。

### 5.6.4 序列流程图

序列的步骤执行列表显示在图像显示区域的右侧。流程图显示当前所选序列中定义的所有视觉对象。

流程图的顺序即为视觉对象的执行顺序。

### 5.6.5 运行视觉序列

Vision Guide窗口底部的运行面板的左侧，则是序列的<Run>按钮。点击该按钮，可以执行整个序列。

创建并测试了序列中的视觉对象后，点击运行面板左侧的<Run>按钮，测试整个视觉序列。

### 5.6.6 多次(循环)运行序列

使用<Run Sequence>按钮时，应特别注意可通过右键点击运行面板上<Run>按钮右侧的▼按钮使用的[Cycles]文本框。

在文本框中输入的数字表示要运行视觉序列多少次。多次运行序列的功能非常方便，用户可以在控制机器人的代码命令之前，先测试视觉序列的可靠性。

多次运行视觉序列，然后使用Vision Guide7.0的统计功能，可以查看序列每个视觉对象的平均值、标准偏差、范围、最小值和最大值。此功能可以帮助您全面的了解视觉序列的执行情况。


如在[Cycles]文本框中输入某个数字，以进行多次运行时，[Cycles]文本框将从白色变为黄色。(在[Cycles]文本框中输入任何大于1的数，文本框颜色都会变黄色。)颜色的变化是为了提示您，按下<Run Sequence>按钮后就将开始执行多个循环。

### 5.6.7 执行视觉序列过程中的图像显示区域

执行视觉序列时，在图像显示区域，以不同的颜色显示检测到的视觉对象，和未检测到的视觉对象。

对于检测到的视觉对象，搜索窗口以及检测到的位置都会显示为绿色。

对于未检测到的视觉对象，则会在搜索窗口中显示红色边框。(由于没有查找到，所以不会显示检测到的位置。)

当Graphics属性设置为None时，不会显示任何图形。为确保所有视觉对象都显示图形，请点击Vision Guide工具栏  <Force All Graphics On >按钮。这可以显示所有图表，无论每个Graphics属性设置是什么。

### 5.6.8 中止视觉序列循环

点击<Abort>按钮将立即停止执行视觉序列。

## 5.7 测试和调试视觉序列

查看视觉序列中是否所有项目都正确运行，请检查视觉序列AllPassed结果的状态。

如果任一视觉对象被检测出不良，则AllPassed视觉序列结果将返回False。另一方面，如果有视觉对象设置不正确，AllPassed结果也可能错误地返回到“accepted”。

最后，测试视觉序列后发现，视觉序列中的某些内容没有正常运行，则需要进行测试。


本章的其余部分介绍了与Vision Guide序列中使用的一些调试功能和方法。

### 5.7.1 检查视觉对象的合格颜色

为了明确问题产生的原因，首先要确认的一点就是，图像显示区域中视觉对象的合格/不合格的颜色。

检测结果为合格的视觉对象，将在搜索窗口中与检测位置一起呈现绿色(可以通过PassColor属性修改颜色)。

不合格的视觉对象的搜索窗口则会呈现为红色(可以通过FailColor属性修改)。如果视觉序列中有视觉对象显示了红色的搜索窗口，这就提醒您需要进一步检查某个视觉对象。

当视觉对象的Graphics属性设置为None，则不会显示任何图形。为确保显示所有视觉对象的图形，点击Vision Guide工具栏 <Force All Graphics On>按钮。这样无论单独的Graphics属性设置如何，都将显示所有图形。

### 5.7.2 检查单个视觉对象结果

运行视觉序列时，可以在序列窗口上显示视觉序列结果，或者在对象窗口上显示视觉对象的结果。

如果您怀疑某一视觉对象存在问题，或者想在运行视觉序列时仔细监控某个对象，您只需要点击几下鼠标，便可轻松实现。

- (1) 从流程图或序列目录树选择视觉对象。
- (2) 检查该视觉对象的结果。

您还可以在运行视觉序列时切换视觉对象，或者在对象窗口和序列窗口之间进行切换。

这使您能开始 1 个或多个循环的视觉序列运行，然后检查该视觉序列中的多个不同视觉对象的结果，而无需终止视觉序列。

### 5.7.3 单步执行视觉序列

Vision Guide 7.0开发环境的另一个重要调试功能之一就是单步执行功能。

视觉序列的运行面板上有一个<Step>按钮。点击该按钮将使视觉序列一次只对一个视觉对象进行单步调试。

在单步调试过程中，您还可以在执行视觉对象之前改变任何视觉对象的属性。

点击<Step>按钮，可以使所有视觉对象处于不活动状态。视觉对象会变成蓝色。再次点击<Step>按钮将执行序列中的第一个视觉对象。如果查找到了对象，则显示为绿色，如果没有则显示为红色。

### 5.7.4 使用视觉统计功能

视觉统计功能是另一种调试工具。

在测试视觉序列的可靠性时，请先将循环数设置为较大的数字，让视觉序列运行一段时间。

然后在Statistics对话框中检查所有视觉对象的统计结果。您可以看到哪些视觉对象执行可靠，哪些的结果不同。


有关视觉统计功能详情，请参阅 [9. 使用Vision Guide统计](#)。

### 5.7.5 在RUN窗口和Vision Guide窗口之间切换

通过多次运行测试，您可能会发现从RUN窗口到运行最终测试的所有准备已经就绪，或者会发现任然需要进行调试。这种情况下，可以在RUN窗口运行时，也可以访问Vision Guide 7.0开发窗口。

我们来看一个例子。

假设您通过Run窗口运行Vision Guide 7.0应用，并且想要检查上次运行时出问题的Correlation对象“pin1”的Score结果。

首先，请点击EPSON RC+ 7.0主工具栏上的 <Vision>按钮。点击后会弹出Vision Guide 7.0开发窗口，且程序会继续运行。

从流程图选择“Pin1”对象。现在您就可以查看名为“Pin1”的对象结果，而无需停止程序运行。

要返回至Run窗口，只需点击EPSON RC+7.0主工具栏上的<Open run window>按钮。

## 5.8 通过SPEL+运行视觉序列

Vision Guide 7.0 的基本设计理念就是，通过 SPEL+语言可以完全访问，通过鼠标指向和点击的操作在 Vision Guide 7.0 开发环境中创建的内容。也就是说，可将 Vision Guide 7.0 开发环境用作原型环境，无需重新编写即可在 SPEL+语言中使用。

因此，如果您在 Vision Guide 7.0 开发环境中创建了一个视觉序列，则可以在 SPEL+ 中通过 VRun 执行该视觉序列。

VRun 的语法就是“VRun seqname”。“seqname”是您想要运行的视觉序列的名称。在下述代码片段中，您将会看到 2 种不同的 VRun 语句，每个语句都执行了一个不同的视觉序列。

“VRun findpin”执行了一个序列来检测销的中心位置，以使机器人拾取该销。

“VRun housing”则执行了一个序列，检测出销钉应被放置在外壳中的位置。

```
VRun findpin
VGet findpin.pincntr.RobotXYU, found, X, Y, U
.
.
VRun housing
VGet housing.setpos.RobotXYU, found, X, Y, U
```

这只是VRun命令用法的几个简单示例。有关如何将SPEL+语言与Vision Guide序列和机器人联用的更多详情，请参阅 11. 在SPEL+中使用Vision Guide。

## 5.9 图像采集

### 5.9.1 何时采集图像？

很多视觉系统都会要求您指定某一命令或步骤，以采集需要处理的图像。Vision Guide 7.0 帮助删除了这一额外的步骤，因为图像在启动视觉序列时就已经采集。

您只需要设置恰当的 RuntimeAcquire 属性值。但是大多数应用中，不需要设置 RuntimeAcquire 属性，因为默认为 Stationary。默认设置在开始执行视觉序列的时候采集图像。除此之外，还有其他一些视觉序列属性用于采集图像。

### 5.9.2 在多个视觉序列中使用相同的图像

如果您想对两个或多个视觉序列使用相同的图像，您只需要将第一个视觉序列的 RuntimeAcquire 属性设置为 Stationary，然后将其他视觉序列的 RuntimeAcquire 属性设置为 None。RuntimeAcquire 属性被设置为 None 的视觉序列，不会采集任何其他图像，而是会对从第一个视觉序列采集的图像进行处理。

### 5.9.3 使用图像缓冲区

每台相机只有一个图像缓冲区(缓冲区 0)，序列之间可以共用 10 个图像缓冲区(缓冲区 1 - 10)。

设置 `ImageBuffer` 序列属性，规定要使用哪个缓冲区。默认缓冲区为 0。您可以使用图像缓冲区捕捉多个图像，并将其存储在内存中，然后再对其进行处理。

例如，您可以创建一个没有对象的序列，可以将图像捕捉进多个缓冲区，然后在其他序列中处理图像。

```
' seq1 没有视觉对象 – 是为了在多个缓冲区中获取图的序列
VSet seq1.RuntimeAcquire, VISION_ACQUIRE_STATIONARY
VSet seq1.ImageBuffer, 1
VRun seq1 ' 将图像加载到缓冲区 1
Go Image2Pos
VSet seq1.ImageBuffer, 2
VRun seq1 ' 将图像加载到缓冲区 2
Go Image3Pos
VSet seq1.ImageBuffer, 3
VRun seq1 ' 将图像加载到缓冲区 3
...
' 开始处理之前获取的图像
VSet seq2.RuntimeAcquire, VISION_ACQUIRE_NONE
VSet seq2.ImageBuffer, 1
VRun seq2 ' 处理缓冲区 1 的图像
VGet seq2.AllPassed, allPassed
...
```

如果序列使用无需采集图像(仅图像处理)的图像缓冲区 1 - 10，则在 `VRun` 过程中，将忽略 `Camera` 属性值。



### 5.9.4 使用外部触发器采集图像

Vision Guide 7.0支持触发输入，使视觉序列通过外部信号获取并处理图像。要使用触发输入的步骤如下。

- (1) 将触发信号连接到相机接口。如果同时使用了闪光灯，可以连接到相机的闪光输出信号。
- (2) 将使用触发输入的视觉序列的 `RuntimeAcquire` 属性设置为“`Strobed`”。
- (3) 在 SPEL+程序中，照常执行 `VRun`，然后等待 `AcquireState` 属性变为值 3，表明图像已获取完成。在本例中，触发信号是从外部装置发送的。

```
Integer state
Boolean found

#define PICTURE_DONE 3
TmReset 0
VRun seq1
Do
  Wait 0.01
  VGet seq1.AcquireState, state
  If Tmr(0) > 10 Then
    Error ER_STROBE_OT ' User defined overtime error
  EndIf
Loop Until state = PICTURE_DONE
VGet seq1.obj1.Found, found
```

如果您没有等待图像获取就检查 `AcquireState`，则任务中的下一视觉命令将在执行之前自觉等待图像采集。在这种情况下，必须在继续处理图像之前采集图像，否则，您必须中止任务。建议您检查 `AcquireState`，以便在触发器未触发且没有采集图像时程序可以继续运行。

使用来自 Vision Guide 7.0 GUI 的触发输入，执行序列时，系统会等待直到触发输入。点击<Abort>按钮中止触发输入等待。



注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。

### 5.9.5 使用彩色图像

如果配置了彩色相机则可以采集彩色图像，并从磁盘读取图像。

部分视觉工具可以处理彩色图像，其他工具只处理黑白图像，如下表所示。

视觉工具	彩色处理	黑白处理
Blob		×
Correlation		×
Geometric		×
Edge		×
Polar		×
Code Reader		×
OCR		×
ImageOp	×	×
ColorMatch	×	×
LineFinder		×
LineInspector		×
ArcFinder		×
ArcInspector		×
DefectFinder		×
BoxFinder		×
CornerFinder		×
Contour		×

ImageOp工具具有ColorFilter和ColorStretch操作功能，可以处理彩色图像。所有其他ImageOp操作功能都只能处理黑白图像。

ColorMatch工具通常用于处理彩色图像。但是，也可用于处理黑白图像(识别不同灰度的颜色)。

采集彩色图像以后，同时会在内部创建一个黑白图像，供那些只能处理黑白图像的工具使用。运行序列时，可执行彩色处理的对象将使用彩色图像，只能执行灰度处理的对象将使用黑白图像。

彩色图像包括红色、绿色和蓝色的色调。使用 ImageColor 序列属性选择需要采集的色调。默认设置为 All，表示将使用所有三种色相带采集全彩图像。

您还可以选择红色、绿色、蓝色或灰度。如果选择了红色、绿色或蓝色，则将从单色图像中获得黑白图像。这使您可以用灰度处理工具搜索一个色带。

当您点击[Vision Guide]窗口上的<Run Object>按钮，如您使用的对象是黑白处理工具，则视频将显示为灰色，正如工具在处理过程中所见的一样。

## 6. 视觉对象

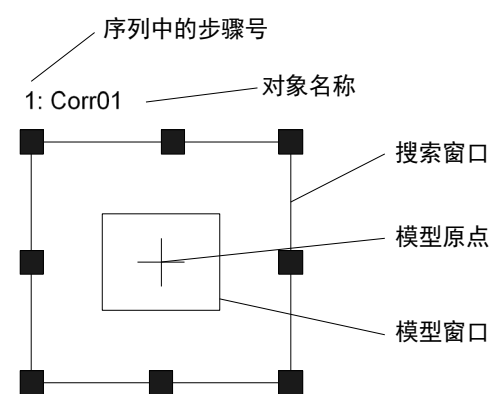
### 6.1 视觉对象的基本事项

在详细了解视觉对象之前，先简单的说明一下视觉对象的布局、以及如何调整视觉对象的位置和大小等基础知识。

创建了一个新的视觉对象并将其放置在图像显示区域里以后，它就具有了一些特有的视觉特性，我们必须理解这些特性。

例如，Correlation 对象会和搜索窗口、模型窗口和模型原点一起显示。这些都可以通过鼠标拖动来调整显示的位置。

以下是基本的视觉对象和其操作方法。



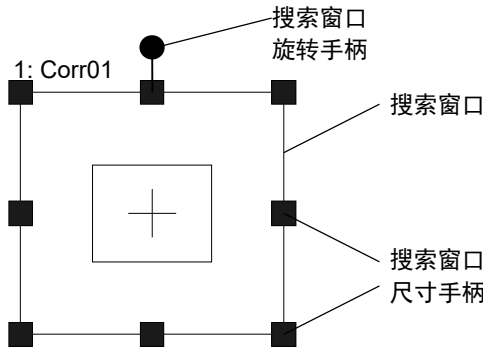
视觉对象布局示例

#### 6.1.1 搜索窗口

大多数视觉对象都有一个搜索窗口，如下所示。搜索窗口就是上图中所示的外框，当处于活动状态(当前正在处理的视觉对象)时为洋红色(浅紫色)；处于非活动状态(是序列中的对象，但不是当前正在进行处理的对象)时为蓝色。在运行时，跟根据是否检测出对象而变成绿色或红色。

搜索窗口用来定义对象区域(检测工件或图案的区域)。

点击视觉对象的名称(或搜索窗口)并将其拖动到图像显示区域中所需的新位置，即可轻松地移动搜索窗口。(图中的“Corr01”是对象的名称。)



在设置视觉对象时，搜索窗口的大小很重要。搜索窗口越大，搜索时间越长。所以搜索窗口应尽可能小，但又要足够大到处理零件的不同变换位置。



将 SearchWinType 属性设置为“RotatedRectangle”或“Circle”，即可指定搜索窗口的角度或设置圆形搜索区域。

使用 6.2.15 Frame 对象 可以帮助减小搜索窗口尺寸。

### 确定搜索窗口的位置

点击视觉对象的名称，然后将视觉对象拖拽到要移动的位置，即可修改搜索窗口的位置。

移动视觉对象时，请确保左上角的位置正确。

### 调整搜索窗口的尺寸

使用搜索窗口的尺寸手柄，调整搜索窗口的大小。当创建视觉对象或单击选择视觉对象时，将会显示搜索窗口的尺寸手柄。

尺寸手柄就是位于搜索窗口边侧和四角的小框。可以拖拽尺寸手柄轻松调整搜索窗口的尺寸。

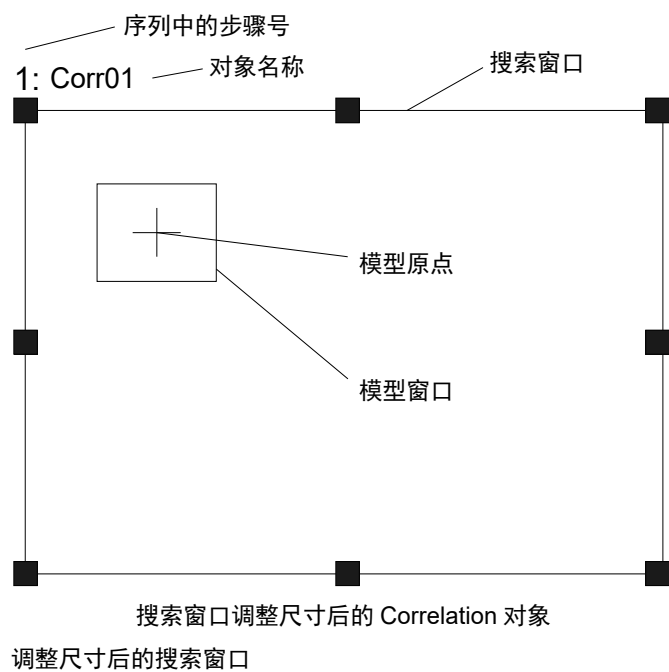
当您将鼠标指针移动到搜索窗口尺寸手柄上，鼠标指针将变成双向箭头。此时请单击鼠标，选中手柄。

您可以点击搜索窗口的任何一个尺寸手柄，以拖动边缘或四角来调整搜索窗口的尺寸。

点击搜索窗口左右两侧的尺寸手柄，即可水平调整视觉对象的大小。

点击搜索窗口上下两边的尺寸手柄，即可垂直调整视觉对象的大小。

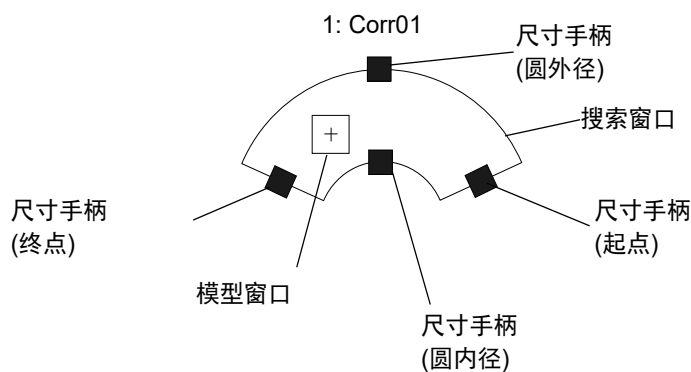
点击搜索窗口四角上的尺寸手柄，即可同时水平和垂直调整搜索窗口的大小。



使用SearchWinType属性设置搜索窗口的形状。

除了“Rectangle”、“RotatedRectangle”和“Circle”，还可以设置为“Arc”或“Polygon”。设置为“Arc”或“Polygon”，可以用尺寸手柄调整的属性，和可以使用的属性，与其他三种设置不同。

设置为“Arc”，可以设置弧形搜索区域。



### 确定搜索窗口的位置

点击视觉对象的名称，然后将视觉对象拖拽到要移动的位置，即可修改搜索窗口的位置。

移动视觉对象时，请确保中心点的位置正确。

### 调整搜索窗口的尺寸

使用搜索窗口的尺寸手柄，调整搜索窗口的大小。在“Arc”类型的搜索窗口中，搜索窗口的尺寸手柄是一个小框，显示在搜索窗口圆弧的外径和内径的中心，以及弧形的起点和终点。拖拽尺寸手柄即可调整搜索窗口的大小。

当您将鼠标指针移动到搜索窗口尺寸手柄上，鼠标指针将变成双向箭头。此时请单击鼠标，选中手柄。

使用圆弧起点和终点的尺寸手柄，可调整搜索窗口圆弧的长度。

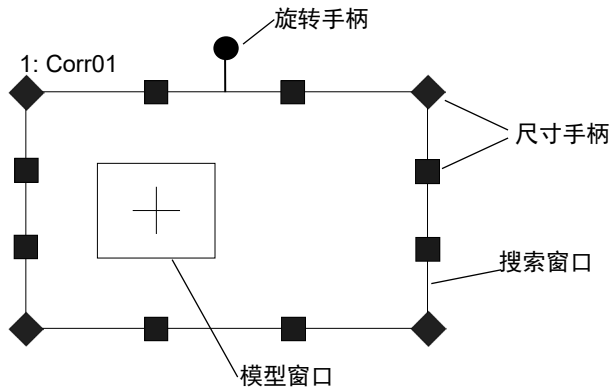
使用圆外径和圆内径中心的尺寸手柄，可以调整搜索窗口圆弧的宽度。



“Arc”类型的搜索窗口，可以使用“Rectangle”，“RotatedRectangle”，“Circle”的搜索窗口形状，无法使用SearchWinLeft属性、SearchWinTop属性、SearchWinHeight属性和SearchWinWidth属性。

取代这些属性的是，SearchWinAngleStartt属性、SearchWinAngleEnd属性、SearchWinRadiusInner属性和SearchWinRadiusOuter属性。也可以使用SearchWinCenterX属性和SearchWinCenterY属性，所以可以调整这6个属性来设置圆弧的位置和大小。

可以将SearchWinType设置为“Polygon”，来设置12边形的搜索区域。它比其他4中类型具有更多属性，可以表现高度自由的形状。



### 定位搜索窗口

请单击视觉对象名称，然后将视觉对象拖拽到您需要设置的何位置。

当移动视觉对象时，请注意确保中心点位于正确位置。

### 调整搜索窗口的大小

可使用搜索窗口的尺寸手柄调整大小。在“Polygon”型的搜索窗口中，尺寸手柄是一个小框，位于12边形的每个顶点处，可以移动手柄自由调整搜索窗口的大小和形状。当鼠标的指针移动到搜索窗口的尺寸手柄上时，指针会变为双箭头。此时点击鼠标可选中手柄。

使用“Rectangle”和“RotatedRectangle”型的搜索窗口，可以使用尺寸手柄调整搜索窗口的宽度、高度或两者。当您移动尺寸手柄时，其他手柄也会根据操作同时移动。但是在“Polygon”型的搜索窗口中，只有选定的手柄移动，其他手柄不动。而是用高度自由的调整搜索区域的形状。



可以在“Rectangle”，“RotatedRectangle”和“Circle”型的搜索窗口形状中使用的SearchWinLeft属性和SearchWinTop属性，不能在“Polygon”型的搜索窗口中使用。

取而代之，您可以使用SearchWinPolygonPointX1属性, SearchWinPolygonPointY1属性 ~ SearchWinPolygonPointX12属性和SearchWinPolygonPointY12属性。

还可以使用SearchWinAngle属性, SearchWinCenterX属性, SearchWinCenterY属性, SearchWinHeight属性和SearchWinWidth属性，可以调整一共29个属性来表示搜索区域的位置、大小和形状。



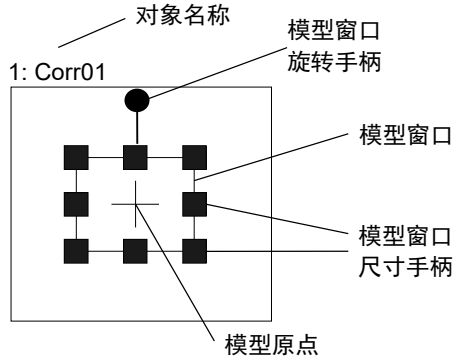
注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。

### 6.1.2 模型窗口

Correlation 对象、Geometric 对象和 Polar 对象有模型窗口和搜索窗口。本节中将使用 Correlation 对象模型窗口进行说明。

Correlation 对象的模型窗口如下图所示。下图中的内框即为模型窗口，颜色为洋红色。(在运行时，若检测到对象则颜色会变为绿色。)模型窗口用于规定模型的边界。(模型是指，被搜索对象的理想状态。通常是您要搜索的对象。)



模型窗口

#### 确定模型窗口的位置

设置模型的尺寸和位置对于示教新模型是至关重要的，因此必须理解如何准确地设置模型窗口的位置和调整模型窗口的尺寸。

若要移动模型窗口，请点击并拖动组成模型窗口的四条侧边之一。建议您先将模型窗口的左上角移动到适当的位置，然后使用模型窗口尺寸手柄调整窗口的大小。

与搜索窗口一样，在设置视觉对象时，模型窗口的尺寸也很重要。通常，当搜索窗口的大小相同，模型越小，搜索时间越长。模型窗口的大小需满足能够处理零件的各种变换的同时，设置的越小越好，但同时也需要尽可能缩小搜索窗口的尺寸。

#### 提示：

将 ModelWinType 属性设置为 “RotatedRectangle” 或 “Circle”，即可确定模型窗口的角度或使用圆形模型窗口。

#### 调整模型窗口的尺寸

选择模型窗口时，会显示模型窗口的尺寸手柄。

但是，如果选定了搜索窗口，则无法显示模型窗口的尺寸手柄。如果手柄未显示在模型窗口中，则说明搜索窗口可能是活动窗口，此时请点击模型窗口的边缘，以切换至模型窗口。

模型窗口的尺寸手柄就是位于模型窗口边侧和四角的小框。可以拖拽尺寸手柄轻松调整模型窗口的尺寸。当您将鼠标光标移到模型窗口的尺寸手柄上时，光标就会变成一个双向箭头光标。此时单击鼠标即可选中手柄。

您可以点击模型窗口的任何一个尺寸手柄，以拖动边缘或四角来调整模型窗口的尺寸。

点击模型窗口左右两侧的尺寸手柄，即可水平调整视觉对象的大小。

点击模型窗口上下两边的尺寸手柄，即可垂直调整视觉对象的大小。

点击模型窗口四角上的尺寸手柄，即可同时水平和垂直调整模型窗口的大小。

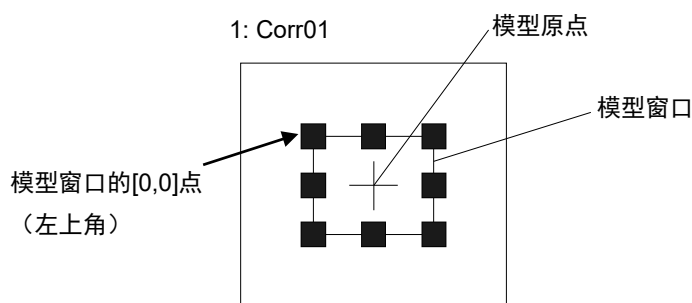


### 6.1.3 模型原点

每个模型都有一个模型原点。模型原点就是一个固定基准点，指示图像中模型的位置。

请注意，模型原点的坐标位置是以模型窗口左上角为基准来表示的。

模型窗口的左上角的起始位置为[0,0]。



模型原点

#### 确定模型原点的位置

可以自动或手动设置模型原点的位置。

若要自动设定模型原点的位置，请将目标对象的 `ModelOrgAutoCenter` 属性设置为“True”。在这种情况下，模型原点会自动设置到模型窗口的中心位置。

NOTE



`ModelOrgAutoCenter` 的默认设置为“True”。

若要以手动设置模型原点的位置，请将 `ModelOrgAutoCenter` 属性设置为“False”。可通过目标对象的对象窗口的属性列表中进行设置。在属性列表中设置属性时，亚像素可用作设置值。

NOTE



即使使用亚像素，图像中的模型原点也用像素表示。

若要移动模型原点，则将目标对象的模型窗口设置为活动窗口。如果搜索窗口处于活动状态，则无法移动模型原点。此时请点击模型窗口的任一侧边，以显示模型窗口的尺寸手柄。

如果 `ModelOrgAutoCenter` 属性设置为“False”，则可利用鼠标移动模型原点。若要移动模型原点，则点击模型原点的十字光标，并将其拖动到某个新的位置。

NOTE



利用鼠标移动了模型原点时，模型原点将设为用像素表示。要以亚像素设置模型原点，请在属性列表中进行设置。

在当前正在处理的模型中，找到一个合适的点，并将其设置为模型原点，将会十分便利。例如，将模型原点设置为机器人抓取零件的位置。

## 6.2 使用视觉对象

本节说明了视觉对象的详细内容，例如各种视觉对象布局，以及操作方法之间的区别。



- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。  
当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。  
请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。

### 6.2.1 ImageOp对象

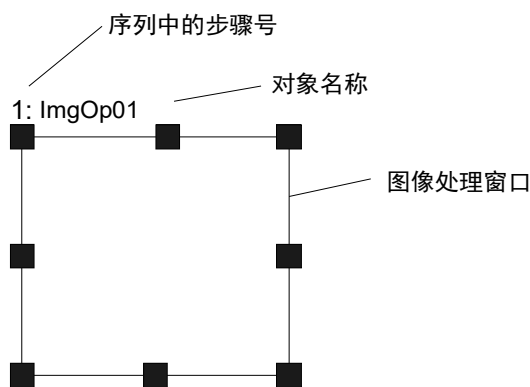
#### ImageOp对象描述

ImageOp对象可以对指定的区域，进行形态调整(开、闭、侵蚀、膨胀)、卷积(钝化、平滑)、翻转、二进制化以及旋转。

ImageOp 区域内的其他视觉对象，在 ImageOp 的输出结果上执行其各自的操作。例如，您可以使用 ImageOp 工具对整个视频画面进行收缩操作，然后将 Blob 对象放置在 ImageOp 的搜索窗口内，以搜索缩小的图像。

#### ImageOp对象布局

Image象有一个图像处理窗口，如下所示。



ImageOp对象布局

## ImageOp对象的属性

以下列表汇总了ImageOp对象的属性，并附有简要说明。有关每条属性的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual*。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序
AngleObject	规定了要执行自动旋转的那个对象。使用指定对象的Angle结果可以逆时针方向旋转图像。请将Operation属性设置为Rotation，否则这条属性会被忽略。 默认：None
AngleObjectResult	用于指定使用AngleObject属性时的结果
Caption	为ImageOp对象指定标题。 默认：Empty String
ColorMode	设置颜色处理时使用的颜色空间(RGB/HSV)。 默认：RGB
CurrentModel	仅限运行时间。 若 Operation 设定为 ColorFilter，则此属性规定了 ModelColor 属性和 VTeach 将要使用的哪个模型。若 CurrentModel=0，则背景颜色已选定。 默认：0
Description	设定用户指定的描述 默认：空白
EditWindow	定义被检测区域的不要紧的像素
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。
Frame	指定要使用哪个位置的定位框。 默认：None
FillHoles	指定是否填充二进制图像的孔洞。 仅在Operation设为Binarize时显示该属性。 默认：False
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
ImageBuffer1	规定了Operation属性设定为SubtractAbs时，按缓冲器编号指定第一个图像。 SubtractAbs计算出(第一个图像 - 第二个图像)的绝对值图像。

属性	说明
ImageBuffer1File	当ImageBuffer1设置为“File”时，指定要存放缓冲图像文件的路径 默认: None
ImageBuffer2	规定了Operation属性设定为SubtractAbs时，按缓冲器编号指定第二个图像。
ImageBuffer2File	当ImageBuffer2设置为“File”时，指定要存放缓冲图像文件的路径 默认: None
KeepRGBRatio	指定是否保持ColorStretch操作的RGB比例。 默认: True
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
MaxRGB	设置ColorStretch处理时的颜色最大值。 默认: 255, 255, 255
MinRGB	设置ColorStretch处理时的颜色最小值。 默认: 0, 0, 0
ModelColor	仅限运行时间。 当 Operation 设定为 ColorFilter 时，在运行过程中，直接输入 RGB 颜色值，以示教模型和背景颜色。 默认: RGB(0, 0, 0)
ModelColorTol	仅限运行时间。 设置模型颜色的容差量。 如果像素颜色在模型颜色的容差范围以内，那么像素会保持不变。 默认: 10 (ColorMode = RGB); 0,0,50 (ColorMode = HSV)
ModelName	仅限运行时间。 设置当前正在使用的模型的名称。
ModelWin	仅限运行时间。 在一次调用中设置或返回模型窗口的左侧参数、顶部参数、高度参数和宽度参数。
ModelWinHeight	指定模型窗口的高度
ModelWinLeft	指定模型窗口左侧位置
ModelWinTop	指定模型窗口顶部位置
ModelWinWidth	指定模型窗口的宽度
Name	用于为ImageOp对象分配一个唯一的名称。 默认: ImgOp01

属性	说明
NumberOfModels	<p>仅限运行时间。</p> <p>这是所用的颜色模型的数目。</p> <p>在运行时间里，如果设置了该属性，则可以利用 CurrentModel 和 VTeach 示教颜色模型。</p> <p>默认：1</p>
Operation	<p>设置将执行的图像处理类型。</p> <p>Operation属性的详细信息，请参考<i>Vision Guide 7.0 Properties and Results Reference</i></p> <p>默认：Open</p>
PassColor	<p>选择当对象被检测为良品时，搜索窗口显示的颜色。</p> <p>默认：LightGreen</p>
PassType	可定义检测结果良品/不良品的判断标准
Polarity	可定义被检测对象和背景色的差异(在浅色背景下检测深色对象(DarkOnLight)，或在深色背景下检测浅色对象(LightOnDark)。)
RotationAngle	<p>规定操作属性设定为Rotation时图像旋转的度数。</p> <p>默认：0</p>
RotationDirection	可规定旋转时的旋转方向
SearchWin	<p>仅限运行时间。</p> <p>在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。</p>
SearchWinAngle	定义待搜索区域的角度
SearchWinCenterX	定义待搜索区域中心的X轴坐标值
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值
SearchWinHeight	<p>定义待搜索区域的高度(单位：pixel)。</p> <p>默认：100</p>
SearchWinLeft	定义待搜索区域的最左边位置(单位：pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位：pixel)。
SearchWinType	定义待搜索区域的类型(Rectangle, RotatedRectangle, Circle)
SearchWinWidth	<p>定义待搜索区域的宽度(单位：pixel)。</p> <p>默认：100</p>
ShiftObject	设定进行Shift处理的对象
ShiftX	设定X方向的Shift值
ShiftY	设定Y方向的Shift值
ThresholdAuto	可定义是否自动设置被检测对象、背景、图像的边缘灰度的阈值
ThresholdBlockSize	<p>定义了相邻区域的参考范围，以决定当Operation属性设定为BinarizeAdaptive时的阈值。</p> <p>默认：1/16ROI</p>

属性	说明
ThresholdColor	定义阈值范围内像素的颜色。 默认: Black
ThresholdHigh	规定了Operation属性设定为Binarize时所采用的上限阈值。 默认: 128
ThresholdLevel	定义了当Operation属性设定为BinarizeAdaptive时, 与相邻区域的亮度差异的比率。 默认: 15%
ThresholdMethod	设置二值化处理的方法。
ThresholdLow	规定了Operation属性设定为Binarize时所采用的下限阈值。 默认: 0
ZoomFactor	规定了Operation属性设定为Zoom时所采用的缩放值。 默认: 1

### ImageOp对象结果

以下列表汇总了ImageOp对象的结果, 并附有简要说明。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Passed	返回测试结果合格与否。
Time	返回处理对象所需要的时间(单位: 毫秒)。
FocusValue	返回相对对焦等级。 该值达到最小时为最佳对焦。 仅当Operation属性设为DetectFocus时显示该结果。

### 使用ImageOp对象

现在我们已经为理解Vision Guide ImageOp对象的使用方法打下了基础。本节将介绍使用ImageOp对象的每个步骤, 如下所示:


- 创建一个 ImageOp 对象
- 确定搜索窗口的位置和尺寸
- 设置与 ImageOp 对象相关的属性
- 测试 ImageOp 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前, 首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列, 点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列, 请在Vision Guide窗口中单击序列目录树, 选择已创建的视觉序列。

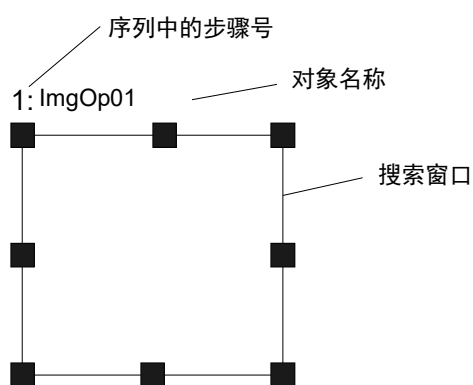
有关如何创建新的视觉序列, 以及如何选择已创建视觉序列的详细咨询, 请参阅 5. 视觉序列。

**第1步：创建一个新的ImageOp对象**

- (1) 点击Vision Guide工具栏中的<All Tools> -  <New ImageOp>按钮。
- (2) 将鼠标移动到图像显示区域。鼠标光标将变为ImageOp对象图标。
- (3) 继续移动鼠标，直至该图标位于图像显示区域中的要求位置，然后再点击鼠标左键，创建该对象。
- (4) 请注意，对象名称将自动创建。本示例中，由于是在本序列中创建的第一个ImageOp对象，所以对象名称为“ImgOp01”。(稍后将说明如何修改对象名称)
- (5)

**第2步：确定搜索窗口的位置和尺寸**

一种类型的ImageOp对象如下所示：

**New ImageOp对象布局**

- (1) 点击ImageOp对象的名称标签，并按住鼠标左键，将ImageOp对象拖动至搜索窗口的左上方的位置后松开鼠标。
- (2) 使用搜索窗口的尺寸手柄调整ImageOp对象搜索窗口的大小。将鼠标光标移动到尺寸手柄上，然后在按住鼠标左键的同时，拖动手柄以调整窗口的尺寸。

### 第3步：设置ImageOp对象相关的属性

现在设置ImageOp对象的属性值。下文说明了ImageOp对象常用的一些专用属性。很多不同视觉对象上也使用的Graphics等其他属性的说明详见 *Vision Guide 7.0 Properties and Results Reference Manual* 或ImageOp属性列表。

#### Name 属性

- 新创建的 ImageOp 对象中，默认的项目名称是“ImgOp\*\*”。在一个视觉序列中要使用多个 ImageOp 对象时，使用\*\*的编号对 ImageOp 进行区分。

默认情况下，会将在视觉序列序列中使用的第一个 ImageOP 对象命名为“ImgOp01”。

需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 ImageOP 对象名称的地方都会变为新的名称。

#### Operation 属性

- 确定将要执行的哪个图像操作。  
这是 ImageOp 对象的最重要的一条属性。

#### Iterations 属性

- 确定重复执行的次数。

#### Polarity 属性

- 有时会需要从浅色背景上检测深色对象(DarkOnLight)，有时会需要从深色背景上检测浅色对象(LightOnDark)。

Polarity 属性的基本目的就是做出这个选择。

默认设置为“DarkOnLight”(深色对象在浅色背景上)。如果要更改此默认设置，请点击 Polarity 属性的值字段，将出现一个下拉列表，包含“DarkOnLight”和“LightOnDark”两个选项。单击您要选择的项目。

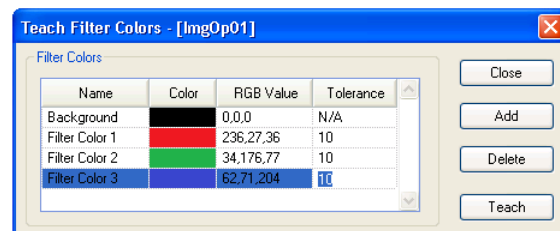
您可以先测试ImageOp对象，然后回头设置稍后需要的其他任何属性。



#### 第4步：示教ImageOp对象的颜色

当 Operation 属性设置为“ColorFilter”时，您就需要示教一种或多种将要过滤的颜色以及背景颜色。Vision Guide 窗口上的<Teach>按钮会变为可用，并有一个矩形模型窗口会出现在 ImageOp 主窗口内。

点击 Teach 按钮，就会显示以下画面。最初的状态下还没有模型。



点击<Add>按钮即可添加一种新的过滤颜色，默认颜色为黑色。选择背景颜色或其中一种过滤颜色之后，可以使用模型窗口内像素的平均颜色，或者您也可以直接输入 RGB 值。

您可以在模型窗口打开时调整其大小和位置。添加颜色或修改大小或位置后，在不关闭对话框的情况下，点击<Teach>按钮，则可示教新的颜色。

您可以在 Teach 对话框打开时更改 ColorMode 属性。

当 ColorMode 为 RGB 时，每种颜色都有一个 Tolerance 值。当 ColorMode 为 HSV 时，每种颜色都有三个 Tolerance 值(hTol、sTol、vTol)。

点击<Delete>按钮删除过滤颜色。背景颜色无法删除。

#### 第5步：测试ImageOp对象

如果要运行 ImageOp 对象，则点击 [Object] 选项卡左下方的 <Run Object> 按钮。您就能看到 ImageOp 工具在图像上产生的效果。

#### 第6步：调整属性并再次测试

ImageOp 对象运行几次后，您可能会遇到一些问题，或需要对一些属性设置进行微调。以下是一些常见问题以及微调的方法。

某些应用中可能需要微调 ImageOp 对象。与 ImageOp 对象的微调相关的主要属性如下。

Iterations - 确定执行目标图像操作的次数。

ThresholdColor, ThresholdLow, ThresholdHigh, ThresholdAuto - 调整 Binarize 操作的参数。

请参阅 *Vision Guide 7.0 Properties and Results Reference Manual* 中“ThresholdColor 属性”、“ThresholdLow 属性”、“ThresholdHigh 属性”、“ThresholdAuto 属性”的说明。

完成属性调整并测试 ImageOp 对象，并得到满意结果后，创建视觉对象的步骤就完成了。您可以继续创建其它视觉对象或设置并测试整个视觉序列。

6.2.2  Geometric对象

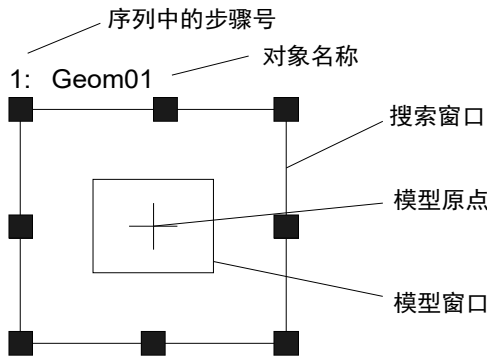
**Geometric对象描述**

Geometric 对象可根据几何特征找到某个模型。该对象基于被检测对象边缘的几何特征搜索模型，而不是利用像素间的关联搜索模型。Geometric 对象相较于关联性搜索有若干优势，例如它对光线变化(包括镜面反射)和比例和角度的变化，以及对于模型具有更高的容差量。

Geometric 对象通常用于通过查找对象的特征确定对象的位置和方向。此方法通常用来检测零件位置，以协助将机器人移动到抓取和放置零件的位置。

**Geometric对象布局**

Geometric对象具有搜索窗口和模型窗口，如下所示。



Geometric对象布局

**Geometric对象属性**

下表给出了Geometric对象属性的基本说明。

欲了解每条属性的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual*。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。如果设定值偏低，可能会导致错误检测。 默认: 700
AngleEnable	规定了Geometric搜索时，是否进行角度(旋转)搜索。 该属性要在Geometric对象的模型示教之前设定。
AngleOffset	指定旋转的偏移值。 默认: 0.000
AngleRange	规定了一系列旋转模型的登录范围。
AngleStart	规定了角度搜索的中心。

属性	说明
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为Geometric对象指定标题。
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则Geometric对象将应用于指定视觉对象的所有结果(NumberFound)。 默认：1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。
CenterPntRotOffset	指定是否旋转以CenterPointObject的Angle结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的XY偏移。 如果将SearchWinType设置为RotatedRectangle，搜索窗口则以Angle结果为基础进行旋转。 默认：False
CheckClearanceFor	设定确认间隙的对象。
ClearanceCondition	设定间隙的判断方法。
Confusion	指出被检测图像的预期混淆量。 此处显示的值是对于被检测对象可获得的最大形状的分，而不是您正在检测的对象。
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认：空白
DetailLevel	选定等级，搜索时某个边线达到该等级即认为被找到。
EditWindow	定义被检测区域的不要紧的像素
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。

属性	说明
Frame	规定了与指定框有关的当前对象搜索位置。 (允许Geometric对象相对于框架进行定位。)
FrameResult	指定需使用Frame的第几个结果。 默认: 1
Graphics	设置显示的图表模式。
LabelBackColor	设置对象标签的背景颜色。
ModelObject	规定使用哪种模型进行搜索。
ModelOrgAutoCenter	模型有固定的基准点, 并且在模型窗口中模型的位置由该参考点表示。该固定基准点被称为模型原点。 ModelOrgAutoCenter属性可将模型原点设置在模型窗口的中心。
ModelOrgFindCenter	将模型原点设于已登记模型的边缘中心。
ModelOrgX	模型原点的X坐标值。(可使用亚像素)
ModelOrgY	模型原点的Y坐标值。(可使用亚像素)
ModelWin	仅限运行时间。 在一次调用中设置或返回模型窗口的左侧参数、顶部参数、高度参数和宽度参数。
ModelWinAngle	指定了模型窗口的角度。
ModelWinCenterX	指定了模型窗口中心点的X坐标值。
ModelWinCenterY	指定了模型窗口中心点的Y坐标值。
ModelWinHeight	指定模型窗口的高度
ModelWinLeft	指定模型窗口左侧位置
ModelWinTop	指定模型窗口顶部位置
ModelWinType	指定了模型窗口类型。
ModelWinWidth	指定模型窗口顶部位置宽度。
Name	用于为Geometric对象分配一个唯一的名称。 默认: Geom01
NumberToFind	规定了在当前搜索窗口中待查找的对象的数目。 (Geometric对象一次可找到不止1个对象。)
PassColor	选择当对象被检测为良品时, 搜索窗口显示的颜色。
PassType	可定义检测结果良品/不良品的判断标准。 默认: SomeFound。
RejectOnEdge	规定了是否排除与搜索窗口边缘有连接的零件。 通常而言, 此项属性宜设置为True。以避免不完全处在搜索窗口内的零件所造成的错误检测。
SaveTeachImage	设置示教模型时, 是否保存图像文件。
ScaleEnable	启用比例缩放。
ScaleFactorMax	设置或返回ScaleTarget值所应用的最大比例系数。

属性	说明
ScaleFactorMin	设置或返回ScaleTarget值所应用的最小比例系数。
ScaleTarget	设置或返回您正在搜索的那个模型的预期比例。
ScaleTargetPriority	设置或返回是否优先检测ScaleTarget附近的对象。
ScoreMode	设置或返回在Fail时显示结果的阈值。
SearchReducedImage	设置或返回搜索时是否使用尺寸缩小的图像。
SearchPolarity	设置或返回搜索Polarity。
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数，或中心X坐标、中心Y坐标、圆内周长的半径大小、圆外周长的半径大小。
SearchWinAngle	定义待搜索区域的角度。
SearchWinAngleEnd	定义待搜索区域的结束角度。
SearchWinAngleStart	定义待搜索区域的开始角度。
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	定义待搜索区域的最左边位置(单位: Pixel)。
SearchWinLeft	定义被检测区域的最左边缘(单位: Pixel)。
SearchWinPolygonPointX1	当SearchWinType设置为“Polygon”时, 定义被检测区域的第1个顶点的X坐标值
SearchWinPolygonPointY1	当SearchWinType设置为“Polygon”时, 定义被检测区域的第1个顶点的Y坐标值
SearchWinPolygonPointX2	当SearchWinType设置为“Polygon”时, 定义被检测区域的第2个顶点的X坐标值
SearchWinPolygonPointY2	当SearchWinType设置为“Polygon”时, 定义被检测区域的第2个顶点的Y坐标值
SearchWinPolygonPointX3	当SearchWinType设置为“Polygon”时, 定义被检测区域的第3个顶点的X坐标值
SearchWinPolygonPointY3	当SearchWinType设置为“Polygon”时, 定义被检测区域的第3个顶点的Y坐标值
SearchWinPolygonPointX4	当SearchWinType设置为“Polygon”时, 定义被检测区域的第4个顶点的X坐标值
SearchWinPolygonPointY4	当SearchWinType设置为“Polygon”时, 定义被检测区域的第4个顶点的Y坐标值
SearchWinPolygonPointX5	当SearchWinType设置为“Polygon”时, 定义被检测区域的第5个顶点的X坐标值
SearchWinPolygonPointY5	当SearchWinType设置为“Polygon”时, 定义被检测区域的第5个顶点的Y坐标值
SearchWinPolygonPointX6	当SearchWinType设置为“Polygon”时, 定义被检测区域的第6个顶点的X坐标值

属性	说明
SearchWinPolygonPointY6	当SearchWinType设置为“Polygon”时，定义被检测区域的第6个顶点的Y坐标值
SearchWinPolygonPointX7	当SearchWinType设置为“Polygon”时，定义被检测区域的第7个顶点的X坐标值
SearchWinPolygonPointY7	当SearchWinType设置为“Polygon”时，定义被检测区域的第7个顶点的Y坐标值
SearchWinPolygonPointX8	当SearchWinType设置为“Polygon”时，定义被检测区域的第8个顶点的X坐标值
SearchWinPolygonPointY8	当SearchWinType设置为“Polygon”时，定义被检测区域的第8个顶点的Y坐标值
SearchWinPolygonPointX9	当SearchWinType设置为“Polygon”时，定义被检测区域的第9个顶点的X坐标值
SearchWinPolygonPointY9	当SearchWinType设置为“Polygon”时，定义被检测区域的第9个顶点的Y坐标值
SearchWinPolygonPointX10	当SearchWinType设置为“Polygon”时，定义被检测区域的第10个顶点的X坐标值
SearchWinPolygonPointY10	当SearchWinType设置为“Polygon”时，定义被检测区域的第10个顶点的Y坐标值
SearchWinPolygonPointX11	当SearchWinType设置为“Polygon”时，定义被检测区域的第11个顶点的X坐标值
SearchWinPolygonPointY11	当SearchWinType设置为“Polygon”时，定义被检测区域的第11个顶点的Y坐标值
SearchWinPolygonPointX12	当SearchWinType设置为“Polygon”时，定义被检测区域的第12个顶点的X坐标值
SearchWinPolygonPointY12	当SearchWinType设置为“Polygon”时，定义被检测区域的第12个顶点的Y坐标值
SearchWinRadiusInner	定义待搜索区域的圆弧内径。
SearchWinRadiusOuter	定义待搜索区域的圆弧外径。
SearchWinTop	定义将搜索区域的最上边的位置(单位: Pixel)。
SearchWinType	定义要搜索的区域类型。(Rectangle, RotatedRectangle, Circle, Arc, Polygon)
SearchWinWidth	定义待搜索区域的宽度。(单位: Pixel)
SeparationAngle	设置或返回找到的对象之间允许存在的最小角度。
SeparationMinX	设置或返回找到的对象沿X轴方向允许的最小间距。
SeparationMinY	设置或返回找到的对象沿Y轴方向允许的最小间距。
SeparationScale	设置或返回找到的对象之间允许存在的最小比例差值。
SharedEdges	设置或返回是否允许找到的对象之间存在共享边。
ShowModel	在可以缩放的视图中显示之前示教的模型。可用于更改模型原点，及更改需要忽略的像素。

属性	说明
SkewFitEnable	指定是否对模型使用变形适应。 默认: False
Smoothness	设置或返回几何边缘提取滤波的平滑度水平。
Sort	选择对象结果的排序方式。
Timeout	设置或返回Geometric对象的最大搜索时间。

### Geometric对象结果

下表给出了Geometric对象结果的基本说明。

欲了解每条结果的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual*。

结果	说明
Angle	返回与已找到的某个零件相关的旋转量。(即: 此结果规定了某零件相对于初始示教的方向而进行旋转的量)。
CameraX	返回相机坐标系内已找到的零件位置(参见模型原点)的X坐标位置。
CameraY	返回相机坐标系内已找到的零件位置(参见模型原点)的Y坐标位置。
CameraXYU	仅限运行时间。 返回相机坐标系中的已检测到零件位置的CameraX、CameraY和CameraU坐标。
ClearanceOK	返回间隙的判断结果
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了Accept属性的当前设置值。
FoundOnEdge	当发现Geometric对象与搜索窗口边缘距离太近时，则返回True。如果FoundOnEdge为True，则Found的结果设置为False。
NumberFound	返回检测到对象的数目。 (最少是0个，最多是NumberToFind属性中设定的检测数量)。
Overlapped	若检测到的对象是重叠的，则返回True。
Passed	返回测试结果合格与否。
PixelX	返回已找到的零件位置(参见模型原点)的X坐标位置(以像素为单位)。
PixelY	返回已找到的零件位置(参见模型原点)的Y坐标位置(像素表示)。
PixelXYU	仅限运行时间。 返回已找到的零件位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
Reversed	如果检测到的对象的Polarity与模型是相反的，则返回True。
RobotX	返回已查找到零件位置(参见模型原点)相对于机器人坐标系的X坐标位置。

结果	说明
RobotY	返回已查找到的零件位置(参见模型原点)相对于机器人坐标系的Y坐标位置。
RobotU	返回在机器人坐标系内已找到的零件位置的U坐标位置。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的零件位置的RobotX、RobotY和RobotU坐标。
Scale	返回比例系数。 ScaleEnabled必须设定为True才能使此结果有效。
Score	返回一个INTEGER值，这个值代表运行时间内找到的对象与Geometric对象正在搜索的模型的匹配程度。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。 这样便于对比结果。
SkewDirection	返回执行过程中检测到的对象偏斜方向。
SkewRatio	返回执行过程中检测到的对象偏斜率。
Time	返回处理对象所需要的时间(单位：毫秒)。
TimedOut	返回对象执行过程是否因超时而终止。

### 理解Geometric搜索

Geometric对象的目的是在搜索窗口，定位先前示教过的模型。Geometric对象可用于寻找零件、检测对象物是否存在、检测零件缺陷以及各种检测用途。

本节将介绍Geometric对象的基础知识。就以下几个方面进行说明。

- Geometric 对象的模型和被检测对象
- 搜索的基本概念
- 设置 Accept 属性和 Confusion 属性的阈值
- 其他搜索参数
- 采用多结果对话框调试搜索问题
- Geometric 对象和旋转
- Geometric 对象和比例
- 示教模型用于角度搜索
- 搜索重复性和精度
- 校准相机与被检测对象之间的距离

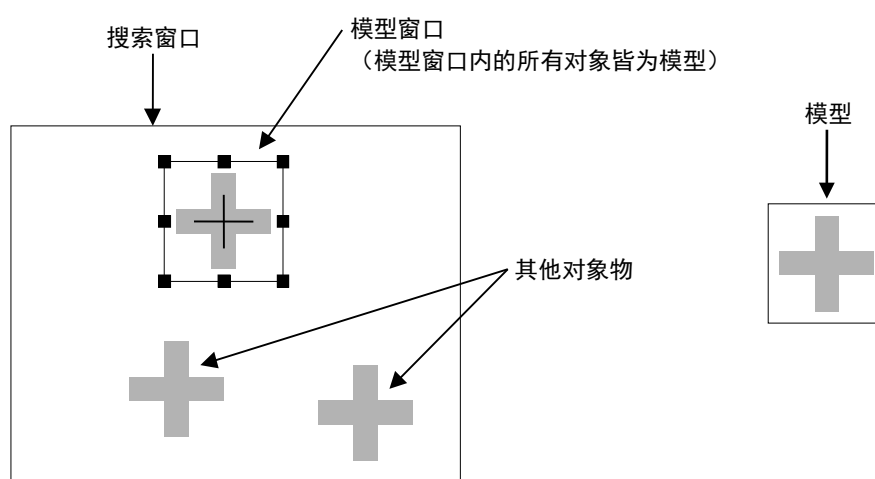


### Geometric对象的模型和被检测对象

使用 Geometric 对象时很重要的一点就是要理解被检测对象与模型之间的区别。被检测对象是指，搜索窗口中的边缘特定的图案。对象物的范围可以是面积只有若干像素的简单线条，也可以是面积达数万像素的复杂图案。Geometric 运算可以测量搜索窗口内的被检测对象与先前示教的该对象模型的匹配程度。模型或者说是模板是对象物的理想化表现，而被检测对象则是实际存在于搜索窗口内的工件。

常见的做法就是，在搜索窗口内注册一个具有代表性的模型，然后在该搜索窗口内搜索与模型类似的对象。下图中是一个包含被检测对象(十字形图案)的搜索窗口。若要注册这个十字形图案的模型，需要定义模型窗口，然后点击执行面板上的 <Teach>按钮。(有关模型示教的详细内容，请参阅 *使用 Geometric 对象*。)

这样就建立了一个如下图右侧所示的模型，且此模型可用于搜索该搜索窗口内的其他十字形图案。



此搜索窗口含有多个对象物(左侧)和根据图像注册的模型(右侧)

根据对象物的几何形状来搜索模型的方法，是一种有效而稳定的方式，但是在定位模型以选择最佳模型时，还应该注意以下几个方面。

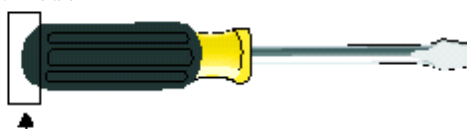
#### 确保图像具有足够的对比度

在以亚像素精度辨别模型源中的边缘和目标图像边缘时，必须考虑对比度。请勿使用对比度低的图像，因为Geometric搜索工具使用一种基于几何边缘的算法。对比度越低，执行搜索所需的与边缘信息越少，搜索精度就越低。因此，请勿使用灰度值渐变的模型。背景和图像边缘之间的灰度值差尽量保持在10以上。

#### 避免使用不正确的几何模型

不正确的几何模型的缺点有，缺少轮廓清晰的几何特征，或者其几何特征无法使它们自身充分区别于其他图像特征。这些模型产生的结果很不可靠。

#### 不正确的几何模型



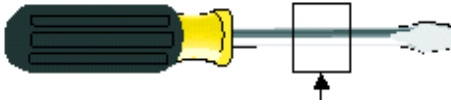
简单的曲线缺乏鲜明的特征，容易产生错误匹配。

### 避免使用模糊的模型

某些类型的几何模型会在位置、角度或比例方面产生模糊的结果。

位置模糊的模型通常仅包含一组或多组平行线。例如，请不要使用仅由平行线组成的模型，因为这样无法确定平行线的准确位置。理论上，任何一条直线上的线段的实际数目都是无限的，因此可以检测到无限匹配。请确保线段上具有形状特征，而该特征那个可以将它与其他图像的细节区分开。

#### 位置模糊的模型

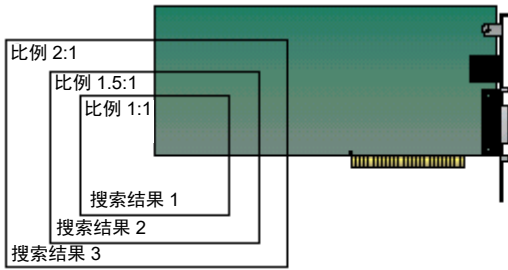


#### 避免使用仅由若干条平行线组成的、无鲜明特征的模型

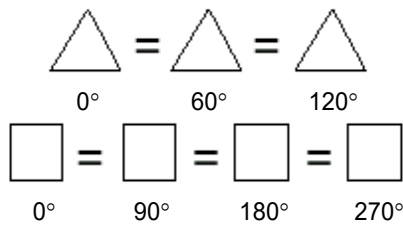
由对象的一小部分组成的模型应进行测试，以确认这些模型的比例不模糊。例如，由孤立的角构成的模型的比例就很模糊。



图中就是一个在比例的范围中进行搜索的模糊图形的示例。由于缺少鲜明的几何特征，因此无法得出正确的结果。



由于被检测对象都是相似的物体，所以对称模型的角度通常是模糊的。例如，圆圈的角度就完全是模糊的。正方形和三角形等其他简单的对称模型的某些角度也是模糊的。



### 几近模糊的模型

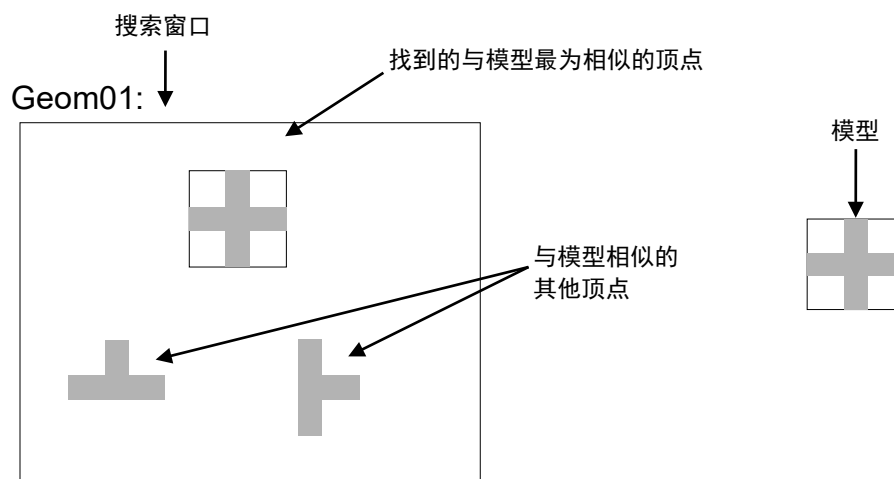
若某个模型的绝大部分都是模糊的特征，则可能会出现错误的匹配，因为构成模糊特征的对象中搜索到结果边缘所占的百分比很高，足以被视为匹配。为避免这一现象，要确保您的模型具有足够的显著特征，以便使这个模型区别于其他的目标图像特征。这样就能确保在搜索结果中返回正确的匹配项。例如，下面的模型可能产生错误的匹配，因为模型中大部分的有效边缘由平行直线构成而不是由有特征的曲线。

#### 几近模糊的模型



### 基本搜索概念

搜索是指，在搜索窗口中，找到与模型最相似的区域，从而定位被检测对象。下图中有个模型和搜索窗口，以及搜索窗口内与模型最为相似的区域。此类模型可以用于搜索，诸如印刷电路板上的对准记号的对象。机器人可以利用搜索功能返回的位置数据来查找电路板的位置，以确定安装位置或电路板本身的位置。



### 设置Accept和Confusion属性的阈值

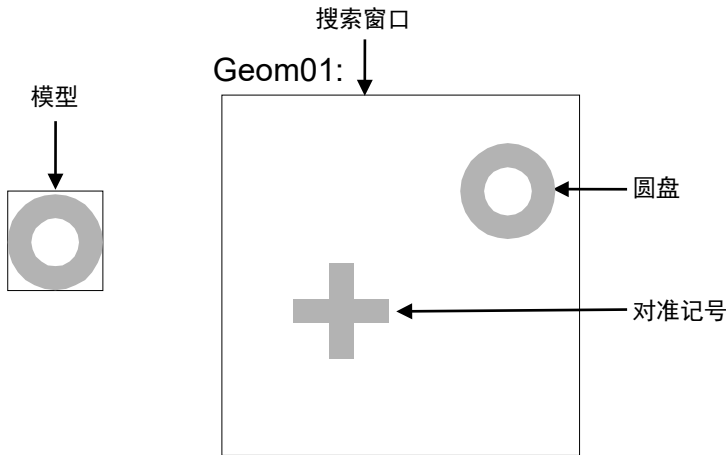
Accept 属性和 Confusion 属性是 Geometric 对象的重要搜索参数。Accept 属性可影响搜索速度，即提供一个提示，说明在何时在某个给定的场景区域内继续进行搜索。若 Accept 属性设置得较高，则要求被检测对象必须与模型非常相似，这样通过粗略检查即可排除掉很多区域，不必继续在这些区域内搜索。如果 Accept 属性设置得较低，则与模型仅稍稍相似的对象均可能超过 Accept 属性阈值，这样就需要对更多的区域进行更详细的检查。所以，Accept 属性值越高，Geometric 对象的运行速度越快，也就是检测速度也就越快。

Confusion 属性与预测的结果数量相互作用，而该数量则会影响搜索速度。当 Confusion 属性和预期的结果数量一致时，则可以在完成搜索所有图像的可搜索区域前，提前中止搜索。

通过设置 Accept 属性，您可以控制系统找到容差范围内最差的对象物。误差可能是由缺陷、缩放、旋转或视频噪音造成的。Vision Guide 7.0 将 Accept 属性的默认值设置为 700。对于大多数应用来说，该值都是一个很好的起点。但是，您可以通过反复的尝试和运行来优化这个值。请注意，并不是说您总需要获得完美或者接近完美的分数才能使应用正常运行。根据被检测对象的变形程度，在有的应用中，即使将形状得分设为 200，也可以获得准确的位置信息。

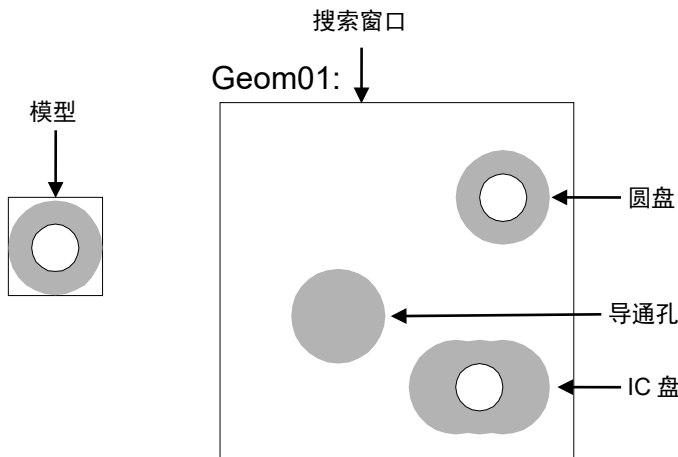
根据您可以接受的“最差的结果”(±容差范围)的最高值, 来设置 Confusion 属性。Confusion 属性阈值宜大于或等于 Accept 属性阈值。Confusion 属性的值越高, 搜索的时间就越长, 但是搜索的准确度也越高。Confusion 属性默认值为 800, 您可以根据实际运用的需要进行调整。

下图所示是一个比较简单的场景。圆盘与对准记号(十字形图案)几乎没有相同特征。此时, 可以将 Confusion 属性设置为一个较低的值(700 左右)。Accept 属性值根据容差量进行设定, 一般都会小于或等于 Confusion 属性。但在这个例子中, 由于几乎没有变形, 所以将形状得分设置为 920 是最合适的。



比较简单的场景

在下图是一个比较复杂的场景: 导通孔和IC盘都与圆盘相似。因此, Confusion 属性宜设置为一个较高的值(820左右)。



比较复杂的场景

具有灰度常数区域的搜索窗口将始终在该区域获得相关值 0。如果某个情形下的背景基本统一(例如一张白纸), 故而大多数位置都不会有几何形状。因此, 如果 Geometric 对象查找到任何东西, 都应该是需要搜索的目标对象, 所以可以将 Confusion 属性设置为一个较低的值。

Accept 和 Confusion 属性可以被视为您提供给系统的提示，以便让系统能够更快地定位特征。总的来说，应保守设置这些属性，但设置不用太准确。最保守的设置方法是，将 Accept 属性设低，将 Confusion 属性设高。如果对要搜索的场景了解较少时，建议采用保守的设置。这样搜索的结果是精准的，但是搜索速度比较慢。(在使用 Geometric 属性的位置结果来控制机器人的动作时，这一点尤其重要。)

如果对要搜索的场景非常了解，则建议采用更灵活的设置。例如，当您正在搜索的对象周围没有其他特征，场景剩余部分都为空白时，则可以灵活设置属性值。这样就不会进行精确搜索，提高搜索速度。

### 其他Geometric搜索参数

<b>DetailLevel</b>	DetailLevel 决定了搜索期间什么特征可被视为边缘。边线是用相邻像素之间的灰度值的过渡部分来定义的。“Medium”的默认设置可以让您即使在有对比度变化、噪声或光线不均匀的情况下，也能正确的从图像中检测到边缘部分。某些情况下检测对象的对比度相较于图像中的高对比度区域显得很低，则可能会错失某些低对比度的边缘。如果您的图像含有低对比度的对象，请提高DetailLevel的设定值，来确保检测到图像中所有的重要边线。VeryHigh 设置下，可将对比度极低的边线全都提取出来。但要注意的是，这种模式对噪声十分敏感。
<b>ScaleTargetPriority</b>	规定了是否要优先检测ScaleTarget附近的对象。如果是要检测工业产品这类尺寸差别很小的对象，则将此属性设置为True。 默认： True
<b>SearchReducedImage</b>	在进行粗略的对象检测时缩小输入图像的尺寸。如果输入图像具有很多特征(例如边缘)，则这项属性可缩短搜索时间。 默认： False
<b>Smoothness</b>	Smoothness属性让您可以控制边线提取滤波的平滑程度。平滑操作可平滑粗糙边缘，并消除噪声。此项控制的范围为0(不光滑)到100(极为光滑)。 默认： 50
<b>SharedEdges</b>	将SharedEdges设置为True，则可以让多个结果共享边缘。否则，多个结果共享的边缘就会被视为得分最高的那个结果的边缘。
<b>TimeOut</b>	在一些时间要求严格的应用中，您可以为Geometric对象设置一个以毫秒为单位的时间限制，以检测指定模型的搜索结果。如果在到达时限之前没有找到要求的数量，则会停止搜索。并返回已找到的搜索结果。但是无法预测将检测到哪些搜索结果。 默认： 2000毫秒

### 采用多结果对话框调试搜索问题

有时，即使在同意生产批次内，加工的零件也可能会发生很大变化，或者一个零件上有多个相似的特征。这种情况下，即设置一个较为合理的Accept属性值，也还是可能会搜索到其他错误的对象。在这类应用中，很能预测到会发生什么问题。

我们创建了 ShowAllResults 对话框来帮您解决上述以及各种其他问题。当您只想检测一个零件上的一个对象物特征，经过多测检测后，您可以找出为什么 Vision Guide 7.0 有时会搜索到其他并不想要得到的对象。通常导致这种情况产生的原因如下。

1. 搜索窗口内有两个或更多对象物非常相似，且有着很接近的 Score 结果分值。
2. 设置的 Confusion 或 Accept 属性值不够高，使得得分低于目标对象的错误对象的得分，也符合 Accept 属性设定的值。

如果您是 Vision Guide 7.0 的新用户，那么请注意，当在搜索窗口内搜索单个特征时，上述两种情况十分容易造成混淆。

如果检测的结果有时是您要找的对象，有时又是错误的对象，则可以用多结果对话框对比结果，确认问题所在。请按照以下步骤进行确认。

- (1) 将NumberToFind属性设置为3或以上。
- (2) 在Vision Guide 7.0开发环境下运行视觉对象。
- (3) 点击<ShowAllResults>属性按钮调出[Show All Results]对话框。
- (4) 检查搜索到的前三个或更多的对象的得分。

如上所述，确认了搜索到的前三个或更多的对象的得分之后，将揭晓问题的根源所在。一般情况下产生这些问题的原因如下。

1. 每个检测到的对象的得分都大于Accept属性中设置的值。  
如果是这种情况，请将Confusion属性的值调高，以确保检测出正确的对象，而不返回那些仅与Accept属性阈值一致的对象。当然也可以通过修改Accpet属性的值来改善这一问题。
2. 每个对象物的得分都非常相似。  
如果是这种情况，请按照以下步骤区分被检测对象。
  - 重新调整搜索窗口，确保作为被检测出对象而随机返回的对象物，不会出现在搜索窗口中。
  - 重新示教要检测对象的模型。
  - 调整周围的照明灯光，以确保要检测的对象的得分，可以高于错误检测对象的得分。

关于使用多结果的更多详细信息，请参阅本章后续章节 [6.2.24 使用来自单个对象的多个结果](#)。

### Geometric对象的间隔

您可以指定搜索结果与同一模型中其他搜索结果之间的最小间隔，以明确的区分(匹配)搜索结果。本质上，这定义了同一模型中可以重叠多少搜索结果。

您可以设置X位置、Y位置、角度和比例这4个标准的最小间隔。要区分出搜索结果，只需满足一个最小间隔即可。例如，如果搜索结果满足角度的最小间隔，那么无论位置和比例如何，都可以区分出检测结果。当然，如果不需要使用最小间隔也能正确的识别对象，您也可以不使用这些间隔标准。

位置的最小间隔属性的SeparationMinX和SeparationMinY定义了，在同一模型中，搜索结果至少要间隔多远才能算做是两个不同的搜索结果。此间隔被指定为标称比例(ScaleTarget)的模型大小的百分比。

默认值为10%。若将该值设为0%，则禁用本属性。例如，如果模型在ScaleTarget为100像素宽，若将SeparationMinX设置为10%时，只有当搜索结果在X方向上的距离超过10像素时，才会被视为是两个不同的对象。

角度的最小间隔(SeparationAngle)定义了，区别两个搜索结果的最小角度差异。该值为绝对角度值。默认值为 10.0°。若将该值设为 0°，则禁用本属性。

比例的最小间隔(SeparationScale)用倍率定义了，区别两个搜索结果的最小比例差异。默认值为 1.1。若将该值设为 1.0，则禁用本属性。

### Geometric对象和比例

模型比例定义了，目标图像中检测到的模型和预期模型的大小。如果预期的搜索结果大于或小于模型，则可以在可用范围内，设置其比例。使用 ScaleTarget 属性设置预期比例(范围: 0.5 ~ 2.0)。

在默认情况下，比例范围的搜索是无效的。您可以根据需要，将 ScaleEnable 设置为 True，启用比例范围搜索功能。这样您就可以根据指定的 ScaleTarget 值(或大或小)，在不同的尺寸范围中，检测出目标图像的模型。使用 ScaleMinFactor 和 ScaleMaxFactor 设置比例范围。ScaleMinFactor(0.5 ~ 1.0)和 ScaleMaxFactor(1.0 ~ 2.0)共同决定标称 ScaleTarget 的比例范围。

最大系数和最小系数按照以下公式，计算出 ScaleTarget 设置：

$$\max \text{ scale} = \text{ScaleTarget} \times \text{ScaleMaxFactor}$$

$$\min \text{ scale} = \text{ScaleTarget} \times \text{ScaleMinFactor}$$

请注意，范围被定义为一个比例，这样即使需要修改预期比例(ScaleTarget)时，也无需修改范围。比例范围的搜索时并行进行的。也就是说搜索结果的实际范围，不会影响第一个被搜索到的对象。但是，比例范围越大，搜索速度越慢。

### Geometric对象和旋转

Geometric 对象是查找旋转零件的理想工具。因为 Geometric 对象搜索的是边缘的几何形状，所以当查找旋转零件时，它比其他视觉工具更为可靠。

如需 Geometric 对象执行带有角度检测功能的搜索，需在示教 Geometric 对象模型时，将 AngleEnable 属性设置为 True。并设置 AngleStart 属性和 AngleRange 属性。

#### 示教用于角度搜索的模型

如需进行带有角度检测功能的搜索，首先需要设置用于角度搜索的 Geometric 搜索对象。首先，将 AngleEnable 属性设置为 True，然后用 AngleRange 属性指定要示教模型的角度范围。您还可以通过设置 AngleStart 属性，来更改角度搜索的中心。这是 AngleRange 所基于的角度。例如，如果 AngleStart 为 45，AngleRange 为 10，则搜索 35° ~ 55°的模型。

请记住，示教带有旋转角度的模型时，搜索窗口必须足够大，以保证模型旋转时，不会超出搜索窗口之外。

### 搜索重复性和精度

影响搜索重复性和精度的因素有，模型的尺寸和细节(形状、被检测对象的粗糙度和模型的对称性)，以及搜索窗口被检测对象的变形(噪声、缺陷、旋转以及缩放的影响)。

若要测量噪声对位置的影响，您可以先在特定的搜索窗口中，搜索没有产生变形的检测对象，然后不更改对象位置，再次实行完全相同的搜索(将第 2 张图像采集到帧缓冲器中)，最后比对两个搜索结果的位置。按照以下步骤进行对比：

- (1) 定义模型后，点击执行面板上的<Run>按钮两次或更多次。
- (2) 点击<Statistics>按钮。
- (3) 然后可利用 Statistics 对话框，查看两次搜索结果之间的位置差异。

如果是无变形检测对象的大模型(30x30)，则重复精度可达到 1/20 像素。然而，在大多数情况下，精度在 1 像素或以下 (1/2、1/3、1/4 等)。

如需测量搜索精度，可在特定的搜索窗口中，搜索没有产生变形的检测对象，然后移动检测对象，将搜索得到的位置差异与实际差异进行比对即可。如果您的模型是大模型(30x30或更大)，且无变形、无旋转也无比例误差，并在X方向和Y方向上都有足够的边缘，则搜索精度可达到1/4像素。(请至于，此搜索精度是在不考虑机器人本身的误差，而只考虑视觉系统时的精度。所以，当您实际运用在机器人上时，还需考虑机器人机构自身的误差。)

### 校准相机和被检测对象之间的距离

为了获得最佳的搜索结果，图像中对象在搜索时的尺寸应与示教模型时的尺寸相同。假定使用相同的相机和镜头，在示教模型时和执行搜索时，如果被检测对象和相机的距离发生变化，则搜索窗口内的被检测对象也会呈现不同的外观尺寸。也就是说，相机与被检测对象离得越近，对象就显得越大；相机与被检测对象离得越远，对象就显得越小。


所以，当被检测对象和相机之间的距离发生改变时，需要重新示教模型。



### 使用Geometric对象

总是，我们已经了解了Geometric对象和搜索的工作原理，为理解如何使用Vision Guide 7.0 的Geometric对象打下了基础。本节将说明如何使用Geometric对象。使用Geometric对象的步骤如下：


- 创建一个新的Geometric对象
- 设置搜索窗口的位置和尺寸
- 设置模型窗口的位置和尺寸
- 定位模型原点
- 设置 Geometric 对象的相关属性
- 示教模型
- 测试 Geometric 对象并检查结果
- 调整属性并再次测试
- 处理单个 Geometric 对象的多种结果

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

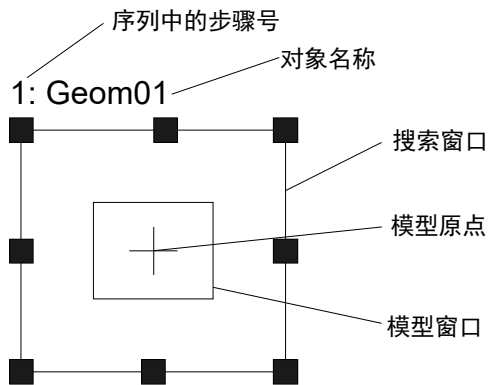
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

#### 第1步：创建一个新的Geometric对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>- <New Geometric>按钮。
- (2) 您将看到一个 Geometric 图标出现在<New Geometric>工具栏按钮上方。
- (3) 点击这个 Geometric 图标，并将其拖至 Vision Guide 窗口的图像显示区域中。
- (4) 对象名称将自动创建。本示例中，由于是在本序列中创建的第一个 Geometric 对象，所以对象名称为“Geom01”。(稍后将说明如何修改名称。)

**第2步：设置搜索窗口的位置和尺寸**

屏幕上将显示一个与下图相似的Geometric对象。



新Geometric对象的布局

- (1) 点击Geometric对象的名称标签，按住鼠标左键，将Geometric对象拖至您想让搜索窗口的左上角所在的位置并按住鼠标。
- (2) 点击搜索窗口的尺寸手柄，拖动鼠标，将Geometric对象搜索窗口调整成合适的大小。(搜索窗口就是要搜索的区域。)



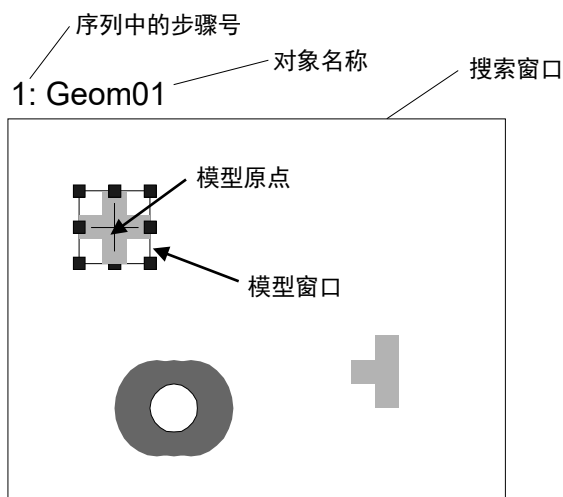
**注意**

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。

**第3步：设置模型窗口的位置和尺寸**

- (1) 正在运行中的Geometric对象搜索窗口会突出显示搜索窗口上的尺寸手柄。如果没有显示尺寸手柄，请点击Geometric对象的名称字段。如果您能看到尺寸手柄，请忽略此步骤，继续下一步。
- (2) 点击模型窗口方框上的任一条边线。此操作会让模型窗口突出显示，并显示模型窗口上的尺寸手柄。
- (3) 点击模型窗口方框上的任一条边线，按住鼠标左键，将模型窗口拖动至您想让模型窗口的左上角所在的位置。
- (4) 点击尺寸手柄拖动鼠标，调整模型窗口的大小，确保其可以囊括被示教为Geometric对象的模型。

Geometric对象布局如下所示，搜索窗口是整个被检测区域，而模型窗口囊括了被搜索对象。实际的搜索窗口和模型窗口的形态可能会和下图显示的有所差异，此处的图像只是个大概的参考。



搜索窗口和模型窗口的位置、和尺寸调整后的Geometric对象

正确设置模型窗口的尺寸和位置的技巧：

模型窗口的尺寸和位置十分重要，因为它们定义了将要搜索的对象。创建 Geometric 对象的模型窗口时，请注意以下两点：

- (1) 搜索窗口既是被搜索的区域，窗口越小搜索的时间越短。特别是如果预先知道被搜索的零件会有较大的角度旋转，那么尽量缩小搜索窗口和模型(零件的一部分等)，则可减少零件旋转造成的影响。
- (2) 使模型窗口的尺寸尽量接近搜索窗口的尺寸，以缩短搜索时间。

另外，如果两个零件位置相邻甚至相互接触时，将模型窗口设置的稍大于(几个像素)实际检测对象时，可以帮助区分目标对象和其他对象。

请注意，模型窗口的最佳尺寸取决于每个实际应用的具体情况。

#### 第4步：定位模型原点

模型原点定义了模型上的位置，当运行Geometric对象时，该位置将返回为被检测对象的位置。也就是说，当位置信息非常关键时，需要将模型原点设置在重要的位置。

例如，在使用Geometric对象检测零件，控制机器人取放零件应用时，机器人能否轻松的抓取零件，模型原点的位置是一个重要的因素。因为这将是机器人根据RobotX, RobotY, RobotU, RobotXYU返回的结果所移动到的那个位置。

在创建新的 Geometric 对象后，ModelOrgAutoCenter 属性的默认设置为 True。此时模型原点会自动被设置在模型窗口的中心，且无法手动移动。


若要以手动修改模型原点，须先将 ModelOrgAutoCenter 属性设置为 False。该属性的设置方法和模型原点定位方法如下。

- (1) 点击 Vision Guide 窗口流程图中的 Geometric 对象。在对象窗口的属性列表中查找 ModelOrgAutoCenter 属性，并点击值字段。
- (2) 下拉列表会显示“True”和“False”两个选项。点击“False”选项。这样就可以将 ModelOrgAutoCenter 属性设置为 False，也可以使用鼠标拖动模型原点。
- (3) 点击模型窗口以突出显示模型窗口。
- (4) 点击模型原点，按住鼠标左键，将模型原点拖至新位置。应注意的是，模型原点只能位于模型窗口范围内。

#### 第5步：设置Geometric对象的相关属性

在步骤 4 中显示的对象窗口中，设置 Geometric 对象属性。设置各项属性，请点击相关属性的值字段并输入新值，或者点击所示的下拉列表中的相应选项。

下表所示为Geometric对象常用的一些属性。其它一些在许多视觉对象中都会经常使用到的属性，如AbortSeqOnFail和Graphics等，请参阅“Geometric对象属性”的相关内容。在测试Geometric对象时，无需设置这些属性。但是，如果您是第一次使用Geometric对象，可参考以下说明。

 <p>注意</p>	<ul style="list-style-type: none"><li>■ 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。 当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。 正确配置Accept和RejectOnEdge等属性值，可降低错误检测的风险。</li></ul>
---	---

<b>Name属性</b>	<p>新创建的 Geometric 对象中，默认的项目名称是“Geom**”。在一个视觉序列中要使用多个 Geometric 对象时，使用**的编号对 Geometric 进行区分。</p> <p>默认情况下，会将在视觉序列序列中使用的第一个 Geometric 对象命名为“Geom01”。</p> <p>需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Geometric 对象名称的地方都会变为新的名称。</p>
<b>Accept属性</b>	<p>Accept 属性用于设置形状分值的标准，被检测对象必须与其一致，或者符合其设定的标准时，才能被找到。</p> <p>Score 结果的返回值将与这个 Accept 属性的值进行对比。</p> <p>默认：700</p>
<b>Confusion属性</b>	<p>如果搜索窗口内有很多相似的特征，则可利用 Confusion 属性帮助您找到正确的搜索对象。</p> <p>默认：800</p>
<b>ModelOrgAutoCenter属性</b>	<p>如果您打算以手动方式改变模型原点的位置，您必须首先将 ModelOrgAutoCenter 属性设置为 False。</p> <p>默认：True</p>
<b>Frame属性</b>	<p>Frame 属性可让您将先前定义的 Frame 对象选为 Geometric 对象的参考框。</p> <p>有关 Frames 的详细信息，请参考 <i>Frame</i> 对象中的说明。</p>
<b>NumberToFind属性</b>	<p>根据想要查找的对象的数目，您可能想将 NumberToFind 属性设置为大于 1 的数字。这样就能让 Geometric 对象在一个搜索窗口内找到多个特征。</p>
<b>AngleEnable属性</b>	<p>如果您打算利用 Geometric 模型进行角度搜索，就将此项属性设置为 True。若要对多个模型进行角度搜索，则在示教 Geometric 对象的模型之前将这条属性设置为 True。</p>
<b>AngleRange属性</b>	<p>这条属性与 AngleEnable 属性共同使用，可用角度搜索 Geometric 模型。</p>
<b>RejectOnEdge属性</b>	<p>利用这条属性，可以将与搜索窗口边界有接触的那些零件排除在外。通常而言，这条属性宜设置为 True。</p>

可以将属性保持为默认属性，并进行下一步。稍后可在必要时设置这些属性。

### 第6步：示教模型

Geometric 对象执行搜索功能，需要一个模型，这通过被称为示教模型的步骤来实现。在这个步骤中，Geometric 对象的模型窗口应该已经被放置在可囊括模型对象的位置。通过以下步骤完成模型示教。

- (1) 确保 Geometric 对象是否为当前显示的对象。查看流程图或序列目录树，确认当前正在处理的对象。或者，检查图像显示区域，查看正以洋红色突出显示的对象。
- (2) 点击执行面板上的 <Teach> 按钮。完成模型示教。多数情况下，示教模型仅需几秒钟。但如果在 AngleEnable 属性设置为“True”时示教某个模型，由于要示教多个角度的模型，则会需要较长时间方能完成。

### 第7步：测试Geometric对象并检查结果

要运行Geometric对象，点击执行面板上对象的<Run>按钮。

Geometric对象的结果将会显示出来。这时需检查的主要结果有：

<b>Found 结果</b>	返回是否检测到了Geometric对象。如果找到目标对象，则返回的结果为“True”。如果未找到目标对象，Found的结果将返回“False”，且以红色突出显示。如果找不到Geometric对象，请参考第8步了解常见原因。
<b>FoundOnEdge 结果</b>	如果找到的对象物的一部分与搜索窗口的边界接触，则这个结果返回为“True”。 在这种情况下，Found结果返回为“False”。
<b>Score 结果</b>	这个结果显示与模型最为相似的那个特征与模型匹配的程度。Score结果的范围为0至1000，1000表示模型和被检测对象完全一致。当要确认被检测出的对象与希望检测出的对象的相似程度时，请先确认运行Geometric对象之后的Score结果。
<b>Time 结果</b>	返回Geometric对象执行所需时间。 搜索窗口和模型窗口越小，搜索速度就越快。
<b>NumberFound 结果</b>	在搜索多个Geometric对象时，NumberFound结果会返回所找到的与Geometric对象模型相匹配的特征的数目。
<b>Angle 结果</b>	Geometric对象的倾斜角度。 这个角度是依据模型的原始角度计算的。但有时只是个粗略值，可能并不可靠。 建议使用Polar对象来确定旋转角度。特别是使用机器人向导时，请使用Polar对象。
<b>PixelX 结果</b>	返回对象的XY坐标值(像素坐标系)。
<b>PixelY 结果</b>	这个位置是被检测出对象的模型原点的位置。如果需要返回其他位置，需要变更模型原点，重新示教模型。
<b>CameraX结果</b>	在相机坐标系中定义检测对象的XY坐标值。
<b>CameraY结果</b>	如果相机尚未完成校准则不会返回任何值，而是返回[No Cal]。
<b>RobotX结果</b>	在机器人坐标系中定义被检测出对象的XY坐标值。
<b>RobotY结果</b>	可用于命令机器人移动至这个XY坐标位置。(不需要进行其它转换或处理。) 这个值是被检测出对象的模型原点的位置。如果您想返回一个不同的位置，您必须首先重新确定模型原点的位置，然后重新示教该模型。如果相机未进行校准则不会返回任何值，而是返回[No Cal]。
<b>RobotU结果</b>	返回被检测出对象的角度，该角度是转换成机器人坐标系后的值。 如果相机尚未完成校准，则不会返回任何值，而是返回[No Cal]
<b>ShowAllResults</b>	当处理多个结果时，点击ShowAllResults设置值字段中的按钮，则会显示ShowAllResults对话框，可以查看当前视觉对象的所有结果。

## NOTE



由于上述示例各步骤中没有包括进行校准的步骤，所以RobotXYU、RobotX、RobotY、RobotU和CameraX、CameraY、CameraXYU的结果会返回“No Cal”。这意味着由于未进行校准，视觉系统将无法计算相对于机器人坐标系或相机坐标系的坐标结果。请参考 [校准](#) 以了解详情。

### 第8步：调整属性并再次测试

在Geometric对象运行若干次之后，您可能会在查找某个Geometric对象时遇到困难，或者您还想微调某些属性设置。下面将介绍一些常见问题的故障排除方法和微调整技巧。

### Geometric对象的故障排除

#### Geometric对象返回的Found结果为False:

- 尝试降低Accept属性的设定值(例如：低于当前Score结果的值)，并再次运行Geometric对象。
- 确认FoundOnEdge结果的返回值是否为True。如果Found结果为“False”，FoundOnEdge结果为“True”，表示已经查找到对象，但其中有一些对象与搜索窗口边缘有接触。此时，请将搜索窗口调大，使其能适应被搜索对象的大小。如果搜索窗口的大小无法调整，请尝试更改相机位置，或调整模型窗口的大小。

#### Geometric检测到错误识别对象时:

- 检查Accept属性是否设置得足够高。如果Accept属性设置得很低，则有可能会检测出搜索对象以外的其他对象。
- 请确认Confusion属性是否设置得足够高。Confusion属性值必须大于或等于Accept属性值。如果搜索窗口内有与被搜索对象非常相似的对象时，则必须将Confusion属性值调为更高的值，以避免检测出错误的识别对象。
- 调整搜索窗口，使它靠近被检测对象，并精准的将其单独隔离出来。

### Geometric对象微调

通常需要进行Geometric对象的微调，才能正常使用。以下是与Geometric对象微调和模型添加相关的主要属性说明：

**Accept属性** Accept属性的值越低，Geometric对象的运行速度就越快。但是Accept过低时，可能会造成被检测出的对象，不是想要查找的对象。可以尝试多运行几次Geometric对象，一旦Score结果返回的形状分值符合预期结果时，调整Accept属性的值。为了以最佳执行速度并得到可靠的结果，通常需要若干次尝试运行才能找到一个合适的值。

**Confusion属性** 如果搜索窗口中有与被搜索对象非常相似的对象时，则需要将Confusion属性设置得相对高一些。这将区分那些有相似特征的对象，正确的检测出目标对象。但是，Confusion属性值设置的过高时，会降低运行速度。如果搜索窗口没有类似的对象物时，可降低Confusion属性的设定值，以缩短执行时间。

**Add another sample** 当进行示教的模型窗口与当前模型的尺寸一致时，可以选择“Add another sample”。当模型具有轻微变化(形状或图案略微不同，阴影表现不同)时，选择已更改的模型来添加新模型可能会稳定对象执行的得分。如果角度位置大幅度偏移，或者因差异较大而无法添加模型，原始模型将保持不变。

当完成调整，并测试Geometric对象得到满意的结果后，创建视觉对象的步骤就完成了。下一步是继续创建其他视觉对象或者配置并测试整个视觉系列。

### Geometric对象的其他实用工具

Vision Guide 7.0中有一个非常实用的Histogram功能，该功能可以在搜索窗口中直观的显示灰度值的分布情况。使用Vision Guide直方图功能，可以检查Geometric对象的图像的状态。



### 6.2.3 Correlation对象

#### Correlation对象描述

Correlation对象是Vision Guide 7.0中最常用的工具。示教完Correlation对象后，便可以非常快速并精准的查找到并测量到先前注册的对象。Correlation对象被广泛应用于以下几种应用中。

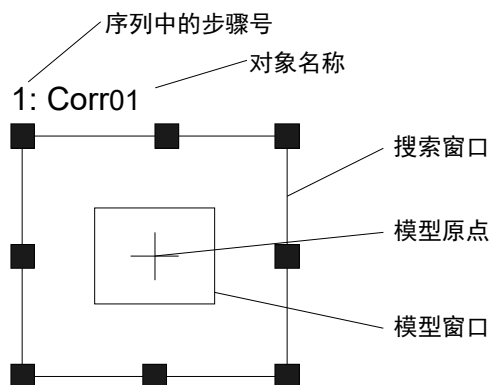
**Alignment** 通过定位对象上的特征(例如登记标志)确定已知对象的位置和方向。此方法通常用来查找零件位置，以协助将机器人引导到拾取位置和布局位置。

**Gauging** 零件检查时查找零件的特征，如直径、长度、角度和其他关键外形尺寸。

**Inspection** 查找简单缺陷，如零件遗失或印刷模糊。

#### Correlation对象布局

Correlation对象有搜索窗口和模型窗口，如下所示。



Correlation对象布局

#### Correlation对象属性

以下列表是对Correlation对象属性的总结。有关详情，请参考*Vision Guide 7.0 Properties and Results Reference Manual*。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。如果设定值偏低，可能会导致错误检测。 默认: 700
AngleAccuracy	指定角度搜索精度(单位: 度)。 默认: 1

属性	说明
AngleEnable	规定了Correlation搜索时，是否进行角度(旋转)搜索。 该属性要在Correlation对象的模型示教之前设定。 默认: False
AngleMaxIncrement	指导相关性模型进行角度搜索的角度最大增量。 最大值: 10。 默认: 10
AngleOffset	指定旋转的偏移值。 默认: 0.000
AngleRange	指定一系列旋转模型的示教范围。 最大值为 45。 默认: 10
AngleStart	指定角度搜索的中心。 默认: 0
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为Correlation对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject结果。 如果指定了All，则Correlation对象将应用于指定视觉对象的所有结果(NumberFound)。 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 如果将 SearchWinType 设置为 RotatedRectangle，搜索窗口则以 Angle 结果为基础进行旋转。 默认: False
CheckClearanceFor	设定确认间隙的对象
ClearanceCondition	设定间隙的判断方法

属性	说明
Confusion	指出被检测图像的预期混淆量。 此处显示的值是对于被检测对象可获得的最大形状的分，而不是您正在检测的对象。 默认：800
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。 默认：1
Description	设定用户指定的描述 默认：空白
EditWindow	定义被检测区域的不要紧的像素
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色 默认：Red
Frame	规定了与指定框有关的当前对象搜索位置 (允许Correlation对象相对于框架进行定位。) 默认：None
FrameResult	指定需使用Frame的第几个结果 默认：1
Graphics	设置显示的图表模式 默认：1 - All
LabelBackColor	设置对象标签的背景颜色 默认：Transparent
ModelObject	规定使用哪种模型进行搜索 默认：Self
ModelOrgAutoCenter	模型有固定的基准点，并且在模型窗口中模型的位置由该参考点表示。该固定基准点被称为模型原点。 ModelOrgAutoCenter属性可将模型原点设置在模型窗口的中心。 默认：True
ModelOrgX	模型原点的X坐标值。(可使用亚像素)
ModelOrgY	模型原点的Y坐标值。(可使用亚像素)

属性	说明
ModelWin	<p>仅限运行时间。</p> <p>在一次调用中设置或返回模型窗口的左侧参数、顶部参数、高度参数和宽度参数。</p>
ModelWinAngle	指定模型窗口的角度。
ModelWinCenterX	指定模型窗口中心的X坐标值。
ModelWinCenterY	指定模型窗口中心的Y坐标值。
ModelWinLeft	指定模型窗口左侧位置。
ModelWinHeight	<p>指定模型窗口的高度。</p> <p>默认：50</p>
ModelWinTop	指定模型窗口顶部位置。
ModelWinType	定义模型窗口的类型。
ModelWinWidth	指定模型窗口的宽度。
Name	<p>用于为Correlation对象分配一个唯一的名称。</p> <p>默认：Corr01</p>
NumberToFind	<p>规定了在当前搜索窗口中待查找的对象的数目。</p> <p>默认：1</p>
PassColor	<p>选择当对象被检测为良品时，搜索窗口显示的颜色。</p> <p>默认：LightGreen</p>
PassType	<p>可定义检测结果良品/不良品的判断标准。</p> <p>默认：SomeFound</p>
RejectOnEdge	<p>规定了是否排除与搜索窗口边缘有连接的零件。</p> <p>通常将属性设置为True。</p> <p>这避免了不完全位于搜索窗口内的部分所导致的错误检测。</p> <p>默认：False</p>
SaveTeachImage	设置示教模型时，是否保存图像文件。
ScoreMode	设置或返回在Fail时显示结果的阈值。
SearchWin	<p>仅限运行时间。</p> <p>在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数或中心X坐标、中心Y坐标、圆内周长的半径大小、圆外周长的半径大小。</p>
SearchWinAngle	定义待搜索区域的角度。
SearchWinAngleEnd	定义待搜索区域的结束角度。
SearchWinAngleStart	定义待搜索区域的开始角度。
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	<p>定义待搜索区域的高度(单位：pixel)。</p> <p>默认：100</p>
SearchWinLeft	定义待搜索区域的最左边位置(单位：pixel)。

属性	说明
SearchWinPolygonPointX1	当SearchWinType设置为“Polygon”时，定义被检测区域的第1个顶点的X坐标值
SearchWinPolygonPointY1	当SearchWinType设置为“Polygon”时，定义被检测区域的第1个顶点的Y坐标值
SearchWinPolygonPointX2	当SearchWinType设置为“Polygon”时，定义被检测区域的第2个顶点的X坐标值
SearchWinPolygonPointY2	当SearchWinType设置为“Polygon”时，定义被检测区域的第2个顶点的Y坐标值
SearchWinPolygonPointX3	当SearchWinType设置为“Polygon”时，定义被检测区域的第3个顶点的X坐标值
SearchWinPolygonPointY3	当SearchWinType设置为“Polygon”时，定义被检测区域的第3个顶点的Y坐标值
SearchWinPolygonPointX4	当SearchWinType设置为“Polygon”时，定义被检测区域的第4个顶点的X坐标值
SearchWinPolygonPointY4	当SearchWinType设置为“Polygon”时，定义被检测区域的第4个顶点的Y坐标值
SearchWinPolygonPointX5	当SearchWinType设置为“Polygon”时，定义被检测区域的第5个顶点的X坐标值
SearchWinPolygonPointY5	当SearchWinType设置为“Polygon”时，定义被检测区域的第5个顶点的Y坐标值
SearchWinPolygonPointX6	当SearchWinType设置为“Polygon”时，定义被检测区域的第6个顶点的X坐标值
SearchWinPolygonPointY6	当SearchWinType设置为“Polygon”时，定义被检测区域的第6个顶点的Y坐标值
SearchWinPolygonPointX7	当SearchWinType设置为“Polygon”时，定义被检测区域的第7个顶点的X坐标值
SearchWinPolygonPointY7	当SearchWinType设置为“Polygon”时，定义被检测区域的第7个顶点的Y坐标值
SearchWinPolygonPointX8	当SearchWinType设置为“Polygon”时，定义被检测区域的第8个顶点的X坐标值
SearchWinPolygonPointY8	当SearchWinType设置为“Polygon”时，定义被检测区域的第8个顶点的Y坐标值
SearchWinPolygonPointX9	当SearchWinType设置为“Polygon”时，定义被检测区域的第9个顶点的X坐标值
SearchWinPolygonPointY9	当SearchWinType设置为“Polygon”时，定义被检测区域的第9个顶点的Y坐标值
SearchWinPolygonPointX10	当SearchWinType设置为“Polygon”时，定义被检测区域的第10个顶点的X坐标值
SearchWinPolygonPointY10	当SearchWinType设置为“Polygon”时，定义被检测区域的第10个顶点的Y坐标值

属性	说明
SearchWinPolygonPointX11	当SearchWinType设置为“Polygon”时，定义被检测区域的第11个顶点的X坐标值
SearchWinPolygonPointY11	当SearchWinType设置为“Polygon”时，定义被检测区域的第11个顶点的Y坐标值
SearchWinPolygonPointX12	当SearchWinType设置为“Polygon”时，定义被检测区域的第12个顶点的X坐标值
SearchWinPolygonPointY12	当SearchWinType设置为“Polygon”时，定义被检测区域的第12个顶点的Y坐标值
SearchWinRadiusInner	定义待搜索区域的圆弧内径。
SearchWinRadiusOuter	定义待搜索区域的圆弧外径。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinType	定义待搜索区域的类型(Rectangle, RotatedRectangle, Circle, Arc, Polygon)
SearchWinWidth	定义待搜索区域的宽度(单位: Pixel)。 默认: 100
ShowModel	在可以缩放的视图中显示之前示教的模型。可用于更改模型原点，及更改需要忽略的像素。
SkewFitEnable	指定是否对模型使用变形适应。 默认: False
Sort	选择对象结果的排序方式。 默认: 0 - None
Timeout	设置或返回Correlation对象的最大搜索时间。

### Correlation对象结果

以下列表是对Correlation对象结果的总结，并附有简要说明。有关详情，请参考 *Vision Guide 7.0 Properties and Results Reference Manual*。

结果	说明
Angle	返回被检测出零件的旋转量，用度数表示。
CameraX	返回相机坐标系内已找到的零件位置(参见模型原点)的X坐标位置(单位: mm)。
CameraY	返回相机坐标系内已找到的零件位置(参见模型原点)的Y坐标位置(单位: mm)。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到零件位置的CameraX、CameraY和CameraU坐标。
ClearanceOK	返回间隙的判断结果
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了Accept属性的当前设置值。

结果	说明
FoundOnEdge	当发现Correlation对象与搜索窗口边缘距离太近时, 返回“True”。 如果FoundOnEdge为True, 则Found结果设置为False。
NumberFound	返回在已检测到对象的数目。 (最少是0个, 最多是NumberToFind属性中设定的检测数量)。
Passed	返回视觉对象的测试结果合格与否
PixelX	返回已找到的零件位置(参见模型原点)的X坐标位置(以像素为单位)。
PixelY	返回已找到的零件位置(参见模型原点)的Y坐标位置(以像素为单位)。
PixelXYU	仅限运行时间。 返回已找到的零件位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
RobotX	返回已查找到的零件位置(参见模型原点)相对于机器人坐标系的X坐标位置。
RobotY	返回已查找到的零件位置(参见模型原点)相对于机器人坐标系的Y坐标位置。
RobotU	返回在机器人坐标系内已找到的零件位置的U坐标位置。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的零件位置的RobotX、RobotY和RobotU坐标。
Scale	返回执行过程中检测到的对象缩放比例值。
Score	返回一个0~1000的INTEGER值, 这个值代表运行时间内找到的对象与Correlation对象正在搜索的模型的匹配程度。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。 这样便于对比结果。
SkewDirection	返回执行过程中检测到的对象偏斜方向。
SkewRatio	返回执行过程中检测到的对象偏斜率。
Time	返回处理对象所需要的时间(单位: 毫秒)。
TimedOut	返回对象运行时间是否超时

### 了解归一化相关

Correlation 的目的是在搜索窗口中定位和测量一个或多个预先注册的对象。Correlation 对象可用于查找零件、检测检查零件是否存在、检测缺陷和其他各种查找应用。

虽然 Vision Guide 7.0 有多种视觉对象工具，但 Correlation 对象是最常用的工具，因为其速度快、整体可靠性高。

例如，在很多应用中会使用 Edge 工具来查找对象的边缘。但是，如果在相同区域内有多个边缘而造成 Edge 对象变得混乱时，可以使用 Correlation 对象来查找边缘位置。

另外，当可以使用注册模型时，则可以使用更可靠的 Correlation 对象代替 Blob 对象。

本节将介绍可应用于 Correlation 对象的搜索工具的基本知识。内容如下

- 对象物和模型：说明
- 搜索的基本概念
- 归一化相关
- 归一化相关形状得分
- 归一化相关优化(Accept 和 Confusion)
- 设置 Accept 和 Confusion 的属性值
- 关于 Accept 属性和 Confusion 属性的应用知识
- Correlation 对象和旋转
- 注册模型用于角度搜索
- 搜索重复性和精度
- 校准相机摄像距离



### Correlation对象的模型和对象物

使用 Correlation 对象时，了解对象物和模型的区别很重要。

对象物是搜索窗口中任何特定的灰度模式。对象物可以是某区域中由少数像素组成的简单的边缘，也可以是某区域中由数以万计的像素组成的复杂图案。

相关性搜索操作，则可以测量相关区域中的对象物，与事前注册的模型的匹配程度。

对象物是存在于实际的搜索窗口内的，而模型或模板则是对对象物的理想化表示。

模型是用于表示对象物的灰度模式。这相当于模板匹配系统中的模板。

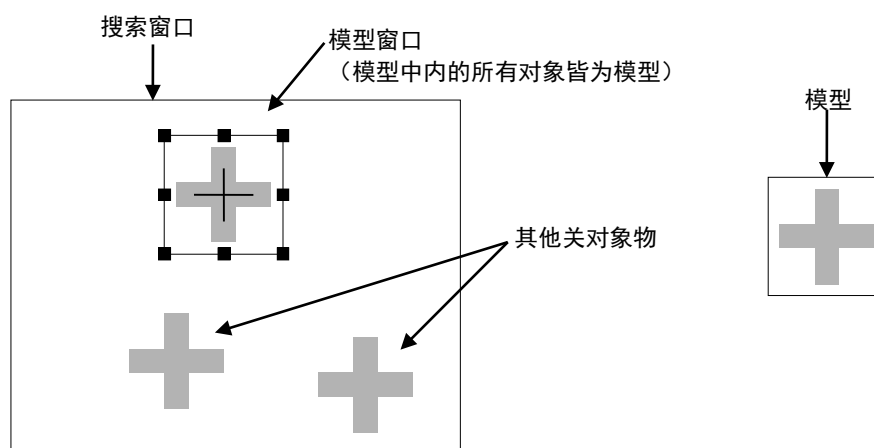
如果模型有两个灰度级时，则为二进制模型。如果模型有两个以上的灰度级时，则为灰色模型。Vision Guide 7.0使用的所有模型均为灰色模型，因为灰色模型比二进制模型的功能更强大，可以更加接近真实的还原测量对象。使用灰色模型，可以获得更可靠的测量结果。

最常见的搜索方法是，在一个搜索窗口注册具有代表性的模型，然后在该搜索窗口内搜索相似的对象物。下图就是一个包含测试对象物(十字形)的搜索窗口。

若要示教这个十字形图案的模型，需要定义模型窗口后，点击执行面板上的<Teach>按钮。

(示教模型详细信息，请参阅本章后面的 *使用Correlation对象*。)

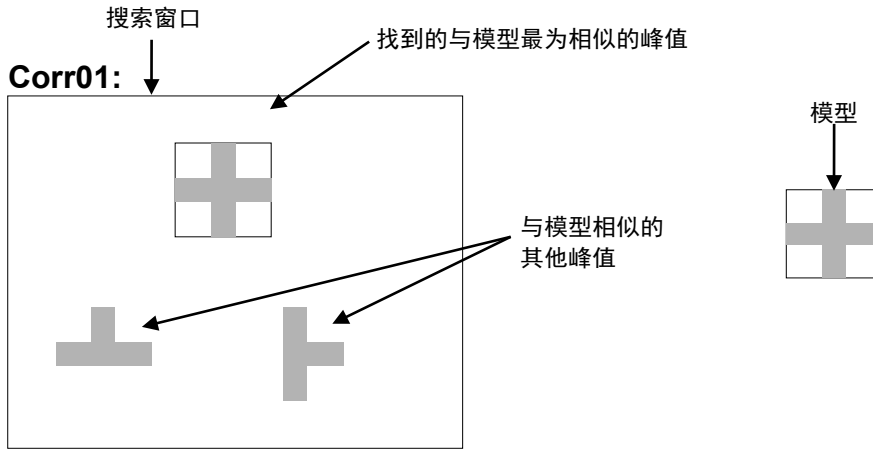
结果，建立的模型如右图所示，且该模型可用于在搜索窗口中搜索其他十字形符号。



此搜索窗口含有多个检测对象(左侧)和根据图像建立的模型(右侧)

### 搜索的基本概念

搜索是指在搜索窗口中找到和模型最相似的区域，从而找到搜索对象。下图为模型、搜索窗口和搜索窗口内与模型最相似(峰值)的区域。这类模型可用于搜索对象，例如印刷电路板上的基准标记。机器人可以利用搜索功能返回的位置数据，确定电路板上零件的安装位置或电路板本身的位置。



搜索窗口、模型和峰值

在搜索窗口内查找模型的方法有很多，但最常见的方法是穷举搜索法。这种方法是在搜索窗口中的所有位置对模型进行评估，然后返回匹配度最高的位置。通过这种穷举搜索法得到的结果准确性很高，但速度可能很慢。例如，假设搜索窗口是一个 36x36 像素的正方形，而模型是一个 6 像素的正方形。这种情况下为了找到和模型一致的位置，则需要对 961 个所有可能的位置进行评估。

但是 Vision Guide 7.0 搜索方法这种穷举搜索法进行了很大改良。首先，扫描搜索窗口以找到潜在的匹配位置，然后仅对这些位置进行模型评估。最后返回匹配度最高的位置。改良后的方法比穷举搜索法的速度要快几千倍。

### 归一化相关

归一化相关是指，排除搜索窗口内线性差异和模型亮度线性差异的影响，仅根据图像和模型之间的几何相似程度来检测对象的方法。

归一化相关的功能非常强大且稳定，所以我们将其用于 Vision Guide 7.0 的 Correlation 对象的搜索运算法则。即使在以下情况，归一化相关的运算值也不会因受到影响而改变。

- 所有搜索窗口或模型像素均乘以某个常数。
- 在所有搜索窗口或模型像素中增加某个常数。

归一化相关最重要的特点是，归一化相关的值，不会受到搜索窗口内线性差异和模型亮度线性差异的影响。这个属性非常重要。

因为图像的整体亮度和对比度都是由诸如光照强度、景物反射率、相机光圈、传感器增益和偏移量(包括可能的自动增益控制电路等)、以及图像数字转换器的增益和偏移量等因素决定的。而所有这些因素在大多数生产情况下都很难予以控制。例如，灯泡会随时间变化，周围的光线会随着一天中时间的推移而改变，相机和数字转换器可能会发生故障需要更换，所以被检测出对象的反射率也有可能发生变化。

归一化相关的另一重要特点是，形状得分值(请参阅下一节，归一化相关形状得分)具有绝对意义。也就是说，可以对所有的模型和搜索窗口，定义完全相同的匹配项。

该属性不仅不会受到搜索窗口内线性差异和模型亮度线性差异的影响，也不会受到模型亮度、对比度和尺寸的影响。因此，形状得分可用于测量检查应用中检测对象的质量。

如果计算出搜索窗口和模型形状得分为 900，则搜索窗口和模型非常相似。

如果形状得分 100，则搜索窗口和模型并不相似。

即使对模型或搜索窗口一无所知，这些属性也仍然有效。也就是说，归相关系数具有绝对意义。

### 归一化相关的形状得分

在归一化相关中，被检测对象的形状得分(与模型的相似程度)定义为0到1000的数值。形状分数越高，检测对象和模型的相似度越高。(形状得分数值为1000说明完全匹配。)

形状得分是Correlation对象返回的作为Score结果的值。有关Score结果的更多信息，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual*。

### 优化归一化相关 (Accept和Confusion)

对于大多数实际的搜索应用中，传统的模板匹配需要很长的处理时间。因为传统的模板匹配会对所有可能的位置进行穷举搜索。

为了解决这个问题，需要采取适当的定向搜索方式。所谓定向搜索是指，在搜索过程中根据检测到的数据来排除没有可能匹配的位置，然后只对有匹配可能的位置进行搜索。

相关搜索采用一种被称为登山法的定向搜索方式。所谓登山法是指，可通过始终在最陡峭的上坡度移动而在本地数据中完全查找到一个或多个变量函数的峰值。当每个邻近点都低于该点时，就达到了峰值。

传统的登山法，有以下缺点：

- 根据起点位置的不同，可能会将错误的最大值误判为峰值。
- 当到达平稳状态时，会无法判断继续搜索的方向。

而当起点太少时，可能会错过一些重要的山丘。而当起点太多时，可能会从不同的方向爬上同一个山丘，从而阻碍了登山法在速度上的优势。

但是 Vision Guide 7.0 则可以通过搜索窗口和模型对山丘进行估算，从而克服这些问题。

相关搜索函数在数学上相当于滤波操作。相关函数是特定的已知过滤器(模型)的输出，该过滤器使特定的空间频率增强并使其他的空间频率减弱。另外，如果搜索窗口的已知部分的频率组成，不同于模型的频率组成，就没有必要在此区域放置登山起点。

通过检查模型的转移函数，系统可以估计相关函数的空间频率组成，并依次估计山丘的最小空间和尺寸。了解最小空间可以让系统计划从哪开始登山。而了解最小尺寸则可以避免系统采用错误的峰值。

除从模型获得的信息外，还可以通过定义一组 Vision Guide 属性(即 Accept 属性和 Confusion 属性)来控制搜索。

Accept 属性指定形状分数，当得分大于或等于此形状分数时，对象才能视为“被找到”(即 Found 结果返回为“True”)。如果预估的山丘高度并未超过搜索窗口指定区域中 Accept 的属性值时，则该区域的登山结束。

Confusion 属性说明搜索窗口中预期的复杂程度。具体来说，就是您正在搜索的对象之外，其他对象物的最大形状得分。这将向系统给出关于待搜索场景的重要提示。即如果对象物获得一个高于混淆阈值的形状得分，其必须是正在搜索的对象。

系统使用 Confusion 属性和预测的结果数量(通过 NumberToFind 指定)，判断有多少山丘要爬，多少山丘不用爬。具体地说，一旦被搜索对象的形状得分高于 Confusion 属性阈值和 Accept 属性阈值时，一旦检测到预期数量时，搜索可以终止。

要开始搜索，必须并行执行多次登山法，并以模型的转移函数确定的多个位置为起点。随着登山法的进行，对每个山丘的预估值精度将逐渐提高，最终找到那个真正的峰值。

登山法是在多个山丘上并行运行，并且每次只进行一个步骤，直到所有山丘的预计高度都低于 Confusion 属性设置的阈值为止。当爬升到 Confusion 属性的阈值时，则是该山坡的峰值。

### 设置Accept和Confusion属性的阈值

Accept属性阈值和Confusion属性阈值会影响Correlation对象的搜索速度。

Accept属性可影响搜索速度，即提供一个提示，说明在何时在某个给定的场景区域内继续进行搜索。

如果Accept属性设置得较高，则要求被检测对象必须与模型非常相似。这样通过粗略检查即可排除掉很多区域，不必继续在这些区域内搜索。

如果Accept属性设置得较低，则与模型仅稍稍相似的对象均可能超过Accept属性阈值，这样就需要对更多的区域进行更详细的检查。

所以，Accept属性值越高，Geometric对象的运行速度越快，也就是检测速度也就越快。

Confusion属性与预测的结果数量相互作用，而该数量则会影响搜索速度。Confusion当Confusion属性和预期的结果数量一致时，则可以在完成搜索所有图像的可搜索区域前，提前中止搜索。

通过设置Accept属性，您可以控制系统找到容差范围内最差的对象物。误差可能是由缺陷、缩放、旋转或视频噪音造成的。

Vision Guide 7.0 中 Accept 属性的默认值设置为 700。对于大多数应用来说，该值都是一个很好的起点。但是，您可以通过反复的尝试和运行来优化这个值。

请注意，并不是说您总需要获得完美或者接近完美的分数才能使应用正常运行。根据被检测对象的变形程度，在有的应用中，即使将形状得分设为 200，也可以获得准确的位置信息。但是，对于大多数的应用，一般建议将 Accept 属性的形状得分设置在 500 以上。

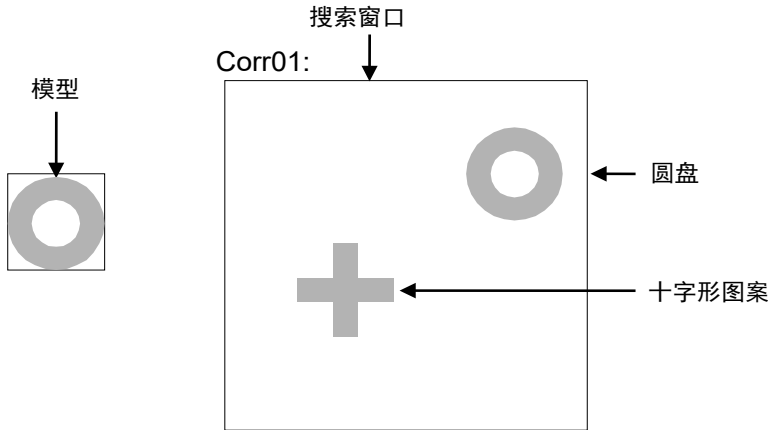
根据您可以接受的“最差的结果”(+容差范围)的最高值，来设置 Confusion 属性。

Confusion属性的阈值宜大于或等于Accept属性阈值。Confusion属性的值越高，搜索的时间就越长，但是搜索的准确度也越高。

Confusion 属性的默认值为 800，您可以根据实际运用的需要进行调整。

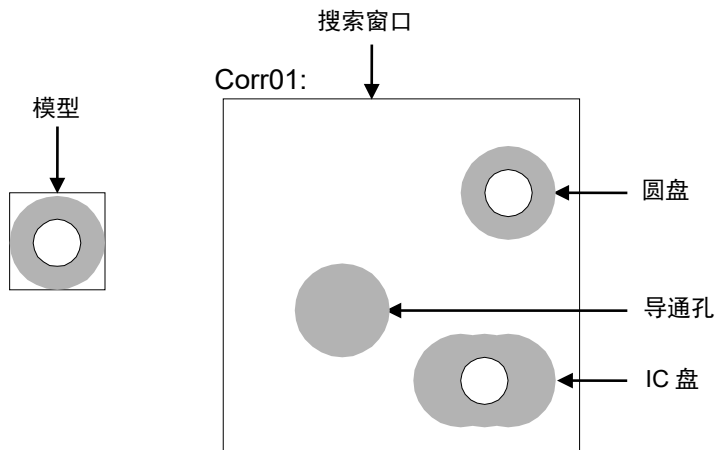
下图是一个比较简单的场景：圆盘与十字形图案几乎没有相似之处。因此，Confusion属性可以设置为很低的值(500左右)。

Accept属性值根据容差量进行设定，一般都会小于或等于Confusion属性。但在这个例子中，由于几乎没有变形，所以将形状得分设置为920是最合适的。



比较简单的场景

在下图是一个比较复杂的场景：导通孔和 IC 盘都与圆盘相似。因此，Confusion 属性宜设置为一个较高的值(820 左右)。



比较复杂的场景

### 关于Accept属性和Confusion属性的应用知识

具有灰度常数区域的搜索窗口将始终在该区域获得相关值 0。如果某个情形下的背景基本统一(例如一张白纸)，则在大多数区域均无关联。因此，如果 Correlation 对象查找到任何东西，都应该是需要搜索的目标对象，所以可以将Confusion属性设置为一个较低的值。

Accept 和 Confusion 属性可以被视为您提供给系统的提示，以便让系统能够更快地定位检测对象。

总的来说，应保守设置这些属性，但设置不用太精确。最保守的设置方法是，将Accept属性设低，将Confusion属性设高。

如果对要搜索的场景了解较少时，建议采用保守的设置。这样搜索的结果是精准的，但是搜索速度比较慢(当使用 `Correlation` 属性的位置结果来控制机器人的动作时，这一点尤其重要)。

如果对要搜索的场景非常了解，则建议采用更灵活的设置。例如，当您正在搜索的对象周围没有其他特征，场景剩余部分都为空白时，则可以灵活设置属性值。这样就不会进行精确搜索，提高搜索速度。

### 采用多结果对话框排除搜索问题

有时，即使是在相同的制作批次内，正在处理的零件也可能变化很大，或者是一个零件上有两个或更多的相似对象。此时，即使将 `Accept` 属性设置为合适的值，也可能会搜索到匹配这个值的错误的对象。在这类应用中，很能预测到会发生什么问题。

我们创建了 `ShowAllResults` 对话框来帮您解决上述以及各种其他问题。

虽然您只想检测一个零件上的一个对象物特征，经过多测检测后，您可以找出为什么 `Vision Guide 7.0` 有时会搜索到其他并不想要得到的对象。通常为情况下，是由于以下原因造成的。

1. 搜索窗口内有两个或更多对象物非常相似，且有着很接近的 `Score` 结果分值。
2. 设置的 `Confusion` 或 `Accept` 属性值不够高，使得得分低于目标对象的错误对象的得分，也符合 `Accept` 属性设定的值。

对于初次使用 `Vision Guide 7.0` 的用户来说，当您在搜索窗口中仅搜索单个对象时，以上两种情况可能会经常发生。

如果遇到有时已查找到您正在搜索的对象，而有时却又查找到其他对象的情况，则请使用 `ShowAllResults` 对话框确认问题所在。以下是查找问题根源的方法：

- (1) 将 `NumberToFind` 属性设置为 3 或以上。
- (2) 在 `Vision Guide 7.0` 开发环境下运行视觉对象。
- (3) 点击<`ShowAllResults`>属性按钮，将显示 `ShowAllResults` 对话框。
- (4) 检查搜索到的前三个或更多对象的得分。
- (5) 如果只查找到一个或两个对象(`Vision Guide 7.0` 将只为这些认为已查找到的特征设置得分)，为了查找多更多对象，请降低 `Accept` 属性并再次运行视觉对象。(确认完 `ShowAllResults` 对话框后，可以将 `Accept` 调回正常设定值)
- (6) 点击<`ShowAllResults`>属性按钮，将出现 `ShowAllResults` 对话框。
- (7) 检查搜索到的前三个或更多的对象的得分。

如上所述，确认了搜索到的前三个或更多的对象的得分之后，将揭晓问题的根源所在。一般情况下产生这些问题的原因如下：

1. 每个检测到的对象的得分都大于 **Accept** 属性中设置的值。如果是这种情况，请将 **Confusion** 属性的值调高，以确保检测出正确的对象，而不返回那些仅与 **Accept** 属性阈值一致的对象。当然也可以通过修改 **Accept** 属性的值来改善这一问题。
2. 每个对象物的得分都非常相似。如果是这种情况，请按照以下步骤区分被检测对象。
  - 重新调整搜索窗口，确保作为被检测出对象而随机返回的对象物，不会出现在搜索窗口中。
  - 重新示教要检测对象的模型。
  - 调整周围的照明灯光，以确保要检测的对象的得分，可以高于错误检测对象的得分。

关于使用多结果的更多详细信息，请参阅本章后续章节 *6.2.24 使用来自单个对象的多个结果*。



### Correlation对象和旋转

在模块匹配过程中，如果检测对象和模型的大小或角度存在差异，则形状得分和定位精准度会降低。如果差异很大，形状得分将很低，或者搜索操作将无法查找到特征。

角度和调整大小的容差量因模型而异，但通常角度的范围在  $3^{\circ} \sim 10^{\circ}$  之间，而大小的范围在  $2 \sim 5\%$  之间。但也有例外情况，例如无关于角度的圆形点对称模型，或是无关于尺寸的仅由一条边和一个角组成的模型等。

在搜索窗口内设想两种不同的场景。

第一个场景是一张人脸的照片，鼻子被示教为模型。由于不认为鼻子是 XY 对称，所以，旋转将大大影响该检测对象的位置精度。第二个场景是一个电路板，上面的基准点(例如前文图示中的十字形)作为模型。由于基准点(十字形)是 XY 对称的，所以，旋转也不会对该检测对象的位置精度产生直接的影响。

基本上，细微特征明显的模型(如鼻子、花或者树的照片等)受旋转的影响较大。而对称的对象(如十字形)受旋转的影响较小。

但是，出于这种考虑，建议使用 Polar 对象确定旋转角度。Correlation 对象可用于检测 XY 位置，Polar 对象可与 Correlation 对象结合使用，这样就可将 XY 位置作为中心以查找对象角度。有关如何关联 Polar 对象和 Correlation 对象以及其他视觉对象的更多信息，请参阅本章的 6.2.6 Polar 对象 。


如果预先知道会出现大角度旋转或是尺寸变化的情况下，有很多方法可以解决。以下介绍几个常用的方法。

- 将复杂的对象分解为更小、更简单的对象。总的来说，与大型的复杂模型相比，小而简单的模型对比例和角度变化的敏感度要低很多。
- 处理 Correlation 对象时，可使用角度相关的属性(AngleEnable、AngleRange 和 AngleMaxIncrement)来检测旋转角度。当不能将对象分解为简单对象时，这对定位复杂模型时很有用。但是，此功能是通过在不同角度的一系列模型来运行的，所以搜索过程会比常规搜索慢许多倍。

#### NOTE



在 Correlation 对象中使用角度检测功能时，在示教模型时，必须将 AngleEnable 属性设置为 True。通过这种设置，则可以在 AngleRange 属性和 AngleMaxIncrement 属性定义的各种角度中，注册 Correlation 对象。

- 同时使用 Polar 对象和 Correlation 对象来确定零件的旋转角度。(有关详细信息，请参阅本章的 6.2.6 Polar 对象 。

### 示教模型用于角度搜索

要执行带有角度检测功能的搜索，需要先在系统中示教一系列的旋转模型。

首先，将 `AngleEnable` 属性设置为 `True`，然后使用 `AngleRange` 属性指定将要进行示教的模型的角度范围。使用这种方式示教旋转的模型时，则会在指定范围内，自动创建一组等间隔的，具有不同角度的旋转模型。

并且，在角度范围内，还可以指定模型注册的最大角度增量。这可以通过设置 `Correlation` 对象的 `AngleMaxIncrement` 属性的增量值来实现。

但是，请记住有关 `AngleMaxIncrement` 属性的以下内容：

- 指定最大的角度增量后，模型注册功能将选择自动角度增量，并在所选择的增量值和指定的最大角度增量中，自动选择那个较小的值并使用。
- 如果将 `AngleMaxIncrement` 属性设置为 0，模型示教功能将选择角度的自动增量并使用该角度增量。在这种情况下，系统通常将角度增量设置为  $2^{\circ} \sim 5^{\circ}$  之间。这导致最低的模型存储要求和最快的搜索次数，但也可能产生大致符合预期的效果。

如果要精确的测量角度，则需要将 `AngleMaxIncrement` 属性设置为与所需角度精度相对应的增量。但需要注意的是，角度增量越小，模型所需的空间越大，搜索速度越慢。

#### NOTE



我们推荐尽可能采用 `Polar` 对象确定角度。`Polar` 对象可以提供更可靠、更精确的结果，而使用机器人向导的视觉功能时需要这些结果。

当示教具有角度检测功能的模型时，搜索窗口必须足够大，以保证模型旋转时，模型的任何部分都不会突出到搜索窗口以外。

### 搜索重复性和精度

搜索重复性和准确度会受到，模型的尺寸、详细信息(形状、对象的粗糙度和形状的对称性)以及在搜索窗口中检测对象的变形(噪音、缺陷以及旋转和比例)的影响。

为测量不相关噪音对位置的影响，请在包含未变形的完整对象的特定搜索窗口内执行搜索，然后在不更改对象位置的情况下再次进行完全相同的搜索(在帧缓冲器中存储第二个图像)，然后再对比已测量的位置。

请按照步骤设置：

1. 模型校准后，多次点击执行面板上对象的<Run>按钮。
2. 点击<Statistics>工具栏按钮。
3. 然后可以使用 **Statistics** 对话框查看两个对象搜索在位置上的区别。

对于没有变形的检测对象的大型模型 (30x30)，可以获得 1/20 像素的重复定位精度。然而，在大多数情况下，获得低于 1 像素的结果更为现实。(1/2、1/3 或 1/4 像素)

可通过对包含降级特征的定向搜索窗口进行搜索而测量搜索准确度，将对象移动一段确切的距离，然后对比已报告的位置差异和实际差异。如果您的模型较大(30x30 或更大)，无降级、无旋转也无比例错误，且在 X 方向和 Y 方向上都有足够的边线，则搜索可精确到 1/4 像素。(记住：这个搜索精度只针对视觉系统，而不考虑所有机器人本身具有的误差。所以，如果您想利用机器人移动这个零件，您就必须考虑机器人机构自身的误差。)

旋转和比例对搜索准确度的影响取决于模型：

- 旋转对称的模型效果好。
- 有很好特征但不对称的模型效果不好。

### 校准相机摄像距离

若要得到最佳的搜索结果，图像内的特征在搜索时的尺寸宜等于它示教模型时的尺寸。

假定使用相同的相机和镜头，如果相机在训练模型时的摄像距离和进行搜索时的摄像距离发生了变化，则搜索窗口内的那个特征就会呈现不同的外观尺寸。也就是说，如果相机与特征离得越近，则特征就显得越大；如果相机与特征离得越远，特征就显得越小。




如果相机摄像距离改变，您就应该重新示教模型。

### 使用Correlation对象

到此您已经了解了归一化相关和搜索的工作原理，也就是已经获得了理解如何使用 Vision Guide 7.0 Correlation 对象的基础知识。

本节将介绍使用 Correlation 对象的每个步骤。


- 新建一个 Correlation 对象
- 确定搜索窗口的位置和尺寸
- 确定模型窗口的位置和尺寸
- 确定模型原点的位置
- 配置与 Correlation 对象相关的属性
- 示教模型
- 测试 Correlation 对象并检查结果
- 调整属性并再次测试
- 处理单个 Correlation 对象的多种结果

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

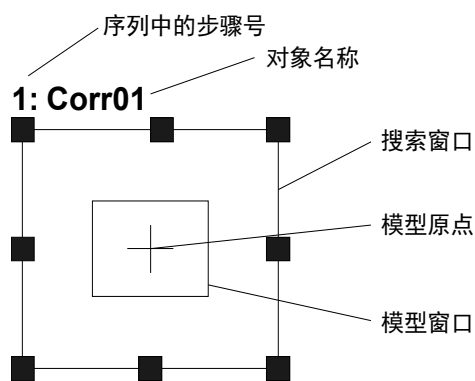
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：新建一个Correlation对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>- <New Correlation >按钮。
- (2) 鼠标光标将变成 Correlation 图标。
- (3) 移动鼠标将 Correlation 拖至 Vision Guide 窗口的图像显示区域。点击鼠标左键将 Correlation 对象放置在图像显示区域上。
- (4) 注意：对象名称将自动创建。本示例中创建的对象名为 Corr01，因为它是在本序列中创建的第一个 Correlation 对象。(稍后将说明如何修改名称。)

### 第2步：确定搜索窗口的位置和尺寸

画面上会显示一个如下所示的Correlation对象：



新Correlation对象布局

- (1) 点击 Correlation 对象的名称标签(或在某一侧)，按住鼠标左键，将 Correlation 对象拖至您想让搜索窗口的左上角所在的位置。
- (2) 点击搜索窗口的尺寸手柄，拖动鼠标，将 Correlation 对象搜索窗口调整成合适的大小。(搜索窗口就是要搜索的区域。)



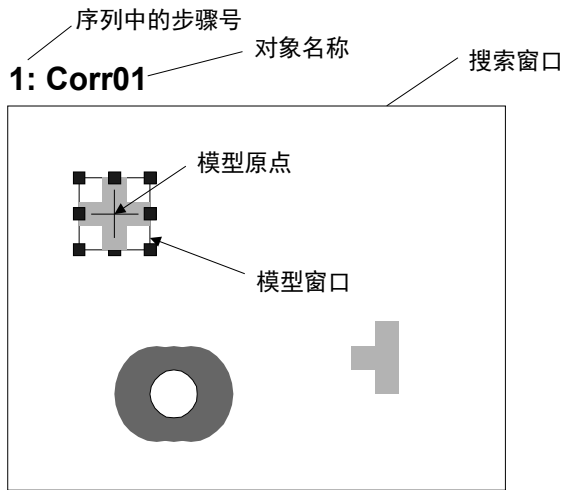
注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。

### 第3步：确定模型窗口的位置和尺寸

- (1) 正在运行中的 Correlation 对象搜索窗口会呈现洋红色，并且在四个角上和四条边的中央位置都会显示尺寸手柄。如果没有显示尺寸手柄，请点击 Correlation 对象名称字段。当运行中的 Correlation 的尺寸手柄均已显示，并窗口呈现洋红色时，则可继续步骤 2 的操作。
- (2) 点击模型窗口方框上的任一条边线。此操作会让模型窗口突出显示，并显示到模型窗口上的尺寸手柄。
- (3) 点击模型窗口方框上的任一条边线，按住鼠标左键，将模型窗口拖动至您想让模型窗口的左上角所在的位置。
- (4) 点击尺寸手柄拖动鼠标，调整模型窗口的大小，确保其可以囊括被示教为 Correlation 对象的模型。

Correlation 对象布局如下所示，搜索窗口是整个被检测区域，而模型窗口囊括了被搜索对象。实际的搜索窗口和模型窗口的形态可能会和下图显示的有所差异，此处的图像只是个大概的参考。



搜索窗口和模型窗口的位置和尺寸调整后的Correlation对象

---

正确设置模型窗口的尺寸和位置的技巧：

模型窗口的尺寸和位置十分重要，因为它们定义了将要搜索的对象。创建 Correlation 对象的模型窗口时，请注意以下两点：

- (3) 搜索窗口既是被搜索的区域，窗口越小搜索的时间越短。特别是如果预先知道被搜索的零件会有较大的角度旋转，那么尽量缩小搜索窗口和模型(零件的一部分等)，则可减少零件旋转造成的影响。
- (4) 使模型窗口的尺寸尽量接近搜索窗口的尺寸，以缩短搜索时间。

另外，如果两个零件位置相邻甚至相互接触时，将模型窗口设置的稍大于(几个像素)实际检测对象时，可以帮助区分目标对象和其他对象。

请注意，模型窗口的最佳尺寸取决于每个实际应用的具体情况。

---

#### 第4步：确定模型原点的位置

模型原点定义了模型上的位置，当运行 **Correlation** 对象时，该位置将返回为被检测对象的位置。也就是说，当位置信息非常关键时，需要将模型原点设置在重要的位置。

例如，在使用 **Correlation** 对象检测零件，控制机器人取放零件应用时，机器人能否轻松的抓取零件，模型原点的位置是一个重要的因素。

因为这将是机器人根据 **RobotX**, **RobotY**, **RobotU**, **RobotXYU** 返回的结果所移动到的那个位置。

创建新的 **Correlation** 对象后，**ModelOrgAutoCenter** 属性的默认设置为 **True**。此时模型原点会自动被设置在模型窗口的中心，且无法手动移动。

若要以手动修改模型原点，须先将 **ModelOrgAutoCenter** 属性设置为 **False**。该属性的设置方法和模型原点定位方法如下。

- (1) 点击 **Vision Guide** 窗口流程图中的 **Correlation** 对象。在对象窗口的属性列表中查找 **ModelOrgAutoCenter** 属性，并点击值字段。
- (2) 下拉列表会显示“**True**”和“**False**”两个选项。点击“**False**”选项。这样就可以将 **ModelOrgAutoCenter** 属性设置为 **False**，也可以使用鼠标拖动模型原点。
- (3) 点击模型窗口以突出显示模型窗口。
- (4) 点击模型原点，按住鼠标左键，将模型原点拖至新位置。应注意的是，模型原点只能位于模型窗口范围内。

#### 第5步：设置Correlation对象属性

设置 **Correlation** 对象属性。要设置属性，仅需点击相关属性值字段，然后再重新输入一个新的值。或者如果显示有下拉列表，则点击该列表中的相应选项即可。

以下是 **Correlation** 对象的一些常用属性。

测试 **Correlation** 对象时，不需要设置这些属性。但如果您是第一次使用 **Correlation** 对象，请参考以下说明。

有关其他常用的视觉对象的详细信息，如 **AbortSeqOnFail** 和 **Graphics** 等，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual* 或 *Correlation 对象属性列表*。



注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。  
当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。  
正确配置 **Accept** 和 **RejectOnEdge** 等属性值，可降低错误检测的风险。

Name 属性	<p>新创建的Correlation对象中，默认的项目名称是“Corr**”。在一个视觉序列中要使用多个Correlation对象时，使用**的编号对Correlation进行区分。默认情况下，会将在视觉序列序列中使用的第一个Correlation对象命名为“Corr01”。</p> <p>需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示Correlation对象名称的地方都会变为新的名称。</p>
Accept 属性	<p>Accept属性用于设置形状分值的标准，被检测对象必须与其一致，或者符合其设定的标准时，才能被找到。</p> <p>Score结果的返回值将与这个Accept属性的值进行对比。</p> <p>默认值为700，该默认值适合首次运行Correlation对象时使用。</p>
Confusion 属性	<p>如果搜索窗口内有很多相似的对象，Confusion属性可帮助您找到正确的被搜索对象。</p> <p>默认值为800，该默认值适合首次运行Correlation对象时使用。</p>
ModelOrgAutoCenter属性	<p>将ModelOrgAutoCenter属性设置为“False”，则可以手动修改模型原点的位置。</p> <p>默认： True</p>
Frame 属性	<p>Frame属性可让您将先前定义的Frame对象选为Correlation对象的参考框。</p> <p>有关Frames的详细信息，请参考Frame对象中的说明。</p>
NumberToFind 属性	<p>根据想要查找的对象的数目，您可能想将NumberToFind属性设置为大于1的数字。这将使一个Correlation对象在一个搜索窗口内查找到多个特征。</p>
AngleEnable 属性	<p>如果您打算利用Correlation模型进行角度搜索，就将此项属性设置为True。若要对多个模型进行角度搜索，则在示教Correlation对象的模型之前将这条属性设置为True。</p>
AngleMaxIncrement、AngleRange 属性	<p>这两个属性可以与AngleEnable属性使用，可以使相关模型开启角度搜索功能。</p>
RejectOnEdge属性	<p>利用这条属性，可以将与搜索窗口边界有接触的那些零件排除在外。通常而言，这条属性宜设置为True。</p>

可以将属性保持为默认属性，并进行下一步。稍后再根据需要修改这些属性。



### 第6步：示教Correlation对象的模型

Correlation 对象执行搜索功能，需要一个模型，这通过被称为示教模型的步骤来实现。在这个步骤中，Correlation 对象的模型窗口应该已经被放置在可囊括模型对象的位置。通过以下步骤完成模型示教。

- (1) 确保Correlation对象是否为当前显示的对象。查看流程图或序列目录树，确认当前正在处理的对象。或者，检查图像显示区域，查看正以洋红色突出显示的对象。
- (2) 点击执行面板上的<Teach>按钮。完成模型示教。多数情况下，示教模型仅需几秒钟。但如果在AngleEnable属性设置为True时示教某个模型，由于要示教多个角度的模型，则会需要较长时间方能完成。

### 第7步：测试Correlation对象并检查结果

要运行 Correlation 对象，点击执行面板上对象的<Run>按钮。

然后将显示Correlation对象的结果。这时需检查的主要结果有：

<b>Found 结果</b>	返回是否检测到了 Correlation 对象。如果找到目标对象，则返回的结果为“True”。如果未找到目标对象，Found 的结果将返回“False”，且以红色突出显示。如果找不到 Correlation 对象，请参考第 8 步了解常见原因。
<b>FoundOnEdge 结果</b>	如果找到的对象物的一部分与搜索窗口的边界接触，则返回的这个结果将为 True。 在这种情况下，Found 结果返回为 False。
<b>Score 结果</b>	这个结果显示与模型最为相似的那个特征与模型匹配的程度。 Score 结果的范围为 0 至 1000，1000 表示模型和被检测对象完全一致。当要确认被检测出的对象与希望检测出的对象的相似程度时，请先确认运行 Correlation 对象之后的 Score 结果。
<b>Time 结果</b>	返回Correlation对象执行所需时间。 搜索窗口和模型窗口越小，搜索速度就越快。
<b>NumberFound 结果</b>	在搜索多个 Correlation 对象时，NumberFound 结果会返回与 Correlation 对象的模型相匹配的对象物的数量。
<b>Angle 结果</b>	Correlation 对象的倾斜角度。 这个角度是依据模型的原始角度计算的。但有时只是个粗略值，可能并不可靠。 建议使用 Polar 对象来确定旋转角度。特别是使用机器人向导时，请使用 Polar 对象。
<b>PixelX 结果</b>	返回测试对象的 XY 坐标值(像素坐标系)。
<b>PixelY 结果</b>	这个计算值反馈的是被检测对象的模型原点的位置。如果要返回其他位置，则需修改模型原点位置，并重新示教模型。
<b>CameraX 结果</b>	在相机坐标系中定义检测对象的 XY 坐标值。
<b>CameraY 结果</b>	如果相机尚未完成校准则不会返回任何值，而是返回[No Cal]。

- RobotX 结果** 在机器人坐标系中定义被检测对象的 XY 坐标值。
- RobotY 结果** 机器人可移动到该 XY 坐标位置(不需要其他转换和操作)。这个计算值反馈的是被检测对象的模型原点的位置。如果您希望返回其他位置，则需修改模型原点位置，并重新示教该模型。如果相机尚未完成校准时不会返回任何值，而是返回[No Cal]。
- RobotU 结果** 返回被检测出对象的角度，该角度是转换成机器人坐标系后的值。如果相机尚未完成校准，则不会返回任何值，而是返回[No Cal]
- ShowAllResults** 当处理多个结果时，点击 ShowAllResults 设置值字段中的按钮，则会显示 ShowAllResults 对话框，可以查看当前视觉对象的所有结果。

### NOTE



在上述步骤示例中，由于尚未完成校准，所以 RobotX,RobotY,RobotU,RobotXYU, CameraX, CameraY, CameraU, CameraXYU 的结果会返回 “No Cal”。这意味着没有进行任何校准，因此视觉系统无法计算与机器人坐标系或相机坐标系相关的坐标结果。想了解更多信息，请参考 7.视觉校准。

### 第8步：调整属性并再次测试

在Geometric对象运行若干次之后，您可能在查找某个Correlation对象时遇到困难，或者您还想微调某些属性设置。下面将介绍一些常见问题的故障排除方法和微调技巧。

#### Correlation 对象故障排除

##### 当 Correlation 对象返回的 Found 结果为 False 时

- 返回的 Score 结果是否低于 Accept 属性设置。如果 Score 结果更低，请降低 Accept 属性的设置值(例如低于当前 Score 结果的值)，并再次运行 Correlation 对象。
- 查看 FoundOnEdge 的结果，是否返回为“True”。如果 Found 结果返回的值为“False”，而 FoundOnEdgeTrue 结果为“True”时，意味着被检测对象的一部分接触到了搜索窗口。此时，请将搜索窗口调大，使其能适应被搜索对象的大小。如果搜索窗口的大小无法调整，请尝试更改相机位置，或调整模型窗口的大小。

##### 当 Correlation 对象检测出错误的对象时

- 检查 Accept 属性是否设置得足够高。如果 Accept 属性设置得很低，则有可能会检测出搜索对象以外的其他对象。
- 请确认 Confusion 属性是否设置得足够高。Confusion 属性值必须大于或等于 Accept 属性值。如果搜索窗口内有与被搜索对象非常相似的对象时，则必须将 Confusion 属性值调为更高的值，以避免检测出错误的识别对象。
- 调整搜索窗口，使它靠近被检测对象，并精准的将其单独隔离出来。

### Correlation 对象微调

通常需要进行Correlation对象的微调，以确保对象能正常动作。

以下是与Correlation对象微调和模型添加相关的主要属性说明：

- |                           |  |
|---------------------------|--|
| <b>Accept属性</b>           | Accept属性的值越低，Geometric对象的运行速度就越快。但是Accept过低时，可能会造成被检测出的对象，不是想要查找的对象。可以尝试多运行几次Geometric对象，一旦Score结果返回的形状分值符合预期结果时，调整Accept属性的值。为了以最佳执行速度并得到可靠的结果，通常需要若干次尝试运行才能找到一个合适的值。 |
| <b>Confusion属性</b>        | 如果搜索窗口中有与被搜索对象非常相似的对象时，则需要将Confusion属性设置得相对高一些。这将区分那些有相似特征的对象，正确的检测出目标对象。但是，Confusion属性值设置的过高时，会降低运行速度。如果搜索窗口没有类似的对象物时，可降低Confusion属性的设定值，以缩短执行时间。                     |
| <b>Add another sample</b> | 当进行示教的模型窗口与当前模型的尺寸一致时，可以选择“Add another sample”。当模型具有轻微变化(形状或图案略微不同，阴影表现不同)时，选择已更改的模型来添加新模型可能会稳定对象执行的得分。如果角度位置大幅度偏移，或者因差异较大而无法添加模型，原始模型将保持不变                            |

当完成调整，并测试Correlation对象得到满意的结果后，创建视觉对象的步骤就完成了。下一步是继续创建其他视觉对象或者配置并测试整个视觉系列

### Correlation对象的其他实用工具

Vision Guide 7.0 中有一个非常实用的 Histogram 功能，该功能可以在搜索窗口中直观的显示灰度值的分布情况。使用 Vision Guide 直方图功能，可以检查 Geometric 对象的图像的状态。关于 Vision Guide 直方图功能的详细信息，请参考 8.1 利用直方图。

还可以使用Vision Guide统计功能检查Correlation对象的统计结果。在9. 使用Vision Guide统计中说明了Vision Guide统计功能。

### 6.2.4 Blob对象

#### Blob对象描述

Blob 对象是利用二进制检查的工具，来计算图像的几何、拓扑和其他特征。

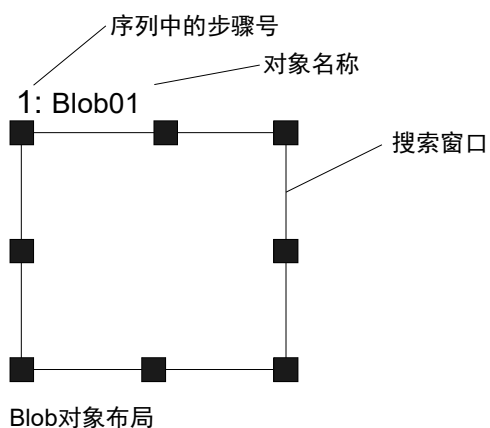
Blob 在确认图片中是否有被检测对象、尺寸和方向上很有用。例如，Blob 对象可用于探测硅片上的墨点的存在、尺寸和位置，以确定零件的方向或甚至用于机器人向导(但是在确认旋转方向时，建议您使用 Polar 对象)。

Blob对象计算的一些几何学特征：

- 面积和周长
- 重心
- 主轴和力矩
- 连接性
- Extrema
- 像素坐标系、相机坐标系、机器人坐标系中重心的坐标位置
- 二进制对象的孔洞数目、粗糙度和密实度

#### Blob对象布局

Blob对象布局与Correlation对象一样是一个矩形。但是，Blob对象没有模型。这意味着Blob对象不需要模型窗口和模型原点。如下图所示，Blob对象仅有一个对象名称和一个搜索窗口。搜索窗口定义了要搜索Blob对象的区域。Blob对象布局如下所示：



### Blob对象属性

以下列表是对Blob对象的属性进行的简单说明。关于各属性的详细信息，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual*。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为Blob对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则Blob对象将应用于指定视觉对象的所有结果(NumberFound) 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 如果将 SearchWinType 设置为 RotatedRectangle，搜索窗口则以 Angle 结果为基础进行旋转。 默认: False
CheckClearanceFor	设定确认间隙的对象
ClearanceCondition	设定间隙的判断方法
CoordObject	指定复制结果的 Coordinates 对象 执行对象时进行复制处理，若没有通过 Decision 等分支功能执行，则不进行复制处理。 默认: None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。 默认: 1
Description	设定用户指定的描述 默认: 空白

属性	说明
EditWindow	定义被检测区域的不要紧的像素
Enabled	指定是否执行对象。 默认: True
FailColor	选择当对象被检测为不良品时, 搜索窗口显示的颜色。 默认: Red
FillHoles	指定是否填充二进制图像的孔洞。 默认: False
Frame	指定要使用哪个位置的定位框。 默认: None
FrameResult	指定需使用Frame的第几个结果。 默认: 1
Graphics	设置显示的图表模式。
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
MaxArea	确定Blob对象的Area上限。被检测到的二进制对象返回的Area结果, 必须小于MaxArea属性的设置值。 默认: 100,000
MinArea	确定Blob对象的Area下限。被检测到的二进制对象返回的Area结果, 必须大于MinArea属性的设置值。 默认: 25
MinMaxArea	仅限运行时间。 用一条语句设置或返回MinArea和MaxArea。
Name	用于为Blob对象分配一个唯一的名称。 默认: Blob01
NumberToFind	规定了在搜索窗口中要查找的Blob对象的数目。 默认: 1
PassColor	选择当对象被检测为良品时, 搜索窗口显示的颜色。 默认: Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认: SomeFound
Polarity	可定义被检测对象和背景色的差异(在浅色背景下检测深色对象(DarkOnLight), 或在深色背景下检测浅色对象(LightOnDark))。 默认: 1 - DarkOnLight
RejectOnEdge	通常情况下设置为True, 排除那些与搜索窗口边缘有接触的零件。 默认: False

属性	说明
SearchWin	仅限运行时间。在一次调用中设置或返回搜索窗口的左侧参数、顶部参数、高度参数和宽度参数或中心X坐标、中心Y坐标、圆内周长的半径大小、圆外周长的半径大小。
SearchWinAngle	定义待搜索区域的角度。
SearchWinAngleEnd	定义待搜索区域的结束角度。
SearchWinAngleStart	定义待搜索区域的开始角度。
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	定义待搜索区域的高度(单位: pixel)。 默认: 100
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。
SearchWinPolygonPointX1	当SearchWinType设置为“Polygon”时, 定义被检测区域的第1个顶点的X坐标值
SearchWinPolygonPointY1	当SearchWinType设置为“Polygon”时, 定义被检测区域的第1个顶点的Y坐标值
SearchWinPolygonPointX2	当SearchWinType设置为“Polygon”时, 定义被检测区域的第2个顶点的X坐标值
SearchWinPolygonPointY2	当SearchWinType设置为“Polygon”时, 定义被检测区域的第2个顶点的Y坐标值
SearchWinPolygonPointX3	当SearchWinType设置为“Polygon”时, 定义被检测区域的第3个顶点的X坐标值
SearchWinPolygonPointY3	当SearchWinType设置为“Polygon”时, 定义被检测区域的第3个顶点的Y坐标值
SearchWinPolygonPointX4	当SearchWinType设置为“Polygon”时, 定义被检测区域的第4个顶点的X坐标值
SearchWinPolygonPointY4	当SearchWinType设置为“Polygon”时, 定义被检测区域的第4个顶点的Y坐标值
SearchWinPolygonPointX5	当SearchWinType设置为“Polygon”时, 定义被检测区域的第5个顶点的X坐标值
SearchWinPolygonPointY5	当SearchWinType设置为“Polygon”时, 定义被检测区域的第5个顶点的Y坐标值
SearchWinPolygonPointX6	当SearchWinType设置为“Polygon”时, 定义被检测区域的第6个顶点的X坐标值
SearchWinPolygonPointY6	当SearchWinType设置为“Polygon”时, 定义被检测区域的第6个顶点的Y坐标值
SearchWinPolygonPointX7	当SearchWinType设置为“Polygon”时, 定义被检测区域的第7个顶点的X坐标值
SearchWinPolygonPointY7	当SearchWinType设置为“Polygon”时, 定义被检测区域的第7个顶点的Y坐标值

属性	说明
SearchWinPolygonPointX8	当SearchWinType设置为“Polygon”时，定义被检测区域的第8个顶点的X坐标值
SearchWinPolygonPointY8	当SearchWinType设置为“Polygon”时，定义被检测区域的第8个顶点的Y坐标值
SearchWinPolygonPointX9	当SearchWinType设置为“Polygon”时，定义被检测区域的第9个顶点的X坐标值
SearchWinPolygonPointY9	当SearchWinType设置为“Polygon”时，定义被检测区域的第9个顶点的Y坐标值
SearchWinPolygonPointX10	当SearchWinType设置为“Polygon”时，定义被检测区域的第10个顶点的X坐标值
SearchWinPolygonPointY10	当SearchWinType设置为“Polygon”时，定义被检测区域的第10个顶点的Y坐标值
SearchWinPolygonPointX11	当SearchWinType设置为“Polygon”时，定义被检测区域的第11个顶点的X坐标值
SearchWinPolygonPointY11	当SearchWinType设置为“Polygon”时，定义被检测区域的第11个顶点的Y坐标值
SearchWinPolygonPointX12	当SearchWinType设置为“Polygon”时，定义被检测区域的第12个顶点的X坐标值
SearchWinPolygonPointY12	当SearchWinType设置为“Polygon”时，定义被检测区域的第12个顶点的Y坐标值
SearchWinRadiusInner	定义待搜索区域的圆弧内径。
SearchWinRadiusOuter	定义待搜索区域的圆弧外径。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinType	定义待搜索区域的类型 (Rectangle,RotatedRectangle,Circle, Arc, Polygon)
SearchWinWidth	定义待搜索区域的宽度(单位: pixel)。 默认: 100
SizeToFind	选择要查找的Blob的尺寸。 默认: 1 - Largest
Sort	选择对象结果的排序方式。 默认: 0 - None
ThresholdAuto	可定义是否自动设置被检测对象、背景、图像的边缘灰度的阈值。 默认: Disables
ThresholdBlockSize	定义了相邻区域的参考范围，以决定当ThresholdMethod属性设定为LocalAdaptive时的阈值。 默认: 1/16ROI
ThresholdColor	定义阈值范围内像素的颜色。 默认: Black



属性	说明
ThresholdHigh	<p>同ThresholdLow属性一起使用，用于定义被检测对象(或对象)、背景和图像边缘的灰度范围。</p> <p>ThresholdHigh属性用于定义图像的被检测对象区域的灰度范围的上限。在ThresholdLow和ThresholdHigh定义的灰度范围内的图像的任何部分的像素权重都会分配为1。(即检测对象的一部分。)</p> <p>如果ThresholdAuto属性为“True”，且ThresholdColor属性为“White”，该属性值将被设为255且无法更改。 默认：128</p>
ThresholdLevel	<p>定义了当ThresholdMethod属性设定为LocalAdaptive时，与相邻区域的亮度差异的比率。 默认：15%</p>
ThresholdLow	<p>同ThresholdHigh属性一起使用，用于确定表示特征(或对象)、背景和图像边缘的像素灰度区域。区域ThresholdLow用于定义图像的被检测对象区域的灰度范围的下限。在ThresholdLow和ThresholdHigh定义的灰度范围内的图像的任何部分的像素权重都会分配为1。(即检测对象的一部分。)</p> <p>如果ThresholdAuto属性为“True”，且ThresholdColor属性为“Black”，该属性值将被设为0且无法更改。 默认：0</p>
ThresholdMethod	设置二值化处理的方法。

### Blob对象结果

以下列表是对Blob对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	返回被检测出零件的旋转量，用度数表示。
Area	返回二进制对象的面积，用像素表示。
CameraX	返回相机坐标系中已找到的零件位置的 X 坐标位置。
CameraY	返回相机坐标系中已找到的零件位置的 Y 坐标位置。
CameraXYU	<p>仅限运行时间。</p> <p>返回相机坐标系中已检测到零件位置的 CameraX、CameraY 和 CameraU 坐标。</p>
ClearanceOK	返回间隙的判断结果。
Compactness	返回二进制对象的密实度。
Extrema	<p>仅限运行时间。</p> <p>返回二进制对象极值的 MinX、MaxX、MinY 和 MaxY 像素坐标。</p>

结果	说明
Found	返回是否检测到了对象。 (即是否查找到了 Area 结果介于 MinArea 和 MaxArea 属性之间的连贯的 Blob。)
FoundOnEdge	当发现 Blob 对象与搜索窗口边缘距离太近时, 则返回“True”。
Holes	返回二进制对象中已查找到的孔洞数目。
MajorDiameter	当检测到的二进制对象近似椭圆时, 返回长直径。
MaxFeretDiameter	返回检测到的缺陷的最大费雷特直径。
MaxX	返回二进制对象极值的最大 X 像素坐标, 用像素表示。
MaxY	返回二进制对象极值的最大 Y 像素坐标, 用像素表示。
MinorDiameter	当检测到的二进制对象近似椭圆时, 返回短直径。
MinX	返回二进制对象极值的最小 X 像素坐标, 用像素表示。
MinY	返回二进制对象极值的最小 Y 像素坐标, 用像素表示。
NumberFound	返回在搜索窗口内检测到的二进制对象数目(最少是 0 个, 最多是 NumberToFind 属性中设定的 Blob 对象检测到的二进制数量)。
Passed	返回测试结果合格与否。
Perimeter	沿已查找到的二进制对象外部边缘的像素数目。
PixelX	返回已找到的零件位置的 X 坐标, 用像素表示。
PixelY	返回已找到的零件位置的 Y 坐标, 用像素表示。
PixelXYU	仅限运行时间。 返回已找到的零件位置的 PixelX、PixelY 和 PixelU 坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到的对象的 X 坐标位置。
RobotY	返回机器人坐标系中检测到的对象的 Y 坐标位置。
RobotU	返回机器人坐标系中检测到的对象的 U 坐标位置。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的零件位置的 RobotX、RobotY 和 RobotU 坐标。
Roughness	返回缺陷的粗糙度。
ShowAllResults	显示允许您在表格中查看指定视觉对象所有结果的一个对话框。当 NumberToFind 属性设置为 0 或大于 1 时, 比较结果将更加容易。
Time	返回处理对象所需要的时间(单位: 毫秒)。
TotalArea	返回所有已查找到的结果的总面积。

#### Blob检查的工作原理

Blob检查过程包括以下步骤:

1. 分割，由以下组成：
  - 阈值
  - 连通性分析
2. Blob 结果计算过程

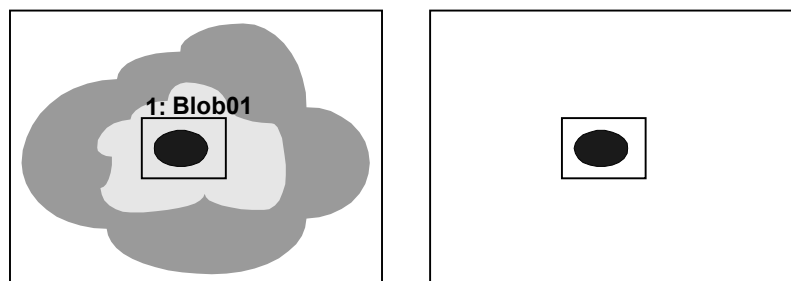
### 分割

在 Blob 检查中，要检测对象的特征，必须先判断出来对象处于图像的哪个位置，也就是要将对象从图像中的其他区域区分开来。这个将图像中的检测对象与其他背景区分出来的过程，则被称为分割。

Blob 可根据每个像素的灰度值用于可以分割的图像。阈值化就是这种分割最简单的一个例子。

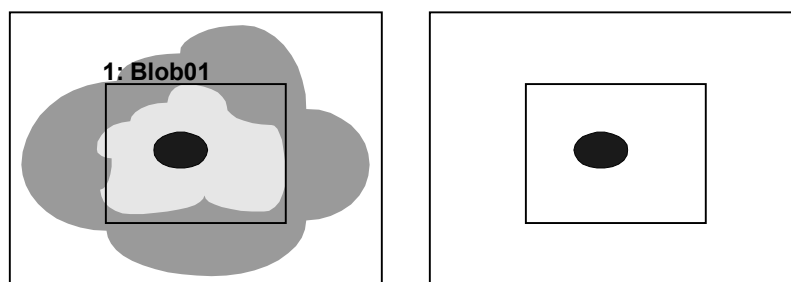
本章内容中所阐述的 Blob 对象，对于那些无法通过灰度值区分的，任何类型的灰度图像都可以提供准确的检测结果。但是这种情况下，这类型图像的使用价值也是有限的。检测的结果会受到定义图像的搜索窗口的尺寸所影响。

在对整个图像进行 Blob 检查时，被检测对象的特征，必须是图像中具有特定的灰度级的唯一对象。如果图像中的另一个对象有相同的灰度值，图像分割将失败。图 A 至图 D 说明在整个图像的 Blob 检查中，哪些可以用灰度值进行分割，哪些不可以用灰度值进行分割。



图A：可以用灰度值进行分割的场景

A 所示为相机的视野(左边)。将要使用 Blob 对象进行处理的场景，将显示在名为“Blob01”的搜索窗口中。使用灰度值完成分割后，可以清楚地区分对象和背景，如图 A 的右边所示。

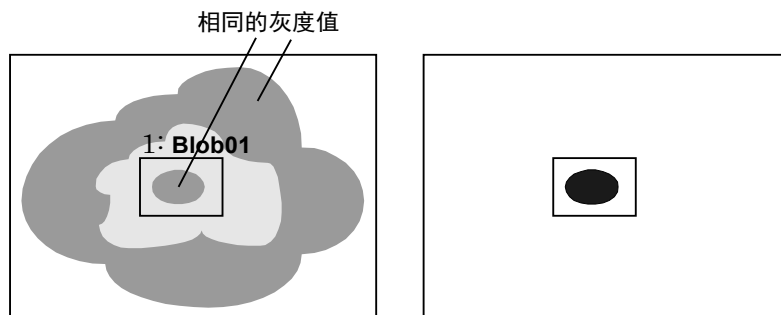


图B：扩大搜索窗口查看图A的场景

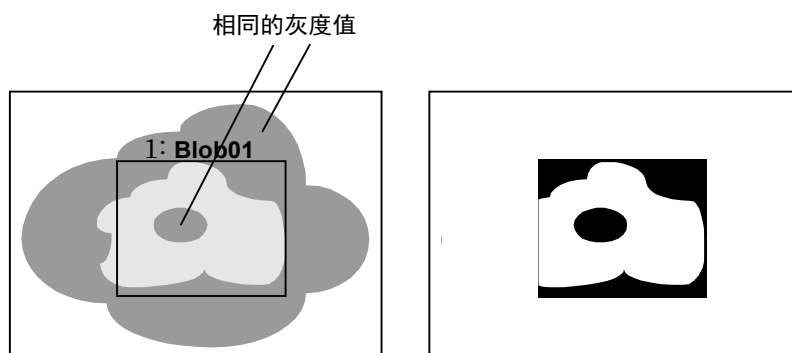
如图 B 所示调整搜索窗口的尺寸时，只有背景的大小发生了变化。对 Blob 对象的特征没有影响。

图C和图D所示为相似的图像视野。但是，在此场景中，图像中有两个相同灰度值的对象。

这种情况下，背景区域和Blob检测特征，两者都会受到搜索窗口尺寸的大小，和被搜索窗口囊括的图像部分的影响。即使可以用灰度值对图C中的图像进行分割，但像图D中那样放大搜索窗口后，被分割的图像也将完全不同。如果不从搜索窗口中删除一个对象，则无法将具有相同灰度值的对象区分开来。

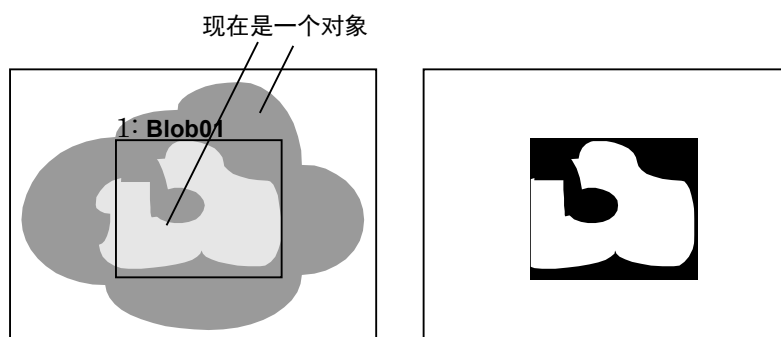


图C：不可以用灰度值进行分割的场景



图D：扩大的搜索窗口中有两个具有相同灰度值的对象

请注意图E中所示的情况。在这个例子中，内部的二进制对象和部分背景相连。所以本来要分割的是中间部分的二进制对象，但却变成完全不同的大型二进制对象。



图E：对象和背景连接在一起

## 阈值

Blob 对象用阈值确定图像中每个像素的权重。

用户定义的阈值有 ThresholdLow 和 ThresholdHigh 两个。灰度值介于两个阈值之间的像素的像素权重为“1”，其他所有的像素权重为“0”。

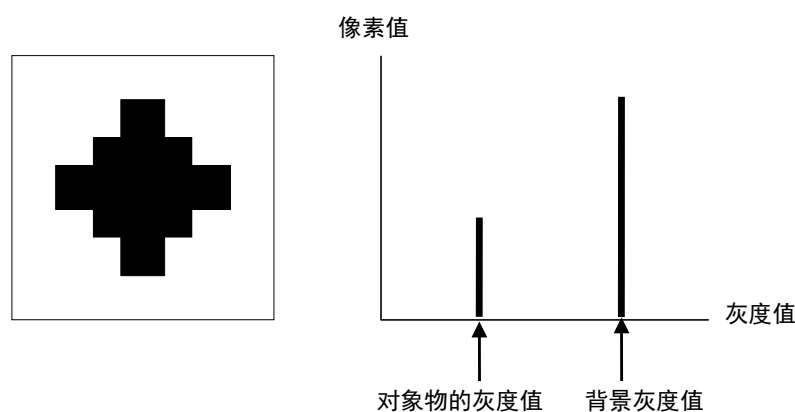
ThresholdColor 属性定义权重为“1”的像素的颜色。这是阈值内的颜色(黑色或白色)。

根据这些像素的权重，Blob 对象将图像分为对象(权重为“1”的像素)和背景(权重为“0”的像素)。Polarity 属性用于配置 Blob 对象，以查找包含黑像素或白像素的二进制对象。Polarity 为“DarkOnLight”时，将会检测包含黑色像素的二进制对象。Polarity 为“LightOnDark”时，将会检测包含白色像素的二进制对象。

## 采用直方图确定阈值

通过使用 Vision Guide 7.0 直方图工具，用户可以确定用于 ThresholdLow 和 ThresholdHigh 的值。

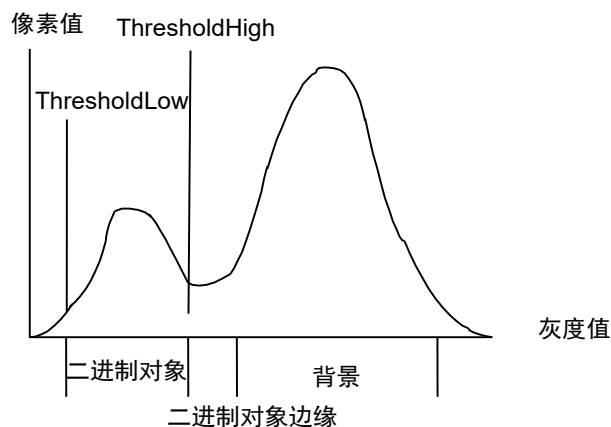
例如，想象在白色背景上有理想的黑色二进制对象的二进制图像。下图为此类图像及其直方图。



理想二进制图像及其直方图

请注意该直方图中仅有两个灰度值为非零值：二进制对象的灰度值和背景的灰度值。

在实际的图像中，永远也不会得到这样的直方图。受到各种噪音的影响(例如不均匀的印刷、不规则的照明设备和电气噪音)峰值会降低。下图所示为更真实的直方图。



使用直方图设置阈值

在上图所示的直方图中，每个峰值都很明显。峰面积与先前的理想直方图的峰面积比例相同，但是此时的峰值在人字形中更加分散，并且有两个或以上的灰度值。

两个主要的峰值之间的山谷部分的灰度值，代表了二进制对象的边缘，并不是完全的深色或浅色。

请调整 Threshold 值，使二进制对象的权重为“1”的像素。

连通性(已连接二进制对象分析)

连通性可以定义为以有非零权重的连贯像素为基础的分析。简单地说，连通性用于查找一组成为二进制对象的已连接像素。

Blob 对象在进行连通的地方自动进行连通，然后在计算已查找到的二进制对象的测量值。根据运行 Blob 对象前设置的 NumberToFind 属性，Blob 对象的连通性返回已查找到的 Blob 对象的数目。

Blob结果计算

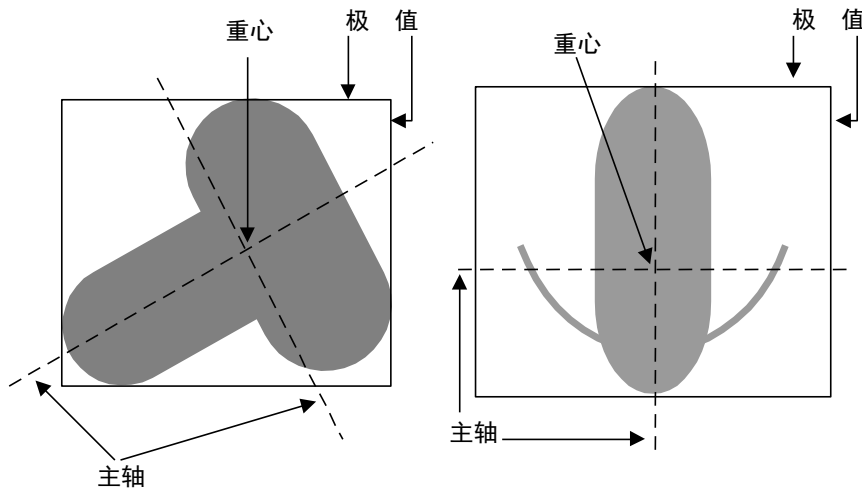
一旦完成二进制对象分析的所有其他步骤，就可以计算已查找到的二进制对象的结果。本节先前介绍的 *Blob 对象结果* 章节的内容为返回的 Blob 对象的所有结果列表。

*Vision Guide 7.0 Properties and Results Reference Manual* 中详细介绍了所有视觉对象的每个结果。

*Vision Guide 7.0 Properties and Results Reference Manual* 中介绍的一些结果应用于许多不同的视觉对象，如 Found、Time 或 PixelX 结果。Found 和 Time 结果为极通用的结果，一般情况下可应用于所有视觉对象，但是一些结果，如位置相关的结果，应用于 Blob 对象时有特别的意义。这些结果描述如下：

MinX、MinY、MaxX、MaxY 结果

把 MinX、MinY、MaxX 和 MaxY 的结果结合在一起，会形成一个二进制对象的边界矩形。边界矩形是指，能囊括二进制对象的最小的长方形。请参考下图帮助您理解边界矩形的概念。



主轴、重心和极值

### 在机器人坐标系、相机坐标系和像素坐标系中的位置数据

坐标位置的结果，返回的是中心的位置。当请注意，重心不一定是零件的中心位置。所以在抓取零件时，如果使用重心参数，可能会无法正常抓取。当从Blob属性获取RobotX、RobotY和RobotU坐标位置的结果，然后将其作为零件抓取的参数时，请确认能否通过重心位置正常抓取。

如果不想将重心作为抓取零件的参数，可以计算偏移量，或使用Correlation对象等其他视觉对象，来检测零件并更加准确的返回抓取位置。

### TotalArea 结果

TotalArea结果是所有已查找到的结果的面积总和。可以用于计算像素。将NumberToFind设置为0，Blob对象将查找到面积介于MinArea和MaxArea之间的所有二进制对象。然后TotalArea将显示所有结果的总面积。

### Blob 对象的 Angle 结果限制

请注意 Blob 对象的 Angle 结果范围有限。

Blob 对象的 Angle 结果返回的角度值在+90 至-90 范围内。Blob 对象不能返回整个360° 的范围的角度结果。



Blob对象返回的Angle结果并不始终都是可靠的，所以在使用机器人时要注意。因为Blob对象的Angle结果的范围有限，所以在某些情况下不可靠。

建议使用Polar对象来代替Blob对象来计算零件的角度方向。


Polar对象可使用从Blob对象中查找到的X、Y位置作为重心，然后根据Blob对象重心来计算角度。6.2.6 Polar对象章节详细介绍了这些内容。

### 使用Blob对象

综上，我们已经了解了二进制对象分析的工作原理，为理解如何使用Vision Guide 7.0 Blob对象打下了基础。本节将说明如何使用Blob对象。

使用Blob对象的所需步骤如下：


- 创建一个新的 Blob 对象
- 设置搜索窗口的位置和尺寸
- 设置 Blob 对象的相关属性
- 测试 Blob 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

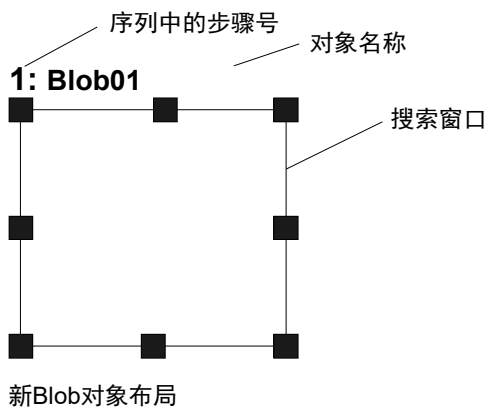
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. 视觉序列。

### 第1步：创建一个新的Blob对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>-  <New Blob>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 Blob 图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“Blob01”，因为它是在本序列中创建的第一个 Blob 对象。(稍后将说明如何修改名称。)

### 第2步：设置搜索窗口的位置和尺寸

屏幕上将显示一个Blob对象，如下所示。



- (1) 点击 Blob 对象的名称标签，然后保持按下鼠标的同时拖动 Blob 对象，以将搜索窗口的左上角与所需位置对齐。
- (2) 点击尺寸手柄并拖动鼠标，更改 Blob 对象搜索窗口的大小，并将其调整为适当尺寸。(搜索窗口就是进行 Blob 搜索的区域。)



注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。



### 第3步：设置Blob对象属性

现在设置Blob对象的属性值。

以下显示的是一些Blob对象特定的属性，也是更常用的属性。许多不同视觉对象所用其它属性的说明，如AbortSeqOnFail和Graphics等，可参见 *Vision Guide 7.0 Properties and Results Reference Manual* 或本章之前的 *BlobObject* 属性列表。



注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。  
当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。  
正确配置MaxArea, MinArea, RejectOnEdge等属性值，可降低错误检测的风险。

**Name 属性** 新创建的Blob对象中，默认的项目名称是“Blob\*\*”。在一个视觉序列中要使用多个Blob对象时，使用\*\*的编号对Blob进行区分。

默认情况下，会将在视觉序列序列中使用的第一个Blob对象命名为“Blob01”。

需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示Blob对象名称的地方都会变为新的名称。

**Polarity 属性** 有时需要从浅色背景上检测深色对象(DarkOnLight)，有时需要从深色背景上检测浅色对象(LightOnDark)。Polarity属性的基本目的就是做出这个选择。

默认设置为“DarkOnLight”(深色对象在浅色背景上)。如果要更改此默认设置，请点击Polarity属性的值字段，将出现一个下拉列表，包含“DarkOnLight”和“LightOnDark”两个选项。单击您要选择的项目。

**MinArea、MaxArea** 定义Blob对象被视为“Found”面积(Found结果值为“True”)。默认范围很大，为25~100,000(MinArea~MaxArea)。这意味着，在不调整MinArea和MaxArea属性的情况下，运行新建的Blob对象时，绝大部分Blob对象将被找到。通常，可以修改这些属性，使要查找到的二进制对象的显示范围合理。这样的话，在范围外检测到任何二进制对象，也可以确定它并不是目标对象的blob。

**RejectOnEdge 属性** 利用这条属性，可以将与搜索窗口边界有接触的那些零件排除在外。通常而言，这条属性宜设置为True。

现在您可以测试Blob对象，测试后，我们将再次设置其他必须的属性。

#### 第4步：测试Blob对象并检查结果

要测试Blob对象，点击执行面板中该对象的<Run>按钮。然后将显示Blob对象的结果。这时需检查的主要结果如下。当然，将来还会有其它有用的结果。

- Found 结果**            回复是否查找到了二进制对象。如果查找到的二进制对象不符合 MinArea 和 MaxArea 属性中确定的面积限制，Found 的结果将返回“False”。
- Area 结果**            已查找到的二进制对象的面积(以像素为单位)。
- Angle 结果**            Blob 的方向角度。  
该角度根据副轴计算得出，且介于+/- 90 度之间。
- Time 结果**            返回 Blob 对象执行所需时间。
- PixelX、 PixelY**        已查找到的二进制对象重心的 XY 坐标位置(以像素为单位)。
- MinX、 MinY、 MaxX、 MaxY**    这四个值定义了二进制对象的边界矩形(Extrema)。(即与二进制对象接触的外接点形成的四边形)



此时， RobotXYU、 RobotX、 RobotY、 RobotU 结果以及 CameraX、 CameraY、 CameraXYU 结果将返回 “no cal”。这就意味着无法进行校准，因此也无法计算机器人坐标系或相机坐标系相关的坐标结果。想了解更多信息，请参阅 7. 视觉校准。

#### 第5步：调整属性并再次测试

Blob 对象运行几次后，您可能会遇到一些二进制对象检测时的问题，或是需要对一些属性设定进行微调。以下是一些常见问题以及微调的方法。


- 问题**            : 如果 Blob 对象返回的 Found 结果为 False，请检查并调整以下事项。
  - Polarity 属性值可能与实际图像不符(该属性有两种设定，即浅色背景上检测深色对象(DarkOnLight)，和深色背景上检测浅色对象(LightOnDark))。确认 Polarity 属性值，务必确保与要检测的对象及其背景的深浅一致。此外，搜索窗口内显示的对象及其背景的深浅必须匹配。
  - 查看 Area 结果，并将该值与 MinArea 和 MaxArea 属性定义的值进行比较。如果 Area 结果不在 MinArea 和 MaxArea 属性定义的范围，请调整这些属性并再次运行 Blob 对象。
  - 采用直方图检查图像中灰度值的分布。有关直方图的详细资讯，请参阅 8. 直方图工具中。


- 微调**            : 某些应用中可能需要微调 Blob 对象。下面介绍 Blob 对象微调的主要属性。
  - MinArea 和 MaxArea。运行几次 Blob 对象后，可以得出 Area 结果中返回的近似值。更改 MinArea 和 MaxArea 属性的值时，请使用这些近似值。一般情况下，建议将 MinArea 和 MaxArea 属性设置成 Found 结果的限制值。这样，可以仅返回您要查找的二进制对象，且 Found 结果将为 “True”。(通过这种设置，可以排除面积与被检测对象不同的 Blob 对象。)

- ThresholdHigh 和 ThresholdLow。这些属性用于调整背景与二进制对象零件区分的灰度阈值设置参数。最好使用直方图工具设置这些属性。请参阅 *Vision Properties and Results Reference Manual* 中对 ThresholdHigh 和 ThresholdLow 属性描述。直方图的详细资讯，请参阅 8. 直方图工具。

完成属性调整测试 Blob 对象，并得到满意的结果后，创建视觉对象的步骤就完成了。您可以继续创建其他视觉对象，或设置并测试整个视觉序列。

### Blob对象的其他实用工具

Vision Guide 工具栏中有一个非常实用的 Histogram 功能，点击  <Histogram> 按钮，即可在搜索窗口中直观的显示灰度值的分布情况。使用 Vision Guide 直方图功能，可以设置 ThresholdLow 和 ThresholdHigh 属性的灰度值，这两个属性定义了二进制对象的组成部分和背景的组成部分。如果在查找二进制对象时遇到问题，Histogram 功能会非常有用。关于 Vision Guide 直方图用途的详细信息，请参阅 8.1 利用直方图。

当您统计确认 Blob 对象的结果时，可以使用  <Statistics> 选项的功能。有关 Vision Guide 统计功能的详细信息，请参阅 9. 使用 Vision Guide 统计。

### 将Blob对象用作像素计数器

Blob 对象可用作像素计数器。像素计算器对图像中二进制对象阈值范围内的，所有像素进行计数。

请按照以下步骤操作。

- (1) 创建 Blob 对象。
- (2) 设置极性。
- (3) 设置上限阈值和下限阈值。
- (4) 将 NumberToFind 设为“0”。这将使 Blob 对象查找到图像中的所有二进制对象。
- (5) 将 MinArea 设置为“1”，MaxArea 设置为“999999”。将计算大于等于 1 像素的二进制对象。
- (6) 运行序列。

使用 TotalArea 结果，读取 Blob 阈值内的像素总和。

### 6.2.5 Edge对象

#### Edge对象描述

Edge 对象用于定位图像的边缘。

图像中的对象边缘由灰度值“从深到浅”或“从浅到深”表示。可以从多个像素上观察到这种变化。

Edge 对象找到“Polarity”属性定义的明暗变化(“从浅到深”或“从深到浅”),然后将检测出的位置定义为单边缘的边缘位置。也可以更改 EdgeType 属性以搜索双边缘。设置为双边缘时,则会搜索两个相对的边,并返回中间点作为结果。Edge 对象支持多个结果,因此您可以指定希望找到的单边缘或双边缘的数量。

可以使用 SearchType 属性设置 Edge 对象,是沿直线或弧线进行搜索。

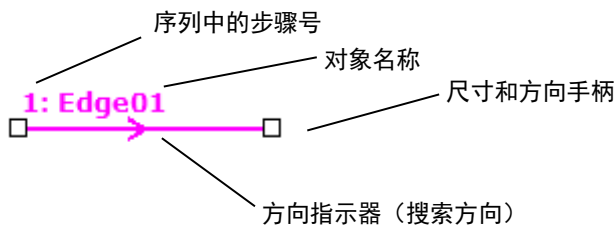
SearchType 为 Line 的 Edge 对象,具有搜索长度,在形状上与 Line 对象相似。搜索长度即为 Edge 对象的长度。SearchType 为 Line 的 Edge 对象的最强大的功能之一,是可以将它们放在任何角度。该功能可通过改变 Edge 对象的角度,使 Edge 对象矢量垂直于要检测边缘的区域。通常可以通过将 Edge 对象与 Frame(和被检测区域同步移动的)关联来实现。

#### Edge对象布局

Edge 对象具有两种布局。

##### SearchType 为 Line 时的布局

当 SearchType 为 Line 时, Edge 对象的搜索窗口即为 Edge 对象上的直线。Edge 对象沿着这条线,朝着方向指向器指示的方向,搜索明暗变化(由浅到深或者由深到浅)。

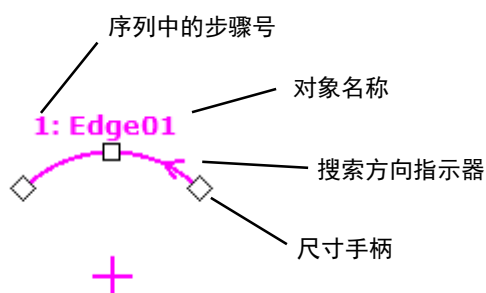


##### Edge对象直线布局

Edge对象可以放在任何方向(不仅仅是垂直或水平方向)进行搜索。您可以使用Edge对象的尺寸和方向手柄,在找到所需边缘的方向(以及用户指定的间隔)上,移动Edge对象的一端来实现。如需移动整个对象,请拖拽标签或直线。

### SearchType 为 Arc 时的布局

当 SearchType 为 Arc 时，Edge 对象的搜索窗口即为 Edge 对象搜索的弧线。Edge 对象沿着弧线，朝着方向指向器指示的方向，搜索明暗变化(由浅到深或者由深到浅)。



### Edge对象弧布局

如需调整圆弧的大小，请拖拽任一端的尺寸手柄。如需更改半径，请拖拽中间的尺寸手柄。如需移动整个对象，请拖拽标签或中间点。

### Edge对象属性

以下列表是对Edge对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

如果更改SearchType属性，设置的属性会根据指定的类型而变化。当SearchType为Line时，以下属性则不会出现在Vision Guide窗口的属性网格中。

AngleEnd  
 AngleStart  
 CenterPointObject  
 CenterPntObjResult  
 CenterPntOffsetX  
 CenterPntOffsetY  
 CenterPntRotOffset

当SearchType为Arc时，则不会出现以下属性：

X1  
 Y1  
 X2  
 Y2

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。 检测对象的得分需要大于或等于设定的形状得分。 默认: 100
AngleEnd	指定执行圆形/椭圆形搜索范围的终止角度。 默认: 135
AngleStart	指定执行圆形/椭圆形搜索范围的起始角度。 默认: 45
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为Edge对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则Edge对象将应用于指定视觉对象的所有结果(NumberFound) 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心 (CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 如果将 SearchWinType 设置为 RotatedRectangle，搜索窗口则以 Angle 结果为基础进行旋转。 默认: False
CheckClearanceFor	设定确认间隙的对象
ClearanceCondition	设定间隙的判断方法
ContrastTarget	设定边缘搜索所需的对比度。 默认: 0(最佳对比度)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认: 0

属性	说明
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认：空白
EdgeSort	设置检测到的边缘结果的排序方式。
EdgeThreshold	设置阈值，低于该值的边缘将被忽略。 默认：2
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认：1 - Single
Enabled	指定是否执行对象。 默认：True
EndPntObjResult	指定使用 EndPointObject 的哪个结果。
EndPointObject	指定使用哪个视觉对象来定义直线的终点。
EndPointType	指定用于定义直线终点的终点类型。
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
Frame	指定要使用哪个位置的定位框。 默认：none
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	选择对象标签的背景颜色。 默认：Transparent
Name	用于为Edge对象分配一个唯一的名称。 默认：Edge01
NumberToFind	规定待查找的边缘数目。 默认：1
PassColor	选择合格对象的颜色。 默认：Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound
Polarity	规定Edge对象是应该搜索浅色到深色的过度(LightToDark)还是深色到浅色的过度(DarkToLight)。 默认：1-LightToDark
Radius	定义从对象中心到对象最外层搜索环的距离。

---

属性	说明
SearchType	设置是使用直线还是圆弧搜索。 默认: Line
ScoreWeightContrast	根据对比度设置得分百分比。 默认: 50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认: 50
SearchWidth	定义边缘搜索的宽度。 范围: 3 - 99。 默认: 3
StartPntObjResult	指定使用StartPointObject的哪个结果。
StartPointObject	指定使用哪个视觉对象来定义直线的起点。
StartPointType	指定用于定义直线起点的起点类型。
StrengthTarget	设置搜索边缘的强度。 默认: 0
StrengthVariation	设置StrengthTarget的变化量总和。 默认: 0
X1	边缘起点位置的X坐标。
X2	边缘终点位置的X坐标。
Y1	边缘起点位置的Y坐标。
Y2	边缘终点位置的Y坐标。



**Edge对象结果**

以下列表是对Edge对象结果的总结，并附有简要说明。有关详情，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual*。

结果	说明
CameraX	返回相机坐标系中已找到的边缘位置的 X 坐标位置。
CameraY	返回相机坐标系中已找到的边缘位置的 Y 坐标位置。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到边缘位置的 CameraX、CameraY 和 CameraU 坐标。
ClearanceOK	返回间隙的判断结果
Contrast	返回查找到 Edge 的对比度。
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了 Accept 属性的当前设置值。
NumberFound	返回查找到的边缘的数目。(最少是 0 个，最多是 NumberToFind 属性中设定的检测数量)。
Passed	返回测试结果合格与否。
PixelX	返回已找到的 Edge 位置的 X 坐标，用像素表示。
PixelY	返回已找到的 Edge 位置的 Y 坐标，用像素表示。
PixelXYU	仅限运行时间。 返回已找到边缘位置的 PixelX、PixelY 和 PixelU 坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到的边缘位置的 X 坐标。
RobotY	返回机器人坐标系中检测到的边缘位置的 Y 坐标。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到 Edge 位置的 RobotX、RobotY 和 RobotU 坐标。
Score	返回一个 INTEGER 值，这个值代表被检测出的边缘的总体得分。
Strength	返回查找到边缘的强度。
Time	返回处理对象所需要的时间(单位：毫秒)。

### 使用Edge对象

下面将介绍如何创建和使用Edge对象。


- 如何创建新的 Edge 对象
- 确定搜索窗口的位置和尺寸
- 设置 Edge 对象属性
- 测试 Edge 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

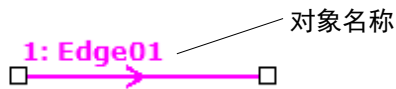
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. 视觉序列。

#### 第1步：创建新的Edge对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>-  <New Edge>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 Edge 对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“Edge01”，因为它是在本序列中创建的第一个 Edge 对象。(稍后将说明如何修改名称。)

#### 第2步：确定搜索窗口的位置和尺寸

屏幕上将显示以下所示Edge对象。



#### 新Edge对象

创建新的 Edge 对象之后，您可以通过 SearchType 属性，设置是沿直线或沿弧线搜索。当 SearchType 为 Line(默认)时，您可以点击任一尺寸柄，再将线端拖动至新位置，从而改变搜索长度和旋转角度。当 SearchType 为 Arc 时，您可以拖拽弧各端的任一柄，从而更改弧。如需更改半径，请拖拽中间的手柄。

您也可以点击 Edge 对象的名称选项卡或边缘线沿线附近的任意位置，按住鼠标将整个 Edge 对象拖动至屏幕上新位置。确定位置后松开鼠标。Edge 对象将停留在屏幕上的新位置。

### 第3步：设置Edge对象属性

现在，我们可以设置 Edge 对象的属性值了。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项即可。

以下显示的是一些 Edge 对象更常用的属性。可参见 *Vision Guide 7.0 Properties and Results Reference Manual*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等。

<b>EdgeType(单边缘)</b>	选择搜索边缘的类型。  如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。
<b>Name 属性 (“Edgexx”)</b>	新创建的Edge对象中，默认的项目名称是“Edge**”。在一个视觉序列中要使用多个Edge对象时，使用**的编号对Edge进行区分。 默认情况下，会将视觉序列序列中使用的第一个Edge对象命名为“Edge01”。  需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示Edge对象名称的地方都会变为新的名称。
<b>NumberToFind (1)</b>	沿着边缘搜索线搜索1条或以上的边缘。
<b>Polarity (LightToDark)</b>	如果您要寻找DarkToLight(深色到浅色)的边缘，请更改Polarity设置。

### 第4步：测试Edge对象并检验结果

运行 Edge 对象仅需做到以下几点：

点击执行面板上对象的<Run>按钮。然后将显示Edge对象的结果。这时需检查的主要结果如下。

<b>PixelX结果</b>	沿着边缘搜索线检测到的边缘的XY坐标位置(以像素为单位)。
<b>PixelY结果</b>	
<b>CameraX结果</b>	在相机坐标系中定义Edge对象的XY坐标值。 如果相机已经被校准，则CameraX和CameraY的结果将仅返回一个值。如果相机尚未完成校准，则将返回“no cal”。
<b>CameraY结果</b>	
<b>RobotX结果</b>	定义了机器人坐标中Edge对象的XY坐标。 可用于命令机器人移动至这个XY坐标位置。(不需要进行其它转换或步骤。)如果相机已经被校准，则RobotX和RobotY的结果将仅返回一个值。如果未进行校准，则将显示“no cal”。
<b>RobotY结果</b>	

### 6.2.6 Polar对象

#### Polar对象描述

Polar对象可将有笛卡尔坐标系的图像，快速的转换为相应的极坐标系图像。Polar对象在判断检测对象是否发生旋转时，是非常实用的工具。Polar对象运行稳定而高速，我们推荐您在计算对象的旋转角度时使用。

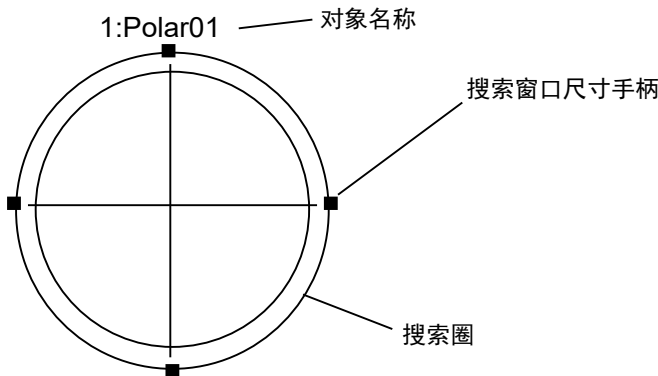
Polar对象检测是否发生旋转的工具，所以绕对象重心的旋转不会影响计算精度，但是对象在 x-y 平面上的平行移动很容易影响计算精度。所以 Polar对象很适合用于计算检测对象的旋转，但前提是该对象的中心位置不会移动。

因此，Vision Guide 7.0 的 Polar对象可使用 CenterPoint 属性。CenterPoint 属性使您可以将 Polar对象的中心点锁定在其它对象(Correlation对象和 Blob对象等)的位置结果上。

换句话说，当首先通过 Correlation对象和 Blob对象找到 XY位置时，Polar对象可将此 XY位置作为中心点，计算检测对象的旋转。

#### Polar对象的布局

Polar对象的布局与目前为止我们看到的其他对象的布局非常不同。但用途非常相似。Polar对象呈圆形布局，且显示中心点和半径。如需修改Polar对象的位置，只需点击对象名称(或点击外围圆上的任意位置)，然后按住鼠标将其推拽到新位置即可。



#### Polar对象的布局



Polar对象中心位置(由 CenterPoint 属性规定)也可以以其他对象的位置为基准进行设置。也就是说即使您可以改变 Polar对象的位置，但只要运行了对象或序列，Polar对象的中心位置也可能改变。详情见本章后面部分 Polar对象的相关内容。

Polar对象的搜索窗口是圆形的。如上图所示，其外边界由外环定义。而内环是一个小于外环的圆形，且位于外圈内的 n 像素的位置。像素的值由 Thickness 属性定义。更改 Thickness 属性时，您会注意到内外环之间的厚度或距离发生了变化。因此这使您搜索的区域是可见的。

如果需要调整 Polar对象搜索窗口外边界的尺寸，应点击 4 个搜索窗口尺寸手柄的其中一个，并根据需要圈向内或向外拖动。

## Polar对象属性

以下列表是对Polar对象属性的总结。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。如果设定值偏低，可能会导致错误检测。 默认: 700
AngleOffset	Polar对象注册完成后，调整被搜索对象得角度指示器图形，并定义适用于Polar对象的偏移角度。 默认: 0
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为Polar对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则Polar对象将应用于指定视觉对象的所有结果(NumberFound)。 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 默认: False
CenterX	指定用作Polar Search Tool中心点位置的X坐标。 当CenterPointObject属性已被其他视觉对象设定时，将自动设置本属性。
CenterY	指定用作Polar Search Tool中心点位置的Y坐标。 当CenterPointObject属性已被其他视觉对象设定时，将自动设置本属性。
CheckClearanceFor	设定确认间隙的对象

属性	说明
ClearanceCondition	设定间隙的判断方法
Confusion	指出被检测图像的预期混淆量。 此处显示的值是对于被检测对象可获得的最大形状的分，而不是您正在检测的对象。 默认：800
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
Frame	指定要使用哪个位置的定位框。 默认：none
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	设置对象标签的背景颜色。 默认：Transparent(透明)
ModelObject	规定使用哪种模型进行搜索。 默认：Self
Name	用于为Polar对象分配一个唯一的名称。 默认：Polar01
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound
Radius	定义从对象中心到对象最外层搜索环的距离。 默认：50
SaveTeachImage	设置示教模型时，是否保存图像文件。
ScoreMode	设置或返回在Fail时显示结果的阈值。
ShowModel	查看示教模型内部的灰度显示。 设置不要紧的像素。

属性	说明
Thickness	规定搜索环的厚度，这是Polar对象的被搜索区域。 厚度以像素为单位，在Radius属性定义的搜索环上，从外环朝内环进行测量。 默认：5

### Polar对象结果

以下列表是对Polar对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	返回被检测出零件的旋转量，用度数表示。
CameraX	返回相机坐标系的 Polar 对象 CenterPoint 位置的 X 坐标位置。
CameraY	返回相机坐标系的 Polar 对象 CenterPoint 位置的 Y 坐标位置。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到零件位置的 CameraX、CameraY 和 CameraU 坐标。
ClearanceOK	返回间隙的判断结果。
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了 Accept 属性的当前设置值。
Passed	返回测试结果合格与否。
PixelX	返回已找到的零件位置(参见模型原点)的X坐标位置(以像素为单位)。
PixelY	返回已找到的零件位置(参见模型原点)的 Y 坐标位置(以像素为单位)。
PixelXYU	仅限运行时间。 返回已找到的零件位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
RobotX	返回机器人坐标系中 Polar 对象的 CenterPoint 位置的 X 坐标。
RobotY	返回机器人坐标系中 Polar 对象的 CenterPoint 位置的 Y 坐标。
RobotU	返回机器人坐标系中 Polar 对象的 Angle 结果的 U 坐标。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的零件位置的 RobotX、RobotY 和 RobotU 坐标。
Score	返回一个 INTEGER 值，这个值代表运行时间内找到的对象与 Polar 对象正在搜索的模型的匹配程度。
Time	返回处理对象所需要的时间(单位：毫秒)。

#### NOTE



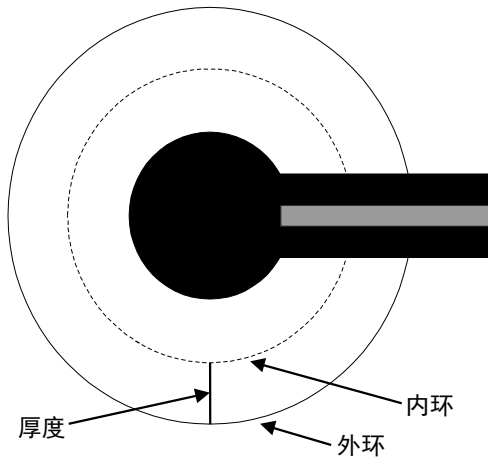
返回X坐标和Y坐标(CameraX、CameraY、PixelX、PixelY、RobotX、RobotY)的Polar对象所有结果，都是从CenterPoint属性读取相关坐标值。如果在运行Polar对象之前设置了CenterPoint属性的值，则这些值也会被传递到Polar对象的结果中。Polar对象只计算旋转角度，而不会计算中心位置的X,Y坐标，但是CenterPoint属性会提供X,Y的坐标位置，所以无需从其他视觉对象中获取X,Y坐标的结果。

### 了解Polar对象

Polar 对象的目的是检测指定对象或图案的旋转量。为此，首先需要创建一个极坐标模型，该模型一般是一个有一定厚度的圆环。创建完成后执行 Polar 对象，通过将圆环与被检测工件当前的旋转进行比对，来计算角度偏差，并将其作为 Polar 对象结果之一返回。

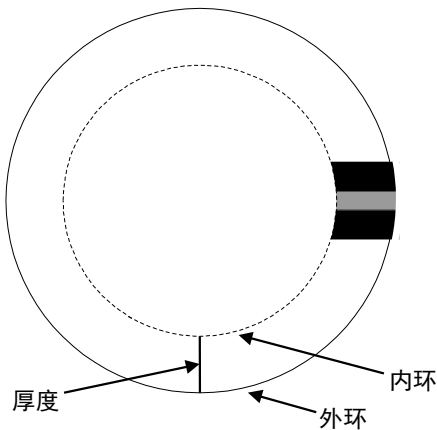
例如，假设有一个围绕中心点旋转的工件，机器人需要拿起工件并将其放置到托盘上。但因为工件总是朝着不同的方向，所以如果不确定对象的旋转量，则会很难将工件拾起。

为了方便理解，以下举例进行说明。在下图A中，被检测对象是图中黑色和灰色部分显示的工件，我们则可以定义一个与对象外侧部分相交的极坐标模型。请注意内环与外环之间的区域。该区域定义了注册了此工件的Polar对象的模型。您可以看到内环中的一些区域可辨性非常高，因为它们与其他区域的差别非常明显。这些区域，就是可以检测出工件旋转量的关键。



图A: Polar对象的工件示例

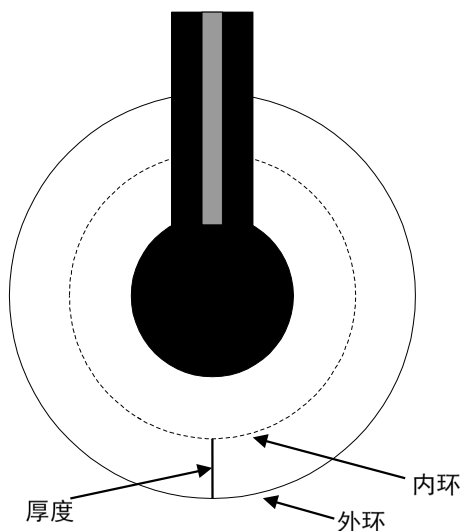
图B所示为一个已示教的模型外观。模型大部分是白色，有一段是黑色和灰色的圈。如果运行Polar对象，将搜索该类模式的图案。即搜索有一小部分是黑色和灰色，其余部分是白色的图案。



图B: 图A零件的极坐标模型

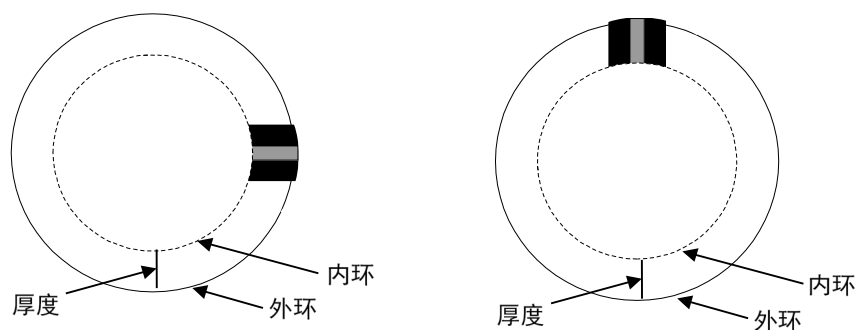


图C所示为相对于模型旋转了约90°的工件。Polar对象将计算模型和对象新位置之间的角度偏差。



图C：工件与原始示教模型旋转90°

图D中左边是0°原始示教模型，右边是旋转90°的工件。



图D：极坐标模型和旋转后的工件的极坐标

#### Polar 对象相关的主要参数

从上一节的示例中可以看出，Polar 对象有 3 个主要参数，每个参数对于能否获得最佳结果都很重要。这些参数由下列属性定义：

- CenterPoint 属性
- Radius 属性
- Thickness 属性
- AngleOffset 属性

**CenterPoint** 属性定义 **Polar** 对象的中心点位置。

如前文示例中所述，**Polar** 对象的中心点必须处于一个适当的位置，才能使示教的模型中心点与要测量旋转角度的被检测对象的中心位置完全一致。否则，则会在 **XY** 平面中平移并导致角度结果不准确。

**Radius** 属性定义了 **Polar** 对象中心点到 **Polar** 对象的搜索圈外环之间的距离。该属性定义了 **Search Area** 的外边界。

**Thickness** 属性定义了从外环到虚拟内环之间的距离(以像素为单位)。该距离实际上是 **Search Area** 的厚度。

**AngleOffset** 属性提供一种机制来设置指示器图形线的角度位置(显示 **Polar** 对象的旋转位置时使用)。

运行 **Polar** 对象后，图形线会显示在 3 点钟位置( $0^\circ$ )。但如果要指示器图形与被检测对象最需关注的旋转位置吻合，可以根据需要修改指示器图形的位置。(记住该 **AngleOffset** 属性通常应在示教模型并运行 **Polar** 对象后进行设置。)

### 如何确定对象的旋转

举一个需要确定对象旋转的经典应用案例。用机器人拾取集成电路模具，此时则必须知道模具的 **XY** 位置和旋转方向。模具与背景相比，具有不同的灰度。模具表面包含显示其从  $0$  到  $360^\circ$  度旋转的灰度信息。

该应用中，可以使用 **Blob** 对象(对象名称为“**Blob01**”)查找模具的中心。也可以使用 **Correlation** 对象代替 **Blob** 对象。

创建 **Polar** 对象，使用 **Blob** 对象的 **X,Y** 位置结果作为 **Polar** 对象的中心位置。也就是将 **Polar** 对象的 **CenterPoint** 属性设置为“**Blob01**”。

然后为 **Polar** 对象设置正确的 **Radius** 和 **Thickness** 属性值。示教灰度的极坐标模型，作为  $0^\circ$  的模具模型。

当搜索方向各异的新模具时，将从 **Blob** 对象“**Blob01**”的 **XY** 位置和 **Radius** 属性以及 **Thickness** 属性构造一个 **Polar** 搜索窗口。

然后在搜索窗口中搜索于示教模型的旋转角度相同的角度。并将模型找到的模型角度，作为 **Angle** 结果(以像素为单位)和 **Robot U** 结果(机器人坐标系)返回。

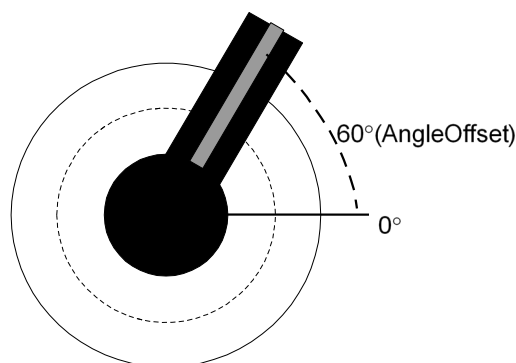


重要提示: 返回的 **RobotU** 结果就是机器人坐标系中工件的实际旋转情况，但是夹具可能没有安装在精确的旋转角度  $0^\circ$  位置。夹具可能存在  $2^\circ \sim 3^\circ$  的误差。因此在移动至零件时，在程序中进行调整。

### 调节已示教模型的角度偏移

在示教 Polar 模型时，默认原始的圆形模型的旋转角度为  $0^\circ$ ，此时旋转角度  $0^\circ$  位置处于时钟的 3 点钟方向。在下图显示的模型中，被搜索区域大致位于 1 点钟方向上。示教此模型后，与所有 Polar 模型一样，该模型的  $0^\circ$  位置将在 3 点钟方向。但是，由于我们想看到零件实际旋转的视觉指示器，所以需要用 Angleoffset 属性对极坐标角指示器图形的位置进行适当的调整。

例如，我们示教大约在 1 点钟方向的零件，则需设置  $60^\circ$  左右的偏移。(即 Polar 对象的 AngleOffset 属性应设置为  $60^\circ$ )



AngleOffset属性设置为 $60^\circ$ 的Polar模型

### 执行Polar对象的难点

示教 Polar 对象时，需要重点考虑的因素是，Polar 模型包含了充足的信息来确定被检测对象的方向。

因此，Polar 对象的预期精度和执行速度，决定了进行 Polar 对象检索时，所用 Polar 模型的大小。

如果 Polar 对象的旋转结果需要  $1/2^\circ$  的精度，则 Polar 对象只需要 180 像素宽即可。(假设检索精度为  $1/4$  像素且每像素  $2^\circ$ ，或每个 Polar 对象分辨率单位为  $0.5^\circ$ )

Polar 对象的精确度也取决于模型的灰度信息。

选择 Polar 模型的厚度(以像素为单位)时，也必须包含足够充分的信息，以实现即使被检索图像的中心位置在识别上稍有偏移，角度检测也能得出正确的结果。

将 Thickness 属性设置为 1 像素时，如果图像存在 1 像素错位，则极坐标转换会向极坐标图像发送完全不同的像素。

因此，为了给源图像位置提供一定的容差量，可以将 Thickness 属性设置为 5 像素，这样如果源图像位置偏移 1 个像素，相应的极坐标图像像素仅偏离  $1/5$ 。(这就是将 Thickness 属性的最小值设置为 5 的原因。)

根据模型中的灰度信息量和所需的搜索速度，选择 Polar 对象的 Radius 和 Thickness 属性。

搜索速度与 Radius 和 Thickness 属性的值成反比。将 Radius 属性设置为一个较低的值，并且把 Thickness 属性设置为 5，则可以得到最快的搜索速度。但在多数情况下，将 Thickness 属性设置为 5，可能不足以精确的查找到 Polar 模型。

### 使用Polar对象

接下来将介绍如何创建并使用 Polar 对象。


- 创建一个新的 Polar 对象
- 确定搜索窗口的位置和尺寸
- 配置与 Polar 对象相关的属性
- 测试 Polar 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

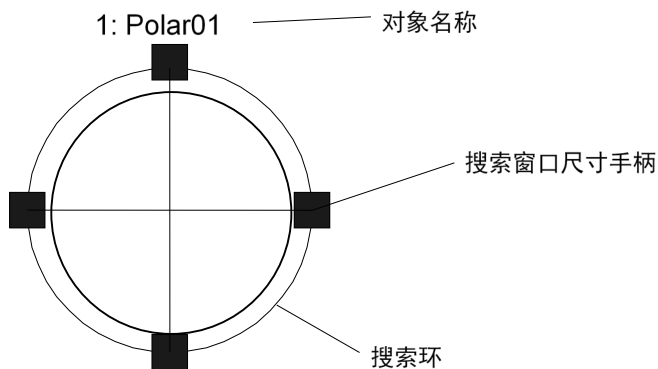
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：创建一个新的Polar对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>-  <New Polar>按钮。
- (2) 将鼠标移动到图像显示区域。鼠标光标将变为 Polar 对象图标。
- (3) 继续移动光标，直至图标位于图像显示区域中的所需位置，然后单击鼠标左键以创建对象。
- (4) 请注意，对象名称将自动创建。本示例中，由于是在本序列中创建的第一个 Polar 对象，所以对象名称为“Polar01” (稍后将说明如何修改对象名称。)

### 第2步：定位Polar对象

屏幕上将显示一个与下图相似的Polar对象：



### 新Polar对象

Polar 对象是圆形搜索窗口。您可以改变整个对象的位置或半径。

如需移动整个对象，请点击对象名称或外环上的任何位置，并按住鼠标左键，即可将整个对象拖动至画面上的新位置。确定位置后松开鼠标，Polar 对象将停留在画面中的新位置上。

如果需要改变半径，应将鼠标指针移动到任一尺寸手柄上，按下鼠标左键，然后移动鼠标即可。

### 第3步：配置与Polar对象相关的属性

现在，我们可以设置 Polar 对象的属性值了。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项即可。

以下显示的是一些 Polar 对象更常用的属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual* 或本章前面所描述的 *Polar 对象属性* 列表，以了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和“图形”等属性。

**Name 属性** 新创建的 Polar 对象中，默认的项目名称是“Polar\*\*”。在一个视觉序列中要使用多个 Polar 对象时，使用\*\*的编号对 Polar 进行区分。

默认情况下，会将在视觉序列序列中使用的第一个 Polar 对象命名为“Polar01”。

需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Polar 对象名称的地方都会变为新的名称。

**CenterPointObject 属性** 通常会在同一个序列中，先前执行的对象中设置此属性。该属性将决定执行 Polar 对象时的中心点位置。

**Thickness 属性** 通常为了获得模型信息，确定工件的角度，会给该属性设置一个足够大的值。

**AngleOffset 属性** 通常设置该属性时，需要确保能将最终角度的结果定位在目标位置上。例如，如果要搜索始终的分针，则需调整 AngleOffset 属性，使显示角度的与分针完全吻合。

### 第4步：测试Polar对象并检查结果

执行以下操作，以运行 Polar 对象。

点击执行面板上对象的<Run>按钮。将首先运行 CenterPointObject。

然后将显示 Polar 对象的结果。这时需检查的主要结果有：

**Angle 结果** 被检测出模型的角度(单位：度)。

### 6.2.7 OCR对象

#### OCR对象描述

OCR(Optical Character Recognition/光学字符识别)对象可用于识别特定字体和字符大小图像中的单行字符串。OCR 对象的 GUI 有一个字体向导，该向导可根据 SEMI 标准创建字体，也可以根据图像和 ASCII 定义文件中的字符，创建用户定义的字体。您可以将创建的字体导出至磁盘，并将数据导入至同一项目，或其它项目的 OCR 对象。

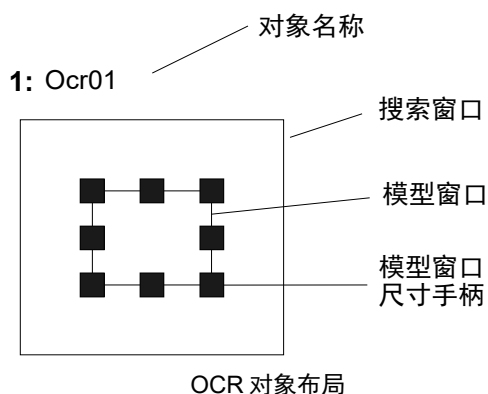
NOTE



OCR对象仅在安装并启用OCR选项时才可使用。

#### OCR对象布局

OCR对象有搜索窗口和模型窗口，如下所示。



NOTE



如果使用“Arc”类型的搜索窗口，可以将沿弧线排列的字符串识别为特定的字体或字符大小的图像。

## OCR对象属性

以下列表是对OCR对象属性的总结。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Caption	为OCR对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则OCR对象将应用于指定视觉对象的所有结果(NumberFound)。 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 默认: False
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。 FindChar=False 时呈灰色。 默认: 1
Description	设定用户指定的描述 默认: 空白
Dictionarymode	指定字典模式。 默认: All
Direction	设置沿弧线排列的字符的方向。 默认: InsideOut
Enabled	指定是否执行对象。 默认: True

属性	说明
ExportFont	将当前字体导出至磁盘。
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
FindChar	指定是否将每个字符视为一个单独的对象。 默认：False
Frame	指定要使用哪个位置的定位框。 默认：None
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
ImportFont	从Vision Guide的GUI中，启动文件对话框并导入字体文件。
InvalidChar	设置或返回用于表示Text结果中无效字符的字符。 默认：“?”
LabelBackColor	选择对象标签的背景颜色。 默认：Transparent
ModelWin	仅限运行时间。 在一次调用中设置或返回模型窗口的左侧参数、顶部参数、高度参数和宽度参数。
ModelWinAngle	指定模型窗口的角度。
ModelWinCenterX	指定模型窗口中心的X坐标值。
ModelWinCenterY	指定模型窗口中心的Y坐标值。
ModelWinLeft	指定模型窗口左侧位置。
ModelWinHeight	指定模型窗口的高度。 默认：50
ModelWinTop	指定模型窗口顶部位置。
ModelWinType	定义模型窗口的类型。
ModelWinWidth	指定模型窗口的宽度。 默认：50
Name	用于为OCR对象分配一个唯一的名称。 默认：Ocr01
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound



属性	说明
Polarity	可定义被检测对象和背景色的差异(在浅色背景下检测深色对象(DarkOnLight), 或在深色背景下检测浅色对象(LightOnDark))。 默认: 1 - DarkOnLight
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinAngleEnd	定义待搜索区域的结束角度。
SearchWinAngleStart	定义待搜索区域的开始角度。
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	定义待搜索区域的高度(单位: pixel)。 默认: 100
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。
SearchWinRadiusInner	定义待搜索区域的圆弧内径。
SearchWinRadiusOuter	定义待搜索区域的圆弧外径。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinType	定义待搜索区域的类型(Rectangle, Arc)
SearchWinWidth	定义待搜索区域的宽度(单位: pixel)。 默认: 100

### OCR对象结果


以下列表是对 OCR 对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	以角度为单位返回检测到字符的旋转量。
Found	返回是否检测到了对象。
Passed	返回视觉对象的测试结果合格与否。
CameraX	返回相机坐标系中已查找到的字符的 X 坐标位置。
CameraY	返回相机坐标系中已查找到的字符的 Y 坐标位置。
NumberFound	返回已查找到的字符数目。
MaxX	返回字符极值的最大 X 像素坐标。(以像素为单位)
MaxY	返回字符极值的最大 Y 像素坐标。(以像素为单位)
MinX	返回字符极值的最小 X 像素坐标。(以像素为单位)
MinY	返回字符极值的最小 Y 像素坐标。(以像素为单位)
PixelX	返回已找查到的字符位置的 X 坐标位置。(以像素为单位)
PixelY	返回已找查到的字符位置的 Y 坐标位置。(以像素为单位)
RobotX	返回机器人坐标系中已查找到的字符的 X 坐标位置。
RobotY	返回机器人坐标系中已查找到的字符的 Y 坐标位置。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。 这样便于对比结果。
Text	返回在搜索过程中查找到到的字符。
Time	返回处理对象所需要的时间(单位：毫秒)。

## 使用OCR对象

本节将介绍使用 OCR 对象所需的步骤，如下所示：

- 创建一个 OCR 对象
- 为新对象创建或导入字体
- 校准字体(根据需要)
- 配置与 OCR 对象相关的属性
- 测试 OCR 对象并检查结果
- 调整属性并再次测试

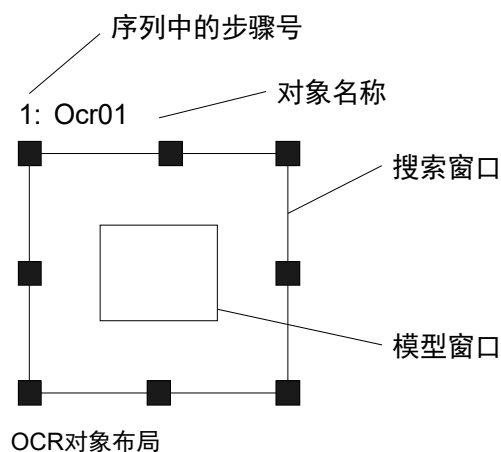
开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. 视觉序列。

## 新建一个OCR对象

1. 点击 Vision Guide 工具栏中的<All Tools>- <OCR>按钮。
2. OCR 图标将出现在 OCR 对象按钮的上方。
3. 点击该 OCR 图标，并将其拖动至 Vision Guide 窗口的图像显示区域中。
4. 将自动创建对象名称。本示例中是在本序列中创建的第一个 OCR 对象，座椅对象被命名为“Ocr01”。(稍后将说明如何修改名称。)



### 创建字体

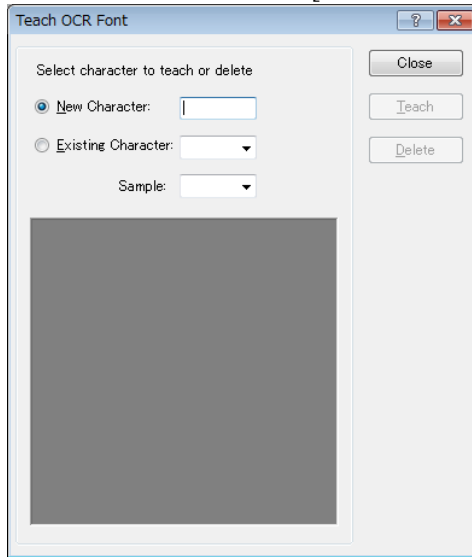
请根据要读取的字符图像创建字体。您还可以使用 **ImportFont** 属性，从其它项目导入字体。在导入字体时，您不需要创建字体。

系统中已自带了常规的西文和日语字体。但根据输入图像的大小，可能还需创建字体。

可使用图像文件创建用户定义的字体。

创建字体步骤如下：

- (1) 点击流程图中的 OCR 对象。将模型窗口与字符对齐。
- (2) 点击<Teach>按钮，显示 [Teach OCR Font] 对话框。



- (3) 如需注册新字符，请选择<New Character>按钮。然后点击<Teach>按钮。您可以为一个字符注册多个图像。  
可以注册的字符包括日语(全角)、半角字母数字字符和半角符号。  
如需从字体中删除注册字体，请选择<Existing Character>按钮，从下拉表中选择要删除的字体，然后点击<Delete>按钮。

### 6.2.8 CodeReader对象

#### CodeReader对象描述

CodeReader 对象用于读取条形码或二维码。

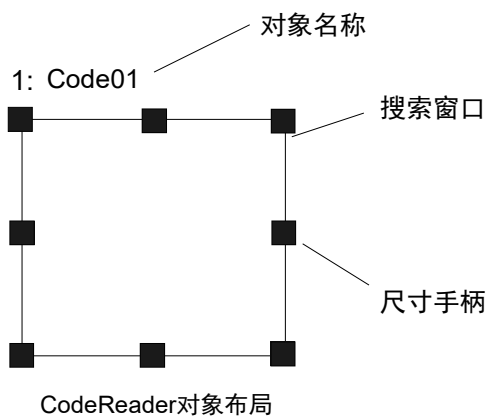
支持的代码为：

EAN13	数字 0 ~ 9, 固定长度
Code39	0 ~ 9, A ~ Z, ./+-%\$SpC, 可变长度
DataMatrix	二维码 *1
Interleaved 2 of 5	数字 0 ~9, 固定长度
Code128	Full ASCII, 可变长度
Codabar	数字 0 ~ 9, \$ - : / . +
PDF417	二维码
QR	二维码
EAN	8 位数字 0 ~ 9, 固定长度
UPC A	数字 0 ~ 9, 固定长度
UPC E	数字 0 ~ 9, 固定长度

\*1: 仅支持 ECC200 标准。

#### CodeReader对象布局

CodeReader 对象仅有一个搜索窗口，如下图所示。



## CodeReader对象属性

以下列表是对 CodeReader 对象属性的总结。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Caption	为 CodeReader 对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在 PixelX, PixelY 上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则CodeReader对象将应用于指定视觉对象的所有结果(NumberFound) 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用 CenterPointObject 属性定位了搜索窗口的中心点后，设置或返回 Y 偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 默认: False
CodeType	设置或返回使用 CodeReader 对象搜索条形码或二位代码的类型。 默认: 0 - Auto
CoordObject	指定复制结果的 Coordinates 对象 执行对象时进行复制处理，若没有通过 Decision 等分支功能执行，则不进行复制处理。 默认: None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认: 空白

属性	说明
Enabled	指定是否执行对象。 默认: True
FailColor	选择当对象被检测为不良品时, 搜索窗口显示的颜色。 默认: Red
Frame	根据指定框架设置当前对象的搜索位置。 默认: none
FrameResult	指定需使用Frame的第几个结果。 默认: 1
Graphics	设置显示的图表模式。 默认: 1 - All
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
Name	用于为 CodeReader 对象分配一个唯一的名称。 默认: Code01
NumberToFind	规定待查找的代码数目。 最大值: 8 默认: 1
Orientation	设置条形码的方向。
PassColor	选择当对象被检测为良品时, 搜索窗口显示的颜色。 默认: LightGreen
PassType	可定义检测结果良品/不良品的判断标准。 默认: SomeFound
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinHeight	定义待搜索区域的高度(单位: pixel)。 默认: 100
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinWidth	定义待搜索区域的宽度(单位: pixel)。 默认: 100

## CodeReader对象结果

以下列表是对 CodeReader 对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	返回检测到的条码倾斜角度。
CameraX	返回相机坐标系中检测到的对象的 X 坐标。(单位: mm)
CameraY	返回相机坐标系中检测到的对象的 Y 坐标。(单位: mm)
Found	返回是否检测到了对象。
FoundCodeType	返回检测到的条码类型。
NumberFound	返回检测到的代码数目。 (最少是 0 个, 最多是 NumberToFind 属性中设定的检测数量)。
Passed	返回视觉对象的测试结果合格与否。
PixelX	返回检测到的对象的 X 坐标(以像素为单位)。
PixelY	返回检测到的对象的 Y 坐标(以像素为单位)。
PixelXYU	仅限运行时间。 返回已找到的零件位置的 PixelX、PixelY 和 PixelU 坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到的对象的 X 坐标位置。
RobotY	返回机器人坐标系中检测到的对象的 Y 坐标位置。
RobotU	返回机器人坐标系中检测到的对象的 U 坐标位置。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的零件位置的 RobotX、RobotY 和 RobotU 坐标。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。
Text	返回搜索操作找到的文本。
Time	返回处理对象所需要的时间(单位: 毫秒)。




## 使用CodeReader对象

本节将说明使用 CodeReader 对象所需的步骤，如下：

- 创建一个 CodeReader 对象
- 确定搜索窗口的位置和尺寸
- 设置 CodeReader 对象的相关属性
- 测试 CodeReader 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。如果已经创建了视觉序列，请在 Vision Guide 窗口中单击序列目录树，选择已创建的视觉序列。有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. 视觉序列。

### 第1步：创建一个CodeReader对象

1. 点击 Vision Guide 工具栏中的<All Tools>-  <CodeReader>按钮。
2. CodeReader 图标会出现在 CodeReader 对象按钮的上方。
3. 点击这个 CodeReader 图标，并将其拖至 Vision Guide 窗口的图像区域中。
4. 将自动创建对象名称。本示例中是在本序列中创建的第一个 CodeReader 对象，座椅对象被命名“Code01”。(稍后将说明如何修改名称。)

### 第2步：确定搜索窗口的位置和尺寸

画面上会显示一个如下所示的 CodeReader 对象。

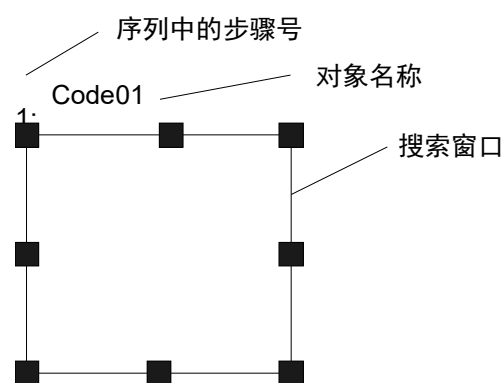


图 1：新CodeReader对象

1. 点击 CodeReader 对象的名称标签，按住鼠标左键，将 CodeReader 对象拖至您想让搜索窗口的左上角所在的位置。
2. 点击搜索窗口的尺寸手柄，拖动鼠标，将 CodeReader 对象搜索窗口调整成合适的大小。(搜索窗口就是要搜索的区域。)



NOTE 请在条形码的两侧留出空白区域。否则会无法完成搜索。如果是二维码，则需在其周围留出一个单元格或更多的空白区域。

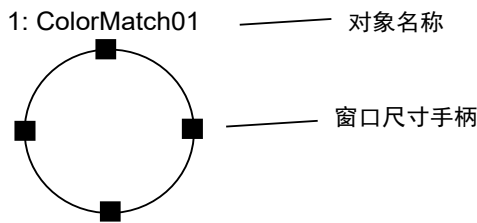
### 6.2.9 ColorMatch对象

#### ColorMatch对象描述

ColorMatch 对象用于查找与颜色模型匹配的颜色。

#### ColorMatch对象布局

ColorMatch 对象的布局有矩形、旋转矩形或圆形，因此具有中心点和半径。点击对象的名称或者点击搜索窗口外周长的任意位置，并将该对象拖动至新位置，即可修改 ColorMatch 对象的位置。



新ColorMatch对象布局



CenterPoint 属性定义 ColorMatch 对象中心位置，中心位置也可根据其它对象的位置进行设置。也就是说即使您调整了 ColorMatch 对象的位置，但如果运行了对象或序列时，对象的中心位置也可能会改变。

ColorMatch 对象的搜索区域可以是矩形、旋转矩形或圆形。

当 SearchWinType 为 Circle，要调节 ColorMatch 对象搜索窗口的外边界的大小，则需点击搜索窗口的任一尺寸手柄，然后根据需要向内或向外拖动圆圈，以更改半径。

#### ColorMatch属性

以下列表是对 ColorMatch 对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。如果设定值偏低，可能会导致错误检测。 默认: 700
Caption	为 ColorMatch 对象指定标题。 默认: Empty String

属性	说明
CenterPointObject	<p>指定对象中心点位置。</p> <p>若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。</p> <p>默认：Screen</p>
CenterPntObjResult	<p>指定要使用的CenterPointObject属性结果。</p> <p>如果指定了All，则ColorMatch对象将应用于指定视觉对象的所有结果(NumberFound)</p> <p>默认：1</p>
CenterPntOffsetX	<p>使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。</p> <p>默认：0</p>
CenterPntOffsetY	<p>使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移</p> <p>默认：0</p>
CenterPntRotOffset	<p>指定是否旋转以CenterPointObject的Angle结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的XY偏移。</p> <p>如果将SearchWinType设置为RotatedRectangle，搜索窗口则以Angle结果为基础进行旋转。</p> <p>默认：False</p>
CenterX	<p>指定用作对象中心点位置的X坐标。</p> <p>当CenterPoint属性已被其他视觉对象设定时，将自动设置本属性。</p>
CenterY	<p>指定用作对象中心点位置的Y坐标。</p> <p>当CenterPoint属性已被其他视觉对象设定时，将自动设置本属性。</p>
ColorMode	<p>设置将使用的颜色空间(RGB/HSV)。</p> <p>默认：RGB</p>
CoordObject	<p>指定复制结果的Coordinates对象</p> <p>执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。</p> <p>默认：None</p>
CurrentModel	<p>仅限运行时间。</p> <p>指定在ModelColor属性和VTeach中使用哪种模型。</p> <p>NumberOfModels的值为1到CurrentModel的值。</p> <p>默认：1</p>
CurrentResult	<p>可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。</p> <p>默认：1</p>

属性	说明
Description	设定用户指定的描述 默认：空白
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
Frame	根据指定框架设置当前对象的搜索位置。 (允许对象相对于框架进行定位。) 默认：None
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	设置对象标签的背景颜色。 默认：Transparent
ModelColor	仅限运行时间。 在运行过程中，直接输入 RGB 颜色值，示教模型。 默认：RGB(0, 0, 0)
ModelColorTol	仅限运行时间。 设置模型颜色的容差量。 如果像素颜色在模型颜色的容差范围以内，那么像素会保持不变。 默认：0 (ColorMode = RGB); 0,0,50 (ColorMode = HSV)
ModelName	仅限运行时间。 设置当前正在使用的模型的名称。
ModelObject	规定使用哪种模型进行搜索。 默认：Self
Name	用于为ColorMatch对象分配一个唯一的名称。 默认：ColorMatch01
NumberOfModels	仅限运行时间。 这是所用的颜色模型的数目。在运行时间里，如果设置了该属性，则可以利用 CurrentModel 和 VTeach 示教各颜色模型。 默认：1
NumberToFind	规定了在当前搜索窗口中待查找的对象的数量。 默认：1
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound

属性	说明
Radius	定义从对象中心到对象最外层搜索环的距离。 默认：50
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinAngle	定义待搜索区域的角度。
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	定义待搜索区域的高度(单位： pixel)。
SearchWinLeft	定义待搜索区域的最左边位置(单位： pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位： pixel)。
SearchWinType	定义待搜索区域的类型(Rectangle,RotatedRectangle,Circle)
SearchWinWidth	定义待搜索区域的宽度(单位： pixel)。

#### ColorMatch结果

以下列表是对 ColorMatch 对象结果的总结，并附有简要说明。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
CameraX	返回相机坐标系内已找到的零件位置(参见模型原点)的 X 坐标位置。(单位： mm)
CameraY	返回相机坐标系内已找到的零件位置(参见模型原点)的 Y 坐标位置。(单位： mm)
ColorIndex	返回查找到的颜色模型的指数。
ColorName	返回查找到的颜色模型的名称。
ColorValue	根据 ColorMode 设置，返回查找到颜色模型的 RGB 或 HSV 值。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到零件位置的 CameraX、CameraY 和 CameraU 坐标。(单位： mm)
Found	返回颜色是否与某一颜色模型相匹配。
NumberFound	返回查找到的对象的数目。(最少是 0 个，最多是 NumberToFind 属性中设定的检测数量)。
Passed	返回测试结果合格与否。
PixelX	返回已找到的零件位置的 X 坐标位置(以像素为单位)。
PixelY	返回已找到的零件位置的 Y 坐标位置(以像素为单位)。


结果	说明
PixelXYU	仅限运行时间。 返回已找到的零件位置的 PixelX、PixelY 和 PixelU 坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到的对象的 X 坐标位置。
RobotY	返回机器人坐标系中检测到的对象的 Y 坐标位置。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的零件位置的 RobotX、RobotY 和 RobotU 坐标。
Score	被检测出对象的颜色与模型颜色的匹配程度。 返回一个 0~1000 的 INTEGER 数值
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。 当 NumberToFind 属性设置为 0 或大于 1 时，比较结果将更加容易。
Time	返回处理对象所需要的时间。

### 使用ColorMatch对象

现在我们已经回顾了 ColorMatch 和搜索的工作方式，并已经为理解如何使用 Vision Guide ColorMatch 对象打下了基础。因此，本节将说明使用 ColorMatch 对象所需的步骤，如下所示：

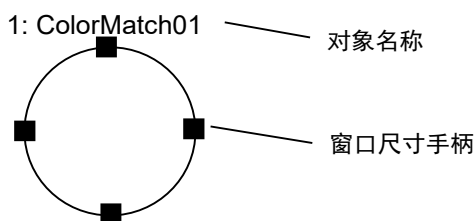
- 新建一个 ColorMatch 对象
- 设置搜索窗口的位置和尺寸
- 设置 ColorMatch 对象的相关属性
- 示教颜色模型
- 测试 ColorMatch 对象并检查结果
- 调整属性并再次测试
- 处理单个 ColorMatch 对象的多个结果

#### 第1步：新建一个ColorMatch对象

- (1) 点击 Vision Guide 工具栏中的 <All Tools>- <ColorMatch>按钮。
- (2) 鼠标指针将变成一个 ColorMatch 图标。
- (3) 将鼠标指针移动至 Vision Guide 窗口的图像显示区域，点击鼠标左键。  
ColorMatch 对象将被防置在图像显示区域中。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“ColorMatch01”，因为它是在本序列中创建的第一个 ColorMatch 对象。(稍后将说明如何修改名称。)

### 第2步：设置搜索窗口的位置和尺寸

屏幕上将显示一个与下图相似的 ColorMatch 对象。



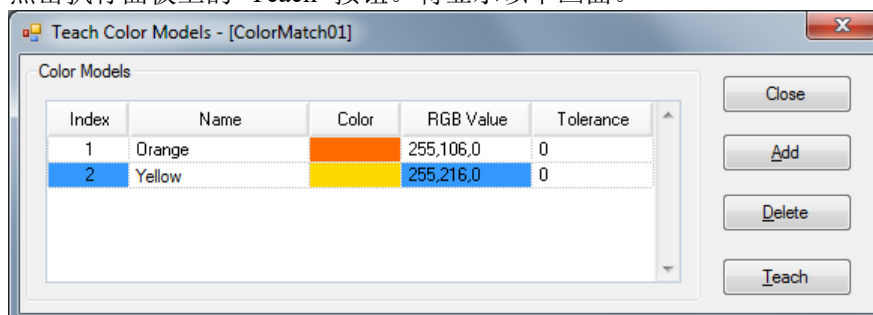
新ColorMatch对象布局

- (1) 点击 ColorMatch 对象的名称标签，并按下鼠标，将该对象拖动至您想让搜索窗口所在的位置。
- (2) 点击搜索窗口的尺寸手柄，拖动鼠标，将 Correlation 对象搜索窗口调整成合适的大小。(搜索窗口就是颜色搜索区域。)

### 第3步：示教模型

要使用 ColorMatch 对象，首先需要示教一个或多个颜色模型。示教颜色模型时，平均颜色取决于 ColorMatch 对象窗口内的所有像素。您可以命名每个模型。

- (1) 确认当前显示的是 ColorMatch 对象。查看流程图或序列目录树，确认当前正在处理的对象。或者，检查图像显示区域，查看正以洋红色突出显示的对象
- (2) 点击执行面板上的<Teach>按钮。将显示以下画面。



当显示示教窗口时，您可以根据需要改变 ColorMatch 对象的位置，以示教各颜色模型。

- (3) 将 ColorMatch 对象放置在您需要示教的颜色上。试着用示教的颜色填满整个窗口。
- (4) 点击<Add>按钮添加新的颜色模型。
- (5) 点击模型列，选择要示教的模型。
- (6) 点击<Teach>按钮示教颜色。
- (7) 命名颜色。此名称将作为 ColorName 结果，因此请设置名称，以分辨颜色。
- (8) 当 ColorMode 为 RGB 时，默认容差值为 0；当 ColorMode 为 HSV 时，默认容差值为 0, 0, 50。当颜色变化很小或亮度不均匀时，修改容差量有助于颜色匹配。
- (9) 如需添加更颜色多模型，请重复步骤(3)至(8)。



NOTE

通常您可以使用 ColorMatch 对象窗口示教颜色模型，但也可以在示教窗口中输入 RGB(或 HSV)值以示教颜色模型。

### 6.2.10 LineFinder

#### LineFinder对象描述

LineFinder对象用于识别图像中直线的位置。

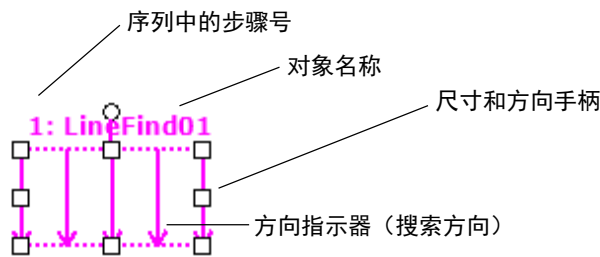
LineFinder对象自动处理多个Edge对象以识别边缘位置，然后获取从每个边缘位置识别出的直线。

图像中的对象的边缘由灰度值“从深到浅”或“从浅到深”的变化表示。可以在多个像素上观察这种变化。

LineFinder对象先根据Polarity属性的定义(“从深到浅”或“从浅到深”的变化)，检测出深浅变化，然后将检测出的位置定义为单边缘的边缘位置。您还可以更改EdgeType属性，搜索双边缘。搜索双边缘时，会查找两个相对的边，并返回中间点作为结果。

#### LineFinder对象布局

LineFinder对象的外观与Correlation对象和Blob对象看起来不同。LineFinder对象的搜索窗口直线，Edge对象沿着直线进行搜索。LineFinder对象沿着直线，朝方向指示器指示的方向，搜索颜色的变化(由浅到深或由深到浅)。



#### LineFinder对象布局

可以将LineFinder对象放在任何方向(不仅是垂直或水平)上，进行搜索。与Blob对象的SearchWinType=AngledRectangle一样，可以使用LineFinder对象搜索窗口的尺寸和方向手柄，将LineFinder对象移动到您需要检测出边缘的方向。



## LineFinder对象属性

以下列表是对LineFinder对象属性的总结。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。 默认: 100
AngleBase	设置基准角度。 默认: 0
AngleMode	设置角度输出格式。 默认: 1-Default
AngleStart	规定了角度搜索的中心。
Caption	为LineFinder对象指定标题。 默认: Empty String
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则LineFinder对象将应用于指定视觉对象的所有结果(NumberFound)。
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 如果将 SearchWinType 设置为 RotatedRectangle，搜索窗口则以 Angle 结果为基础进行旋转。 默认: False
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。
CheckClearanceFor	设定确认间隙的对象
ClearanceCondition	设定间隙的判断方法
ContrastTarget	设定边缘搜索所需的对比度。 默认: 0 (best contrast)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认: 0

属性	说明
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认：空白
Directed	指定是否使用直线方向设置角度。 默认：True
EdgeSort	设置检测到的边缘结果的排序方式。
EdgeThreshold	设置阈值，低于该值的边缘将被忽略。 默认：2
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认：1 - Single
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
FittingThreshold	指定用于直线拟合的边缘结果。
Frame	指定要使用哪个位置的定位框。 默认：none
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	选择对象标签的背景颜色。 默认：Transparent
Name	用于为LineFinder对象分配一个唯一的名称。 默认：LineFind01
NumberOfEdges	指定要查找边缘的数目。 默认：5
PassColor	选择合格对象的颜色。 默认：Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound
Polarity	规定 LineFinder 对象是应该搜索浅色到深色的过度(LightToDark)还是深色到浅色的过度(DarkToLight)。 默认：1-LightToDark
ScoreWeightContrast	根据对比度设置得分百分比。 默认：50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认：50

属性	说明
SearchWidth	定义边缘搜索的宽度。 范围：3 - 99。 默认：3
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinAngle	定义待搜索区域的角度
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	定义待搜索区域的高度(单位： pixel)。
SearchWinLeft	定义待搜索区域的最左边位置(单位： pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位： pixel)。
SearchWinWidth	定义待搜索区域的宽度(单位： pixel)。
StrengthTarget	设置搜搜索边缘的强度。 默认： 0
StrengthVariation	设置StrengthTarget的变化量总和。 默认： 0
X1	边缘起点的X坐标位置。
X2	边缘终点的X坐标位置。
Y1	边缘起点的Y坐标位置。
Y2	边缘终点的Y坐标位置。

## LineFinder对象结果

以下列表是对Edge对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	返回图像坐标系中检测到的边缘角度。
CameraX1	返回相机坐标系中检测到的边缘线起点的X坐标。
CameraY1	返回相机坐标系中检测到的边缘线起点的Y坐标。
CameraX2	返回相机坐标系中检测到的边缘线终点的X坐标。
CameraY2	返回相机坐标系中检测到的边缘线终点的Y坐标。
ClearanceOK	返回间隙的判断结果。
Contrast	返回检测到的边缘的平均对比度。
EdgeCameraXYU	返回在搜索过程中检测出来的边缘的CameraX, CameraY, Angle位置坐标。
EdgePixelXYU	返回在搜索过程中检测出来的边缘的PixelX, PixelY, Angle位置坐标。
EdgeRobotXYU	返回在搜索过程中检测出来的边缘的RobotX, RobotY, Angle位置坐标。
FitError	以均方根(RMS)返回各边缘点与检测到直线之间的距离。
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了Accept属性的当前设置值。
Passed	返回测试结果合格与否。
PixelLine	仅限运行时间。 返回X1、Y1、X2和Y2四个直线坐标，并用像素表示。
Length	以mm为单位返回检测到的边缘线的长度
NumberFound	返回查找到的直线的数目。 (最少是0个，最多是NumberToFind属性中设定的检测数量)。
PixelLength	返回检测到的边缘线的长度，单位为像素。
PixelX1	返回图像坐标系中检测到的边缘线起点的X坐标位置。
PixelY1	返回图像坐标系中检测到的边缘线起点的Y坐标位置。
PixelX2	返回图像坐标系中检测到的边缘线终点的X坐标位置。
PixelY2	返回图像坐标系中检测到的边缘线终点的Y坐标位置。
RobotX1	返回机器人坐标系中检测到的边缘线起点的X坐标位置。
RobotY1	返回机器人坐标系中检测到的边缘线起点的Y坐标位置。
RobotX2	返回机器人坐标系中检测到的边缘线终点的X坐标位置。
RobotY2	返回机器人坐标系中检测到的边缘线终点的Y坐标位置。
RobotU	返回机器人坐标系中检测到的边缘线的角度。
Strength	返回检测到的边缘的平均强度。
Time	返回处理对象所需要的时间(单位：毫秒)。

## 使用LineFinder对象

接下来的几节将指导您如何创建和使用LineFinder对象。


- 创建一个新的 LineFinder 对象
- 配置搜索窗口的位置和尺寸
- 配置 LineFinder 对象的相关属性
- 测试 LineFinder 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

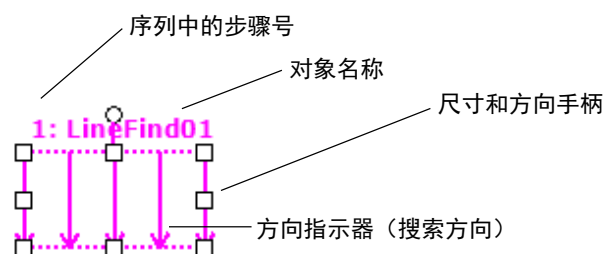
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：新建一个LineFinder对象。

- (1) 点击 Vision Guide 工具栏中的<All Tools>- < LineFinder>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 LineFinder 对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“LineFind01”，因为它是在本序列中创建的第一个 LineFind 对象。(稍后将说明如何修改名称。)

### 第2步：配置搜索窗口的位置和尺寸

屏幕上将显示一个与下图相似的LineFinder对象



#### 新LineFinder对象

1. 点击 LineFinder 对象的名称标签，然后保持按下鼠标的同时拖动 LineFinder 对象，以将搜索窗口的左上角与所需位置对齐。
2. 点击尺寸手柄并拖动鼠标，更改 LineFinder 对象搜索窗口的大小，并将其调整为适当尺寸。

### 第3步：配置LineFinder对象的属性

现在设置 LineFinder 对象的属性值。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项即可。

以下显示的是一些 LineFinder 对象更常用的属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等。使用 AngleMode 属性设置角度输出格式。有关更多详细内容，请参阅 *Line* 对象。

- EdgeType(单边缘)**      选择搜索边缘的类型。
- 如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。
- Name属性 (“LineFindxx”)**      新创建的 LineFinder 对象中，默认的项目名称是“LineFind\*\*”。在一个视觉序列中要使用多个LineFinder对象时，使用\*\*的编号对LineFinder进行区分。
- 默认情况下，会将在视觉序列序列中使用的第一个 LineFinder对象命名为“LineFind01”。
- 需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示LineFinder对象名称的地方都会变为新的名称。
- NumberOfEdges (1)**      沿着边缘搜索线搜索1条或以上的边缘。
- Polarity (LightToDark)**      如果您要寻找DarkToLight(深色到浅色)的边缘，请更改Polarity设置。


#### 第4步：运行LineFinder对象并检查结果

运行 LineFinder 对象仅需做到以下几点：

点击执行面板上对象的<Run>按钮。

然后将显示LineFinder对象的结果。这时需检查的主要结果如下。

<b>Angle结果</b>	返回图像坐标系中检测到的边缘线的角度。
<b>Length结果</b>	返回相机坐标系中检测到的边缘线的长度。 单位： mm
<b>PixelLength结果</b>	返回图像坐标系中检测到的边缘线的长度。 单位： pixels
<b>PixelX1结果</b> <b>PixelY1结果</b>	返回图像坐标系中检测到的边缘线起点的XY坐标位置。
<b>PixelX2结果</b> <b>PixelY2结果</b>	返回图像坐标系中检测到的边缘线终点的XY坐标位置。
<b>CameraX1结果</b> <b>CameraY1结果</b>	返回相机坐标系中检测到的边缘线起点的XY坐标位置。 如果未进行校准，则返回“no cal”。
<b>CameraX2结果</b> <b>CameraY2结果</b>	返回相机坐标系中检测到的边缘线终点的XY坐标位置。 如果未进行校准，则返回“no cal”。
<b>RobotX1结果</b> <b>RobotY1结果</b>	返回机器人坐标系中检测到的边缘线起点的XY坐标位置。 如果未进行校准，则返回“no cal”。
<b>RobotX2结果</b> <b>RobotY2结果</b>	返回机器人坐标系中检测到的边缘线终点的XY坐标位置。 如果未进行校准，则返回“no cal”。
<b>RobotU结果</b>	返回机器人坐标系中检测到的边缘线的角度。

6.2.11  LineInspector对象

**LineInspector对象描述**

LineInspector对象用于检查图像中的直线。

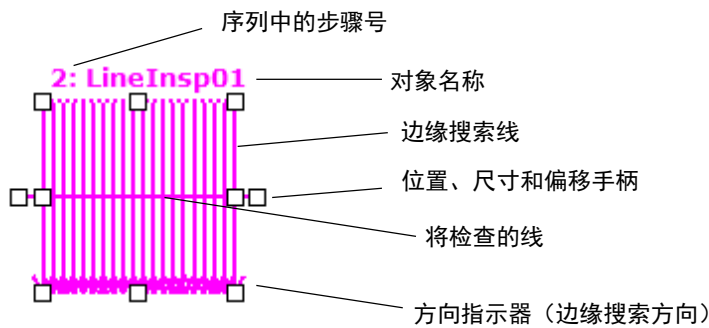
LineInspector对象会自动处理多个Edge对象，以识别所检查直线的缺陷。

图像中的对象的边缘由灰度值“从深到浅”或“从浅到深”的变化表示。可以在多个像素上观察这种变化。

LineFinder 对象先根据 Polarity 属性的定义(“从深到浅”或“从浅到深”的变化)，检测出深浅变化，然后将检测出的位置定义为单边缘的边缘位置。您还可以更改 EdgeType 属性，搜索双边缘。搜索双边缘时，会查找两个相对的边，并返回中间点作为结果。

**LineInspector对象布局**

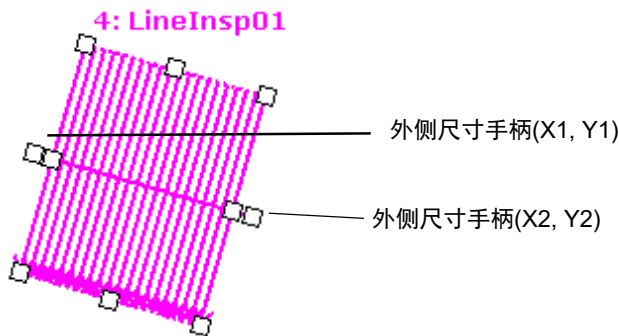
LineInspector 对象的外观与 LineFinder 工具类似。LineInspector 对象的搜索窗口包含多条边缘搜索线。LineInspector 对象沿着直线，朝方向指示器指示的方向，搜索颜色的变化(由浅到深或由深到浅)。通过边缘搜索读取的数据用于确定直线的缺陷。



**LineInspector对象布局**

可以将LineInspector对象放在任何方向(不仅是垂直或水平)上，进行搜索。

如需旋转对象，请顺时针或逆时针拖动外侧的尺寸手柄。LineFinder结果可用于要检查的直线。这种情况下，可将LineFinder设置为LineObject。



如需更改待检查的直线的宽度，请拖拽外部尺寸手柄，使其靠近或远离直线中心即可。

如需更改边缘搜索的尺寸，可拖动内部尺寸手柄。

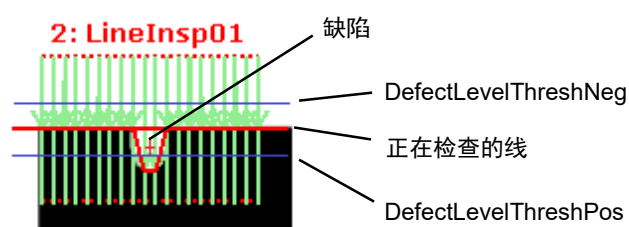


## LineInspector搜索

下图所示为有缺陷的对象的一部分。



LineInspector会找到缺陷，如下所示。所检查的直线中每个边缘搜索位置必须超过DefectLevelThreshPos或DefectLevelThreshNeg属性的值。此外，缺陷面积必须大于MinArea并小于MaxArea。



## LineInspector对象属性

以下列表是对LineInspector对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。 默认: 100
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos
Caption	为LineInspector对象指定标题。 默认: Empty String
ContrastTarget	设定边缘搜索所需的对比度。 默认: 0 (best contrast)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认: 0
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认: None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。

属性	说明
DefectAreaExtended	设置是否扩展缺陷区域插值。 默认: False
DefectLevelThreshNeg	设置直线下方的缺陷阈值。 默认: 2
DefectLevelThreshPos	设置直线上方的缺陷阈值。 默认: 2
Description	设定用户指定的描述 默认: 空白
EdgeSort	设置检测到的边缘结果的排序方式。
EdgeThreshold	设置阈值, 低于该值的边缘将被忽略。 默认: 2
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认: 1 - Single
Enabled	指定是否执行对象。 默认: True
EndPntObjResult	指定使用 EndPointObject 的哪个结果。 默认: 1
EndPointObject	指定使用哪个视觉对象来定义直线的终点。 默认: Screen
EndPointType	指定用于定义直线终点的终点类型。 默认: 0 - Point
FailColor	选择不合格对象的颜色。 默认: Red
Frame	指定要使用哪个位置的定位框。 默认: none
FrameResult	指定需使用Frame的第几个结果。 默认: 1
Graphics	设置显示的图表模式。 默认: 1 - All
InspectEndOffset	设置停止检查的直线终点的偏移量 默认: 15
InspectStartOffset	设置开始检查的直线起点的偏移量 默认: 15
LabelBackColor	选择对象标签的背景颜色。 默认: Transparent
LineObject	规定在检查之前, 用于查找直线的对象。 默认: None
LineObjResult	定义在LineObject中将使用的结果。 默认: 1

属性	说明
MaxArea	确定缺陷的Area上限。被检测到的缺陷对象返回的Area结果，必须小于MaxArea属性的设置值。 默认：100,000
MinArea	确定缺陷的Area下限。被检测到的缺陷对象返回的Area结果，必须大于MinArea属性的设置值。 默认：25
MissingEdgeType	确定如何处理未检测出的边缘。 默认：Interpolate
Name	用于为LineInspector对象分配一个唯一的名称。 默认：LineInsp01
NumberOfEdges	指定要查找边缘的数目。 默认：20
NumberToFind	确定在搜索区域中查找的缺陷数目。 默认：1
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认：AllNotFound
Polarity	规定LineInspector对象是应该搜索浅色到深色的过度(LightToDark)还是深色到浅色的过度(DarkToLight)。 默认：1-LightToDark
ScoreWeightContrast	根据对比度设置得分百分比。 默认：50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认：50
SearchWidth	定义边缘搜索的宽度。 宽度范围为3 - 99。 默认：3
SearchWinHeight	定义待搜索区域的高度(单位：pixel)。
SearchWinLeft	定义待搜索区域的最左边位置(单位：pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位：pixel)。
SearchWinWidth	定义待搜索区域的宽度(单位：pixel)。
SizeToFind	选择要查找的缺陷尺寸。 默认：1 - 最大值
StartPntObjResult	指定使用 StartPointObject 的哪个结果。 默认：1
StartPointObject	指定使用哪个视觉对象来定义直线的起点。 默认：Screen
StartPointType	指定用于定义直线起点的起点类型。 默认：“0 - Point”

属性	说明
StrengthTarget	设置搜索边缘的强度。 默认: 0
StrengthVariation	设置StrengthTarget的变化量总和。 默认: 0
X1	边缘起点的X坐标位置。
X2	边缘终点的X坐标位置。
Y1	边缘起点的Y坐标位置。
Y2	边缘终点的Y坐标位置。

### LineInspector对象结果

以下列表是对LineInspector对象结果的总结，并附有简要说明。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Area	返回缺陷的面积，用像素表示。
CameraX	返回相机坐标系中缺陷的X坐标。
CameraY	返回相机坐标系中缺陷的Y坐标。
Contrast	返回检测到的边缘的平均对比度。
DefectLevel	返回缺陷水平。
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了Accept属性的当前设置值。
Length	返回缺陷的长度，单位为毫米。
Passed	返回测试结果合格与否。
PixelLength	返回缺陷的长度，用像素表示。
NumberFound	返回查找到的缺陷数目。(最少是0个，最多是NumberToFind属性中设定的检测数量)。
PixelX	返回缺陷的X坐标位置。
PixelY	返回缺陷的Y坐标位置。
PixelXYU	仅限运行时间。 返回已找到的缺陷位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
RobotX	返回机器人坐标系中缺陷的X坐标值。
RobotY	返回机器人坐标系中缺陷的Y坐标值。
RobotU	返回机器人坐标系中缺陷的U坐标值。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的缺陷位置的RobotX、RobotY和RobotU坐标。
Strength	返回检测到的边缘的平均强度。
Time	返回处理对象所需要的时间(单位: 毫秒)。
TotalArea	返回所有已查找到的缺陷的总面积，用像素表示。

## 使用LineInspector对象

接下来的几节将介绍如何创建和使用LineInspector对象。


- 创建一个 LineInspector 对象
- 配置搜索窗口的位置和尺寸
- 配置 LineInspector 对象的相关属性
- 测试 LineInspector 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

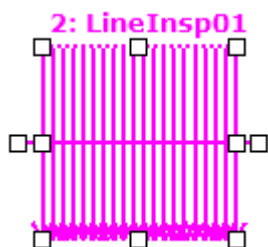
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：创建一个新的LineInspector对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>- <LineInspector>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 LineInspector 对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“LineInsp01”，因为它是在本序列中创建的第一个 LineInspector 对象。(稍后将说明如何修改名称。)

### 第2步：配置搜索窗口的位置和尺寸

屏幕上将显示一个与下图相似的LineInspector对象：



#### 新建LineInspector对象

1. 点击 LineInspector 然后保持按下鼠标的同时拖动 LineInspector 对象，以将搜索窗口的左上角与所需位置对齐。
2. 点击尺寸手柄并拖动鼠标，更改 LineInspector 对象搜索窗口的大小，并将其调整为适当尺寸。

### 第3步：配置LineInspector对象的属性

现在，我们可以设置 LineInspector 对象的属性值了。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项即可。

以下显示的是一些 LineInspector 对象更常用的属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等。

- EdgeType(单边缘)**     选择搜索边缘的类型。  
如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。
  
- Name属性 (“LineInspxx”)**     新创建的 LineInspector 对象中，默认的项目名称是 “LineInsp\*\*”。在一个视觉序列中要使用多个LineInspector对象时，使用\*\*的编号对LineInspector进行区分。  
默认情况下，会将在视觉序列序列中使用的第一个 LineInspector对象命名为 “LineInsp01”。  
需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示LineInspector对象名称的地方都会变为新的名称。
  
- NumberToFind (1)**     沿着边缘搜索线搜索1条或以上的边缘。
  
- Polarity (LightToDark)** 如果您要寻找DarkToLight(深色到浅色)的边缘，请更改Polarity设置。

### 第4步：运行LineInspector对象并检查结果

如果需要运行 LineInspector 对象，仅需做到以下几点：

点击执行面板上对象的<Run>按钮。

然后将显示LineInspector对象的结果。这时需检查的主要结果有：

- Area结果**             查找到的缺陷面积(用像素表示)。
- PixelX结果**            返回图像坐标系中缺陷的XY坐标。
- PixelY结果**
  
- CameraX结果**           在相机坐标系中返回缺陷对象的XY坐标值。
- CameraY结果**           如果相机尚未完成校准则不会返回任何值，而是返回[No Cal]。
  
- RobotX结果**            返回机器人坐标系中缺陷的XY坐标。
- RobotY结果**            如果未进行校准，则返回“no cal”。

## 6.2.12 ArcFinder对象

### ArcFinder对象描述

ArcFinder对象用于识别图像中圆弧/椭圆弧的位置。

LineFinder对象自动处理多个边缘搜索，以识别圆弧的半径和中心点，或椭圆弧的长轴、短轴和角度，然后识别出圆弧/椭圆弧。

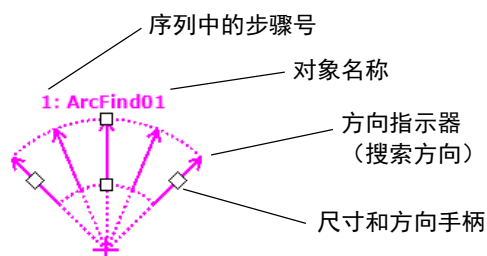
图像中的对象的边缘由灰度值“从深到浅”或“从浅到深”的变化表示。可以在多个像素上观察这种变化。

ArcFinder对象先根据Polarity属性的定义(“从深到浅”或“从浅到深”的变化)，检测出深浅变化，然后将检测出的位置定义为单边缘的边缘位置。您还可以更改EdgeType属性，搜索双边缘。搜索双边缘时，会查找两个相对的边，并返回中间点作为结果。

使用ArcSearchType属性，定义边缘类型(圆弧/椭圆弧)。

### ArcFinder对象布局

ArcFinder对象与Correlation对象和Blob对象不同，ArcFinder对象的搜索窗口为圆形，并通过起点角度、终点角度、外半径和内半径来确定。边缘搜索线均匀分布在起点角度和终点角度之间。边缘搜索线的数目通过NumberOfEdges属性指定。Direction属性可以指定搜索方向(从内径到外径，或从外径到内径)。



### ArcFinder对象布局

ArcFinder对象将中心点移至要搜索的弧形的中心近似点，然后调整RadiusInner和RadiusOuter属性，从而使待查找的弧位于搜索区域内。

### ArcFinder对象属性

以下列表是对ArcFinder对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual*中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认：False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。 默认：100

属性	说明
AngleEnd	指定执行圆形/椭圆形搜索范围的终止角度。 默认: 135
AngleStart	指定执行圆形/椭圆形搜索范围的起始角度。 默认: 45
ArcSearchType	指定要搜索的边缘类型(圆弧/椭圆弧)。
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为ArcFinder对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时, 则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时, 该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All, 则ArcFinder对象将应用于指定视觉对象的所有结果(NumberFound)。 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后, 设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后, 设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以CenterPointObject的Angle结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的XY偏移。 默认: False
CenterX	指定用作对象中心位置的X坐标。 当CenterPointObject属性已被其他视觉对象设定时, 本属性将无效且被自动设置。
CenterY	指定用作对象中心位置的Y坐标。 当CenterPointObject属性已被其他视觉对象设定时, 本属性将无效且被自动设置。
CheckClearanceFor	设定确认间隙的对象
ClearanceCondition	设定间隙的判断方法
ContrastTarget	设定边缘搜索所需的对比度。 默认: 0(最佳对比度)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认: 0



属性	说明
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认：空白
Direction	设置边缘搜索的方向。 默认：InsideOut
EdgeSort	设置检测到的边缘结果的排序方式。
EdgeThreshold	设置阈值，低于该值的边缘将被忽略。 默认：2
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认：1 - Single
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
FittingThreshold	指定用于弧线拟合的边缘结果。
Frame	指定要使用哪个位置的定位框。 默认：none
FrameResult	指定需使用Frame的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	选择对象标签的背景颜色。 默认：Transparent
Name	用于为ArcFinder对象分配一个唯一的名称。 默认：ArcFind01
NumberOfEdges	指定要查找边缘的数目。 默认：5
PassColor	选择合格对象的颜色。 默认：LightGreen
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound
Polarity	规定ArcFinder对象是应该搜索浅色到深色的过度(LightToDark)还是深色到浅色的过度(DarkToLight)。 默认：1-LightToDark
RadiusInner	指定检测范围的内径。

属性	说明
RadiusOuter	指定检测范围的外径。
ScoreWeightContrast	根据对比度设置得分百分比。 默认: 50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认: 50
SearchWidth	定义边缘搜索的宽度。 范围: 3 - 99。 默认: 3
ShowExtension	确定是否显示两端延伸的边缘线。
StrengthTarget	设置搜索边缘的强度。 默认: 0
StrengthVariation	设置StrengthTarget的变化量总和。 默认: 0

### ArcFinder对象结果

以下列表是对ArcFinder对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。


结果	说明
Angle	返回检测到的椭圆弧的角度。
Angle1	返回图像坐标系中检测到的边缘的起点角度。
Angle2	返回图像坐标系中检测到的边缘的终点角度。
CameraX	返回相机坐标系中检测到的圆弧/椭圆弧边缘的中心X坐标。
CameraY	返回相机坐标系中检测到的圆弧/椭圆弧边缘的中心Y坐标。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到零件位置的CameraX、CameraY和CameraU坐标。
ClearanceOK	返回间隙的判断结果
Contrast	返回检测到的圆弧/椭圆弧边缘的对比度。
EdgeCameraXYU	返回在搜索过程中检测出来的边缘的CameraX, CameraY, Angle位置坐标。
EdgePixelXYU	返回在搜索过程中检测出来的边缘的PixelX, PixelY, Angle位置坐标。
EdgeRobotXYU	返回在搜索过程中检测出来的边缘的RobotX, RobotY, Angle位置坐标。
FitError	以均方根(RMS)返回各边缘点与检测到圆弧之间的距离。
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了Accept属性的当前设置值。
FoundMajorDiam	检测到的椭圆形边缘的长轴。
FoundMinorDiam	检测到的椭圆形边缘的短轴。
FoundRadius	检测到的圆形边缘的半径。

结果	说明
MaxError	返回与检测到的圆弧/椭圆弧边缘的最大偏差，以像素表示。
NumberFound	返回查找到的圆弧数目。
Passed	返回视觉对象的测试结果合格与否。
PixelMajorDiam	返回ArcFinder检测到的椭圆弧的长轴的长度。
PixelMinorDiam	返回ArcFinder检测到的椭圆弧的短轴的长度。
PixelRadius	返回检测到的圆弧的半径(单位: 像素)
PixelX	返回图像坐标系中检测到的圆弧/椭圆弧边缘中心的X坐标。
PixelY	返回图像坐标系中检测到的圆弧/椭圆弧边缘中心的Y坐标。
PixelXYU	仅限运行时间。 返回已找到的圆/椭圆形边缘中心位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到的圆弧/椭圆弧边缘中心的X坐标。
RobotY	返回机器人坐标系中检测到的圆弧/椭圆弧边缘中心的 Y 坐标。
RobotXYU	返回机器人坐标系中已找到的圆弧/椭圆弧边缘中心位置的RobotX、RobotY和RobotU坐标。
Strength	返回检测到的边缘的强度。
Time	返回处理对象所需要的时间(单位: 毫秒)。

### 使用ArcFinder对象

接下来的几节将指导您如何创建和使用ArcFinder对象。


- 创建一个新的 ArcFinder 对象
- 配置搜索窗口的位置和尺寸
- 配置 ArcFinder 对象的相关属性
- 测试 ArcFinder 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在 Vision Guide 窗口中单击序列目录树，选择已创建的视觉序列。

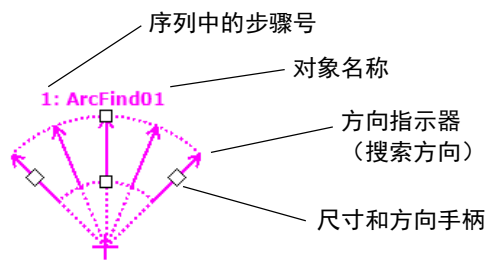
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. 视觉序列。

**第1步：新建一个ArcFinder对象**

- (1) 点击 Vision Guide 工具栏中的<All Tools>--<ArcFinder>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 ArcFinder 对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“ArcFind01”，因为它是在本序列中创建的第一个 ArcFinder。(稍后将说明如何修改名称。)

**第2步：配置搜索窗口的位置和尺寸**

屏幕上将显示一个与下图相似的ArcFinder对象。



**新ArcFinder对象**

1. 点击 ArcFinder 对象的名称标签，然后保持按下鼠标的同时拖动 ArcFinder 对象，以将搜索窗口的左上角与所需位置对齐
2. 点击尺寸手柄并拖动鼠标，更改 ArcFinder 对象搜索窗口的大小，并将其调整为适当尺寸。

### 第3步：配置ArcFinder对象的属性

现在，我们可以设置 ArcFinder 对象的属性值了。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项即可。

以下显示的是 ArcFinder 对象更常用的一些属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等。

<b>EdgeType(单边缘)</b>	选择搜索边缘的类型。  如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。
<b>Name属性 (“ArcFindxx”)</b>	新创建的 ArcFinder 对象中，默认的项目名称是“ArcFind**”。在一个视觉序列中要使用多个 ArcFinder 对象时，使用**的编号对 ArcFinder 进行区分。 默认情况下，会将在视觉序列序列中使用的第一个 ArcFinder 对象命名为“ArcFind01”。 需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 ArcFinder 对象名称的地方都会变为新的名称。
<b>NumberOfEdges(5)</b>	搜索5条边缘，以找到圆弧边缘
<b>Polarity (LightToDark)</b>	如果您要寻找 DarkToLight(深色到浅色)的边缘，则应更改 Polarity。

### 第4步：运行ArcFinder对象并检查结果

运行 ArcFinder 对象仅需完成以下操作：

点击执行面板上对象的 <Run> 按钮。然后将显示 ArcFinder 对象的结果。这时需检查的主要结果如下。

<b>Angle1结果</b>	返回图像坐标系中检测到的圆弧边缘的起点角度。
<b>Angle2结果</b>	返回图像坐标系中检测到的圆弧边缘的终点角度。
<b>FoundRadius结果</b>	返回检测到的圆弧边缘的半径。 单位：像素
<b>MaxError的结果</b>	返回与检测到的圆弧边缘的最大偏差，以像素表示。
<b>PixelX结果</b>	返回图像坐标系中检测到的圆弧边缘中心的XY坐标位置。
<b>PixelY结果</b>	
<b>CameraX结果</b>	在相机坐标系中返回圆弧检测边缘的中心 XY 坐标位置。
<b>CameraY结果</b>	如果未进行校准，则返回“no cal”。
<b>RobotX结果</b>	在机器人坐标系中返回圆弧检测边缘的中心XY坐标位置。
<b>RobotY结果</b>	如果未进行校准，则返回“no cal”。

### 6.2.13 ArcInspector对象

#### ArcInspector对象描述

ArcInspector对象用于检查圆弧/椭圆弧的缺陷。

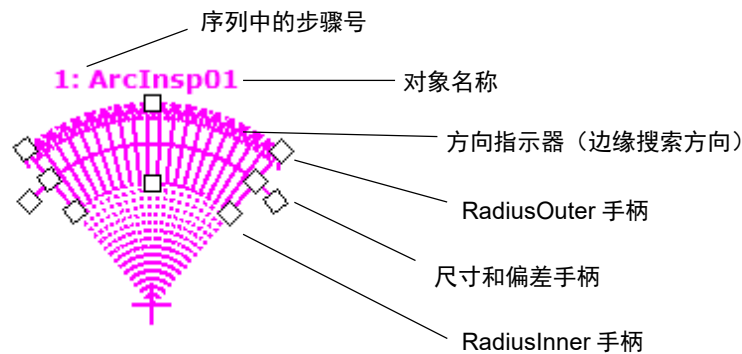
为了查找缺陷，将执行一系列边缘搜索，以确定所检查圆弧/椭圆弧中的异常情况。

图像中的对象的边缘由灰度值“从深到浅”或“从浅到深”的变化表示。可以在多个像素上观察这种变化。

LineFinder 对象先根据 Polarity 属性的定义(“从深到浅”或“从浅到深”的变化)，检测出深浅变化，然后将检测出的位置定义为单边缘的边缘位置。您还可以更改 EdgeType 属性，搜索双边缘。搜索双边缘时，会查找两个相对的边，并返回中间点作为结果。

#### ArcInspector对象布局

ArcInspector对象的外观与 ArcFinder对象类似。ArcInspector对象能从圆弧/椭圆弧的中心沿直线，朝外半径方向搜索边缘。各边缘搜索线会朝方向指示器指示的方向，搜索颜色的变化(由浅到深或由深到浅)。通过 NumberOfEdges 属性设置用于检查的边缘数。AngleStart 属性指定待检查弧的起始角度。AngleEnd 属性指定待检查弧的终点角度。边缘搜索线均匀的分布在(AngleStart + InspectStartOffset)和(AngleEnd - InspectEndOffset)之间。



#### ArcInspector对象布局

ArcInspector对象将CenterX和CenterY与圆弧的中心对齐，然后调整RadiusInner和RadiusOuter属性，使待检查的圆弧半径位于搜索区域以内，从而搜索圆弧上的缺陷。ArcInspector对象可以通过Direction属性，设置缺陷搜索方向(从内半径到外半径(默认)，或从外半径到内半径)。

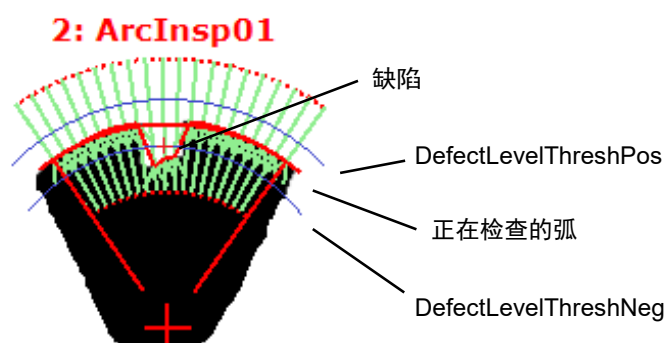
您也可以使用ArcFinder对象先找到一个圆弧，然后使用ArcInspector对象检查该弧。这需要指定ArcObject属性使用ArcFinder对象。

## ArcInspector搜索

下图所示为有缺陷的圆形对象的一部分。



ArcInspector会找到缺陷，如下所示。所检查的弧中每个边缘搜索位置必须超过DefectLevelThreshPos或DefectLevelThreshNeg属性的值。此外，缺陷面积必须大于MinArea并小于MaxArea。



## ArcInspector对象属性

以下列表是对ArcInspector对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认：False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。 默认：100
AngleEnd	指定执行圆弧/椭圆形搜索范围的终止角度。 默认：135
AngleStart	指定执行圆弧/椭圆形搜索范围的起始角度。 默认：45
ArcObject	指定ArcFinder对象，该对象将搜索要检查的圆弧/椭圆形。 默认：None
ArcObjResult	定义在ArcObject中将使用的结果。 默认：1
ArcSearchType	指定要搜索的边缘的类型(圆弧/椭圆形)。指定ArcObject属性时，需要与指定ArcFinder的ArcType匹配。
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为ArcInspector对象指定标题。 默认：Empty String

属性	说明
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认：Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则ArcInspector对象将应用于指定视觉对象的所有结果(NumberFound)。 默认：1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认：0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认：0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 默认：False
CenterX	指定用作对象中心位置的X坐标。 当CenterPointObject属性已被其他视觉对象设定时，本属性将无效且被自动设置。
CenterY	指定用作对象中心位置的Y坐标。 当CenterPointObject属性已被其他视觉对象设定时，本属性将无效且被自动设置。
ContrastTarget	设定边缘搜索所需的对比度。 默认：0 (best contrast)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认：0
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认：空白
DefectAreaExtended	设置是否扩展缺陷区域插值。 默认：False
DefectLevelThreshNeg	设置弧线下方的缺陷阈值。 默认：2



属性	说明
DefectLevelThreshPos	设置弧线上方的缺陷阈值。 默认: 2
Direction	设置边缘搜索的方向。 默认: InsideOut
EdgeSort	设置检测到的边缘结果的排序方式。
EdgeThreshold	设置阈值, 低于该值的边缘将被忽略。 默认: 2
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认: 1 - Single
EllipseAngle	指定ArcInspector的检查基线的椭圆弧的角度。
EllipseMajorDiam	指定ArcInspector的检查基线的椭圆弧的长轴长。
EllipseMinorDiam	指定ArcInspector的检查基线的椭圆弧的短轴长。
Enabled	指定是否执行对象。 默认: True
FailColor	选择当对象被检测为不良品时, 搜索窗口显示的颜色。 默认: Red
Frame	指定要使用哪个位置的定位框。 默认: none
FrameResult	指定需使用Frame的第几个结果。 默认: 1
Graphics	设置显示的图表模式。 默认: 1 - All
InspectStartOffset	设置开始检查的圆弧起点的偏移量。 默认: 5
InspectEndOffset	设置停止检查的圆弧终点的偏移量。 默认: 5
LabelBackColor	选择对象标签的背景颜色。 默认: Transparent
MaxArea	确定缺陷的Area上限。 仅可检测到小于MaxArea属性且大于MinArea属性的缺陷。 默认: 100,000
MinArea	确定缺陷的Area下限。 仅可检测到小于MaxArea属性且大于MinArea属性的缺陷。 默认: 25
MissingEdgeType	确定如何处理未检测出的边缘。 默认: Interpolate
Name	用于为ArcInspector对象分配一个唯一的名称。 默认: ArcInsp01
NumberOfEdges	指定要查找边缘的数目。 默认: 20

属性	说明
NumberToFind	确定在搜索区域中查找的缺陷数目。 默认: 1
PassColor	选择当对象被检测为良品时, 搜索窗口显示的颜色。 默认: LightGreen
PassType	可定义检测结果良品/不良品的判断标准。 默认: AllNotFound
Polarity	规定 ArcInspector 对象是应该搜索浅色到深色的过度 (LightToDark) 还是深色到浅色的过度 (DarkToLight)。 默认: 1-LightToDark
Radius	定义从对象中心到对象最外层搜索环的距离。
RadiusInner	定义被检测出范围的内径。
RadiusOuter	定义被检测出范围的外径。
ScoreWeightContrast	根据对比度设置得分百分比。 默认: 50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认: 50
SearchWidth	定义边缘搜索的宽度。 宽度范围为3 - 99。 默认: 3
SizeToFind	选择要查找的缺陷尺寸。 默认: 1 - 最大值
StrengthTarget	设置搜索边缘的强度。 默认: 0
StrengthVariation	设置StrengthTarget的变化量总和。 默认: 0

#### ArcInspector对象结果

以下列表是对ArcInspector对象结果的总结, 并附有简要说明。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。


结果	说明
Area	返回缺陷的面积, 用像素表示。
CameraX	返回相机坐标系中缺陷的X坐标。
CameraY	返回相机坐标系中缺陷的Y坐标。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到零件位置的CameraX、CameraY和CameraU坐标。
Contrast	返回检测到的边缘的平均对比度。
DefectLevel	返回缺陷水平。
Length	返回缺陷的长度, 单位为毫米。
PixelLength	返回缺陷的长度, 用像素表示。

结果	说明
NumberFound	返回查找到的缺陷数目。(最少是0个, 最多是NumberToFind属性中设定的检测数量)。
Passed	返回视觉对象的测试结果合格与否。
PixelX	返回图像坐标系中缺陷的X坐标。
PixelY	返回图像坐标系中缺陷的Y坐标。
PixelXYU	仅限运行时间。 返回已找到的缺陷位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
RobotX	返回机器人坐标系中缺陷的X坐标。
RobotY	返回机器人坐标系中缺陷的Y坐标。
RobotU	返回机器人坐标系中缺陷的U坐标。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的缺陷位置的RobotX、RobotY和RobotU坐标。
Strength	返回检测到的边缘的强度。
Time	返回处理对象所需要的时间, 单位毫秒。
TotalArea	返回所有已查找到的缺陷的总面积, 用像素表示。

### 使用ArcInspector对象

接下来的几节将指导您如何创建和使用ArcInspector对象。

- 创建一个新的 ArcInspector 对象
- 配置搜索窗口的位置和尺寸
- 配置 ArcInspector 对象的相关属性
- 测试 ArcInspector 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前, 首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列, 点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列, 请在Vision Guide窗口中单击序列目录树, 选择已创建的视觉序列。

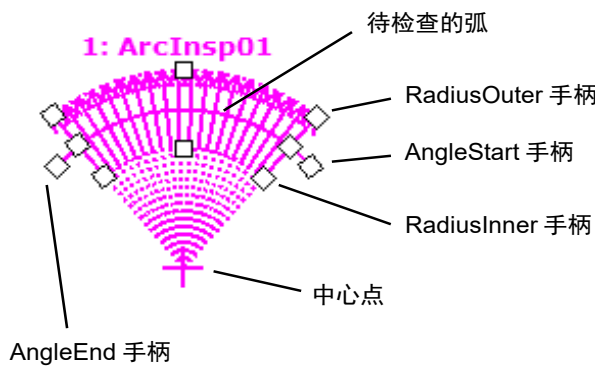
有关如何创建新的视觉序列, 以及如何选择已创建视觉序列的详细咨询, 请参阅 5. *视觉序列*。

### 第1步：创建一个新的ArcInspector对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>-<ArcInspector>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 ArcInspector 对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“ArcInsp01”，因为它是在本序列中创建的第一个 ArcInspector 对象。(稍后将说明如何修改名称。)

### 第2步：配置搜索窗口的位置和尺寸

屏幕上将显示一个与下图相似的ArcInspector对象：



新建ArcInspector对象

1. 点击 ArcInspector 对象的名称标签，然后保持按下鼠标的同时拖动 ArcInspector 对象，以使中心点位于要检查圆弧的中心附近。
2. 点击 RadiusOuter, RadiusInner, AngleStart 和 AngleEnd 尺寸手柄并拖动鼠标，更改 ArcInspector 对象搜索窗口的大小，并将其调整为适当尺寸。

### 第3步：配置ArcInspector对象的属性

现在设置 ArcInspector 对象的属性值了。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项目即可。

以下显示的是一些 ArcInspector 对象更常用的属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等。

**EdgeType(单边缘)**      选择搜索边缘的类型。  
如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。

<b>Name属性</b> (“ArcInspxx”)	新创建的 ArcInspector 对象中，默认的项目名称是“ArcInsp**”。在一个视觉序列中要使用多个 ArcInspector 对象时，使用**的编号对 ArcInspector 进行区分。 默认情况下，会将在视觉序列序列中使用的第一个 ArcInspector 对象命名为“ArcInsp01”。 需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 ArcInspector 对象名称的地方都会变为新的名称。
<b>NumberOfEdges</b>	指定查找缺陷时将使用的边缘搜索数目。 默认为15，最大为99。
<b>Polarity</b>	默认的边缘搜索 Polarity 为 LightToDark (浅色到深色)。如果您要寻找 DarkToLight (深色到浅色) 的边缘，请更改 Polarity 设置。
<b>Direction</b>	指定边缘搜索应是 InsideOut (从 RadiusInner 到 RadiusOuter) 还是 OutsideIn (从 RadiusOuter 到 RadiusInner)。
<b>DefectLevelThreshPos</b>	指定弧线上方，缺陷边缘到弧线边缘的最小距离。
<b>DefectLevelThreshNeg</b>	指定弧线下方，缺陷边缘到弧线边缘的最小距离。
<b>MinArea</b>	指定缺陷的最小面积，用像素表示。
<b>MaxArea</b>	指定缺陷的最大面积，用像素表示。

#### 第4步：运行 ArcInspector 对象并检查结果

如果需要运行 ArcInspector 对象，仅需做到以下几点：

点击执行面板上对象的 <Run> 按钮。然后将显示 ArcFinder 对象的结果。这时需检查的主要结果有：

<b>Area结果</b>	查找到的缺陷面积(用像素表示)。
<b>PixelX结果</b>	返回图像坐标系中缺陷的XY坐标。
<b>PixelY结果</b>	
<b>CameraX结果</b>	在相机坐标系中返回缺陷的XY坐标。
<b>CameraY结果</b>	如果相机尚未完成校准则不会返回任何值，而是返回[No Cal]。
<b>RobotX结果</b>	返回机器人坐标系中缺陷的XY坐标。
<b>RobotY结果</b>	如果未进行校准，则返回“no cal”。

### 6.2.14 DefectFinder对象

#### DefectFinder对象描述

DefectFinder对象用于确定模板图像和输入图像之间的差异。

搜索缺陷过程中，首先会计算搜索区域和模板之间的绝对差异图像，接下来对差异图像进行二进制对象分析，以查找缺陷。

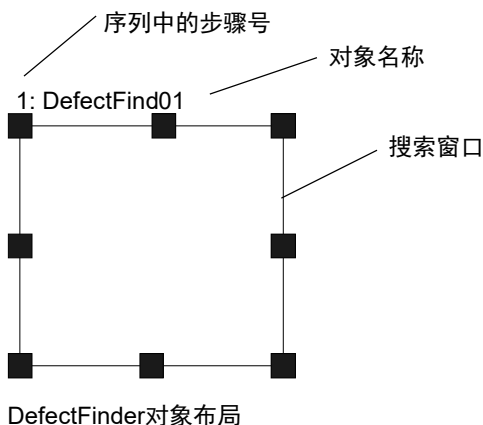
最终获得以下缺陷特征：

- 面积和周长
- 重心
- 主轴和力矩
- 连通性
- 边界矩形
- 重心在像素坐标、相机坐标和机器人坐标系中的坐标位置
- 缺陷(二进制)的孔、粗糙度和密实度

#### DefectFinder对象布局

DefectFinder对象的布局是一个矩形，与Blob对象类似。但是，DefectFinder对象需要示教模型(模板)。在示教模型时，将使用搜索窗口指定的整个区域。没有像Correlation对象那样单独的模型窗口。

搜索窗口会确定DefectFinder搜索缺陷(图像差异)的面积范围。同时，也确定模板图像的面积。DefectFinder对象的示例，如下所示。



**DefectFinder对象属性**

以下列表是对DefectFinder对象属性的总结。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为DefectFinder对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则DefectFinder对象将应用于指定视觉对象的所有结果(NumberFound)。 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心 (CenterPntOffsetX、CenterPntOffsetY) 的 XY 偏移。 如果将 SearchWinType 设置为 RotatedRectangle，搜索窗口则以 Angle 结果为基础进行旋转。 默认: False
CheckClearanceFor	设定确认间隙的对象
ClearanceCondition	设定间隙的判断方法
CoordObject	指定复制结果的 Coordinates 对象 执行对象时进行复制处理，若没有通过 Decision 等分支功能执行，则不进行复制处理。 默认: None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。 默认: 1
Description	设定用户指定的描述 默认: 空白

属性	说明
Enabled	指定是否执行对象。 默认: True
FailColor	选择当对象被检测为不良品时, 搜索窗口显示的颜色。 默认: Red
Frame	指定要使用哪个位置的定位框。 默认: None
FrameResult	指定需使用Frame的第几个结果。 默认: 1
Graphics	设置显示的图表模式。 默认: 1 - All
KernelHeight	计算与模板图形有差异时, 允许的像素差异量。(垂直方向)
KernelWidth	计算与模板图形有差异时, 允许的像素差异量。(水平方向)
LabelBackColor	选择对象标签的背景颜色。 默认: Transparent
LuminanceCorrection	设置使用亮度校正预处理。 默认: None
MaxArea	确定缺陷的Area上限。 被检测到的缺陷对象返回的Area结果, 必须小于MaxArea属性的设置值。 默认: 100,000
MinArea	确定缺陷的Area下限。 被检测到的缺陷对象返回的Area结果, 必须大于MinArea属性的设置值。 默认: 25
MinMaxArea	仅限运行时间。 用一条语句设置或返回MinArea和MaxArea。
Name	用于为DefectFinder对象分配一个唯一的名称。 默认: DefFind01
NumberToFind	确定在搜索窗口中查找的对象数目。 默认: 1
PassColor	选择合格对象的颜色。 默认: Light Green
PassType	可定义检测结果良品/不良品的判断标准。 默认: AllNotFound
Polarity	设置要检测缺陷的Polarity。 默认: Both
RejectOnEdge	如果将属性设置为“True”, 系统将忽略因为与搜索窗口边缘有接触而检测到的缺陷。 默认: False
SaveTeachImage	设置示教模型时, 是否保存图像文件。



属性	说明
SearchWin	<p>仅限运行时间。</p> <p>在一次调用中设置或返回搜索窗口的左侧参数、顶部参数、高度参数和宽度参数。</p>
SearchWinAngle	定义待搜索区域的角度。
SearchWinCenterX	定义待搜索区域中心的X轴坐标值。
SearchWinCenterY	定义待搜索区域中心的Y轴坐标值。
SearchWinHeight	<p>定义待搜索区域的高度(单位: pixel)。</p> <p>默认: 100</p>
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinType	定义待搜索区域的类型(Rectangle,RotatedRectangle,Circle)
SearchWinWidth	<p>定义待搜索区域的宽度(单位: pixel)。</p> <p>默认: 100</p>
ShowModel	<p>显示已登录的图像。</p> <p>设置不要紧的像素。</p>
SizeToFind	<p>选择要查找的缺陷尺寸。</p> <p>默认: 1 - Largest</p>
Sort	<p>选择对象结果的排序方式。</p> <p>默认: 0 - None</p>
ThresholdHigh	<p>同ThresholdLow属性一起使用, 用于定义被检测对象 (或对象)、背景和图像边缘的灰度范围。</p> <p>ThresholdHigh属性用于定义图像的被检测对象区域的灰度范围的上限。在ThresholdLow和ThresholdHigh定义的灰度范围内的图像的任何部分的像素权重都会分配为1。(即检测对象的一部分。)</p> <p>如果ThresholdAuto属性为“True”, 且ThresholdColor属性为“White”, 该属性值将被设为255且无法更改。</p> <p>默认: 128</p>
ThresholdLow	<p>同ThresholdHigh属性一起使用, 用于确定表示特征(或对象)、背景和图像边缘的像素灰度区域。区域ThresholdLow用于定义图像的被检测对象区域的灰度范围的下限。在ThresholdLow和ThresholdHigh定义的灰度范围内的图像的任何部分的像素权重都会分配为1。(即检测对象的一部分。)</p> <p>如果ThresholdAuto属性为“True”, 且ThresholdColor属性为“Black”, 该属性值将被设为0且无法更改。</p> <p>默认: 0</p>

## DefectFinder对象结果

以下列表是对DefectFinder对象结果的总结，并附有简要说明。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	返回被检测出零件的旋转量，用度数表示。
Area	返回缺陷的面积，用像素表示。
CameraX	返回相机坐标系中已找到的零件位置的 X 坐标位置。
CameraY	返回相机坐标系中已找到的零件位置的 Y 坐标位置。
CameraXYU	仅限运行时间。 返回相机坐标系中已检测到零件位置的 CameraX、CameraY 和 CameraU 坐标。
ClearanceOK	返回间隙的判断结果
Compactness	返回缺陷的密实度。
Extrema	仅限运行时间。返回缺陷极值的 MinX、MaxX、MinY 和 MaxY 像素坐标。
Found	回复是否找到了缺陷。
FoundOnEdge	当发现检测出的缺陷与搜索窗口边缘有接触时，则返回“True”。
Holes	返回缺陷中已查找到的孔洞数目。
MajorDiameter	当检测到的缺陷近似椭圆时，返回长直径。
MaxFeretDiameter	返回检测到的缺陷的最大费雷特直径。
MaxX	返回缺陷极值的最大 X 像素坐标，用像素表示。
MaxY	返回缺陷极值的最大 Y 像素坐标，用像素表示。
MinorDiameter	当检测到的缺陷近似椭圆时，返回短直径。
MinX	返回缺陷极值的最小 X 像素坐标，用像素表示。
MinY	返回缺陷极值的最小 Y 像素坐标，用像素表示。
NumberFound	返回查找到的缺陷数目。(最少是 0 个，最多是 NumberToFind 属性中设定的检测数量)。
Passed	返回测试结果合格与否。
Perimeter	沿已查找到的缺陷外部边缘的像素数目。
PixelX	返回已查找到的零件位置的 X 坐标位置，用像素表示。
PixelY	返回已查找到的零件位置的 Y 坐标位置，用像素表示。
PixelXYU	仅限运行时间。 返回已找到的零件位置的 PixelX、PixelY 和 PixelU 坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到的对象的 X 坐标位置。
RobotY	返回机器人坐标系中检测到的对象的 Y 坐标位置。
RobotU	返回机器人坐标系中检测到的对象的 U 坐标位置。


结果	说明
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的对象的 RobotX、RobotY 和 RobotU 坐标。
Roughness	返回缺陷的粗糙度。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。方便比较结果。
Time	返回处理对象所需要的时间(单位：毫秒)。
TotalArea	返回所有已查找到的缺陷的总面积，用像素表示。

DefectFinder对象对模板图像的图像差异进行二进制对象处理。想了解更多的结果相关信息，请参阅 *Blob* 对象。

### 使用DefectFinder对象

现在我们已经回顾了二进制对象分析的作用，并已经为理解如何使用 Vision Guide 7.0 DefectFinder对象打下了基础。因此，本节将介绍使用DefectFinder对象所需的步骤，如下所示：


- 新建一个 DefectFinder 对象
- 确定搜索窗口的位置和尺寸
- 配置 DefectFinder 对象的相关属性
- 示教并确认模板图像
- 测试 Blob 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

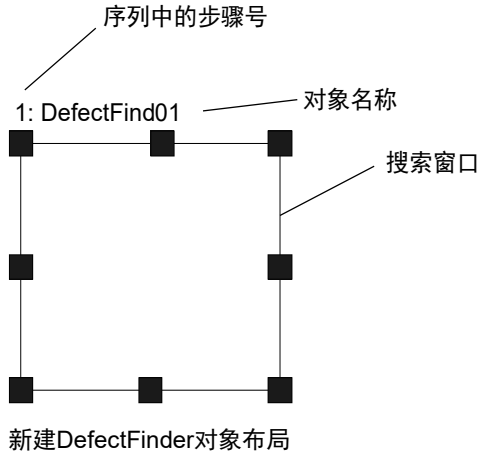
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：新建一个DefectFinder对象


- (1) 点击 Vision Guide 工具栏中的<All Tools>- <DefectFinder>按钮。
- (2) 将鼠标移动到图像显示区域。鼠标光标将变为 DefectFinder 对象图标。
- (3) 继续移动光标，直至图标位于图像显示区域中的所需位置，然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中，由于是在本序列中创建的第一个 DefectFinder 对象，所以对象名称为“DefFind01”对象名称将自动创建。(稍后将说明如何修改名称。)

### 第2步：配置搜索窗口的位置和尺寸

屏幕上将显示一个与下图相似的DefectFinder对象。




- (1) 点击 DefectFinder 对象的名称标签，然后保持按下鼠标的同时拖动 DefectFinder 对象，以将搜索窗口的左上角与所需位置对齐。
- (2) 点击尺寸手柄并拖动鼠标，更改 DefectFinder 对象搜索窗口的大小，并将其调整为适当尺寸。(搜索窗口就是我们要在其中搜索 Blob 的那个区域。)

 注意	<ul style="list-style-type: none"><li>■ 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。</li></ul>
--	---

### 第3步：配置DefectFinder对象属性

现在，我们可以设置DefectFinder对象的属性值了。以下显示的是DefectFinder对象特定且更常用的一些属性。参见*Vision Guide 7.0 Properties and Results Reference Manual* 或本章之前的“DefectFinder对象属性”列表，了解许多不同视觉对象所用其它属性的说明，如AbortSeqOnFail和Graphic等。

 注意	<ul style="list-style-type: none"><li>■ 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。正确配置MaxArea, MinArea, RejectOnEdge等属性值，可降低错误检测的风险。</li></ul>
---	--

<b>Name属性</b>	<p>新创建的DefectFinder对象中，默认的项目名称是“DefFind**”。</p> <p>在一个视觉序列中要使用多个DefectFinder对象时，使用**的编号对DefectFinder进行区分。</p> <p>默认情况下，会将在视觉序列序列中使用的第一个DefectFinder对象命名为“DefFind01”。</p> <p>需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示DefectFinder对象名称的地方都会变为新的名称。</p>
<b>KernelWidth属性</b>	设置当计算与模板图形有差异时，允许多少像素的差异量。
<b>KernelHeight属性</b>	设置的值越大，被检测对象的图像与模板图像的容差量就越大，但是将无法找到较小的缺陷。请根据被检测缺陷的大小和图像的偏差量进行设置。
<b>MinArea属性</b>	规定了DefectFinder对象的面积，在这个范围中则视为“Found”
<b>MaxArea属性</b>	(即返回的Found结果显示为“True”)。
	默认范围很大，设置为25~100,000 (MinArea ~ MaxArea)。这就意味着，在调整MinArea和MaxArea属性前，运行新建的DefectFinder对象时，即使很多细微的缺陷也会被检测出来。所以需要根据被检测对象的缺陷大小，合理调整该属性的设定值。
<b>RejectOnEdge属性</b>	不包括因接触搜索窗口而检测出来的缺陷。
<b>PassType属性</b>	可定义DefectFinder对象的检测结果良品/不良品的判断标准。
	正常情况下，如果没有查到任何缺陷，则判断为良品。将该属性设置为“AllNotFound”。

现在可以测试DefectFinder对象了。测试后将设置其它必要属性。

#### 第4步：示教并确认模板图像

如果要示教DefectFinder对象的模板图像，点击执行面板上的<Teach>按钮。如果要确认已示教完的模板图像，在属性列表中点击ShowModel属性。

#### 第5步：测试 DefectFinder对象并检查结果

要运行DefectFinder对象，点击执行面板上对象的<Run>按钮。然后将显示DefectFinder对象的结果。这时需检查的主要结果如下。当然，将来您还会发现其它有用的东西。

<b>Found结果</b>	<p>回复是否查找到了缺陷对象。</p> <p>如果查找到的缺陷对象不符合MinArea和MaxArea属性中确定的面积限制，则返回的查到结果将为“False”。</p>
<b>Passed结果</b>	返回DefectFinder对象的测试结果合格与否。
<b>Area结果</b>	检测出的缺陷对象的面积(单位：pixel)。
<b>Angle 结果</b>	<p>缺陷对象的方向角度。</p> <p>该角度根据短轴计算而得出，且介于 +/- 90度之间。</p>

Time 结果	返回DefectFinder对象执行所需时间。
PixelX、PixelY	已查找到缺陷对象重心的XY坐标位置(单位: pixel)。
MinX、MinY、MaxX、MaxY	这四个值定义了缺陷对象的边界矩形(Extrema)。

在上述步骤示例中, 由于尚未完成校准, 所以RobotXYU, RobotX, RobotY, RobotU, CameraX, CameraY, CameraXYU的结果会返回 “No Cal”。这意味着没有进行任何校准, 因此视觉系统无法计算与机器人坐标系或相机坐标系相关的坐标结果。想了解更多信息, 请参考 7. 视觉校准。

### 第6步: 调整属性并再次测试

DefectFinder 对象运行若干次之后, 您可能在查找缺陷时遇到一些问题, 或者您还想微调某些属性设置。下面将介绍一些常见问题的故障排除方法和微调技巧。

问题:

当DefectFinder对象的Found结果返回“True”时, 请确认并调整以下项目。

- 确认 Polarity 属性中设置的值。该属性定义查找方式, 是在深色背景上查找浅色对象, 还是在浅色背景上查找深色对象。确保 Polarity 属性符合您正在搜索的对象属性。
- 查看 Area 结果, 并将该值与 MinArea 和 MaxArea 属性中定义的值进行比较。如果 Area 结果不在 MinArea 和 MaxArea 属性定义的范围, 请调整 MinArea 和 MaxArea 属性, 并再次运行 Blob 对象。
- 请调整 KernelWidth 和 KernelHeight 属性。增加属性的设定值, 避免检测出一些细微的缺陷。
- 采用直方图检查图像中灰度值的分布。直方图工具对设置 ThresholdHigh 和 ThresholdLow 属性十分有用。在 8. 直方图工具 中对直方图进行了详细介绍。

微调:

某些应用中可能需要微调DefectFinder对象才能达到最佳效果。下面介绍用于微调DefectFinder对象的关键属性。

- MinArea 和 MaxArea——运行几次 DefectFinder 对象后, 可以在 Area 结果中看到一些返回的近似值。如需更改 MinArea 和 MaxArea 属性, 请使用这些近似值。一般情况下, MinArea 和 MaxArea 属性用于限制 Found 结果, 设置合理的属性值, 可以使 Found 为“True”的结果更加准确的查找到您的目标对象。(设置这两个属性, 有助于排除与想要查找的二进制对象面积不同的 Blob 对象。)
- ThresholdHigh和ThresholdLow——这两个属性通过调整灰度等级阈值的参数, 来识别被检测对象。有关详细资讯, 请参阅*Vision Properties and Results Reference Manual* 中对ThresholdHigh和ThresholdLow属性描述。

完成属性调整测试DefectFinder对象, 并得到满意的结果后, 创建视觉对象的步骤就完成了。您可以继续创建其他视觉对象, 或设置并测试整个视觉序列。

### 6.2.15 Frame对象

#### Frame对象描述

Frame 对象为视觉对象提供一种相对关系的基准。

创建新的 Frame 对象后，其他视觉对象就可以根据 Frame 对象来进行定位。因此，很多情况下都可以使用 Frame 对象。

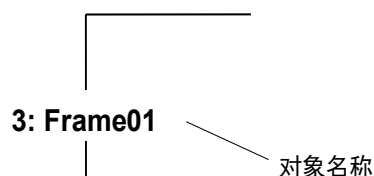
另外，如使用 Frame 对象，检测出了大概的位置并定义了框架后，基于 Frame 对象运行的其他视觉对象，则无需太大的搜索窗口，有助于缩短视觉的处理时间。

如果零件上有某种通用的参考图案(如印刷电路板上的标准符号等)，且这种参考图案被用作其它对象搜索窗口的的基本位置，则最适合使用 Frame 对象。

#### Frame对象布局

Frame 对象看起来像两个相交的 Line 对象。

如需调整 Frame 对象的位置，请单击 Frame 对象名称，然后将对象拖到新位置即可。但是，大多数情况下，Frame 对象的位置和方向基于其它视觉对象的位置来设置。



Frame对象布局

#### Frame对象属性

以下列表是对 Frame 对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Caption	为Frame对象指定标题。 默认: Empty String
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。 默认: 1
Description	设定用户指定的描述 默认: 空白
Enabled	指定是否执行对象。 默认: True
FailColor	选择颜色，某个对象呈现该颜色即被拒绝。 默认: Red

属性	说明
Graphics	设置显示的图表模式。 默认: 1 - All
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
Name	用于为Frame对象分配一个唯一的名称。 默认: Frame01
OriginAngleEnabled	启用单点框架, 使框架以原点对象的角度旋转。而不是像两点框架那样, 基于OriginPoint和YaxisPoint之间矢量而旋转。 默认: False
OriginPntObjResult	从OriginPoint属性指定的视觉对象中, 指定要使用的结果。 如果指定为“All”, 则指定的所有视觉对象都将适用于Frame对象。 默认: 1(使用第一个结果)
OriginPoint	指定将用作Frame对象原点的视觉对象。 默认: Screen
PassColor	选择当对象被检测为良品时, 搜索窗口显示的颜色。 默认: LightGreen
PassType	可定义检测结果良品/不良品的判断标准。 默认: SomeFound
ShowExtensions	指定是否显示框架的延长线。 默认: False
YAxisPoint	指定将用作Frame对象Y轴上的点的视觉对象。(指定Frame对象的方向。) 默认: Screen
YAxisPntObjResult	从YAxisPoint属性指定的视觉对象中, 指定要使用的结果。 默认: 1(使用第一个结果)



## Frame对象结果

*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了与 Frame 对象连用的各结果详细情况。但是，为方便起见，我们提供了一份 Frame 对象的结果列表，并对各结果进行了以下简要说明：

结果	说明
Angle	返回被检测出零件的旋转角度。
Found	返回是否检测到了对象。

## 两点框架

Two point Frame 对象需要一个原点(由 OriginPoint 属性定义)和一个 Y 轴方向(由 YAxisPoint 属性定义)。原点和 Y 轴方向的组合定义了一种本地坐标系，其他视觉对象会参照这个本地坐标系。

Frame 对象最大的优势就是，当 Frame 对象移动时，该框架内定义的所有视觉对象均随其移动。(即：根据 OriginPoint 定义的视觉对象的 XY 变化，以及通过移动 YaxisPoint 产生的旋转，会自动调整搜索窗口。)这样可以缩小搜索窗口的尺寸，从而缩短视觉的处理时间。

## 定义Frame对象

创建新的 Frame 对象时，需要两个视觉对象，即框架的原点和 Y 轴上点的参考点。原点和 Y 轴方向由 OriginPoint 和 YAxisPoint 属性定义。任何具有 XY 坐标位置结果的视觉对象，均可用于定义框架的原点和 Y 轴上的点。

也就谁说，所有 Blob 对象、Correlation 对象、Edge 对象、Polar 对象和 Point 对象均可用于定义 Frame 对象的 Origin 和 YAxisPoint 属性。

## 单点框架

单点框架是 Frame 对象的一种可选使用方法。如选择单点框架，OriginPoint 属性则可以选择的视觉对象，作为 XY 位置原点的参考来使用。

如果将 OriginAngleEnabled 属性设置为“False”，当原点定位时使用的视觉对象的 XY 位置发生变化，框架的位置也会随之改变。无需考虑旋转。当某个视觉对象(blob 和 correlation 等)检测出工件的 XY 位置后，要将框架中的其他对象对齐这个 X 和 Y 的位置时，使用单点框架则可以简单实现 XY 的移动。

在某些情况下，您可能还需要考虑框架内的旋转。假设使用 Blob 对象查找工件的件的 X、Y 和 U 位置(XY 坐标+旋转)。然后其他视觉对象用于查找该工件的特征。Blob 对象用于定义包括框架旋转在内的单点框架。而该框内的其它对象，则朝 X 和 Y 移动，并根据 Blob 对象返回的旋转而旋转。这样，只需要一个视觉对象就能同时进行 XY 方向移动和旋转工件。所以，单点框架无需设置 YaxisPoint 属性。

### 使用Frame对象


接下来的几节将指导您如何创建和使用 Frame 对象。

- 创建一个新的 Frame 对象
- 确定搜索窗口的位置和尺寸
- 配置 Frame 对象的相关属性
- 测试 Frame 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

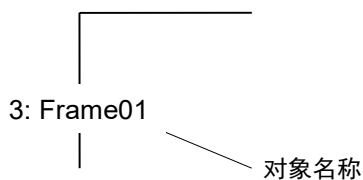
如果已经创建了视觉序列，请在 Vision Guide 窗口中单击序列目录树，选择已创建的视觉序列。

#### 第1步：创建一个新的Frame对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>- <Frame>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 Frame 图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“Frame01”，因为它是在本序列中创建的第一个Blob对象。(稍后将说明如何修改名称。)

#### 第2步：配置搜索窗口的位置和尺寸

屏幕上将显示一个类似下图的 Frame 对象。



#### 新Frame对象

Frame 对象没有大窗口。点击 Frame 对象的名称标签或轴，按住鼠标不放，将整个 Frame 对象拖动至画面上的新位置。确定位置后松开鼠标。Frame 对象将停留在画面的新位置上。

### 第3步：配置Frame对象的属性

现在可以设置 Frame 对象的属性值。设置各项属性，请点击相关属性的值字段并输入新值，或者点击所示的下拉列表中的相应选项。

以下显示的是 Frame 对象更常用的一些属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual* 或本章前面所描述的 *Frame 对象属性*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等属性。

#### Name 属性

新创建的Frame对象中，默认的项目名称是“Frame\*\*”。在一个视觉序列中要使用多个Frame对象时，使用\*\*的编号对Frame进行区分。

默认情况下，会将视觉序列序列中使用的第一个Frame对象命名为“Frame01”。

需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Frame 对象名称的地方都会变为新的名称。

#### OriginPoint 属性

通常会将本属性设置为序列中上一步完成的一个对象。该属性决定了运行 Frame 时的原点。

#### YAxisPoint 属性

通常会将本属性设置为序列中上一步完成的一个对象。该属性决定了运行 Frame 时的 Y 轴方向。

### 第4步：测试Frame对象并检验结果

按照以下步骤，运行 Frame 对象。

点击执行面板上对象的<Run>按钮。如果 OriginPoint 属性或 YAxisPoint 属性其中一个不是“Screen”，则会先执行属性中设置的对象。例如，如果将 OriginPoint 属性设置为 Blob 对象，则会先执行 Blob 对象，以确定框架的原点位置。

然后将显示 Frame 对象的结果。此时需检查的主要结果如下。

#### Angle

框架的角度

## 6.2.16 Line对象

### Line对象描述

Line 对象用于定义两点之间的直线。

这两个点可以是画面上的位置，也可以基于其它视觉对象的位置。以下是一些可以创建 Line 对象的情况。

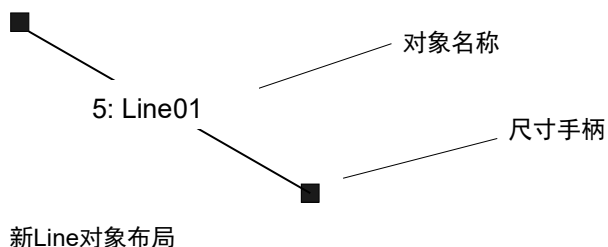
- 两个 Blob 对象之间
- 两个 Correlation 对象之间
- 两个 Point 对象之间
- Blob 对象和 Correlation 对象之间
- 一个 Blob 对象的两个结果之间
- Blob 对象的结果 1 和 Correlation 对象的结果 3 之间
- Point 对象和 Correlation 对象之间
- 具有 XY 位置结果的任意两个视觉对象之间

Line 对象在以下情况时非常有用：

- 测量视觉对象之间的距离(或使用多个结果时的视觉对象结果)。  
也可以用于检查您实际应用时的距离，以确保其介于最大距离和最小距离之间。
- 计算两个视觉对象之间的旋转角度  
(使用机器人坐标系中返回的直线的角度，被称为 RobotU 结果)
- 创建用于计算一条线的中点或两条线的交叉点的构造块。

### Line对象布局

Line 对象的布局如其名，是一条直线。是一条有起点、终点和对象名称的直线。如需修改 Line 对象的位置，只需点击该 Line 对象的名称(或该线上的任一位置)并将其拖至新位置即可。要重新调整 Line 对象的尺寸，则点击该线的起点或终点(尺寸手柄显示的位置)，然后将其拖至新位置即可。



**Line对象属性**

以下列表是对 Line 对象属性的总结。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
AngleBase	设置基准角度。 默认: 0
AngleMode	设置角度输出格式。 默认: 1 - Default
AngleStart	规定了角度搜索的中心。
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
Caption	为 Line 对象指定标题。 默认: Empty String
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认: 空白
Directed	指定是否使用线方向设置角度。 默认: True
Enabled	指定是否执行对象。 默认: True
EndPointObjResult	指定使用 EndPointObject 的哪个结果。 默认: 1
EndPointObject	指定使用哪个视觉对象来定义直线的终点。 默认: Screen
EndPointType	指定用于定义直线终点的终点类型。 默认: 0 - Point
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认: Red
Frame	指定要使用哪个位置的框架。 默认: none
Graphics	设置显示的图表模式。 默认: 1 - All
MaxLength	定义 Line 对象长度的上限。 Line 对象的 Length 结果，必须小于 MaxLength 属性的设置值，才能被找到。 默认: 9999

属性	说明
MaxPixelLength	定义 Line 对象像素长度的上限。 Line 对象的 PixelLength 结果必须小于 MaxPixelLength 属性的设置值，才能被找到。 默认：9999
MinLength	定义 Line 对象长度的下限。 Line 对象的 Length 结果，必须大于 MinLength 属性的设置值，才能被找到。 默认：0
MinPixelLength	定义 Line 对象像素长度的下限。 Line 对象的 PixelLength 结果必须大于 MinPixelLength 属性的设置值，才能被找到。 默认：0
Name	用于为 Line 对象分配一个唯一的名称。 默认：Line01
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：LightGreen
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound
ShowExtensions	设置为“True”时，图像显示区域边缘的延长线，会以图表的方式显示。 默认：False
StartPntObjResult	指定使用 StartPointObject 的哪个结果。 默认：1
StartPointObject	指定使用哪个视觉对象来定义直线的起点。 默认：Screen
StartPointType	指定用于定义直线起点的起点类型。 默认：“0 – Point”
X1	直线起点的 X 坐标位置。
X2	直线终点的 X 坐标位置。
Y1	直线起点的 Y 坐标位置。
Y2	直线终点的 Y 坐标位置。

**Line对象结果**

以下列表是对 Line 对象结果的总结，并附有简要指定。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	以时钟 3 点钟方向作为 0° 的基准，返回起点和终点所形成的直线的角度。
CameraX1	返回相机坐标系中直线起点的 X 坐标位置。
CameraX2	返回相机坐标系中直线终点的 X 坐标位置。
CameraY1	返回相机坐标系中直线起点的 Y 坐标位置。
CameraY2	返回相机坐标系中直线终点的 Y 坐标位置。
Found	返回是否检测到了对象。  (如果找不到用于定义 Line 对象的视觉对象，就找不到该 Line 对象。)
Length	返回直线的长度，以毫米为单位。  必须重新校准相机，否则返回的长度为“no cal”。  如果 Line 对象因 MaxLength 和 MinLength 的限制而失败，结果列表中的 Length 结果则显示为红色。
NumberFound	返回查找到的直线数目。
PixelLength	返回直线的长度，用像素表示。如果 Line 对象因 MaxPixelLength 和 MinPixelLength 的限制而失败，结果列表中的 PixelLength 结果则显示为红色字体。
PixelLine	仅限运行时间。  返回 X1、Y1、X2 和 Y2 四个直线坐标，并用像素表示。
PixelX1	返回线起点的 X 坐标位置，用像素表示。
PixelX2	返回线终点的 X 坐标位置，用像素表示。
PixelY1	返回线起点的 Y 坐标位置，用像素表示。
PixelY2	返回线终点的 Y 坐标位置，用像素表示。
RobotX1	返回机器人坐标系中，检测到的边缘直线的起点 X 坐标。
RobotX2	返回机器人坐标系中，检测到的边缘直线的终点 X 坐标。
RobotY1	返回机器人坐标系中，检测到的边缘直线的起点 Y 坐标。
RobotY2	返回机器人坐标系中，检测到的边缘直线的终点 Y 坐标。
RobotU	返回机器人坐标系中，从起点到终点所形成直线的角度。

### 使用Line对象

接下来的几节将指导您如何创建和使用 Line 对象。


- 创建一个新的 Line 对象
- 确定搜索窗口的位置和尺寸
- 配置 Line 对象的相关属性
- 测试 Line 对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. 视觉序列。

#### 第1步：创建一个新的Line对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>- <Line>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为 Line 对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。然后再点击鼠标左键，创建对象。
- (4) 请注意，对象名称将自动创建。本示例中创建的对象名为“Line01”，因为它是在本序列中创建的第一个 Line 对象。(稍后将说明如何修改名称。)

#### 第2步：配置搜索窗口的位置和尺寸

屏幕上将显示一个与下图相似的 Line 对象。



#### 新Line对象

Line 对象没有窗口。点击任一尺寸手柄，并将直线的一端拖动至新位置，则可以修改长度或旋转直线。您也可以点击 Line 对象的名称标签或该直线的任意位置，然后保持按下鼠标的同时拖动 Line 对象修改位置。确定位置后，放开鼠标即可。Line 对象将停留在画面上的新位置。



### 第3步：配置Line对象的属性

现在设置 Line 对象的属性值。要设置任何属性，仅需点击相关属性值字段，然后再重新输入一个新的值，或者如果显示有下拉列表，则点击该列表中的一项即可。

以下显示的是一些 Line 对象更常用的属性。参见 *Vision Guide 7.0 Properties and Results Reference Manual* 或本章前面所介绍的 *Line 对象属性*，了解许多不同视觉对象所用其它属性的说明，如 AbortSeqOnFail 和 Graphics 等属性。

#### Name 属性

新创建的Line对象中，默认的项目名称是“Line\*\*”。在一个视觉序列中要使用多个Line对象时，使用\*\*的编号对Line进行区分。

默认情况下，会将视觉序列序列中使用的第一个Line对象命名为“Line01”。

需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Line 对象名称的地方都会变为新的名称。

#### StartPointObject 属性

通常会将本属性设置为序列中上一步完成的一个对象。该属性决定了运行 Line 对象时的起点。

#### EndPointObject 属性

通常会将本属性设置为序列中上一步完成的一个对象。该属性决定了运行 Line 对象时的终点。

将 AngleMode 属性设置为 “2-UseAngleBase”，然后设置 Directed 和 AngleBase 属性，指定角度的输出格式。Directed 属性指定角度是否取决于 Line 对象的方向。以 AngleBase 中设置的角度作为基准输出角度。

例如，基于示教时设置的角度执行测量时，输出以 AngleBase 属性设置的角度为中心测量的角度。

将 AngleMode 属性设置为 “1-Default” 时，角度的输出范围是 0 至 360 度。(在这种情况下，操作将与进行以下设置时相同。)

AngleMode 属性: 2-UseAngleBase

AngleBase: 0

Directed: True)

### 第4步：测试Line对象并检验结果

请按照以下步骤，运行 Line 对象。

点击执行面板上对象的<Run>按钮。如果 StartPointObject 属性和 EndPointObject 属性中有一个不是 “Screen”，将首先执行指定的对象。例如，如果 StartPointObject 是 Blob 对象，则首先执行二进制对象，以确定直线的起点位置。

显示 Line 对象的结果。这时需检查的主要结果有：

#### Length 结果

返回线的长度，以毫米为单位。

必须有序列的相关校准，其中包括定义了 Length 的直线。

#### PixelX1 结果

直线两端的 XY 位置(用像素表示)。

#### PixelY1 结果

#### PixelX2 结果

#### PixelY2 结果

#### PixelLength 结果

直线的长度，用像素表示。

### 6.2.17 Point对象

#### Point对象描述

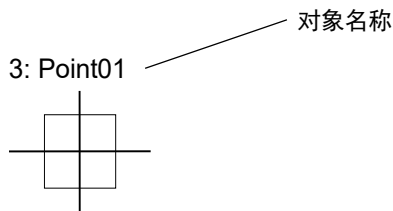
Point 对象用于定义点。Point 对象通常会与其他视觉对象一起使用，例如定义 Line 对象的中点等，一般不会单独使用。

如下表所示，Point 对象经常用于定义 Polar 对象和 Line 对象的参考位置。

<b>Polar 对象</b>	可使用 Point 对象定义 Polar 对象的中心(CenterX 和 CenterY 属性)。
<b>Line 对象</b>	可使用 Point 对象定义直线的起点，中点，终点和两条直线的交点。

#### Point对象布局

Point 对象的布局是一个十字线。如需修改位置，点击 Point 对象然后将其拖拽到要变更的位置即可。但由于大多情况下，Point 对象都是与其他对象相关联的，所以一般都根据相关对象的位置进行定位。



Point 对象布局

#### Point对象属性

以下列表是对 Point 对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
AngleObject	指定带有输出角度的对象。 默认: Screen
AngleObjectResult	指定AngleObject属性使用的结果
CalRobotPlacePos	在设计和程序运行时校准RobotPlacePos。
Caption	为 Point 对象指定标题。 默认: Empty String
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了All，则Point对象将应用于指定视觉对象的所有结果(NumberFound)。
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。

属性	说明
CenterPntOffsetY	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回Y偏移。
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在PixelX, PixelY上。
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认: None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认: 空白
Enabled	指定是否执行对象。 默认: True
FailColor	选择不合格对象的颜色。 默认: 1 - Red
Frame	指定要使用哪个位置的定位框。 默认: none
FrameResult	指定需使用Frame的第几个结果。
Graphics	设置显示的图表模式。 默认: 1 - All
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
LineObj1Result	指定LineObject1属性将使用对象的哪个结果。
LineObj2Result	指定LineObject2属性将使用对象的哪个结果。
LineObject1	当在 Point 对象中定义了直线的中点时，用本属性指定 Line 对象。 默认: none
LineObject2	当在 Poin 对象中定义了两条直线的交点时，用本属性指定第二个 Line 对象。 默认: none
Name	用于为 Point 对象分配一个唯一的名称。 默认: Point01
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认: LightGreen

属性	说明
PassType	可定义检测结果良品/不良品的判断标准。 默认: SomeFound
PointType	在 LineObject1 和 LineObject2 上指定直线的中点和交点 默认: Screen
X	Point 对象的 X 坐标位置, 用像素表示。
Y	Point 对象的 Y 坐标位置, 用像素表示。

### Point对象结果

以下列表是对Point对象结果的总结, 并附有简要说明。 *Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了各结果的详细情况。

结果	说明
Angle	返回被检测出的点的角度。
CameraX	返回相机坐标系中, 检测到对象的 X 坐标。
CameraY	返回相机坐标系中, 检测到对象的 Y 坐标。
CameraXYU	仅限运行时间。 返回在相机坐标系中已检测到对象位置的 CameraX、CameraY 和 CameraU 坐标。
ColorValue	返回位于当前位置的像素的灰度值或颜色值。
NumberFound	返回查找到的点数目。
Found	返回是否查找到了对象。 如果找不到附带的视觉对象, 则也找不到 Point 对象。
Passed	返回测试结果合格与否。
PixelX	返回像素坐标系中, 检测到对象的 X 坐标。
PixelY	返回像素坐标系中, 检测到对象的 Y 坐标。
PixelXYU	仅限运行时间。 返回已找到的零件位置的 PixelX、PixelY 和 PixelU 坐标(以像素为单位)。
RobotX	返回机器人坐标系中, 检测到对象的 X 坐标。
RobotY	返回机器人坐标系中, 检测到对象的 Y 坐标。
RobotU	返回机器人坐标系中检测到的对象的旋转量。 (Point对象不会旋转, 所以旋转量始终为“0”。)
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的对象的 X 坐标、Y 坐标和旋转量。(由于 Point 对象不会旋转, 旋转量始终为“0”)

### 了解Point对象

可以把 Point 对象理解为一个帮助用户标记位置的对象，可以是屏幕上的位置和 Line 对象，或者是其他视觉对象的关联位置。

为了更好的理解 Point 对象，这里说明两个基本知识点。

1. 定义 Point 对象的位置
2. 使用 Point 对象用作其它对象的参考位置

### 定义Point对象的位置

Point 对象的位置可以基于 Point 对象在操作画面上的位置、Line 对象的中点或两个 Line 对象之间的交点来确定。

PointType 属性可用于定义 Point 对象的位置。也就是说可以将 PointType 属性设置为“Screen”、“MidPoint”或“Intersection”中的一个。

### 将PointType属性设置为“Screen”时

当新建一个 Point 对象时，PointType 的默认设置为“Screen”，即 Point 对象的位置就是 Point 对象在屏幕中显示的那个位置。

将 PointType 设置为“Screen”后，只有在您用鼠标拖动 Point 对象或改变 X 属性或 Y 属性值时，其位置才会改变。

### 将PointType属性设置为“MidPoint”时

如要将 Point 对象的位置定义为直线的中点，请将 PointType 属性设置为“MidPoint”。

但应当注意的是，将 PointType 设置成“MidPoint”之前，请先确定要使用哪条直线。否则 Vision Guide 7.0 在计算中点时将不知道该使用哪条线。

请先在 LineObject1 属性中定义要使用的 Line 对象，然后将该直线的中点作为 Point 对象的位置。

在视觉序列运行列表中，运行 Point 对象之前，已执行的有效的对象，会显示在 LineObject1 属性的下拉列表中，请选择您要使用的对象。

实际上，Vision Guide 会自动检测序列运行列表中 Point 对象之前有效的 Line 对象，并且只有这些 Line 对象才会出现在 LineObject1 的下拉列表中。这就简化了系统操作。

如果您将 PointType 属性设置为“MidPoint”，而未在 LineObject1 属性中设置要使用的 Line 对象，则会弹出一条错误信息，提示您必须先在 LineObject1 属性中选择一条直线。

### 将PointType属性设置为“Intersection”时

如果要将 Point 对象的位置定义为两条直线的交点，则需首先确定使用哪两条直线。使用 LineObject1 和 LineObject2 属性定义这两条直线。

必须将 LineObject1 和 LineObject2 属性定义成不同的 Line 对象。定义了两个属性的 Line 对象后，即可将 PointType 属性设置为“Intersection”。这样就可以将 Point 对象的位置指定为 LineObject1 和 LineObject2 属性中定义的两条直线的交点。

首先，从 LineObject1 属性的下拉列表中，会显示 Point 对象前序，已执行的有效的对象，选择您要使用的对象。该对象则是 2 条直线交点中使用的第一条直线。

其次，从 LineObject2 属性的下拉列表中，会显示 Point 对象前序，剩余的有效对象，选择您要使用的对象。该对象则是 2 条直线交点中使用的第二条直线。

Vision Guide7.0 会自动检测序列运行列表中 Point 对象之前有效的 Line 对象，并且只有这些 Line 对象才会出现在 LineObject1 和 LineObject2 的下拉列表中。

如果您将 PointType 属性设置为“Intersection”，而未在 LineObject1 属性和 LineObject2 中设置要使用的 Line 对象，则会弹出一条错误信息，提示您必须先在 LineObject1 属性和 LineObject2 属性中选择要使用的直线。

弹出的错误提示会因未定义的 Line 对象不同而异。



将位置定义为两条直线的交点，会便于计算对象的中心。例如，假设有一个长方形的物体。当检测出四个顶点时，则可以创建在矩形中心相交的对角线。而 Point 对象就位于该交叉点上，即长方形的中心。

### 将Point对象用作其它对象的参考位置

Point 对象的主要用途就是作为其它视觉对象(Line 对象, Polar 对象等)的基准位置。

换句话说，既是 Point 对象的位置是 Line 对象和 Polar 对象的参考位置。

借助如此强大的功能，定义为两条直线交点的 Point 对象，将计算对象的中心，并将其用作 Line 的中点，以计算两个对象的中心之间的距离。

### 将Point对象用作Polar对象的基准位置

Polar 对象在进行极坐标搜索时，需要一个 CenterPoint 位置作为基准。首次使用 Polar 对象时，您可能会将屏幕上的点作为 CenterPoint 属性的基准点。但是您很快就会发现，由于被检测对象在移动，Polar 对象也会随着被检测对象的 CenterPoint 一起移动。

这种情况下，将 Point 对象等其他视觉对象的 XY 位置，应用到 Polar 对象中，则可以充分发挥其强大的功能。

很多情况下您可能想将 Point 对象的位置用作 Polar 对象的 CenterPoint。但最适合的两种情况如下。

1. 将 CenterPoint 定义为直线的中点
2. 将 CenterPoint 定义为两条直线的交点

比如，找到线的中点后，想从该中点开始 Polar 搜索，或希望将 Polar 搜索的 CenterPoint 的基准点设置在两条直线的交点上。

将 Polar 对象与 Point 对象一起使用是非常常见的一种用法。

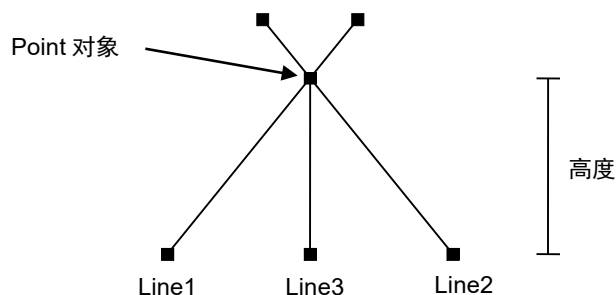
### 将Point对象用作Line对象的基准位置

一条直线需要一个起点和一个终点。多数情况下，线的起点和终点位置是基于其它视觉对象(如 Blob 或 Correlation 对象等)的 XY 位置结果。

但是，您也可将 Point 对象的位置用作直线的基准位置。您可以用 Point 对象定义直线的起点和终点，也可以仅定义起点或终点之一。

Point 对象的最常见用途之一，就是将 Point 对象作为两条直线交点的基准。

下图所示为两个在不同高度位置相交的 Line 对象的例子。Line3 用于计算交点的高度。Point 对象则位于 Line1 和 Line2 的交点上。




Point对象定义为Line1和2交点

### 使用Point对象

接下来的几页将向您介绍如何创建和使用 Point 对象。主要有以下几项。


- 创建一个新的 Point 对象
- 定位 Point 对象
- 配置 Point 对象的属性
- 测试 Point 对象并检查结果

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

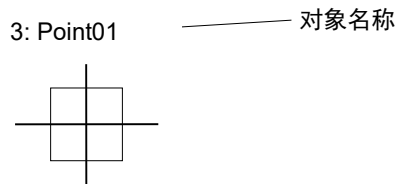
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：新建一个Point对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>-  <New Point>按钮。
- (2) 您将看到一个 Point 图标出现在 Point 对象工具栏按钮的上方。
- (3) 点击该点图标，并将其拖至 Vision Guide 窗口的图像显示区域。
- (4) 注意：对象名称将自动创建。本示例中创建的对象名为“Pnt01”，因为它是在本序列中创建的第一个 Point 对象。(稍后将说明如何修改名称。)

### 第2步：定位Point对象

画面上会显示一个如下所示的 Point 对象：



新建Point对象布局

因为 Point 对象只是一个点，既没有高度也没有厚度，因此不能调整大小。但是，可以使用鼠标在 X 和 Y 属性中设置位置值对其进行定位。

创建 Point 对象时，PointType 属性的默认设置为“0 - Screen”，因此，可以通过执行以下操作将 Point 对象稍微移动一下。

- (1) 点击Point对象的标签名称，按住鼠标左键并将Point对象拖至画面中的新位置。确认位置后释放鼠标，Point对象将停留在画面上的新位置上。

### 第3步：设置Point对象属性

设置 Point 对象的属性值。

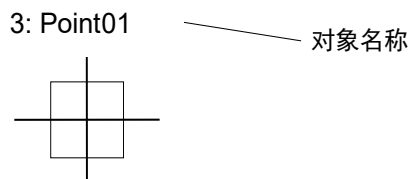
要设置属性，仅需点击相关属性值字段，然后再重新输入一个新的值。或者如果显示有下拉列表，则点击该列表中的相应选项即可。

以下显示的是一些 Point 对象的常用属性。其他各种可以在视觉对象中使用的属性，如 AbortSeqOnFail 属性、Graphics 属性等的详细信息，请参考 *Vision Guide 7.0 Properties and Results Reference* 手册或本章节的 *Point 对象属性* 相关内容。



### Point 对象布局

Point 对象在画面上显示为十字线。Point 对象只能调整位置。点击 Point 对象并将其拖至新位置即可。由于 Point 对象在绝大多数情况下都附属于其它对象，因此它们的位置是根据相关对象的位置来计算的。



Point对象布局

### Point 对象属性

测试 Point 对象时，不需要设置这些属性。因为测试阶段还没有将 Point 对象定位在直线的中点或两条直线的交点上，因此使用默认值即可。但如果您是第一次使用 Point 对象，请参考以下说明。

#### Name 属性 (“Pointxx”)

新创建的Point对象中，默认的项目名称是“Point\*\*”。在一个视觉序列中要使用多个Point对象时，使用\*\*的编号对Point进行区分。

默认情况下，会将在视觉序列序列中使用的第一个Point对象命名为“Point01”。

需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Point 对象名称的地方都会变为新的名称。

#### LineObject1 (None)

通过将 Point 对象的 PointType 属性设置为“Midpoint”(直线的中点)，则可指定要使用哪条直线。

如果 PointType 属性设置为“Intersection”(两条直线的交点)，则本属性也用于指定两条直线中的第一条直线。

默认：“None”

#### LineObject2 (None)

将 Point 对象的 PointType 属性设置为“Intersection”(两条直线的交点)时，本属性用于指定两条直线中的第二条直线(第一条直线由 LineObject1 属性指定。)

默认：“None”

#### PointType (Screen)

定义 Point 对象的位置。

可以定义为屏幕上的位置、LineObject1 属性指定的线中点或 LineObject1 和 LineObject2 属性指定的两条线的交叉点。

默认：“Screen”

在本示例中，由于未指定 Line 对象，因此 LineObject1、LineObject2 和 PointType 属性无法修改默认值。

如果要将直线的中点指定为 Point 对象的位置，请在 LineObject1 属性中指定一个 Line 对象。

如果要将两条直线的交点指定为 Point 对象的位置，请在 LineObject1 属性中指定第一个 Line 对象，并在 LineObject2 属性中指定第二个 Line 对象。

有关详细内容，参见上文 *定位 Point 对象* 中的内容。

使用 AngleObject 属性指定的对象角度，可以设为 Point 对象的输出角度。这不仅是要输出 XY 位置，还可以输出 Point 对象上一级对象中，作为 Point 对象的输出角度，可以同时获得 RobotXYU 等 XY 位置和 U 角度信息。

### 第4步：测试 Point 对象并检查结果

要运行 Point 对象，点击执行面板上对象的 <Run> 按钮。

(1) 点击执行面板上对象的 <Run> 按钮。

然后将显示 Point 对象的结果。这时需检查的主要结果有：

<b>PixelX 结果</b>	Point 对象的 XY 坐标位置，用像素表示。
<b>PixelY 结果</b>	如果 Point 对象的 PointType 属性设置为中点，则这些结果将返回 LineObject1 所指定 Line 对象中点的 XY 坐标位置，并用像素表示。
<b>CameraX 结果</b>	在相机坐标系中定义 Point 对象的 XY 坐标值。
<b>CameraY 结果</b>	如果相机尚未完成校准则不会返回任何值，而是返回 [No Cal]。
<b>RobotX 结果</b>	在机器人坐标系中定义 Point 对象的 XY 坐标值。
<b>RobotY 结果</b>	机器人可以将此 XY 坐标点作为移动的目标位置。不需要进行其它转换或处理。如果相机已经被校准，则 RobotX 和 RobotY 的结果将仅返回一个值。如果未进行校准，则将返回 [No Cal]。

## 6.2.18 BoxFinder对象

### BoxFinder对象描述

BoxFinder对象用于确定图像中矩形边缘(含正方形)的位置。

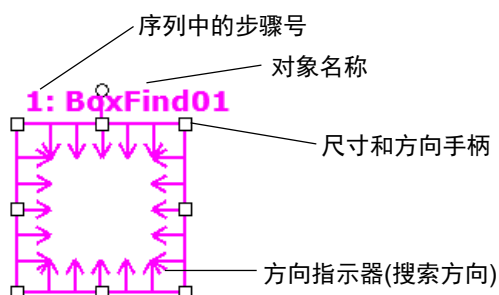
BoxFinder对象自动处理多个Edge对象以确定边缘位置，并获得从每个边缘位置确定的矩形。

图像中对象的边缘由灰度值“从深到浅”或“从浅到深”的变化表示。可以在多个像素上观察到这些变化。

由Polarity属性定义的“从深到浅”或“从浅到深”的变化后，BoxFinder对象会搜索这个变化，并将检测到的边缘连接成最佳直线。您也可以通过修改EdgeType属性，以搜索双边缘。如果搜索双边缘，则可检测到两个相对的边缘，返回的中点。

### BoxFinder对象布局

BoxFinder对象与LineFinder对象一样，在搜索窗口内中有方向指示器，表示边缘搜索的方向。与LineFinder的区别在于，搜索窗口四个侧边的每个方向都有指示器。边缘搜索线的数目通过NumberOfEdges属性指定。您可以使用Direction属性规定搜索方向。



### BoxFinder对象布局

BoxFinder对象可以放置为任何方向进行搜索(不仅仅是垂直和水平方向)。与Blob对象的SearchWinType=AngledRectangle类似，使用旋转BoxFinder对象搜索窗口的旋转手柄，可以将BoxFinder对象移动到所需要查找边缘的方向。

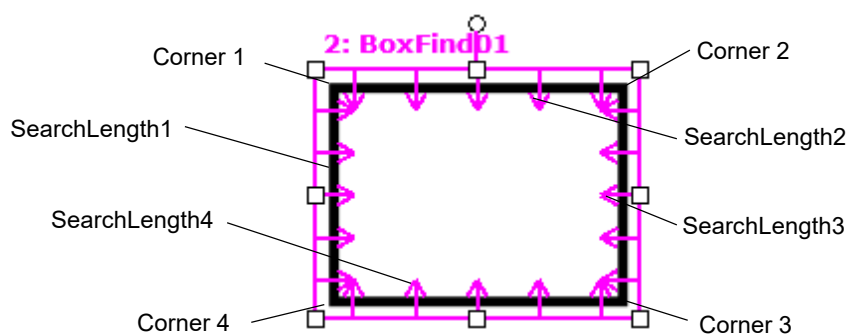
**BoxFinder对象属性**

以下列表是对BoxFinder对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认：False
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。 默认：100
Caption	为 BoxFinder 对象指定标题。 默认：Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在 PixelX, PixelY 上。 默认：Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了 All，则 BoxFinder 对象将应用于指定视觉对象的所有结果(NumberFound)。 默认：1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。 默认：0
CenterPntOffsetY	使用 CenterPointObject 属性定位了搜索窗口的中心点后，设置或返回 Y 偏移。 默认：0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 默认：False
ContrastTarget	设定边缘搜索所需的对比度。 默认：0(best contrast)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认：0
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理，若没有通过Decision等分支功能执行，则不进行复制处理。 默认：None

属性	说明
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认：空白
Direction	设置边缘搜索的方向。 默认：InsideOut
EdgeSort	设置检测到的边缘结果的排序方式。 默认：Score
EdgeThreshold	设置阈值，低于该值的边缘将被忽略。 默认：2
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认：1 - Single
Enabled	指定是否执行对象。 默认：True
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red
FittingThreshold	指定用于直线拟合的边缘结果。 默认：10
Frame	指定要使用哪个位置的定位框。 默认：none
FrameResult	指定需使用 Frame 的第几个结果。 默认：1
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	设置对象标签的背景颜色。 默认：Transparent
MissingEdgeType	确定如何处理未检测出的边缘。 默认：Interpolate
Name	用于为 BoxFinder 对象分配一个唯一的名称。 默认：BoxFind01
NumberOfEdges	指定要查找边缘的数目。 默认：5
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：LightGreen
PassType	可定义检测结果良品/不良品的判断标准。 默认：SomeFound

属性	说明
Polarity	指定 BoxFinder 对象是应该搜索浅色到深色的过度 (LightToDark) 还是深色到浅色的过度 (DarkToLight)。 默认: 1-LightToDark
ScoreWeightContrast	根据对比度设置得分百分比。 默认: 50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认: 50
SearchLength	规定边缘搜索范围的长度。 以下 SearchLength1~4 可一起设置。
SearchLength1	设置下图中“BoxFinder 对象属性、结果的位置关系”的 SearchLength1。
SearchLength2	设置下图中“BoxFinder 对象属性、结果的位置关系”的 SearchLength2。
SearchLength3	设置下图中“BoxFinder 对象属性、结果的位置关系”的 SearchLength3。
SearchLength4	设置下图中“BoxFinder 对象属性、结果的位置关系”的 SearchLength4。
SearchWidth	定义边缘搜索的宽度。 范围: 3 - 99。 默认: 3
SearchWin	仅限运行时间。在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinAngle	定义待搜索区域的角度。
SearchWinCenterX	定义待搜索区域中心的 X 轴坐标值。
SearchWinCenterY	定义待搜索区域中心的 Y 轴坐标值。
SearchWinHeight	定义待搜索区域的高度(单位: pixel)。
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinWidth	定义待搜索区域的宽度(单位: pixel)。
StrengthTarget	设置搜索边缘的强度。 默认: 0
StrengthVariation	设置 StrengthTarget 的变化量总和。 默认: 0



BoxFinder 对象属性、结果的位置关系

### BoxFinder对象结果

以下列表是对BoxFinder对象结果的总结，并附有简要说明。欲了解每条结果的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference*。

结果	说明
Angle	返回检测到矩形的角度。
CameraX	返回相机坐标系中检测到的矩形边缘的中心 X 坐标。
CameraY	返回相机坐标系中检测到的矩形边缘的中心 Y 坐标。
CameraXYU	仅限运行时间。 返回相机坐标系中的已检测到矩形边缘中心位置的 CameraX、CameraY 和 CameraU 坐标。
CameraX1	返回上图“BoxFinder 对象属性、结果的位置关系”Corner1 在相机坐标系中的 X 坐标。
CameraY1	返回上图“BoxFinder 对象属性、结果的位置关系”Corner1 在相机坐标系中的 Y 坐标。
CameraX2	返回上图“BoxFinder 对象属性、结果的位置关系”Corner2 在相机坐标系中的 X 坐标。
CameraY2	返回上图“BoxFinder 对象属性、结果的位置关系”Corner2 在相机坐标系中的 Y 坐标。
CameraX3	返回上图“BoxFinder 对象属性、结果的位置关系”Corner3 在相机坐标系中的 X 坐标。
CameraY3	返回上图“BoxFinder 对象属性、结果的位置关系”Corner3 在相机坐标系中的 Y 坐标。
CameraX4	返回上图“BoxFinder 对象属性、结果的位置关系”Corner4 在相机坐标系中的 X 坐标。
CameraY4	返回上图“BoxFinder 对象属性、结果的位置关系”Corner4 在相机坐标系中的 Y 坐标。
Contrast	返回检测到的矩形边缘的对比度。
FitError	以均方根(RMS)返回各边缘点与检测到矩形的距离。

结果	说明
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了 Accept 属性的当前设置值。
MaxError	返回与检测到的矩形边缘的像素长度的最大偏差。
Passed	返回视觉对象的测试结果合格与否。
Perimeter	返回检测到的矩形边界的像素数目。
PixelX	返回图像坐标系中检测到的矩形边缘的中心 X 坐标。
PixelY	返回图像坐标系中检测到的矩形边缘的中心 Y 坐标。
PixelX1	返回上图“BoxFinder 对象属性、结果的位置关系”Corner1 在图像坐标系中的 X 坐标。
PixelY1	返回上图“BoxFinder 对象属性、结果的位置关系”Corner1 在图像坐标系中的 Y 坐标。
PixelX2	返回上图“BoxFinder 对象属性、结果的位置关系”Corner2 在图像坐标系中的 X 坐标。
PixelY2	返回上图“BoxFinder 对象属性、结果的位置关系”Corner2 在图像坐标系中的 Y 坐标。
PixelX3	返回上图“BoxFinder 对象属性、结果的位置关系”Corner3 在图像坐标系中的 X 坐标。
PixelY3	返回上图“BoxFinder 对象属性、结果的位置关系”Corner3 在图像坐标系中的 Y 坐标。
PixelX4	返回上图“BoxFinder 对象属性、结果的位置关系”Corner4 在图像坐标系中的 X 坐标。
PixelY4	返回上图“BoxFinder 对象属性、结果的位置关系”Corner4 在图像坐标系中的 Y 坐标。
PixelXYU	返回已找到的矩形边缘中心的 PixelX、PixelY 和 PixelU 坐标 (以像素为单位)。
RobotX	返回机器人坐标系中检测到的矩形边缘的中心 X 坐标。
RobotY	返回机器人坐标系中检测到的矩形边缘的中心 Y 坐标。
RobotU	返回机器人坐标系中检测到的矩形边缘的中心 U 坐标。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的矩形边缘的中心位置的 RobotX、RobotY 和 RobotU 坐标。
RobotX1	返回上图“BoxFinder 对象属性、结果的位置关系”Corner1 在机器人坐标系中的 X 坐标。
RobotY1	返回上图“BoxFinder 对象属性、结果的位置关系”Corner1 在机器人坐标系中的 Y 坐标。




结果	说明
RobotX2	返回上图“BoxFinder 对象属性、结果的位置关系”Corner2 在机器人坐标系中的 X 坐标。
RobotY2	返回上图“BoxFinder 对象属性、结果的位置关系”Corner2 在机器人坐标系中的 Y 坐标。
RobotX3	返回上图“BoxFinder 对象属性、结果的位置关系”Corner3 在机器人坐标系中的 X 坐标。
RobotY3	返回上图“BoxFinder 对象属性、结果的位置关系”Corner3 在机器人坐标系中的 Y 坐标。
RobotX4	返回上图“BoxFinder 对象属性、结果的位置关系”Corner4 在机器人坐标系中的 X 坐标。
RobotY4	返回上图“BoxFinder 对象属性、结果的位置关系”Corner4 在机器人坐标系中的 Y 坐标。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。
Strength	返回查找到边缘的强度。
Time	返回处理对象所需要的时间(单位：毫秒)。

### 使用BoxFinder对象

接下来的几节将介绍如何创建和使用BoxFinder对象。

- 如何创建新的BoxFinder对象
- 确定搜索窗口的位置和尺寸
- 设置BoxFinder对象的相关属性
- 测试BoxFinder对象并检查结果
- 调整属性并再次测试


开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果

您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

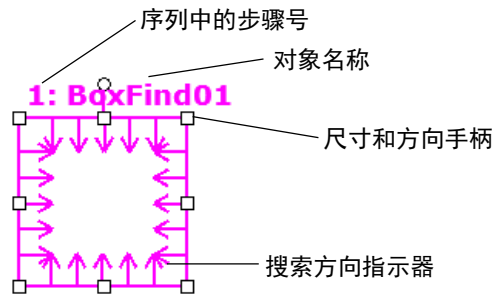
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：新建一个BoxFinder对象

- (1) 点击Vision Guide工具栏中的<All Tools>- <BoxFinder>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为BoxFinder图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。
- (4) 点击鼠标左键，创建对象。
- (5) 注意：对象名称将自动创建。  
本示例中创建的对象名为“BoxFind01”，因为这是在本序列中创建的第一个BoxFinder对象。(稍后将说明如何修改名称。)

### 第2步：配置搜索窗口的位置和尺寸

画面上会显示一个如下所示的BoxFinder对象：



#### 新建BoxFinder对象布局

- (1) 点击BoxFinder对象的名称标签，按住鼠标左键，将BoxFinder对象拖至您想让搜索窗口的所在的位置。
- (2) 点击搜索窗口的尺寸手柄，拖动鼠标，将BoxFinder对象搜索窗口调整成合适的大小。(搜索窗口就是要搜索的区域。)

### 第3步：设置BoxFinder对象的属性

设置BoxFinder对象的属性值。

要设置属性，仅需点击相关属性值字段，然后再重新输入一个新的值。或者如果显示有下拉列表，则点击该列表中的相应选项即可。

有关其他常用的视觉对象的详细信息，如AbortSeqOnFail和Graphics等，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual* 或 *Correlation* 对象属性列表。

<b>Name属性</b>	<p>新创建的 BoxFinder 对象中，默认的项目名称是“BoxFind**”。在一个视觉序列中要使用多个BoxFinder对象时，使用xx的编号对BoxFinder进行区分。</p> <p>默认情况下，会将在视觉序列序列中使用的第一个BoxFinder对象命名为“BoxFind01”。</p> <p>需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示BoxFinder对象名称的地方都会变为新的名称。</p>
<b>EdgeType(单边缘)</b>	<p>选择搜索边缘的类型。</p> <p>如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。</p>
<b>NumberOfEdges(5)</b>	在搜索窗口的每条边上搜索 5 个边缘，以查找边缘。
<b>Polarity (LightToDark)</b>	<p>使用“LightToDark”属性搜索边缘。</p> <p>如果您要寻找 DarkToLight(深色到浅色)的边缘，请更改Polarity 设置。</p>

### 第4步：测试BoxFinder对象并检查结果

接下来的几节将介绍如何运行BoxFinder对象。

点击执行面板上对象的<Run>按钮。

然后将显示BoxFinder对象的结果。这时需检查的主要结果有：

<b>Angle 结果</b>	返回图像坐标系中检测到的矩形边缘的角度。
<b>MaxError 结果</b>	返回与检测到的矩形边缘像素长度的最大差异。
<b>PixelX 结果</b>	返回图像坐标系中检测到的矩形边缘中心的 XY 坐标位置。
<b>PixelY 结果</b>	
<b>CameraX 结果</b>	在相机坐标系中返回检测到的矩形边缘中心的 XY 坐标值。
<b>CameraY 结果</b>	如果未进行 XY 坐标位置校准，则返回“no cal”。
<b>RobotX 结果</b>	在机器人坐标系中返回检测到的矩形边缘中心的 XY 坐标值。
<b>RobotY 结果</b>	如果未进行 XY 坐标位置校准，则返回“no cal”。

6.2.19  CornerFinder对象

CornerFinder对象描述

CornerFinder对象用于确定图像中两条直线的交点(角)的位置。

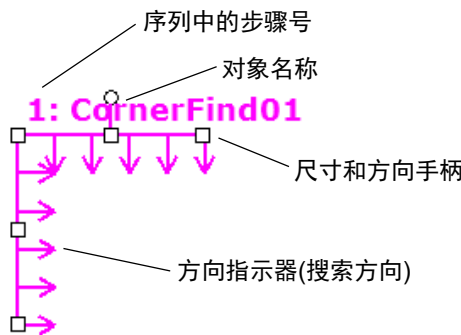
CornerFinder对象自动处理多个Edge对象以识别边缘位置，并获得从每个边缘位置识别出的两条直线的交角。

图像中对象的边缘由灰度值“从深到浅”或“从浅到深”的变化表示。可以在多个像素上观察到这些变化。

由Polarity属性定义的“从深到浅”或“从浅到深”的变化后，CornerFinder对象会查搜索这个变化，并将检测到的边缘连接成最佳直线。您也可以通过修改EdgeType属性，以搜索双边缘。如果搜索双边缘，则可检测到两个相对的边缘，返回的中点。

CornerFinder对象布局

CornerFinder对象与LineFinder对象一样，在搜索窗口内中有方向指示器。与LineFinder的区别在于，搜索窗口两个侧边的每个方向都有指示器。边缘搜索线的数目通过NumberOfEdges属性指定。您可以使用Direction属性规定搜索方向。



CornerFinder对象布局

CornerFinder对象可以放置为任何方向进行搜索(不仅仅是垂直和水平方向)。与Blob对象的SearchWinType=AngledRectangle类似，使用旋转CornerFinder对象搜索窗口的旋转手柄，可以将CornerFinder对象移动到所需要查找边缘的方向。

CornerFinder对象属性

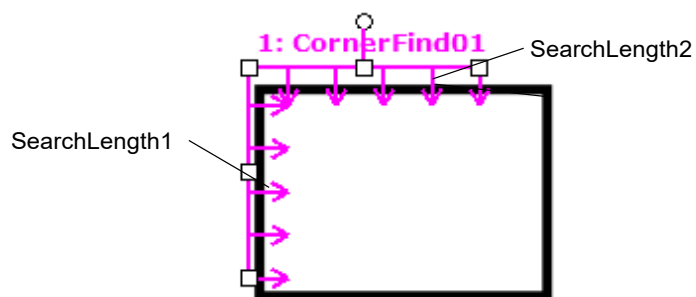
以下列表是对CornerFinder对象属性的总结。有关详情，请参考*Vision Guide 7.0 Properties and Results Reference Manual*。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认: False
Accept	指定形状得分，用于判断是否以检测到检测对象。 检测对象的得分需要大于或等于设定的形状得分。 默认: 100

属性	说明
Caption	为 CornerFinder 对象指定标题。 默认: Empty String
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时, 则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时, 该参数的中心点将会设定在 PixelX, PixelY 上。 默认: Screen
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了 All, 则 CornerFinder 对象将应用于指定视觉对象的所有结果(NumberFound)。 默认: 1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后, 设置或返回X偏移。 默认: 0
CenterPntOffsetY	使用 CenterPointObject 属性定位了搜索窗口的中心点后, 设置或返回 Y 偏移。 默认: 0
CenterPntRotOffset	指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。 默认: False
ContrastTarget	设定边缘搜索所需的对比度。 默认: 0(best contrast)
ContrastVariation	选择ContrastTarget允许的对比度变化量。 默认: 0
CoordObject	指定复制结果的Coordinates对象 执行对象时进行复制处理, 若没有通过Decision等分支功能执行, 则不进行复制处理。 默认: None
CurrentResult	可定义要在对象窗口的结果列表中显示的结果, 或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述 默认: 空白
Direction	设置边缘搜索的方向。 默认: InsideOut
EdgeSort	设置检测到的边缘结果的排序方式。 默认: Score
EdgeThreshold	设置阈值, 低于该值的边缘将被忽略。 默认: 2

属性	说明
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认: 1 - Single
Enabled	指定是否执行对象。 默认: True
FailColor	选择当对象被检测为不良品时, 搜索窗口显示的颜色。 默认: Red
FittingThreshold	指定用于直线拟合的边缘结果。 默认: 10
Frame	指定要使用哪个位置的定位框。 默认: none
FrameResult	指定需使用 Frame 的第几个结果。 默认: 1
Graphics	设置显示的图表模式。 默认: 1 - All
LabelBackColor	设置对象标签的背景颜色。 默认: Transparent
MissingEdgeType	确定如何处理未检测出的边缘。 默认: Interpolate
Name	用于为 CornerFinder 对象分配一个唯一的名称。 默认: CornerFind01
NumberOfEdges	指定要查找边缘的数目。 默认: 5
PassColor	选择合格对象的颜色。 默认: LightGreen
PassType	选择用以规定对象是否合格的规则。 默认: SomeFound
Polarity	指定 ArcFinder 对象是应该搜索浅色到深色的过度 (LightToDark)还是深色到浅色的过度(DarkToLight)。 默认: 1 - LightToDark
ScoreWeightContrast	根据对比度设置得分百分比。 默认: 50
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认: 50
SearchLength	规定边缘搜索范围的长度。 以下 SearchLength1 和 2 可一起设置。
SearchLength1	设置图“CornerFinder 对象属性的位置关系”中 SearchLength1 的长度。

属性	说明
SearchLength2	设置图“CornerFinder 对象属性的位置关系”中 SearchLength2 的长度。
SearchWidth	定义边缘搜索的宽度。 宽度范围为 3 - 99。 默认: 3
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinAngle	定义待搜索区域的角度。
SearchWinCenterX	定义待搜索区域中心的 X 轴坐标值。
SearchWinCenterY	定义待搜索区域中心的 Y 轴坐标值。
SearchWinHeight	定义待搜索区域的高度(单位: pixel)。
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。
SearchWinWidth	定义待搜索区域的宽度(单位: pixel)。
StrengthTarget	设置搜索边缘的强度。 默认: 0
StrengthVariation	设置 StrengthTarget 的变化量总和。 默认: 0



CornerFinder对象属性的位置关系

**CornerFinder对象结果**

以下列表是对CornerFinder对象结果的总结，并附有简要说明。欲了解每条结果的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference*。


结果	说明
Angle	返回检测到交角的角度。
CameraX	返回相机坐标系中检测到交角的 X 坐标。
CameraY	返回相机坐标系中检测到交角的 Y 坐标。
CameraXYU	仅限运行时间。 返回相机坐标系中的已检测到交角位置的 CameraX、CameraY和CameraU坐标。
Contrast	返回查找到边缘的对比度。
FitError	以均方根(RMS)返回各边缘点与检测到交角的距离。
Found	返回是否检测到了对象，即被检测对象或零件的形状得分是否超过了 Accept 属性的当前设置值。
MaxError	返回与检测到的线边缘的最大像素长度差异。
Passed	返回视觉对象的测试结果合格与否。
PixelX	返回图像坐标系中检测到交角的 X 坐标。
PixelY	返回图像坐标系中检测到交角的 Y 坐标。
PixelXYU	仅限运行时间。 返回已检测到交角位置的PixelX、PixelY和PixelU坐标(以像素为单位)。
RobotX	返回机器人坐标系中检测到交角的 X 坐标。
RobotY	返回机器人坐标系中检测到交角的 Y 坐标。
RobotU	返回机器人坐标系中检测到交角的 U 坐标。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到交角的RobotX、RobotY和RobotU坐标。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。
Strength	返回查找到边缘的强度。
Time	返回处理对象所需要的时间(单位：毫秒)。



## 使用CornerFinder对象

接下来的几节将介绍如何创建和使用CornerFinder对象。


- 新建一个CornerFinder对象
- 确定搜索窗口的位置和尺寸
- 设置CornerFinder对象的相关属性
- 测试CornerFinder对象并检查结果
- 调整属性并再次测试

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。如果您没有可使用的视觉序列，点击  <New Sequence>按钮来创建一个新的视觉序列。

如果已经创建了视觉序列，请在Vision Guide窗口中单击序列目录树，选择已创建的视觉序列。

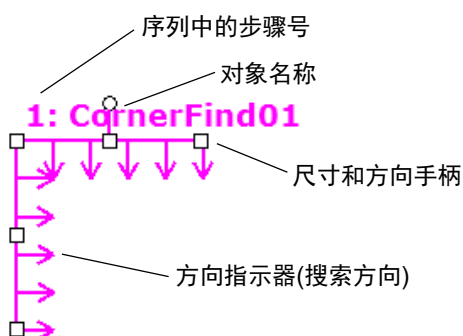
有关如何创建新的视觉序列，以及如何选择已创建视觉序列的详细咨询，请参阅 5. *视觉序列*。

### 第1步：新建一个CornerFinder对象

- (1) 点击Vision Guide工具栏中的<All Tools>- <CornerFinder>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为CornerFinder对象图标。
- (3) 继续移动鼠标，直至图标位于图像显示屏中的目标位置。
- (4) 点击鼠标左键，创建对象。
- (5) 请注意，对象名称将自动创建。本示例中创建的对象名为“CornerFind01”，因为它是在本序列中创建的第一个CornerFinder对象。(稍后将说明如何修改名称。)

### 第2步：配置搜索窗口的位置和尺寸

画面上会显示一个如下所示的CornerFinder对象：



#### 新建CornerFinder对象布局

- (1) 点击CornerFinder对象的名称标签，按住鼠标左键，将CornerFinder对象拖至您想让搜索窗口的所在的位置。
- (2) 点击搜索窗口的尺寸手柄，拖动鼠标，将CornerFinder对象搜索窗口调整成合适的大小。

**第3步：设置CornerFinder对象属性**

设置CornerFinder对象的属性值。

设置CornerFinder对象属性，仅需点击相关属性值字段，然后再重新输入一个新的值。或者如果显示有下拉列表，则点击该列表中的相应选项即可。

以下是CornerFinder对象的一些常用属性。有关其他常用的视觉对象的详细信息，如AbortSeqOnFail和Graphics等，请参阅 *Vision Guide 7.0 Properties and Results Reference Manual* 或 *Correlation* 对象属性列表

<b>Name属性</b>	<p>新创建的 CornerFinder 对象中，默认的项目名称是“CornerFind**”。在一个视觉序列中要使用多个 CornerFinder 对象时，使用**的编号对 CornerFinder 进行区分。</p> <p>默认情况下，会将在视觉序列序列中使用的第一个 CornerFinder 对象命名为“CornerFind01”。</p> <p>需要修改对象名称时，请选择Name属性工具栏，输入新的名称然后按回车键。这样所有显示CornerFinder对象名称的地方都会变为新的名称。</p>
<b>EdgeType(单边缘)</b>	<p>选择搜索边缘的类型。</p> <p>如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。</p>
<b>NumberOfEdges(5)</b>	<p>在搜索窗口的每条边上个搜索 5 个边缘，以查找边缘。</p>
<b>Polarity (LightToDark)</b>	<p>使用“LightToDark”属性搜索边缘。</p> <p>如果您要寻找 DarkToLight(深色到浅色)的边缘，请更改 Polarity 设置。</p>

**第4步：测试CornerFinder对象并检查结果**

要运行CornerFinde对象，请点击执行面板上对象的<Run>按钮。

然后将显示CornerFinder对象的结果。这时需检查的主要结果有：

<b>Angle 结果</b>	返回图像坐标系中检测到的角的角度。
<b>MaxError 结果</b>	返回与检测到的直线的最大偏差值(单位: pixel)。
<b>PixelX 结果</b>	返回图像坐标系中检测到的角的 XY 坐标位置。
<b>PixelY 结果</b>	
<b>CameraX 结果</b>	返回相机坐标系中检测到的角的 XY 坐标位置。
<b>CameraY 结果</b>	如果未进行 XY 坐标位置校准，则返回“no cal”。
<b>RobotX 结果</b>	返回机器人坐标系中检测到的角的 XY 坐标位置。
<b>RobotY 结果</b>	如果未进行 XY 坐标位置校准，则返回“no cal”。

## 6.2.20 Contour对象

### Contour对象描述

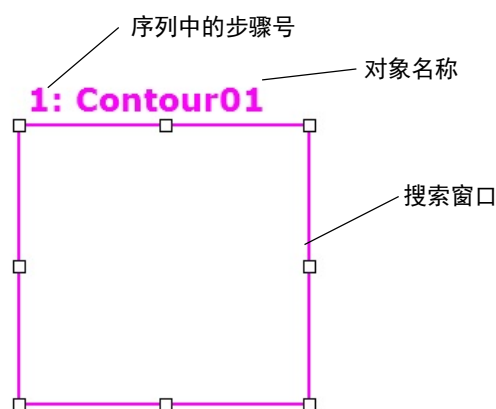
Contour对象沿着工件轮廓输出轨迹。当您希望机器人的夹具沿工件轮廓移动时，则可以轻松的从图像信息中获取路径。

Contour对象有三种获取轨迹的方法，可以根据应用需要选择使用。可以通过更改ContourMode属性修改获取方法，每种模式下的GUI和有效属性有所不同。

下面简要说明每种模式的显著特征。

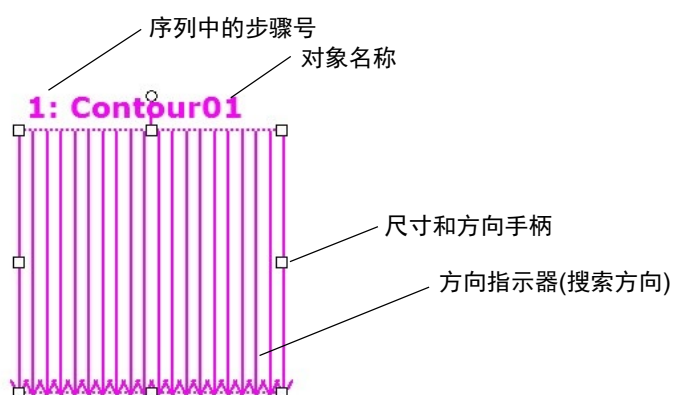
#### 1. Blob格式

在设置好的搜索窗口内，检测目标工件的Blob，并获取轮廓。该模式用于从复杂形状的工件获取轮廓。



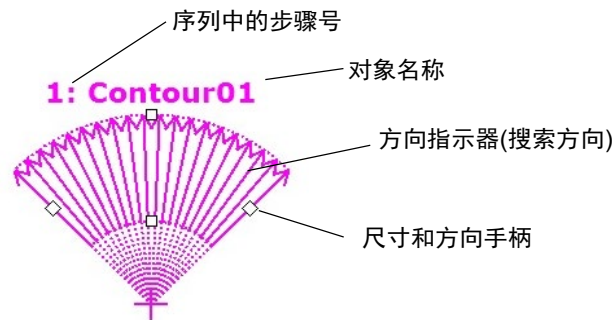
#### 2. Line格式

通过并排布置的多条边缘搜索线获取轮廓。该模式适用于从目标工件的一部分，获取起伏较小的轮廓。



## 3. Arc格式

通过放射状排列的多条边缘搜索线获取轮廓。该模式适用于从目标工件的一部分，获取起伏较小的圆弧轮廓。



## Contour对象属性

以下列表是对Contour对象结果的总结，并附有简要说明。有关详情，请参考*Vision Guide 7.0 Properties and Results Reference*。

此外，可用的属性因ContourMode值而不同。列表所示为应用各属性所需的ContourMode值。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认：False 支持格式：Blob、Line、Arc
Accept	指定形状得分，用于判断是否以检测到检测对象。检测对象的得分需要大于或等于设定的形状得分。 默认：100 支持格式：Line、Arc
AngleEnd	指定执行圆形/椭圆形搜索范围的终止角度。 默认：135 支持格式：Arc
AngleStart	指定执行圆形/椭圆形搜索范围的起始角度。 默认：45 支持格式：Arc
Caption	为 Contour 对象指定标题。 默认：Empty String 支持格式：Blob、Line、Arc
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在 PixelX, PixelY 上。 支持格式：Blob、Line、Arc

属性	说明
CenterPntObjResult	<p>指定要使用的CenterPointObject属性结果。</p> <p>如果指定了 All, 则 Contour 对象将应用于指定视觉对象的所有结果(NumberFound)。</p> <p>默认: 1</p> <p>支持格式: Blob、Line、Arc</p>
CenterPntOffsetX	<p>使用CenterPointObject属性定位了搜索窗口的中心点后, 设置或返回X偏移。</p> <p>支持格式: Blob、Line、Arc</p>
CenterPntOffsetY	<p>使用 CenterPointObject 属性定位了搜索窗口的中心点后, 设置或返回 Y 偏移。</p> <p>支持格式: Blob、Line、Arc</p>
CenterPntRotOffset	<p>指定是否旋转以 CenterPointObject 的 Angle 结果为基础的中心(CenterPntOffsetX、CenterPntOffsetY)的 XY 偏移。</p> <p>如果将 SearchWinType 设置为 RotatedRectangle, 搜索窗口则以 Angle 结果为基础进行旋转。</p>
CenterX	<p>指定用作对象中心点位置的 X 坐标。</p> <p>当 CenterPoint 属性已被其他视觉对象设定时, 将自动设置本属性。</p> <p>支持格式: Arc</p>
CenterY	<p>指定用作对象中心点位置的 Y 坐标。</p> <p>当 CenterPoint 属性已被其他视觉对象设定时, 将自动设置本属性。</p> <p>支持格式: Arc</p>
ContourMode	定义了Contour对象边缘的检测方法。
ContrastTarget	<p>设定边缘搜索所需的对比度。</p> <p>默认: 0(best contrast)</p> <p>支持格式: Line、Arc</p>
ContrastVariation	<p>选择 ContrastTarget 允许的对比度变化量。</p> <p>默认: 0</p> <p>支持格式: Line、Arc</p>
CoordObject	<p>指定复制结果的 Coordinates 对象</p> <p>执行对象时进行复制处理, 若没有通过 Decision 等分支功能执行, 则不进行复制处理。</p> <p>默认: None</p>
CurrentResult	<p>可定义要在对象窗口的结果列表中显示的结果, 或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。</p> <p>支持格式: Blob、Line、Arc</p>
Description	<p>设定用户指定的描述</p> <p>默认: 空白</p>
Direction	<p>设置边缘搜索的方向。</p> <p>默认: InsideOut</p> <p>支持格式: Arc</p>

属性	说明
EdgeThreshold	设置阈值，低于该值的边缘将被忽略。 默认：2 支持格式：Line、Arc
EdgeType	选择搜索的边缘类型(单边缘或双边缘)。 默认：1 - Single 支持格式：Line、Arc
Enabled	指定是否执行对象。 默认：True 支持格式：Blob、Line、Arc
EndPntObjResult	指定使用 EndPointObject 的哪个结果。 默认：1 支持格式：Blob、Line、Arc
EndPointObject	指定使用哪个视觉对象来定义直线的终点。 默认：Screen 支持格式：Blob、Line、Arc
EndPointType	指定用于定义直线终点的终点类型。 默认：0 - Point 支持格式：Blob、Line、Arc
FailColor	选择当对象被检测为不良品时，搜索窗口显示的颜色。 默认：Red 支持格式：Blob、Line、Arc
FillHoles	指定是否填充二进制图像的孔洞。 默认：False 支持格式：Blob
FittingThreshold	规定直线和圆弧的拟合阈值。 支持格式：Line、Arc
Frame	指定要使用哪个位置的定位框。 默认：none 支持格式：Blob、Line、Arc
FrameResult	指定需使用 Frame 的第几个结果。 默认：1 支持格式：Blob、Line、Arc
Graphics	设置显示的图表模式。 默认：1 - All 支持格式：Blob、Line、Arc
LabelBackColor	设置对象标签的背景颜色。 默认：Transparent 支持格式：Blob、Line、Arc
LineDirection	规定轮廓点输出方向。 默认：LeftToRight 支持格式：Line

属性	说明
MaxArea	确定对象的上限领域。 默认: 100,000 支持格式: Blob
MinArea	确定对象的下限领域。 默认: 25 支持格式: Blob
MinMaxArea	仅限运行时间。 用一条语句设置或返回MinArea和MaxArea。
Name	用于为 Contour 对象分配一个唯一的名称。 默认: Contour01 支持格式: Blob、Line、Arc
NumberOfEdges	指定要查找边缘的数目。 默认: 20 支持格式: Line、Arc
NumberToFind	规定输出的 Contour 点最大数目。 默认: 1 支持格式: Blob、Line、Arc
PassColor	选择当对象被检测为良品时, 搜索窗口显示的颜色。 默认: LightGreen 支持格式: Blob、Line、Arc
PassType	可定义检测结果良品/不良品的判断标准。 默认: SomeFound 支持格式: Blob、Line、Arc
Polarity	指定 Contour 对象是应该搜索浅色背景上的深色对象 (DarkOnLight)、深色背景上的浅色对象(LightOnDark), 还是浅色到深色的过度(LightToDark)、深色到浅色的过度 (DarkToLight)。 默认_Blob: 1 – DarkOnLight 默认_Line/Arc: 1 – LightToDark 支持格式: Blob、Line、Arc
RadiusInner	指定被检测出范围的内径。 支持格式: Arc
RadiusOuter	指定被检测出范围的外径。 支持格式: Arc
RejectOnEdge	如果将属性设置为“True”, 系统将忽略与搜索窗口边缘有接触的 Blob 对象。 默认: False 支持格式: Blob
RotationDirection	指定轮廓点的旋转方向。 默认: 0 - CW 支持格式: Blob、Arc

属性	说明
RuntimeContour	指定是否在对象运行时检测轮廓点。 默认: True(在运行时间检测) 支持格式: Blob、Line、Arc
SamplingPitch	设置轮廓点减少的程度。 默认: 0(不减少) 支持格式: Blob、Line、Arc
SaveTeachImage	设置示教模型时, 是否保存图像文件。
ScoreWeightContrast	根据对比度设置得分百分比。 默认: 50 支持格式: Line、Arc
ScoreWeightStrength	根据边缘强度设置得分百分比。 默认: 50 支持格式: Line、Arc
SearchWidth	定义边缘搜索的宽度。 范围: 3 - 99。 默认: 3 支持格式: Line、Arc
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。 支持格式: Blob、Line
SearchWinAngle	定义待搜索区域的角度。 支持格式: Blob、Line
SearchWinCenterX	定义待搜索区域中心的 X 轴坐标值。 支持格式: Blob、Line
SearchWinCenterY	定义待搜索区域中心的 Y 轴坐标值。 支持格式: Blob、Line
SearchWinHeight	定义待搜索区域的高度(单位: pixel)。 默认: 100 支持格式: Blob、Line
SearchWinLeft	定义待搜索区域的最左边位置(单位: pixel)。 支持格式: Blob、Line
SearchWinTop	定义待搜索区域的最上边位置(单位: pixel)。 支持格式: Blob、Line
SearchWinType	定义待搜索区域的类型(Rectangle, RotatedRectangle, Circle)。 支持格式: Blob
SearchWinWidth	定义待搜索区域的宽度(单位: pixel)。 默认: 100 支持格式: Blob



属性	说明
SizeToFind	选择要查找的缺陷尺寸。 默认：1 - 最大值 支持格式：Blob
ShowModel	显示轮廓的示教模型。 支持格式：Blob
Sort	选择对象结果的排序方式。 支持格式：Blob
StartPntObjResult	指定使用 StartPointObject 的哪个结果。 默认：1 支持格式：Blob、Line、Arc
StartPointObject	指定使用哪个视觉对象来定义直线的起点。 默认：Screen 支持格式：Blob、Line、Arc
StartPointType	指定用于定义直线起点的起点类型。 默认：“0 - Point” 支持格式：Blob、Line、Arc
StrengthTarget	设置搜索边缘的强度。 默认：0 支持格式：Line、Arc
StrengthVariation	设置 StrengthTarget 的变化量总和。 默认：0 支持格式：Line、Arc
ThresholdAuto	可定义是否自动设置被检测对象、背景、图像的边缘灰度的阈值。 默认：Disables 支持格式：Blob
ThresholdBlockSize	定义了相邻区域的参考范围，以决定当ThresholdMethod属性设定为LocalAdaptive时的阈值。 默认：1/16ROI 支持格式：Blob
ThresholdColor	定义阈值范围内像素的颜色。 默认：Black 支持格式：Blob

属性	说明
ThresholdHigh	<p>同ThresholdLow属性一起使用，用于定义被检测对象 (或对象)、背景和图像边缘的灰度范围。</p> <p>ThresholdHigh属性用于定义图像的被检测对象区域的灰度范围的上限。在ThresholdLow和ThresholdHigh定义的灰度范围内的图像的任何部分的像素权重都会分配为1。(即检测对象的一部分。)</p> <p>如果ThresholdAuto属性为“True”，且ThresholdColor属性为“White”，该属性值将被设为255且无法更改。</p> <p>默认：128</p> <p>支持格式：Blob</p>
ThresholdLevel	<p>定义了当ThresholdMethod属性设定为LocalAdaptive时，与相邻区域的亮度差异的比率。</p> <p>默认：15%</p>
ThresholdLow	<p>同ThresholdHigh属性一起使用，用于确定表示特征(或对象)、背景和图像边缘的像素灰度区域。区域ThresholdLow用于定义图像的被检测对象区域的灰度范围的下限。在ThresholdLow和ThresholdHigh定义的灰度范围内的图像的任何部分的像素权重都会分配为1。(即检测对象的一部分。)</p> <p>如果ThresholdAuto属性为“True”，且ThresholdColor属性为“Black”，该属性值将被设为0且无法更改。</p> <p>默认：0</p> <p>支持格式：Blob</p>
ThresholdMethod	<p>设置二值化处理的方法。</p>
ContourTolerance	<p>设置减少轮廓点时的容差。</p> <p>默认：0</p> <p>支持格式：Blob、Line、Arc</p>

**Contour对象结果**

以下列表是对Contour对象结果的总结，并附有简要说明。欲了解每条结果的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference*。

结果	说明
CameraX	返回相机坐标系中检测到的轮廓点的 X 坐标位置。
CameraY	返回相机坐标系中检测到的轮廓点的 Y 坐标位置。
CameraXYU	仅限运行时间。 返回相机坐标系中的已检测到轮廓点的CameraX、CameraY和CameraU坐标。
Found	返回是否检测到了对象。
NumberFound	返回轮廓点的总数。
Passed	返回测试结果合格与否。
PixelX	返回轮廓点的 X 坐标，用像素表示。
PixelY	返回轮廓点的 Y 坐标，用像素表示。
PixelXYU	仅限运行时间。 返回已检测到轮廓点的PixelX、PixelY和PixelU坐标(单位: 像素)
RobotX	返回机器人坐标系中轮廓点的 X 坐标位置。
RobotY	返回机器人坐标系中轮廓点的 Y 坐标位置。
RobotXYU	仅限运行时间。 返回机器人坐标系中已找到的轮廓点的RobotX、RobotY和RobotU坐标。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。 这样便于对比结果。
Time	返回处理对象所需要的时间(单位: 毫秒)。

### 轮廓提取原理

轮廓提取大致可分为两个步骤。每种步骤的细节如下。

#### 1. 创建初始轮廓点

检测二进制对象(Blob)

描绘轮廓(Blob)

检测边缘(Line/Arc)

#### 2. 编辑轮廓点

减少轮廓点

获取轮廓

### 检测二进制对象(Blob)

如果ContourMode设为“Blob”时，将使用与Blob对象相同的功能搜索窗口内的二进制对象。有关原理的更多详细内容，请参阅以下内容。

#### 6.2.4 Blob对象 – Blob检查的工作原理

Contour对象在搜索窗口中找到所有二进制对象，然后进入下一步。

### 描绘轮廓(Blob)

当ContourMode设置为“Blob”时，描绘轮廓的步骤将在“检测二进制对象”步骤结束后进行。在这一步中，描绘“检测二进制对象”步骤中检测到的第一个结果的二进制对象轮廓，产生初始轮廓点。因此，需要调整搜索窗口的位置，并调整Sort、MinArea、MaxArea等其他属性的设置，以便将要获取轮廓的工件检测为第一结果。

初始轮廓点以无间隙的连续轨迹输出。初始轮廓点从最接近StartPointObject指定的坐标位置开始，到最接近EndPointObject指定的坐标位置点结束。

### 检测边缘(Line/Arc)

当ContourMode设置为“Line”或“Arc”时，边缘搜索线的检测结果用作初始轮廓点。初始轮廓点从最接近StartPointObject指定的坐标位置的点开始，到最接近EndPointObject指定的坐标位置的点结束。

### 减少轮廓点


创建初始轮廓点后，不需要的点将被移除。轮廓点减少的程度基于SamplingPitch和ContourTolerance值设置。SamplingPitch设置用于确定获取一个轮廓点所需的初始轮廓点的数目。例如，如果设置为“10”，将从最多10个初始轮廓点中提取一个轮廓点。此外，ContourTolerance的值还可以确定减少初始轮廓点时的容许误差。该值越大，不符合工件形状的轮廓点就越少。

### 获取轮廓

此步骤获取的结果数据，即为上一步中减少轮廓点。获取时，轨迹的方向由RotationDirection/LineDirection的设置决定。

在Blob格式的情况下，如果RejectOnEdge设为“False”，则目标工件有时可能会超出搜索窗口。发生这种情况，不会获取搜索窗口的边框作为轮廓。在获取轨迹的过程中，接触到搜索窗口的部分，会停止轨迹。如果搜索窗口可以获取多个轨迹，除非设置了StartPointObject和EndPointObject，否则将获取最长轨迹。

#### 第1步：新建一个Contour对象

- (1) 点击Vision Guide工具栏中的<All Tools>- <Contour>按钮。
- (2) 将鼠标移动至图像显示区域。鼠标指针将变为Contour对象图标。
- (3) 继续移动鼠标，直至该图标位于图像显示区域中的所需位置。
- (4) 点击鼠标左键，创建对象。
- (5) 请注意，对象名称将自动创建。本示例中创建的对象名为“Contour01”，因为它是在本序列中创建的第一个Contour对象。(稍后将说明如何修改名称。)

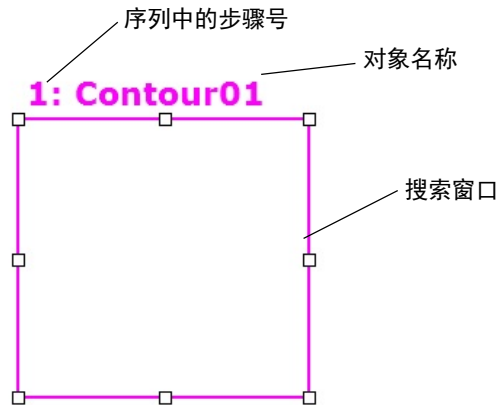
根据所需获得的轨迹类型更改ContourMode。ContourMode的可选类型请参见下表。

#### ContourMode 的设置值 判断标准

Blob	用于获取目标工件完整一周的轨迹。 用于获取复杂形状的轨迹。
Line	用于获取目标工件的部分轨迹。 用于获取几乎无起伏的简单线条轨迹。
Arc	用于获取目标工件的部分轨迹。 用于获取几乎无起伏的圆弧轨迹。

第2步：设置轮廓检测位置(ContourMode: Blob时)

如果ContourMode为Blob，屏幕上将显示一个类似下图的Contour对象：

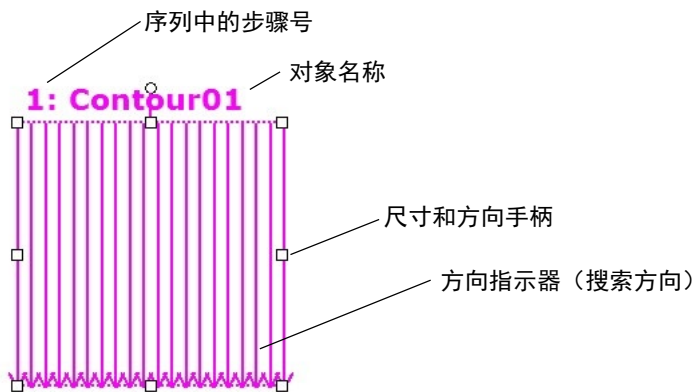


新Contour对象布局(ContourMode: Blob)

- (1) 点击Contour对象的名称标签，然后保持按下鼠标的同时拖动Contour对象，以将搜索窗口的左上角与所需位置对齐。
- (2) 点击尺寸手柄并拖动鼠标，更改Contour对象搜索窗口的大小，并将其调整为适当尺寸。(搜索窗口就是进行Blob搜索的区域。)

第2步：设置轮廓检测位置(ContourMode: Line时)

如果ContourMode为Line，屏幕上将显示一个类似下图的Contour对象：

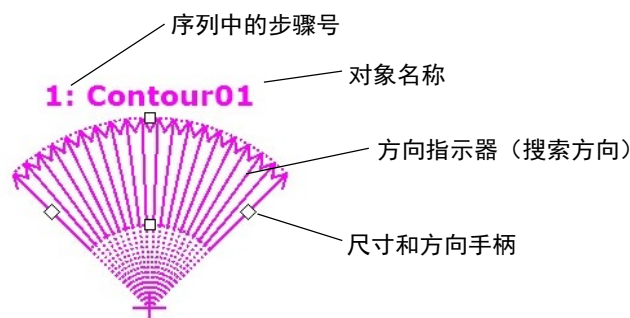


新Contour对象布局(ContourMode: Line)

- (1) 点击Contour对象的名称标签，然后保持按下鼠标的同时拖动Contour对象，以将搜索窗口的左上角与所需位置对齐。
- (2) 点击尺寸手柄并拖动鼠标，更改Contour对象搜索窗口的大小，并将其调整为适当尺寸。  
画面中会显示Edge对象。

### 第2步：设置轮廓检测位置(ContourMode: Arc时)

如果ContourMode为Arc，现屏幕上将显示一个类似下图的Contour对象：



新Contour对象布局(ContourMode: Arc)

- (1) 点击Contour对象的名称标签，然后保持按下鼠标的同时拖动Contour对象，以将搜索窗口的左上角与所需位置对齐。
- (2) 点击尺寸手柄并拖动鼠标，更改Contour对象搜索窗口的大小，并将其调整为适当尺寸。

### 第3步：设置Contour对象属性(ContourMode: Blob时)

接下来设置Contour对象(ContourMode: Blob)的属性值。

以下是ContourMode设为“Blob”时一些常用的属性。

请参阅 *Vision Guide 7.0 Properties and Results Reference* 或“Correlation对象属性”列表，了解许多不同视觉对象所用其它属性的详细信息，如AbortSeqOnFail和Graphics等。



注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。正确配置MaxArea、MinArea、RejectOnEdge和其他属性值，可降低错误检测的风险。

#### Name 属性

新创建的Contour对象中，默认的项目名称是“Contour\*\*”。在一个视觉序列中要使用多个Contour对象时，使用\*\*的编号对Contour进行区分。

默认情况下，会将视觉序列中使用的第一个Contour对象命名为“Contour01”。

需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Contour 对象名称的地方都会变为新的名称。

#### Polarity 属性

有时需要从浅色背景上检测深色对象(DarkOnLight)，有时需要从深色背景上检测浅色对象(LightOnDark)。Polarity 属性的基本目的就是做出这个选择。

默认设置为“DarkOnLight”(深色对象在浅色背景上)。如果要更改此默认设置，请点击 Polarity 属性的值字段，将出现一个下拉列表，包含“DarkOnLight”和“LightOnDark”两个选项。单击您要选择的项目。

MinArea	定义获取轮廓时二进制对象的面积。
MaxArea	默认范围为 25 至 100,000(MinArea 至 MaxArea)。范围很大, 也就意味着, 在不调整 MinArea 和 MaxArea 属性的情况下运行新创建的 Blob 对象, 绝大部分 Blob 对象将报告为 Found。通常, 您需要调整这些属性以反应要检测的二进制对象的合理范围。这样的话, 如果在范围之外检测到任何二进制对象, 都可以确认它并不是您要搜索的目标对象。
RejectOnEdge 属性	利用这条属性, 可以将与搜索窗口边界有接触的那些零件排除在外。通常而言, 这条属性宜设置为 True。
RuntimeContour	定义在执行对象时是否提取轮廓。工件形状改变时, 将其设为“True”。如果工件形状不改变, 将其设为“False”后, 在对象运行时间前示教轮廓信息。您可以通过 ShowModel 属性检查已示教的形状。

您现在可以测试Blob对象, 测试后, 将再次设置其他必须的属性。

### 第3步: 设置Contour对象属性(ContourMode: Line时)

接下来设置Contour对象(ContourMode: Line)的属性值。

以下是ContourMode设为“Line”时一些常用的属性。

请参阅 *Vision Guide 7.0 Properties and Results Reference* 或“Correlation对象属性”列表, 了解许多不同视觉对象所用其它属性的详细信息, 如AbortSeqOnFail和Graphics等。

EdgeType(单边缘)	选择搜索边缘的类型。 如果设置为双边缘, 则会分别搜索每个方向的边缘, 然后返回中心点位置。
Name 属性 (“Contourxx”)	新创建的Contour对象中, 默认的项目名称是“Contour**”。在一个视觉序列中要使用多个Contour对象时, 使用**的编号对Contour进行区分。 默认情况下, 会将在视觉序列序列中使用的第一个Contour对象命名为“Contour01”。 需要修改对象名称时, 请选择 Name 属性工具栏, 输入新的名称然后按回车键。这样所有显示 Contour 对象名称的地方都会变为新的名称。
NumberOfEdges(1)	沿着边缘搜索线搜索 1 条或以上的边缘。
Polarity (LightToDark)	使用 “LightToDark” 属性搜索边缘。 如果您要寻找 DarkToLight(深色到浅色)的边缘, 请更改 Polarity 设置。



**第3步：设置Contour对象属性(ContourMode: Arc)**

接下来设置Contour对象(ContourMode: Arc)的属性值。

以下是ContourMode设为“Arc”时一些常用的属性。

请参阅 *Vision Guide 7.0 Properties and Results Reference* 或“Correlation对象属性”列表，了解许多不同视觉对象所用其它属性的详细信息，如AbortSeqOnFail和Graphics等。

<b>EdgeType(单边缘)</b>	选择搜索边缘的类型。 如果设置为双边缘，则会分别搜索每个方向的边缘，然后返回中心点位置。
<b>Name 属性 (“Contourxx”)</b>	新创建的Contour对象中，默认的项目名称是“Contour**”。 在一个视觉序列中要使用多个Contour对象时，使用**的编号对Contour进行区分。 默认情况下，会将在视觉序列序列中使用的第一个Contour对象命名为“Contour01”。 需要修改对象名称时，请选择 Name 属性工具栏，输入新的名称然后按回车键。这样所有显示 Contour 对象名称的地方都会变为新的名称。
<b>NumberOfEdges(5)</b>	搜索 5 条边缘，以找到圆弧边缘。
<b>Polarity (LightToDark)</b>	使用 “LightToDark” 属性搜索边缘。 如果您要寻找 DarkToLight(深色到浅色)的边缘，请更改 Polarity 设置。

**第4步：设置轮廓点属性**

设置用于调整轨迹精度、轮廓点数目等属性。

使用的属性如下所示。欲了解每条属性的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference*。

<b>SamplingPitch</b>	定义减少的轮廓点的最大数目。 设为“5”时，则最大数目为最初检测到数量的 1/5。
<b>ContourTolerance</b>	定义可减少轮廓点的数量。 该值越小，允许减少的工件形状轮廓点的数量就越少。请在检查对象运行时结果同时，根据精度需要调整该值。
<b>StartPointObject、 EndPointObject</b>	指定轨迹的开始位置和结束位置。 如设定本属性，请在创建了 Point 对象后，调整视觉序列中的运行顺序，使其在 Contour 对象之前运行。将 Point 对象放置在开始/结束轨迹的位置附近。
<b>RotationDirection、 LineDirection</b>	指定轨迹方向。 当 ContourMode 为 Blob 或 Arc 时，设置轨迹的旋转方向 (CW/CCW)。当 ContourMode 为 Line 时，请以边缘搜索线垂直向下为基准，指定左右方向(LeftToRight/RightToLeft)。

### 第5步：测试Contour对象并检查结果

要运行Contour对象，点击执行面板上对象的<Run>按钮。

然后将显示Contour对象的结果。这时需检查的主要结果有：

**PixelX、PixelY**      检测到的轮廓点位置(单位:像素)。

**CameraX、CameraY**   基于相机坐标的轮廓点位置(单位:毫米)。

**RobotX、RobotY**     基于机器人坐标的轮廓点位置(单位:毫米)。

**Time 结果**            返回 Contour 对象执行所需时间。

#### NOTE



此时RobotX和RobotY以及CameraX和CameraY的结果将返回“no cal”。这意味着由于未进行校准，视觉系统将无法计算相对于机器人坐标系或相机坐标系的坐标结果。有关详细内容，请参阅 7. *视觉校准*。

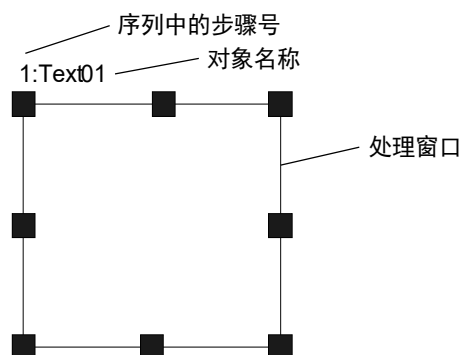
## 6.2.21 Text对象

### Text对象描述

Text对象将视觉对象的执行结果，作为文本显示在屏幕上。

### Text对象布局

Text对象有一个图像处理窗口，如下所示。



### Text对象属性

以下列表是对Text对象属性的总结。*Vision Guide 7.0 Properties and Results Reference Manual* 中阐述了每个属性的详细情况。

属性	说明
AbortSeqOnFail	当对象运行错误(判定结果为不良)时，整个序列会立即停止，并不再处理任何程序。 默认：False
CenterPointObject	指定对象中心点位置。 若将该属性设置在“Screen”上时，则可将该对象放在画面上的任何位置上。如指定了其他视觉对象时，该参数的中心点将会设定在 PixelX, PixelY 上。
CenterPntObjResult	指定要使用的CenterPointObject属性结果。 如果指定了 All，则 Text 对象将应用于指定视觉对象的所有结果(NumberFound)。 默认：1
CenterPntOffsetX	使用CenterPointObject属性定位了搜索窗口的中心点后，设置或返回X偏移。
CenterPntOffsetY	使用 CenterPointObject 属性定位了搜索窗口的中心点后，设置或返回 Y 偏移。
Caption	为 Text 对象指定标题。 默认：Empty String
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。

属性	说明
Description	设定用户指定的描述 默认：空白
Enabled	指定是否执行对象。 默认：True
FailColor	选择不合格对象的颜色。
Font	设置文本显示的格式。
FontBold	字体加粗显示。(仅对 SPEL+程序有效)
FontItalic	字体倾斜显示。(仅对 SPEL+程序有效)
FontName	设置字体的名称。(仅对 SPEL+程序有效)
FontSize	以磅为单位，设置字体的大小。(仅对 SPEL+程序有效)
Graphics	设置显示的图表模式。 默认：1 - All
LabelBackColor	设置对象标签的背景颜色。
TextBackColor	选择文本的背景颜色。 默认：Transparent
Name	用于为 Text 对象分配一个唯一的名称。 默认：Text01
PassColor	选择当对象被检测为良品时，搜索窗口显示的颜色。 默认：LightGreen
ResultObject	指定要呈现的视觉对象。
ResultText1	指定要呈现的结果。
ResultText2	指定要呈现的结果。
ResultText3	指定要呈现的结果。
ShowLabel	指定是否在文本中添加标签。
SearchWin	仅限运行时间。 在一次调用中设置或返回搜索窗口左侧参数、顶部参数、高度参数和宽度参数。
SearchWinHeight	定义待搜索区域的高度(单位：pixel)。
SearchWinLeft	定义待搜索区域的最左边位置(单位：pixel)。
SearchWinTop	定义待搜索区域的最上边位置(单位：pixel)。
SearchWinWidth	定义待搜索区域的宽度(单位：pixel)。
UserText	设置用户定义的文本。

### Text对象结果

以下列表是对Text对象结果的总结，并附有简要说明。欲了解每条结果的详情，请参阅 *Vision Guide 7.0 Properties and Results Reference*。

结果	说明
Passed	返回视觉对象的测试结果合格与否。
Found	返回是否获取了结果。
PixelX	返回文本显示的 X 坐标位置，用像素表示。
PixelY	返回文本显示的 Y 坐标位置，用像素表示。


### 使用Text对象

以上介绍的基础知识，方便了解如何使用Vision Guide Text对象。

以下是使用Text对象所需的步骤。

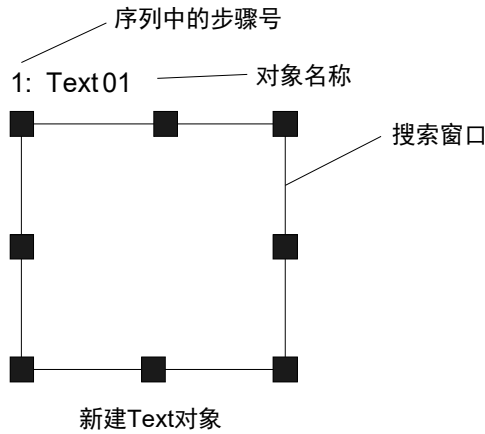
- 创建一个新的Text对象
- 确定搜索窗口的位置和尺寸
- 设置文本
- 测试Text对象并检查结果

#### 第1步：创建一个新的Text对象

- (1) 点击Vision Guide工具栏中的<All Tools>-  <Text>按钮。
- (2) <Text>图标会出现在Text对象工具栏的按钮上方。
- (3) 点击该<Text>图标，并将其拖至Vision Guide窗口的图像显示区域中。请注意，对象名称将自动创建。本示例中创建的对象名为“Text01”，因为它是在本序列中创建的第一个Text对象。(稍后将说明如何修改名称。)

### 第2步：确定搜索窗口的位置和尺寸

画面上会显示一个如下所示的 Text 对象：

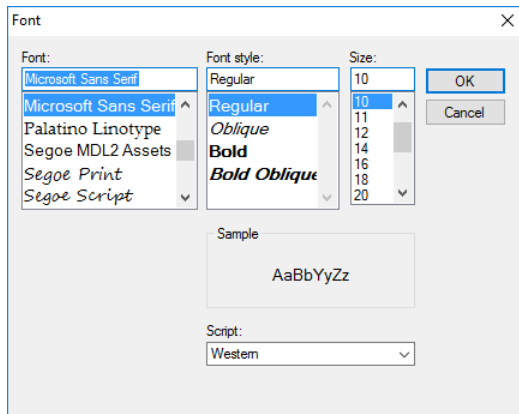


- (1) 点击Text对象的名称标签，然后保持按下鼠标的同时拖动Text对象，以将搜索窗口的左上角与所需位置对齐。
- (2) 点击尺寸手柄并拖动鼠标，更改Text对象搜索窗口的大小，并将其调整为适当尺寸。  
(以搜索窗口的左上角为基准显示文本)

### 第3步：设置文本

- (1) 使用ResultObject指定要呈现结果的视觉对象。可在ResultObject中选择的视觉对象，必须是视觉序列中在Text对象之前执行的视觉对象。
- (2) 从ResultText1到3中选择所需呈现的结果。
- (3) 要显示结果文本以外的字符文本，编辑UserText属性。
- (4) 编辑Font属性调整字体大小。

点击Font属性输入字段，可显示以下设置窗口。您可以设置字体、字形和字号。



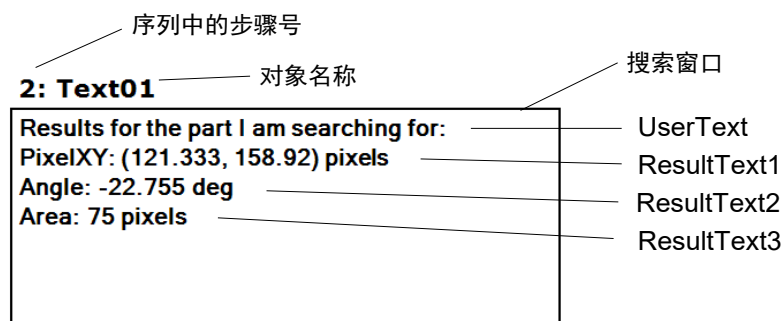
#### 第4步：测试Text对象并检查结果

点击执行面板上对象的<Run>按钮。

作为Text对象设置的文本显示在画面上。如果文本显示有问题，请调整属性设置。

#### Text对象显示的文本

作为Text对象显示的文本样式如下所示。



文本按照UserText、ResultText1、ResultText2、ResultText3的顺序从上到下显示。如果UserText字段留空，且ResultText 1到3的值设为“None”，则不显示该部分。如果字符文本过长，无法在搜索窗口内完全显示，则不会显示该部分。

## 6.2.22 Decision对象

### Decision对象的概要

Decision对象用于根据指定视觉对象的成功或失败，控制执行序列的流程。

### Decision对象的属性

用于Decision对象的属性一览如下所示，下面进行简单的说明。有关各属性的详细信息，请参阅《Vision Guide 7.0 Properties and Results Reference》。

属性	说明
ConditionObject	指定同一序列内的Decision对象之前的视觉对象。 使用该结果，决定使用Decision对象的True分歧或False分歧的序列流程。
Description	设定用户指定的描述。 默认：空白
Enabled	指定是否执行对象。 默认：True
Name	用于为Decision对象分配一个唯一的名称。 默认：Decision01
TrueCond	使用以ConditionObject指定的视觉对象的Passed结果，指定进至True分歧的条件。 TargetPassed：当Passed结果为True时，进至True分歧。 TargetFailed：当Passed结果为False时，进至False分歧。 TargetNoExec：当未执行视觉对象时，进至True分歧。 默认：0-TargetPassed

### Decision对象结果

Decision对象中无结果。

### Decision对象的使用方法

以上介绍的基础知识可帮助用户了解Vision Guide Decision对象的使用方法。

使用Decision对象所需的步骤如下所示。

- 新的Text对象的创建方法
- 条件分歧的设置
- 执行对象的设置及执行
- Coordinates对象的使用方法



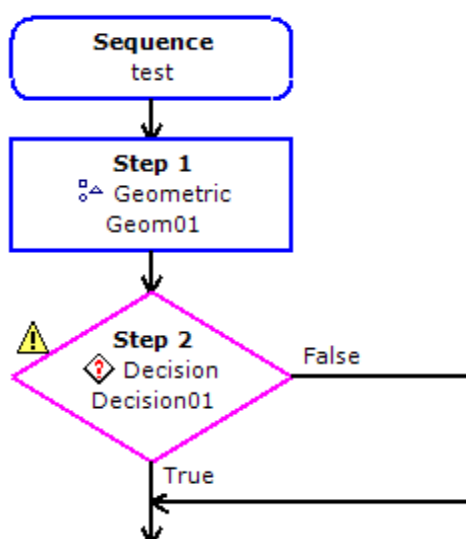
### Step 1: 创建新的Decision对象

- (1) 事先添加要设置到ConditionObject属性的Decision对象的对象。本示例中，将添加Geometric对象。

如果不设置需要在Decision对象之前执行的ConditionObject，无法使用Decision对象，因此无法添加至序列的开头。

- (2) 点击Vision Guide工具栏 - <All Tools> -  <Decision>按钮。

- (3) 点击< Decision >图标，拖动至Vision Guide窗口的图像显示区域。  
对象名称将自动创建。例如，如果是在序列中创建的最初的Decision对象，名称将为“Decision01”。（稍后将说明如何修改对象名称。）  
因为Decision对象无搜索窗口，请从流程图和序列树确认步骤位置。



### Step 2: 条件分歧的设置

- (1) 从Decision对象的ConditionObject属性的下拉列表选择要确认结果的对象。  
ConditionObject仅可设置在序列内于Decision对象之前执行的对象。本示例中，将ConditionObject设置到Geom01。

- (2) 设置Decision对象的TrueCond属性，使用在（1）选择的对象的Passed结果值，指定进至True分歧的条件。

如果已指定默认的0-TargetPassed，（1）的对象的Passed结果为True时，将进至True分歧，并执行视觉对象。

Passed结果为False时，如要进至True分歧并执行视觉对象，则将TrueCond属性指定为TargetFailed。

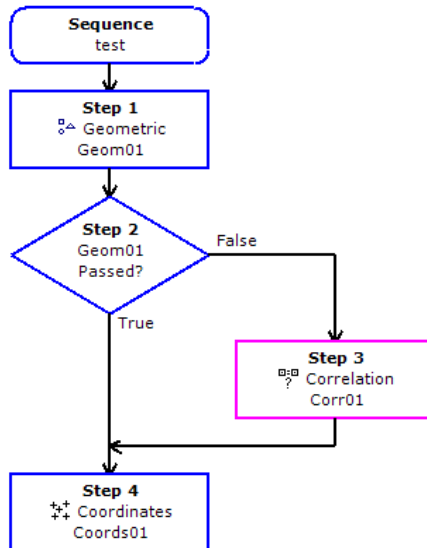
**Step 3: 添加及执行对象**

- (1) 添加通过Decision对象的True及False分歧执行的对象。  
 从Vision Guide工具栏选择想要添加的对象后，将其拖动至流程图，放置在Decision对象的分歧内。可在各分歧内放置所需数量的对象。  
 无法在分歧内放置Decision对象。
- (2) 在Decision对象的分歧聚合之后，添加对象。  
 根据需要，可在Decision对象分歧聚合后，将对象添加到序列中。
- (3) 确认视觉序列的动作。  
 执行的分歧因Decision对象的ConditionObject指定的对象的结果与TrueCond属性设置值的组合而发生变化。  
 也可执行整个序列，以及单步执行序列，以确认动作。

**Step 4: Coordinates对象的使用方法**

在某些情况下，根据Decision对象的在True或False分歧的对象处理，需要获取像素、相机或机器人的坐标。在这种情况下，将提供坐标结果的对象的结果保存到Coordinates对象后，在程序中从Coordinates对象的结果访问坐标。

- (1) 将Coordinates对象添加到序列中。
- (2) 设置保存坐标的对象的CoordObject属性。任意数量的对象可将坐标储存在相同的Coordinates对象中。执行使用CoordObject的每个对象时，将覆盖保存坐标。  
 本示例中，在Geom01与Corr01两者中，CoordObject均已被设置为Coords01。



- (3) 在程序中，使用VGet，从Coordinates对象获取坐标。本示例中，如果Geom01通过，Geom01坐标结果将被复制到Coords01中。如果Geom01通过，将执行Corr01。如果Corr01通过，Corr01坐标结果将被复制到Coords01中。  
 在SPEL程序中，可从Coords01获取机器人的坐标。

```
VGet test.Coords01.RobotXYU, found, x, y, u
```

## 6.2.23 Coordinates对象

### Coordinates对象的概要

Coordinates对象用于储存其他对象的结果。主要在含有Decision对象的序列中使用。

在含有Decision对象的序列中，获得目标结果的对象随着分歧功能的执行而变化。通过将用于储存坐标结果的Coordinates对象设置到各分歧的对象，可从Coordinates对象获取目标的结果。

Coordinates对象从想要储存结果的对象中选择。从下述属性选择Coordinates对象。

属性	说明
CoordObject	指定储存结果的Coordinates对象 在执行对象时进行储存处理，如果未执行Decision的分歧功能等，将不进行复制处理 默认：None

可指定CoordObject属性的对象仅限于含有Pixel、Camera及Robot X、Y、U的对象。无关设置的对象的步骤前后顺序，在任一步骤均可将Coordinates对象指定到CoordObject属性。

可从多个对象的CoordObject属性指定Coordinates对象。在此情况下，执行每个对象时，将执行储存程序，并覆盖保存储存的结果。

### Coordinates对象的属性

以下列表是简单介绍Coordinates对象属性的总结。有关各结果的详细信息，请参阅《Vision Guide 7.0 Properties and Results Reference》。

属性	说明
CurrentResult	可定义要在对象窗口的结果列表中显示的结果，或者是当一个搜索窗口中的对象物检测到了多个结果时需要返回的数据。
Description	设定用户指定的描述。 默认：空白
Enabled	指定是否执行对象。 默认：True
Name	用于为Coordinates对象分配一个唯一的名称。 默认：Coords01

**Coordinates对象的结果**

以下列表是简单介绍Coordinates对象结果的总结。有关各结果的详细信息，请参阅《Vision Guide 7.0 Properties and Results Reference》。

结果	说明
Angle	返回已找到的点的角度。
CameraX	返回相机坐标系内的查找对象的X坐标。
CameraY	返回相机坐标系内的查找对象的Y坐标。
CameraXYU	仅执行时 返回相机坐标系内查找对象位置的CameraX坐标、CameraY坐标和CameraU坐标。
NumberFound	返回已找到的点的数目。
Found	返回是否查找到了对象。
Passed	返回测试结果合格与否。
PixelX	返回像素坐标系内的查找对象的X坐标。
PixelY	返回像素坐标系内查找对象的Y坐标。
PixelXYU	仅执行时 以像素返回查找对象位置的PixelX坐标、PixelY坐标、PixelU坐标。
RobotX	返回机器人坐标系内的查找对象的X坐标。
RobotY	返回机器人坐标系内的查找对象的Y坐标。
RobotU	返回机器人坐标系内的查找对象的旋转量。
RobotXYU	仅执行时 返回机器人坐标系内的查找对象位置的X坐标、Y坐标、旋转量。
ShowAllResults	以表格形式在对话框中显示指定视觉对象的所有结果。 这样便于对比结果。

**Coordinates对象的使用方法**

下面对Coordinates对象的创建方法及使用方法的以下项目进行说明。

- 创建一个新的Coordinates对象
- 设置CoordObject
- 执行序列

开始进行以下步骤前，首先创建一个新的视觉序列或选择要使用的视觉序列。

**Step 1: 创建新的Coordinates对象**

- (1) 请点击Vision Guide工具栏 - <All Tools> -  <Coordinates>按钮。
- (2) <Coordinates>对象工具栏按钮上显示Coordinates图标。
- (3) 点击Coordinates图标，拖动至Vision Guide窗口的图像显示区域。
- (4) 对象名称将自动创建。本示例中创建的对象名称为“Coords 01”，因为它是在本序列中创建的第一个Coordinates对象。（稍后将说明如何修改对象名称。）

**Step 2: 设置CoordObject**

- (1) 添加想要将结果储存到Coordinates对象的对象。  
在前文中已添加Coordinates对象，但因为想要复制结果的对象和Coordinates对象的步骤位置的前后关系不设限制，请根据情况调换步骤顺序。
- (2) 添加对象后，从CoordObject属性选择及指定Coordinates对象。

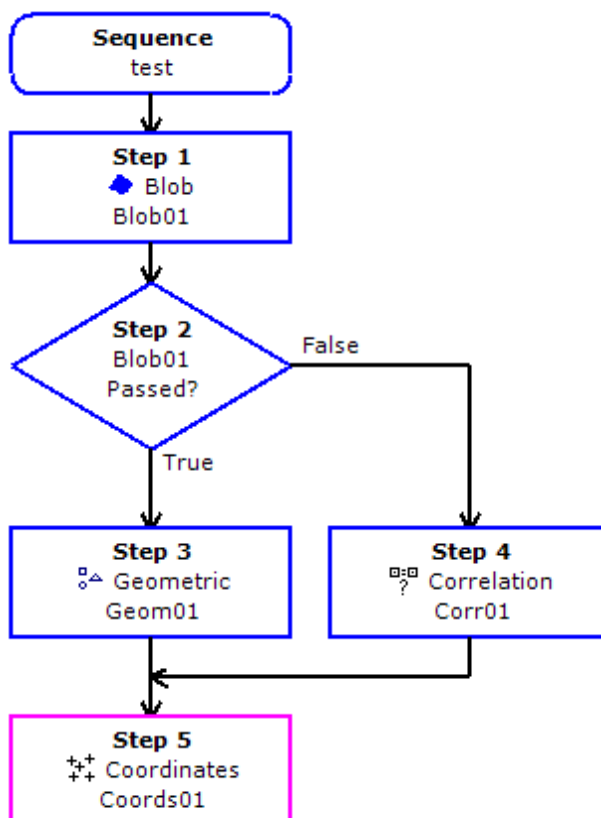
**Step 3: 执行序列**

执行序列。

执行序列之后，请确认Coordinates对象的结果。已储存Coordinates对象被设置为CoordObject的对象的结果。

在如下所示流程图的示例中，Geom01和Corr01两者的CoordsObject属性已被设置为Coords01。如果Blob01通过，将执行Geom01，并将结果储存到Coords01中。如果Blob01未通过，将执行Corr01，并将结果储存到Coords01中。在SPEL程序中，可从Coords01获取机器人的坐标。

VGet test.Coords01.RobotXYU, found, x, y, u



### 6.2.24 从一个对象返回多个结果

Blob 对象、Geometric 对象、Edge 对象、Correlation 对象和 DefectFinder 对象可以在一个搜索窗口中查找多个被检测对象。

为了更好的理解如何处理单个视觉对象的多个结果，接下来将进一步说明以下内容。

- CurrentResult 属性
- NumberToFind 属性
- NumberFound 结果
- ShowAllResults 结果

#### 视觉对象可查找到特征的默认数和最大可设置数

Blob 对象、Geometric 对象、Edge 对象、Correlation 对象或 DefectFinder 对象的默认属性配置，使其是在一个搜索窗口中只能查找到一个被检测对象。这是因为 NumberToFind 属性的默认设置为 1。

如果将 NumberToFind 属性设置为大于 1 的数字，视觉对象会尝试查找相同数量的对象。

如果将 NumberToFind 属性设置为 All，则会尝试查找到最大对象数量(100)。

设置 `NumberToFind` 属性并运行视觉对象。则会检测出并显示多个符合查找标准的对象。(带十字光标的绿框表示已查找到对象的返回位置)。此外，在检测到的二进制中，被判断为 `CurrentResult` 的对象，会区别于其他二进制对象，用比较浅的绿色突出显示。

### 已查找对象的排序

首次运行多结果对象时，`CurrentResult` 属性将自动设置为 1，这使视觉对象的多个结果中的第一结果将显示在对象窗口的结果列表中。

如果是 `Correlation` 对象和 `Geometric` 对象，第一个结果就是得分最高的对象(与本 `Correlation` 对象的所有已查找到的特征相比)。第二个结果就是得分次之的结果，以此类推。

如果是 `Blob` 对象和 `DefectFinder` 对象，第一个结果就是基于 `SizeToFind` 属性和 `Sort` 属性值返回的结果。(例如，如果 `SizeToFind` 属性设置为“Largest”，则第一个结果就是查找到的最大对象。)想了解更多关于结果排序的信息，请参阅 *Vision Guide 7.0 Properties and Results Manual* 中的 `SizeToFind` 属性和 `Sort` 属性。

### 检查视觉对象的多个结果

如果仔细查看结果列表的标题，您会注意到有些地方显示“1/10”。(为方便说明，此处假设已经将 `Blob` 对象的 `NumberToFind` 属性设置为 10。)

这个显示代表 `CurrentResult` 是 10 个(在 `NumToFind` 属性中定义)被检测到对象中的第一个。

请注意，结果列表标题中所显示的“1/10”中第二个数字(此处为数字 10)，这是在 `NumberToFind` 属性中设置的值，而非实际查找到的对象的数量。

当设置的检测对象为 10 时，实际上可能只检测到了 5 个对象。此时您可能会产生疑问，本应该显示排名前 10 个结果，但是为什么只显示了 5 个对象。以下将说明的方法，可以帮助您确认未检测到另 5 个对象的原因。

您可以通过更改 `CurrentResult` 属性，来切换当前显示的二进制对象，以查看其中任何一个二进制的结果。

您可以通过直接在 `CurrentResult` 属性值字段中，输入编号来查看您要确认的结果。也可以将光标移至 `CurrentResult` 属性值字段，然后使用 `[Shift] + [↓]` 或 `[Shift] + [↑]` 键搜索结果。

通过查看 `CurrentResult` 属性或结果列表标题，可以确认结果列表中显示的顺序。例如，当 `NumberToFind` 属性设置为 10，则第一个结果将显示为“1/10”，第二个结果是“2/10”，第三个结果是“3/10”，以此类推。



注意

- 修改 `CurrentResult` 属性时，也会改变程序中的 `VGet` 结果。比如，如果一个 SPEL 程序中，如果使用 `VGet` 从视觉对象获取值，当 `CurrentResult` 属性设置为 3，那么 `VGet` 指令将返回第三个 `RobotXYU` 的值。

所以需注意，因为如果 `CurrentResult` 的设置有误，最终可能会导致整个 SPEL+程序错误。

提示：如需获取多个结果，请在 `VGet` 命令中指定明确的结果编号。

示例：`VGet seqname.objname.RobotXYU(1), found, X, Y, U`

### 如何使用NumberFound结果

NumberFound 属性可以显示实际检测到的二进制对象的数量，非常实用。而 EPSL+ 程序也可以获取该结果，所以可以通过 SPEL+ 程序，确认是否找到了正确数量的结果，或计算找到的结果数量，或者计算还有多少其他结果可用。

以下这一小段代码，可用于检查查找到的结果的数目是否小于 5。

```
VGet seqname.objname.NumberFound, numfound
If numfound < 5 Then
    'put code to handle this case here
```

考虑使用视觉在一个 VRun 命令中，检测到尽可能多的工件。机器人会逐个抓取工件，直至将检测到的工件全部移动到传送带上。以下的一小段代码，则可用于控制机器人逐个抓取被检测到的工件，然后移动到传送带的相同位置上。

```
VRun seqname
VGet seqname.objname.NumberFound, numfound
For i = 1 to numfound
    VGet seqname.objname.RobotXYU(i), found, X, Y, U
    If found = True Then
        'Set coordinates found from vision
        VPick = XY(X, Y, -100.000, U)
    Else
        Print "Vision Error: part not found"
    EndIf

    Jump Vpick           'Jump to Vision Pickup position
    On Gripper           'Turn on vacuum
    Wait .1
    Jump Vconvey         'Jump to Conveyor Position to drop part
    Off Gripper          'Turn off vacuum
    Wait .1
Next count
```



### 一次性检查所有多结果

当 Blob 对象和 Correlation 对象查找到多个结果时，可以使用另一个便利的工具来确认结果，这就是 ShowAllResults 结果。

例如，当您想比较最高得分和第二个等分的结果时，或者是查找到第三个结果后发现得分大幅下降时，则可以使用 ShowAllResults 结果，来一次性查看所有结果。

点击 ShowAllResults 结果的值字段后，将出现一个按钮。点击该按钮，则会弹出一个对话框，显示当前视觉对象的所有结果。

ShowAllResults 对话框如下所示。

Result	Found	Area	PixelX	PixelY	Angle	RobotX	RobotY	RobotU	CameraX	CameraY
1	True	2507.0	542.771	394.182	70.146	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
2	True	2479.0	513.804	253.456	85.176	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
3	True	2461.0	131.518	111.677	-59.733	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
4	True	2457.0	526.077	121.878	15.584	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
5	True	2434.0	412.145	357.687	-20.011	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
6	True	2433.0	123.443	264.624	49.105	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
7	True	2394.0	309.274	106.27	-44.788	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
8	True	2381.0	228.311	350.134	22.2	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
9	True	2371.0	284.449	231.224	-58.645	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
10	True	2321.0	396.725	186.843	-5.727	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)

### 使用多结果对话框调试搜索问题

有时，即使在同一生产批次中，正在处理的工件也存在明显差异，或者可能一个工件上有两个或两个以上，相似的被查找对象的特征。

这种情况下，即使已经为 Accept 属性设置了一个适当的值，仍可能导致搜索到错误的对象。遇到这种问题，很难找到原因，可很难预测 Accept 属性正确的值。

多结果对话框则可以帮助您解决这一烦恼。

当您想在一个工件上只检测一个被检测对象，然而却返回了多个结果。则可以使用多结果对话框，来确认返回错误对象的原因。常见的原因有以下两种。

1. 搜索窗口内有两个或两个以上，非常类似的被检测对象，因此返回的 Score 结果十分接近。
2. 设置的 Confusion 或 Accept 属性的值较低，导致即使得分较低的对象也符合 Accept 属性的设定。

当在搜索窗口中仅搜索单个检测对象时，以上两种情况对于初次使用 Vision Guide 的用户来说，可能会十分令人困惑。

当碰到有时可以正确的检测出对象，而有时又错误的检测出其他对象时，请使用多结果对话框来排查问题所在。请按照以下步骤确认问题原因。

- (1) 将 `NumberToFind` 属性设置为“3”或以上。
- (2) 在 `Vision Guide 7.0` 开发环境下运行视觉对象。
- (3) 点击 `ShowAllResults` 属性设置值字段中的按钮，以显示多结果对话框。
- (4) 检查查找到的前三个或更多对象的得分。
- (5) 如果只检测到一个或两个对象(`Vision Guide zhi` 设置了那些被认为是已查找到对象的得分)，为了检测一个以上的对象，请降低 `Accept` 属性设定值，并再次运行视觉对象。(您可以在检查完 `ShowAllResults` 对话框的结果后，恢复 `Accept` 的设置。)
- (6) 点击<`ShowAllResults` 属性>按钮，显示多结果对话框。
- (7) 检查查找到的前三个或更多对象的得分。

当确认了上述的已查找到的前三个或更多对象的得分后，应该已经大致明白问题所在。常见的问题有以下两种。

1. 已检测到的每个对象都有大于 `Accept` 属性设置的得分。  
如果是这种情况，请将 `Confusion` 属性值调高，以确保始终仅检测合适的对象，而不返回仅与 `Accept` 阈值匹配的对象。您还可以调整 `Accept` 属性值。
2. 每个对象的得分都很接近。  
如果是这种情况，请执行以下步骤来区分被检测对象。
  - 重新调整搜索窗口，使非检测对象的部分都不在搜索窗口的范围内。
  - 重新示教模型，使其匹配您要检测的对象特征。
  - 调整应用中照明，使得被检测对象可以得到比非检测对象更高的得分。

### 利用SPEL+语言读取多个结果

综上所述，`CurrentResult` 属性可用于设置结果列表中将显示的结果。

该属性也能用于指定返回第几个结果。换句话说，如需从 `Blob` 对象返回的第三个结果中，获取 `Area` 结果，则需将 `CurrentResult` 设置为 3。

上文中已经说明了，如何从对象窗口属性列表中设置 `CurrentResult` 属性。现在，将介绍如何使用 SPEL+语言读取多个结果。

要利用 SPEL+语言读取多个结果，应将结果当做一种数组处理，在此数组中，应参考带下标编号的 `CurrentResult`。如下所示，第一个示例显示的是如何获得第三个 `Area` 结果，以及如何利用 SPEL+语言将其放入称作 `area` 的变量中。

```
VGet seqname.objname.Area(3), area
```

如下所示，第二个示例显示的是如何获得相同的第三个 `Area` 结果，但是，这次将其确定为 `Area()` 阵列中第三个元素的值。

```
VGet seqname.objname.Area(3), area(3)
```

可使用不同的名称表示阵列中的元素，但不可以表示固定元素，如上述示例 2。注意，将称作“`var`”的变量用作 `Area` 结果的下标。

```
VGet seqname.objname.Area(var), area(var)
```

第 4 个示例假设您已采用一个视觉对象查找多个零件(假如多达 10 个)。为了让机器人识别这些零件(假设是一支笔)，所以您就需要将 X、Y 和 U 坐标存储到变量中，用于表示各个已查找到的零件的坐标值。利用以下代码，可从 `RobotXYU` 结果中输出这些坐标，并将其输入 X、Y 和 U 阵列，机器人随后将移动至这些阵列中的相应坐标。

```
Function test
  Boolean found(10)
  Integer numfound, i
  Real X(10), Y(10), U(10)

  Jump camshot 'move camera into position snap shot

  VRun seq01 'run the vision sequence to find the pens

  VGet seq01.blob01.NumFound, numfound 'how many found

  For i = 1 to numfound 'get robot coords
    VGet seq01.blob01.RobotXYU(I), found(i), X(i), Y(i), Z(i)
  Next i
  'Add code for robot motion here.....
Fend
```

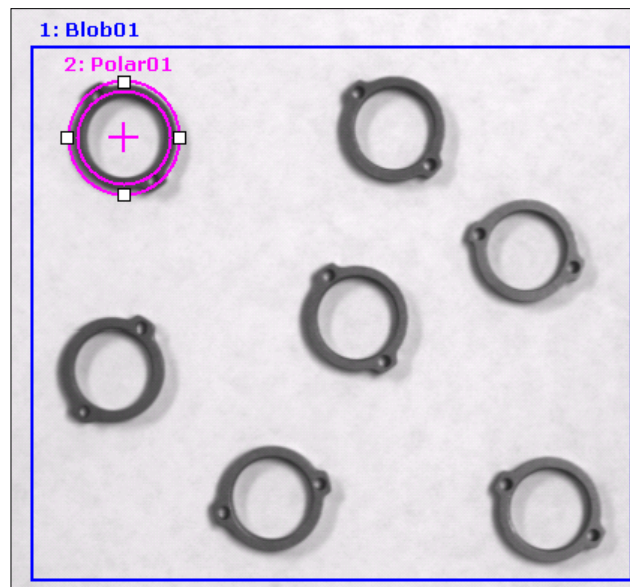
## 6.2.25 多对象自动搜索

您可以对几个视觉对象使用多对象自动搜索。当您指定使用其他对象的所有结果时，将自动创建搜索对象。这样您即可配置一个或多个对象，以进行搜索，并在运行时，将为上一级对象的所有结果自动创建和运行对象。

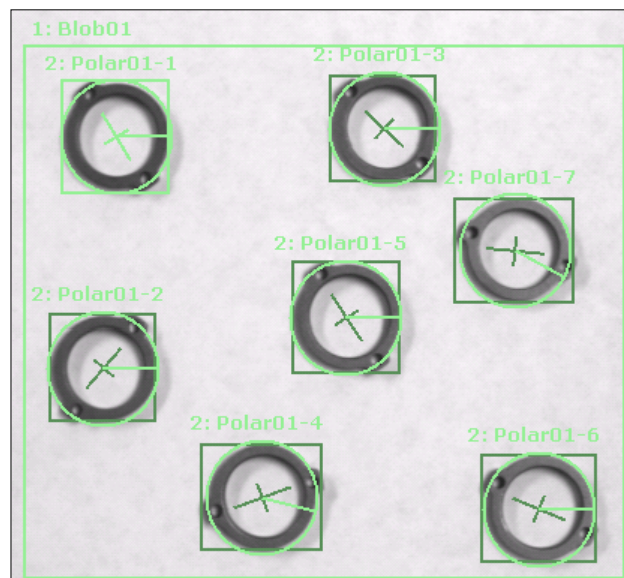
多对象自动搜索可以与CenterPointObject和Frame一起使用。

使用示例：CenterPointObject

1. 创建视觉序列并添加一个Blob对象。将NumberToFind设为“10”。
2. 创建一个Polar对象。  
将Blob对象设置为CenterPointObject，并将CenterPntObjResult设为All。



3. 示教Polar对象。
4. 运行序列。在检测到的二进制对象中，创建并运行Polar对象实例。如果查找到10个二进制对象，将以每个Blob对象的结果为中心，创建10个Polar对象。在下图中，已为检测到的二进制创建了7个Polar对象。



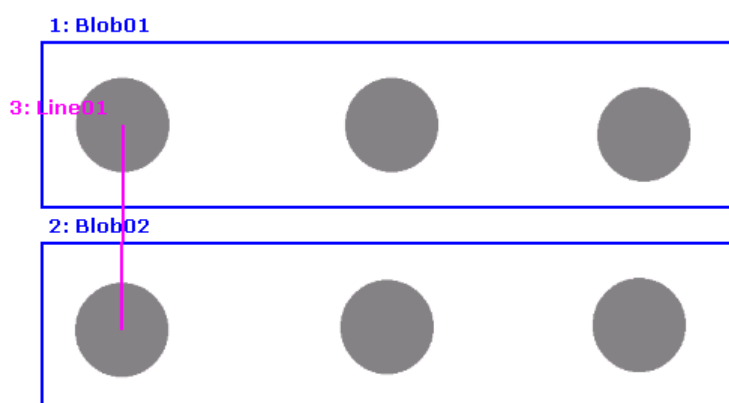
当使用多对象自动搜索时，如果子对象可查找到多个结果，那么将为每个子对象实例找到一个结果。

您还可以将Line、Edge和LineInspector对象，用于多对象自动搜索。

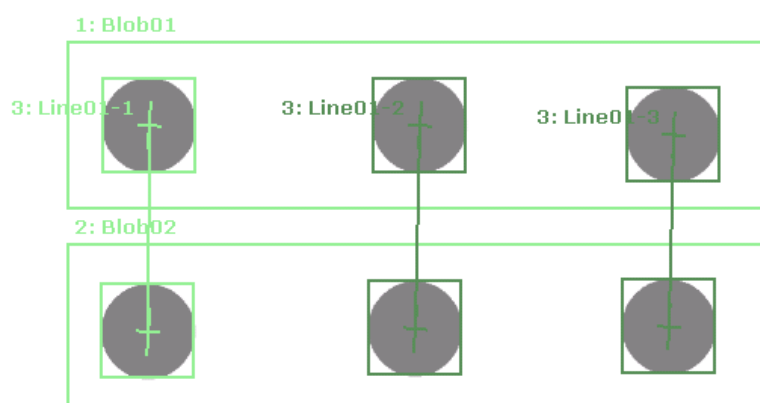
指定StartPointObject和EndPointObject，并将StartPntObjResult和EndPntObjResult设为“All”。

例如：

1. 创建一个新序列，并选择与ImageFile平行的2行二进制(见下图)。
2. 创建Blob01，将NumberToFind设置为“3”，Sort设置为“PixelX”。调整大小和位置以找到第一行二进制对象。
3. 复制并粘贴Blob01以创建Blob02。调整大小和位置以找到第二行二进制对象。
4. 创建Line对象。将StartPointObject设置为“Blob01”，StartPntObjResult设置为“All”。并将EndPointObject设置为“Blob02”，EndPntObjResult设置为“All”。

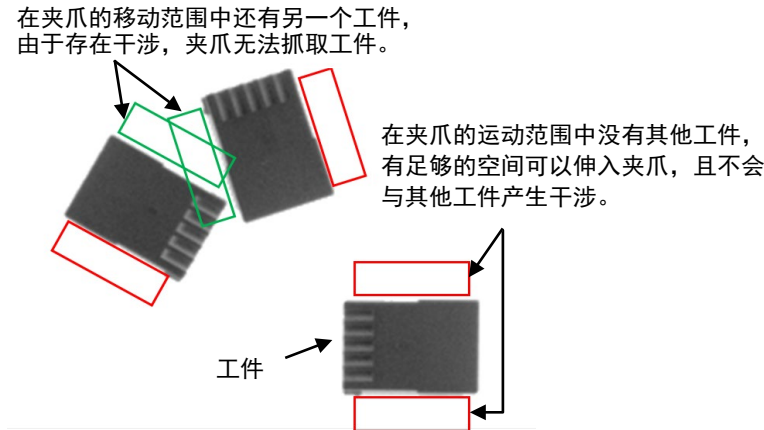


5. 运行序列。将为StartPointObject和EndPointObject的每对结果创建Line对象实例。



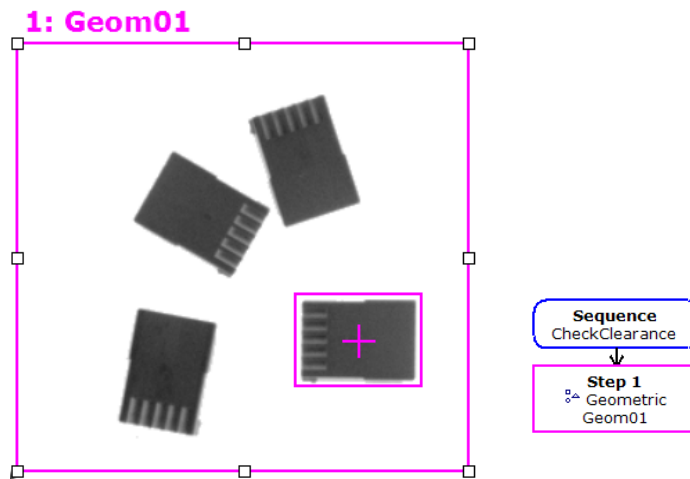
使用示例：CheckClearanceFor

下面以使用两点抓取的夹爪来取放工件的应用为例，来说明自动搜索之一的CheckClearanceFor功能。该视觉序列由2个对象组成，分别是用于检测工件的“Geometric”对象，和检测是否有足够的间隙，可以伸入夹爪而不会产生干涉的“Blob”对象。



例：

1. 创建Geometric (Geom01)作为父对象以检测工件，并设置模型窗口的位置和大小(参考下图)。同时设置相关属性。本案例中，将NumberToFind设置为“All”以检测多个工件。有关CheckClearanceFor功能下可使用的父对象，请参阅*Vision Guide 7.0 Properties & Results Reference*。



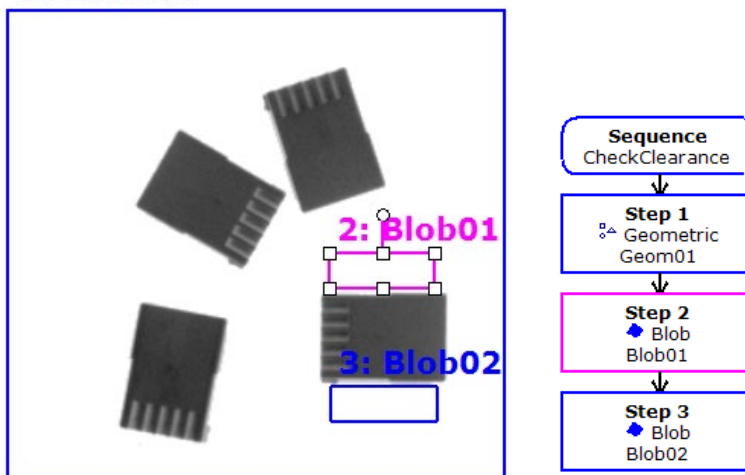
2. 创建子对象Blob (Blob01, Blob02)，用于判断夹具干涉，并将其定位在Geometric窗口的两端 (参考下图)。同时设置相关属性。



如需旋转子对象中检测到的工件，请在子对象中将SearchWinType设置为“RotatedRectangle”。

如设置为“Rectangle”，则搜索窗口无法旋转，也无法跟踪旋转角度。

### 1: Geom01



设置子对象Blob01和Blob02的属性。

3. 将CheckClearanceFor设置为“Geom01”。(参考下图)  
此功能可以在序列执行期间，根据父对象的所有结果，自动创建并运行子对象。

Step 2: Blob01	
Property	Value
AbortSeqOnFail	False
Caption	
CheckClearanceFor	Geom01
ClearanceCondition	Found
CurrentResult	1
Enabled	True
FailColor	Red
FillHoles	False
Graphics	All

有关CheckClearanceFor功能下可使用的子对象，请参阅*Vision Guide 7.0 Properties & Results Reference*。



设置了CheckClearanceFor的子对象，无法通过其他对象进行设置。(在本案例中，将Blob01的CheckClearanceFor设置为“Geom01”时，则Blob01将不会显示在Blob02的CheckClearanceFor的下拉列表中。)

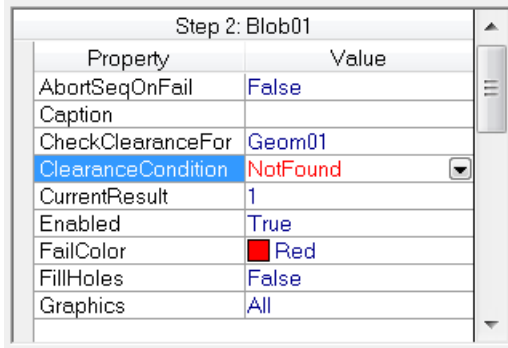
4. 设置ClearanceCondition。

根据子对象的检测见过，设置是否抓取工件。

将ClearanceCondition设置为“NotFound”时，若Found结果为“False”，则子对象的ClearanceOK为“True”，判断的结果是可以抓取。

所ClearanceCondition设置为“Found”时，则结果相反。

在本案例中，需要确认工件的两侧没有其他物体，且有足够空间伸入夹爪，所以将ClearanceCondition设置成“NotFound”。(参考下图)



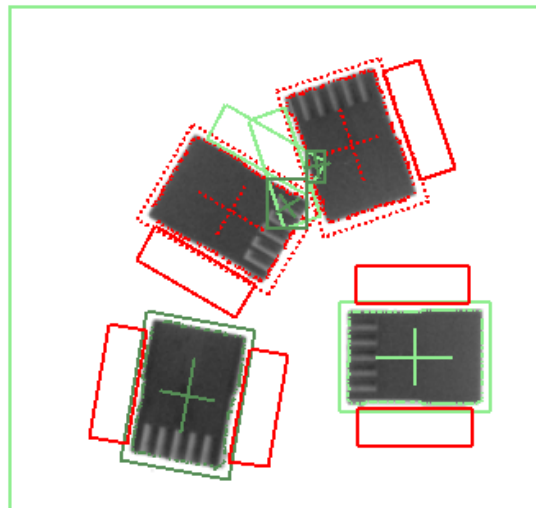
Property	Value
AbortSeqOnFail	False
Caption	
CheckClearanceFor	Geom01
ClearanceCondition	NotFound
CurrentResult	1
Enabled	True
FailColor	Red
FillHoles	False
Graphics	All

3. 运行序列。

在下图中，绿色实线和红色虚线包围的对象，即为父对象Geom01的检测结果。

绿色实线包围的检测结果，是可以抓取的工件。(子对象判断能否进行抓取)

红色虚线包围的检测结果，是无法抓取的工件。



如需确认判断是否抓取工件设置的值，请参考父对象Geom01的ClearanceOK的设置。



NOTE 当使用CheckClearanceFor功能并判断可抓取工件时，请确保执行序列并参考父对象的ClearanceOK设置。并在执行子对象后设置父对象的ClearanceOK。



### 6.2.26 显示/隐藏所有视觉对象标签



#### Force Labels Off(仅适用于视觉工具栏)

当在一个视觉序列中处理的多个视觉对象时，可以使用<Force Labels Off>功能关闭标签，避免屏幕显示过于杂乱。

Vision Guide 工具栏中的<Force Labels Off>是个开启/关闭的按钮。按下按钮时，所选视觉对象以外的标签均会消失，可以清晰的显示选择的视觉对象。

当未按下<Force Labels Off>按钮时，会显示此按钮，且会为图像显示区域中显示的所有视觉对象，显示一个标签。



<Force Labels Off>有时会与<Force All Graphics On>功能一起使用。当按下<Force Labels Off>时，即使按<Force All Graphics On>按钮，也不会显示标签。



如果您在使用视觉序列时，并没有按下<Force Labels Off>按钮，但仍无法看见某个视觉对象，则可能是因为该视觉对象的 Graphics 属性设置为了“none”。所以导致该对象的所有图形都无法显示。

如果当前程序中无视觉序列，<Force Labels Off>按钮将呈现暗灰色，且无法使用。

### 6.2.27 打开所有视觉对象图形



#### Force All Graphics On(仅适用于视觉工具栏)

<Force All Graphics On>按钮是一个快捷键，只需点击该按钮，即可快速显示当前视觉序列中视觉对象的所有图形(搜索窗口、模型原点、模型窗口、直线和标签)。

该按钮的优先度高于单个视觉对象的 Graphics 属性设置，因此您可以快速查看所有视觉对象，而不必单独修改每个视觉对象的 Graphics 属性。



<Force Labels Off>有时会与<Force All Graphics On>功能一起使用。<Force Labels Off>的功能优先。当按下<Force Labels Off>时，即使按<Force All Graphics On>按钮，也不会显示标签。

如果当前程序中无视觉序列，<Force All Graphics On>工按钮将呈现暗灰色，且无法使用。

### 6.2.28 仅显示当前对象



#### Show Only Current Object(仅适用于视觉工具栏)

当序列中具有多个对象时，有时会难以选择和操作所需的对象。此时可通过点击<Show Only Current Object>按钮，仅显示当前视觉对象。如需重新显示所有对象，再次点击<Show Only Current Object>即可。如需仅显示当前对象，可以在对象列表中选择要显示的对象。



## 7. 视觉校准

通过使用视觉校准，您可以将相机和机器人系统之间的位置关系记录到机器人控制器中，并将图像加工系统所确认的对象的图像坐标位置转换为机器人坐标位置。

视觉校准通过以下步骤进行：

- 相机安装
- 为校准创建视觉序列
- 使用校准向导创建校准。
- 点示教
- 进行校准

为高精度执行校准，可执行镜头失真和相机倾斜(作业平面与相机光轴的偏差)校正。

镜头失真和相机倾斜校正通过以下步骤进行：

1. 相机安装
2. 为镜头失真和相机倾斜校正创建视觉序列
3. 执行镜头失真和相机倾斜校正
4. 为校准创建视觉序列
5. 通过校准向导创建校准
6. 点示教
7. 进行校准

### 使用相机的追加校准

- 可检测置于工件平面上的校准板的本地平面。  
(仅垂直 6-轴机器人的第 6 机械臂(J6)安装的可移动式相机。)
- 可以自动检测水平多关节型(SCARA)机器人第 4 轴(J4)或垂直 6-轴机器人第 6 机械臂(J6)安装的可移动式相机的相机安装位置工具坐标。  
此外，可以自动检测作为追加机械臂设定参数的水平多关节型(SCARA)机器人第 2 机械臂(J2)安装的可移动式相机的相机安装位置。
- 可以使用固定向上相机自动检测机器人终端执行器安装的工具的工具坐标。

## 7. 视觉校准

可以通过点击 [Robot Manager] 上的 [Locals]、[Tools] 或 [Arms] 选项卡运行这些校准向导。也可以从相机校准向导执行这些向导。  
无论哪种情况都需要预先创建所需视觉序列。

### 使用相机的本地检测功能配置

硬件配置	使用相机的本地检测功能
紧凑型视觉 CV1	否
紧凑型视觉 CV2-S、CV2-H(固件 2.x.x.x)	否
紧凑型视觉 CV2-SA、CV2-HA(固件 3.0.0.0 或更高版本)	是
计算机视觉系统 PV1	是

上述以外的其他视觉校准(包括使用相机的工具检测和机械臂检测)对于任何硬件配置均可用。

使用利用相机的本地检测功能时需要选装部件校准板。

参考下面的规格表，根据工作表面大小选择合适的校准板。

### 校准板

	大 (CP01-L)	中 (CP01-M)	小 (CP01-S)	超小 (CP01-XS)
外形尺寸(宽×深×厚)[mm]	152×152×2	78×78×2	40×40×2	28×28×2
有效尺寸(宽×深)[mm]	128×128	64×64	32×32	21×21
点距(宽/深)[mm]	4	2	1	0.5
点数	1089(宽 33×深 33)			1849(43×43)
材质	白色玻璃			

## 7.1 相机安装

为每个校准选择一种相机安装方式。相机安装方式不同，校准方式也不同(视觉序列、校准程序要求)。

请注意，错误的设置可能导致校准不当。

Vision Guide 7.0 支持以下相机安装方式：

相机安装	说明
可移动式相机 (第 2 机械臂安装)	相机安装在 SCARA 机器人或直角坐标型机器人的第 2 机械臂上。
可移动式相机 (第 4 机械臂安装)	相机安装在 SCARA 机器人或直角坐标型机器人的第 4 机械臂上。
可移动式相机 (第 5 机械臂安装)	相机安装在 6-轴机器人的第 5 机械臂上。
可移动式相机 (第 6 机械臂安装)	相机安装在 6-轴机器人的第 6 机械臂上。
固定向下相机	相机和目标对象不移动并固定于机器人工作范围。视觉系统使用相机获取机器人坐标系中的位置信息。 如果相机未相对于指定坐标系的 XY 平面垂直安装，必须执行镜头失真和相机倾斜校正。 指定坐标系为机器人底座坐标系和本地坐标系。
固定向上相机	固定相机使其不会移动，并且将机器人的工作区域(部分)囊括在视野中。 例如，该安装方式用于检查机器人运载对象的位置。 校准目标位于终端执行器上或者是机器人把持的对象。
独立相机	相机可以在任何地方进行安装。相机和机器人没有关系。通过该方法，无法读取机器人坐标系中的位置信息。然而，它可以从图像坐标系转换至相机坐标系。即，可以进行简单的长度测量。

提示：

所有相机安装方式均可配置镜头失真和相机倾斜校正。

## 7.2 镜头失真和相机倾斜校正

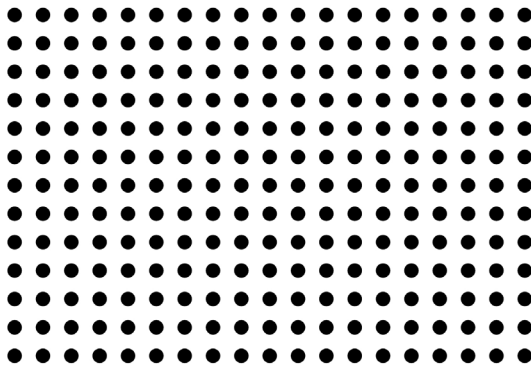
为提高Geometric和Correlation对象的检测成功率，以及以更高精度操作机器人和测量尺寸，进行镜头失真和相机倾斜校正。

定位如下示例所示的正方格子图案，使其占满整个视野，然后执行镜头失真校准。格子图案必须为正方形。也可以使用选装部件校准板。

正方格子图案对于失真校正必不可少。

准备失真尽可能小的网格图案，并将其放置于相机视野中的工作平面上。放置格子图案时，也需考虑从相机的工作距离(WD)。必须将图案放置在与实际检测工件的工件高度相同的高度。

格子图案的角度无需与相机视野对齐。



使正方格子图案占满整个相机视野非常重要。如果执行校准时格子图案只占了视野中的一部分，由于没有格子图案的部分不考虑在内，将无法正确校正失真。

执行校准时，确保格子图案至少100个点位于视野范围内。

完成镜头失真和相机倾斜校正后，后续的相机图像将在拍摄时进行校正，可以获得失真较少的图像。

提示：

执行校准时，可能会因格子图案的拍摄方式而发生错误。这种情况下，通过调整格子图案并再次执行校准，可以正确执行校准。

此外，设定相机时使相机光轴和格子图案之间的角度为45°至90°。如果相机角度为45°度或更小，校准将为错误。

## 7.3 基准点和相机点

基准点是用来校准图像坐标和相机或机器人坐标系关系的重要点。

相机点是用于拍摄检测对象图形的点。

校准是把坐标与基准点关联。基准点和相机点通过在点示教模式下步进机器人配置。

所需基准点和相机点因相机安装类型而异。

使用机器人指定的基准点有以下三种类型。

TaughtPoints(示教点)

EndEffector(终端执行器)

UpwardCamera(向上相机)

可以根据下表所示的相机安装类型选择基准点和相机点。对于不使用相机的独立相机校准，手动将基准点的坐标值(基准点之间的水平和垂直距离)输入系统。

按相机安装类型划分的参考点类型

相机安装类型	可选基准点类型	基准点示教方法	目标检测方法	所需视觉序列
可移动式相机	TaughtPoints	一个点示教	九个相机点示教*	检测一个目标
	UpwardCamera	不需要	九个相机点示教*	检测一个目标
固定向下相机	TaughtPoints	九个点示教	一次拍摄检测九个点	检测九个目标
	EndEffector	不需要	九个相机点示教*	检测一个目标
固定向上相机	EndEffector	不需要	九个相机点示教*	检测一个目标
独立相机	-	指定九个点坐标	一次拍摄检测九个点	检测九个目标

\* 可使用相机点自动生成。(参见以下说明)

### 基准点示教方法

基准点类型设为“TaughtPoints”时步进机器人，使工具位置和目标匹配，示教一或九个点。

如何设定可移动式相机的基准点：

参见：7.3.1 可移动式相机基准点

如何设定固定向下相机和独立相机的基准点：

参见：7.3.2 固定向下相机和独立相机的基准点

### 九个相机点示教

使用“九个相机点示教”检测目标时，每次拍摄图像时检测一个点，共检测九个点的位置。

校准过程中，移动目标至相机视野中指定的九个区域并拍摄检测对象的图像。

在点示教模式下，确认图像的同时步进机器人至示教点，以便在每张图像中的适当位置检测到九个相机点。

如何创建序列：

参见：*7.4.1 检测一个目标的视觉序列*

### 相机点自动生成

使用“九个相机点示教”检测目标时，可使用相机点自动生成。

通过在视野中心仅示教一个点自动生成八个相机点，而非示教九个相机点。

**NOTE:** 使用相机点自动生成时，校准过程中机器人自动移动生成相机点。注意机器人与外围装置的相互干扰。此外，为防止相机点自动生成过程中发生错误，应避免使用各轴延伸的奇异点附近的方向。

### 一次拍摄检测九个点

使用“一次拍摄检测九个点”校准时，一次拍摄即检测九个目标的位置。

必须预先创建使用相机检测九个目标的视觉序列。

如何创建序列：

参见：*7.4.2 检测九个目标的视觉序列*



### 7.3.1 可移动式相机基准点

该方案需要一个基准点。

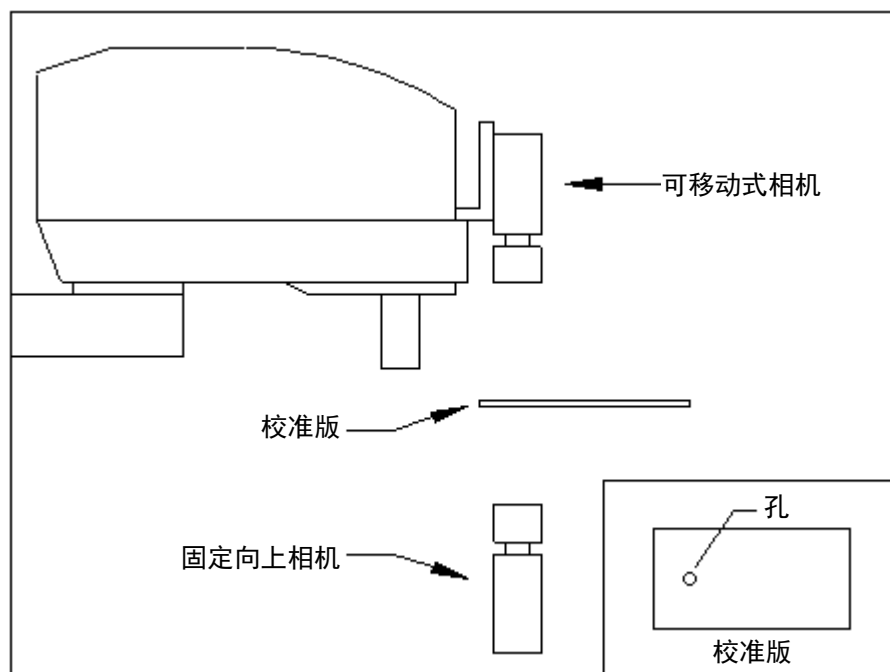
基准点，可以是步进机器人示教的点(基准点类型=示教点)，或者使固定向上相机检测出来的点(基准点类型=向上相机)。后者的精度较高且可以自动执行，因此请尽可能选择固定向上相机检测到的点。

下面是示教基准点的一些例子：

- 机器人工作范围中的一个零件或一个校准目标。
- 机器人工作范围中某处的一个孔，以便安装在机器人终端执行器上的工具可以滑入。

当使用固定向上相机来寻找基准点时，需要一个带孔的薄板，可移动式相机和固定向上相机均可看见这个孔。在校准过程中，固定向上相机(之前已经过了校准)定位薄板上的基准点孔。然后通过搜寻九个位置中的基准孔来对可移动式相机进行校准。

利用固定向上相机可以更加精确地寻找可移动式相机校准的基准点，因为在校准固定向上相机时，每个相机点的机器人工具旋转  $180^\circ$ ，因此可以确定机器人 U 轴的精确中心。这使得校准后的固定向上相机能够更加精确地找到可移动式相机在校准过程中将定位的基准孔。

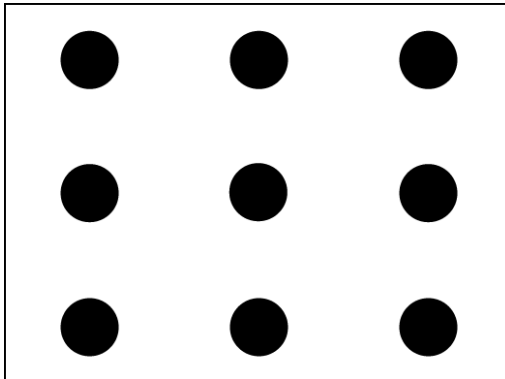


固定向上相机用于校准可移动式相机

### 7.3.2 固定向下相机和独立相机的基准点

基准点类型设为“TaughtPoints(示教点)”的以下校准方案需要包含九个目标的校准目标板或靶纸。下图展示了九个校准目标的图案。

固定向下相机  
独立相机



固定向下相机和独立相机的校准目标

对于固定向下相机校准，目标可以是板上的孔，便于机器人终端执行器上的杆滑入。目标之间的距离不必很精确。当示教了点之后，校准软件将读取机器人位置。

对于独立相机校准，可以使用纸或聚酯薄膜等上面的图案。必须知道目标之间的垂直和水平距离。当您通过 [Calibration] 对话框示教这些点时，需要输入这些数据。

### 7.3.3 通过TwoRefPoints示教

示教可移动式相机校准或固定向下相机校准的基准点时，系统将提示您示教该点，然后任意地从第一个位置旋转 180°，再次示教该点。这使得系统能够更加精确地确定机器人坐标系中基准点的位置。然而，如果您正在使用的是已经在校准设置中准确定义和指定了的机器人工具，那么您可以跳过旋转 180° 再次示教该点这一步骤。

若要跳过旋转 180° 的步骤，应确保 TwoRefPoints 校准属性设为 False。

关于定义工具的详细内容，请参阅以下内容。

11. 在 SPEL+ 语言中使用 Vision Guide 7.0 中定义一项工具。

## 7.4 为校准创建视觉序列

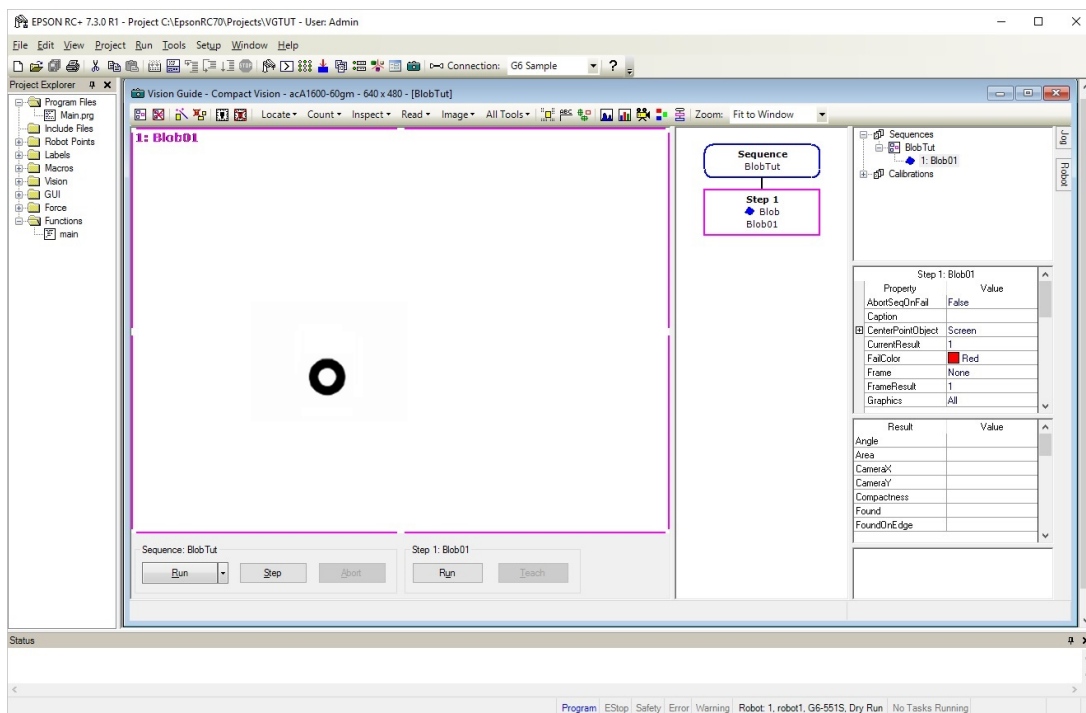
对相机进行校准之前，您必须创建一个可以找到校准目标的视觉序列。

### 7.4.1 检测一个目标的视觉序列

使用可移动式相机、固定向上相机或基准点类型为“EndEffector”的固定向下相机时，序列将在整个视野内搜索一个校准目标。

保证将搜索窗口的大小确定为与整个图像显示区域一样大。

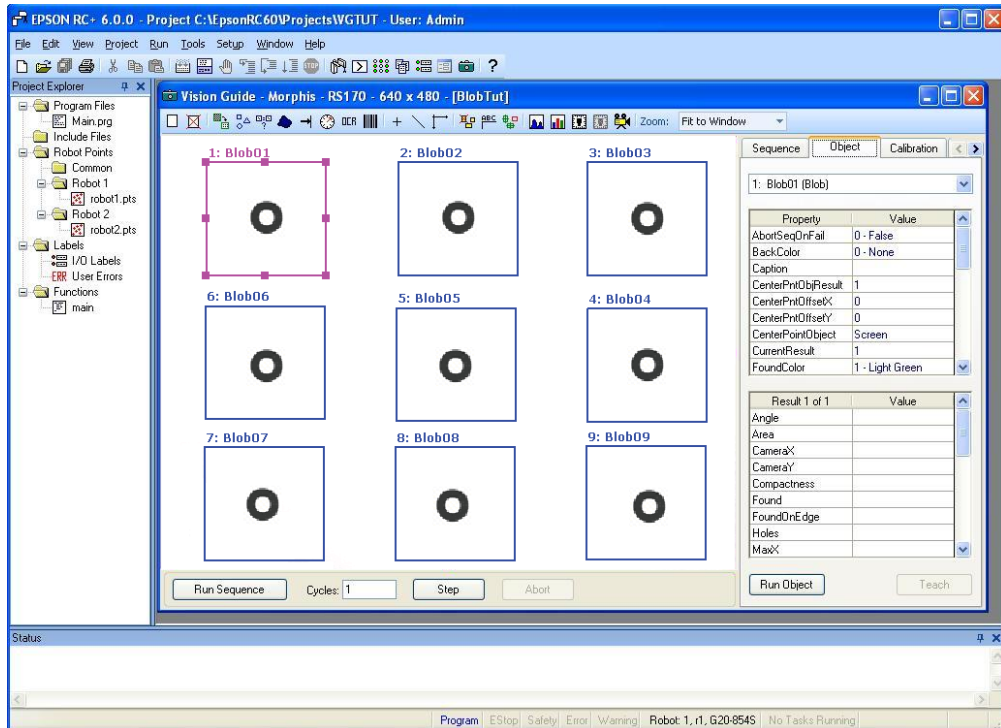
序列中最后一个对象的 X 和 Y 结果将在校准过程中由校准软件所使用。有关可用的对象，参见 *Vision Guide 7.0 Properties and Results Reference* 的 *PixelXYU* 结果。



## 7.4.2 检测九个目标的视觉序列

基准点类型为“TaughtPoints”的固定向下相机或独立相机校准所需的视觉序列必须定位九个校准目标。可以采用两种方式进行。

1. 为每个目标创建一个对象，总共创建九个对象。
- 或
2. 利用其中一个多结果对象(Blob 对象或 Correlation 对象等)来返回九个结果。将 NumberToFind 设置为“9”，以便返回九个结果。



为独立相机校准而放置视觉对象

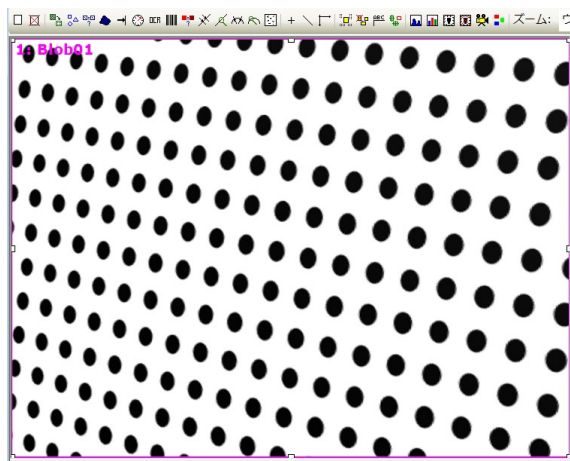
使用上述第一个方法时，将对象放置在图像窗口，放置顺序必须与目标的示教顺序相同。第一个对象将放在左上角，第二个放在上方中央的位置，第三个放在右上角，等等，依此类推。

对于固定向下相机校准，中间一排的对象的顺序是相反的：6、5、4。这是因为以 1、2、3、6、5、4、7、8、9 的顺序移动到各个位置会在机器人上产生出最有效的动作，并更快地完成校准。

### 7.4.3 用于失真校正的视觉序列

要执行失真校正，必须预先按照以下步骤创建视觉序列。

1. 设定正方网格图案。
2. 创建检测正方网格图案的序列。
3. 根据需要选择Lamp属性。
4. 创建 Blob 对象，然后将搜索区域扩大至整个窗口。
5. 将 NumberToFind 属性设为“All”。
6. 将 RejectOnEdge 属性设为“True”。
7. 设定 ThresholdAuto、ThresholdHigh、ThresholdLow、MinArea、MaxArea 和其他属性以检测点。
8. 执行创建的序列。确认检测到的点是否已超过 100。



带倾斜角度的相机拍摄的正方网格图案图像示例

### 7.4.4 用于本地、工具和机械臂设定的视觉序列

要使用相机检测校准板的本地坐标，需要用于识别校准板的视觉序列。


设定序列的属性(ExposureTime等)，以便清晰识别选装部件校准板上的点。不需要视觉对象。

要执行使用相机的工具或机械臂设定的校准时，需要检测校准目标位置的视觉序列。按照与 7.4.1 检测一个目标的视觉序列 相同的方法创建含有检测一个目标位置的视觉对象的序列。

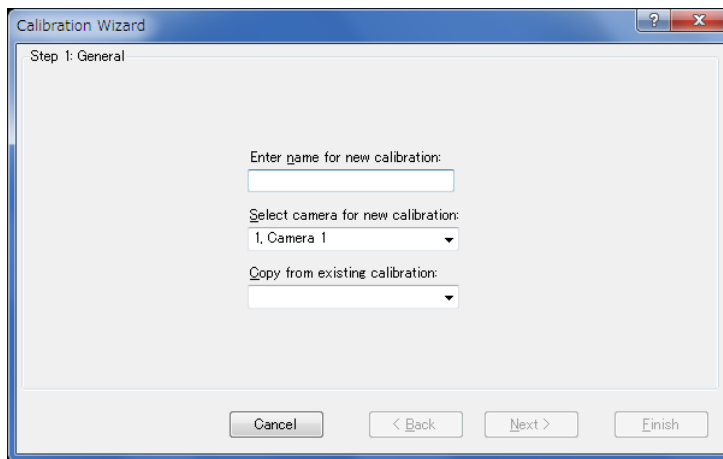
## 7.5 校准图形用户界面

本节将介绍校准使用的图形用户界面。

### 7.5.1 创建一个新的校准

要创建一个新的校准，点击 Vision Guide 工具栏上的  <New calibration>按钮。

出现以下对话框，开始校准向导。



输入新校准的名称。选择要校准的相机后，可以通过指定[Copy from existing calibration]复制一个现有的校准。

输入校准名后，可以在任意画面中点击<Finish>按钮退出向导。这种情况下，参见 7.5.3 校准属性和结果 设定所需属性。

也可以点击<Back>按钮修改之前的设定。

点击<Next>按钮移到下一步骤。

如果名称已被另一个校准使用，将出现报错信息。

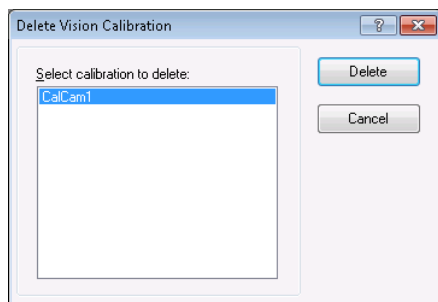
遵循各个步骤的指示完成向导。设定因相机位置和方向而异。有关设定的详细内容，参见 7.6 校准程序。

完成向导后，向导中配置的校准将添加至校准目录树。在目录树中选择了校准时，显示校准的属性和结果。向导中配置的项目可随时在属性面板中更改。

### 7.5.2 删除校准

点击 Vision Guide 工具栏上的  <Delete calibration>按钮。

出现以下对话框：



选择您想要删除的校准并点击<Delete>按钮。

### 7.5.3 校准属性和结果

校准向导中配置的设定反映至校准属性。创建校准后，可以在 Vision Guide 窗口的校准窗口中更改校准属性。

属性	说明
ApproachPoint	设定接近点作为校准中移动至各相机点时的起点。
AutoCamPoints	自动生成相机点。
AutoReference	该属性指定校准可移动式相机时是否自动计算校准基准点。
AutoRefFinalRotation	自动计算基准点时，设定工具旋转最终角度。
AutoRefInitRotation	自动计算基准点时，设定机械臂和工具旋转最初角度。
AutoRefMode	该属性将基准点的自动计算模式指定为Rough、Fine或Manual。使用Fine时，大幅移动机器人并以更高精度计算。使用Manual时手动输入旋转角度和容差值。
AutoRefMoveMode	指定自动计算基准点时机器人的动作模式 (仅6轴机器人)。
AutoRefTolerance	设定自动计算基准点时视觉检测偏差的容差度。
Camera	为当前选择的校准指定相机。

属性	说明
CameraOrientation	<p>选择相机的方向。有如下七个选择：</p> <p><b>Mobile J2 (可移动式相机: 安装在机器人第2轴上)</b> 安装在水平多关节机器人(SCARA)的第2机械臂，或直角坐标机器人的第2轴上。</p> <p><b>Mobile J4 (可移动式相机: 安装在机器人第4轴上)</b> 安装在水平多关节机器人(SCARA)的第4机械臂，或直角坐标机器人的第4轴上。</p> <p><b>Mobile J5 (可移动式相机: 安装在机器人第5轴上)</b> 安装在垂直型6轴机器人的第5机械臂上。</p> <p><b>Mobile J6 (可移动式相机: 安装在机器人第6轴上)</b> 安装在垂直型 6 轴机器人的第 6 机械臂上。</p> <p><b>Fixed downward (固定向下相机)</b> 相机无法移动，返回机器人坐标系中的位置信息。</p> <p><b>(固定向上相机)</b> 相机无法移动，返回机器人坐标系中的位置信息。</p> <p><b>Standalone (独立相机)</b> 相机无法移动，返回相机坐标系的位置信息，而不是机器人坐标系中的位置信息。</p>
DistCorrectCal	该属性执行镜头失真和相机倾斜校正。
DistCorrectEnable	该属性启用/禁用镜头失真和相机倾斜校正。
DistCorrectTargetSeq	该属性指定用于镜头失真和相机倾斜校正的视觉序列。 可使用当前项目中的任意序列。
DistCorrectType	设置要使用的失真校正类型。
Lamp	此为可选的输出通道，在校准开始时会自动打开。
LampDelay	此为在开灯之前的延时时间(秒)。这考虑到了荧光灯开启所需的时间。
LJMMode	设定点数据的方向标志控制模式。
MaxMoveDist	指定机械臂端部的移动距离限制。
MotionDelay	此为每个校准周期中，每个机器人动作后的延时时间(毫秒)。
PointsTaught	该属性确定是否已经示教了校准点。
ReferenceType	<p>选择基准点，指定示教时使用的基准点的类型。</p> <p>示教移动式相机时，有两种方法可以指定基准点。一种是使用示教点作为基准点，另一种是使用固定向上相机获取的点作为基准点。</p>
RobotArm	<p>该属性规定了校准过程中使用的机械臂的编号。通常设置为零。</p> <p>通常设置为“0”。</p>
RobotAccel	此为校准过程中使用的加速度。



属性	说明
RobotLimZ	该属性规定了 Scara 可移动式相机校准周期内第一个动作命令的 LimZ 值。
RobotLocal	该属性规定了当前校准使用的本地编号。所用的本地必须事先确定好。
RobotNumber	该属性规定了当前校准使用的机器人编号。
RobotSpeed	该属性规定了校准过程中机器人移动的速度。要读取更高速度，您需要设置更高的加速度。
RobotTool	该属性规定了校准过程中使用的工具编号。所用的工具编号必须事先确定好。
RobotXOffset	在机器人坐标系中，为检测到的零件 X 坐标位置设置偏移值。
RobotYOffset	在机器人坐标系中，为检测到的零件 Y 坐标位置设置偏移值。
RobotXYRotateOffset	指定是否根据角度结果旋转 XY 偏移值(RobotXOffset, RobotYOffset)。
RobotUOffset	此属性设置了所检测到的零件在机器人坐标系中U坐标位置的偏移值。
ShowConfirmation	此属性设置了是否显示校准结果对话框，以便操作员在运行时接受。
TargetSequence	该属性指定了使用哪个视觉序列进行校准。该序列可以是当前项目中的任何序列。
TwoRefPoints	该属性指定是否在校准中使用两个基准点。
UpwardLamp	当固定向上相机序列开始执行时，自动打开任意的 I/O 输出位。
UpwardSequence	指定用于校准固定向上相机时，使用的视觉序列。 仅在执行使用固定向上相机捕获的点作为基准点，示教可移动式相机时，此列表才有效。
CalComplete	校准的完成状态。
CallImageSize	执行校准时的图像尺寸。
DistCorrectCalComplete	镜头失真和相机倾斜校正是否完成的状态。
FOVHeight	视野的高度(单位: mm)。
FOVWidth	视野的宽度(单位: mm)。
XAvgError	相机坐标 X 方向的平均偏差。
XMaxError	相机坐标 X 方向的最大偏差。
XmmPerPixel	相机坐标 X 方向的，每个像素的长度(mm)。
XTilt	相机坐标 X 方向的倾斜。
YAvgError	相机坐标 Y 方向的平均偏差。
YMaxError	相机坐标 Y 方向的最大偏差。
YmmPerPixel	相机坐标 Y 方向的，每个像素的长度(mm)。
YTilt	相机坐标 Y 方向的倾斜。

## 7.5.4 失真检测

为执行失真校正，使用校准属性检测失真。失真检测需要在执行其他视觉校准的点示教前执行。

检测失真的步骤如下所示。在校准向导中配置了失真校正时，可以跳过步骤(1)。

Calibration: calib1	
Property	Value
Index	1
DistCorrect	
Enabled	True
TargetSeq	seq1
Cal	Click to calibrate ->
Lamp	None
LampDelay	0 sec

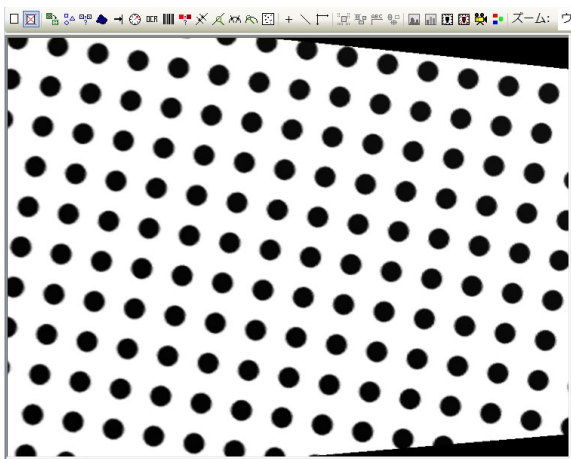
- (1) 将 DistCorrect 属性下的 [Enabled] 设为“True”，然后在 TargetSeq 属性中选择检测正方网格图案的序列。

如何创建序列：

参见：7.4.3 用于失真校正的视觉序列

- (2) 选择 DistCorrect 属性下的 Cal 属性，执行正方网格图案检测。

将执行了失真检测的校准指定为视觉序列的“Calibration”后执行序列，可确认校正失真后的图像。



校正镜头失真和相机倾斜后的图像示例

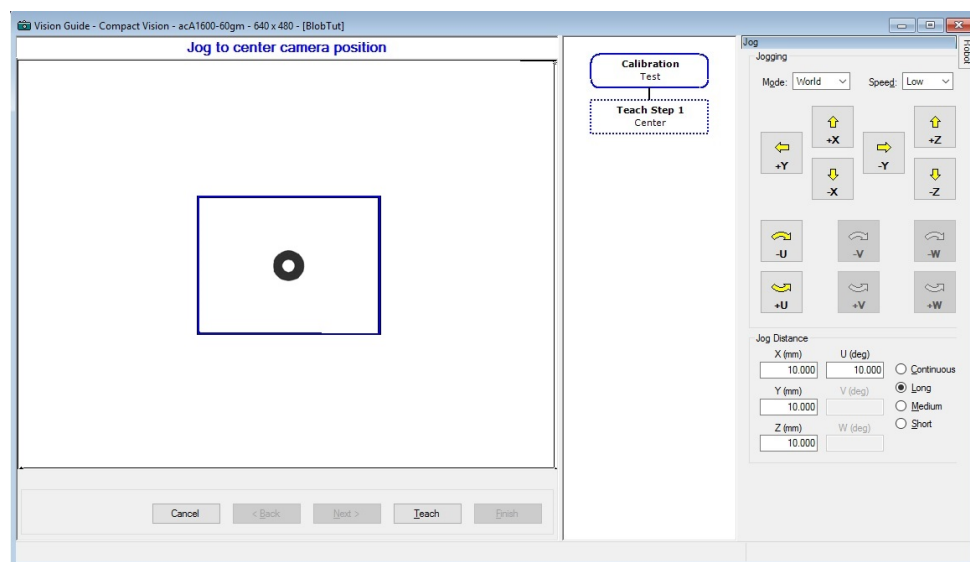
### 7.5.5 点示教

在Vision Guide窗口的序列或校准目录树中选择校准后，将出现<Teach Points>按钮。点击<Teach Points>按钮将Vision Guide窗口切换到 [Teach Calibration Points] 模式，可以为当前所选择的校准示教点。

[Teach Calibration Points] 窗口用于为当前选择的校准方案示教校准点。

示教的点因校准设定而异。

根据校准设定出现所需点示教面板。按照画面说明步进机器人并示教点。



[Teach Calibration Points]对话框

图像显示在窗口左侧。点击右端的[Jog]选项卡打开步进窗口。

在显示器的底部有一个消息框用于显示指令。执行了要求的指令以后，点击位于消息框右侧的<Next>按钮来继续下一个步骤。完成了所有步骤以后，会出现一个消息框指示您已完成。

显示的指令随着示教点的校准类型的不同而各异。

指令	说明
步进到固定基准点位置	对于使用已示教点作为基准的可移动式相机校准(与使用固定向上相机来自动搜寻基准点相反)而言, 该指令是必需的。步进机器人, 直到终端执行器与校准目标对准。
旋转 180° 步进到固定基准点位置	此为一项可选步骤。首先将 <b>TwoRefPoints</b> 属性设为 <b>True</b> , 然后再对点进行示教, 以启用此功能。如果您正在使用已经精确定义的校准工具, 则可以将 <b>TwoRefPoints</b> 设为 <b>False</b> 。如果当前未使用校准工具, 则将 <b>TwoRefPoints</b> 设为 <b>True</b> 。将提示您从当前位置步进机器人 180°, 将终端执行器与校准目标对齐。
步进到左上相机位置	步进机器人, 直到校准目标位于视觉显示区域的左上角。
步进到上中央相机位置	步进机器人, 直到校准目标位于视觉显示区域的中央上方位置。
步进到右上相机位置	步进机器人, 直到校准目标位于视觉显示区域的右上角。
步进到右中央相机位置	步进机器人, 直到校准目标位于视觉显示区域的中央偏右位置。
步进到中央相机位置	步进机器人, 直到校准目标位于视觉显示区域的中央位置。
步进到左中央相机位置	步进机器人, 直到校准目标位于视觉显示区域的中央偏左位置。
步进到左下相机位置	步进机器人, 直到校准目标位于视觉显示区域的左下角。
步进到下中央相机位置	步进机器人, 直到校准目标位于视觉显示区域的中央偏下位置。
步进到右下相机位置	步进机器人, 直到校准目标位于视觉显示区域的右下角。

### 7.5.6 示教校准点

在序列或校准目录树中选择校准后，将出现<Calibrate>按钮。点击<Calibrate>按钮开始校准。

如果将要移动机器人，在操作开始前会显示一条确认消息。

### 7.5.7 校准完成对话框

完成一个校准周期后，出现如下所示的校准完成对话框。

该对话框显示当前校准和上一次校准的校准值概要。

如果这是第一次校准，则之前的校准值为空白。

校准完成后出现的校准状态对话框

下表描述了 [Calibration Complete] 对话框的显示值。

检查完这些结果以后，点击<OK>以使用新的校准值或点击<Cancel>放弃。

**提示：** 如果误差大于1mm或者倾斜值大于1，系统将提示并未正确完成校准。检查在校准循环过程中是否正确地检测到了校准点，并检查机器人校准点和基准点在校准时是否对准。要在运行时禁用此对话框，可将ShowConfirmation属性设为False。

**NOTE：** 在使用远心镜头时，可能会显示异常倾斜值。

## 7. 视觉校准

---

参考值	说明
X mm per pixel, Y mm per pixel	相机的分辨率(每个像素的平均宽度和高度)
Max X error, Max Y error	校准期间产生的最大偏差。 每个像素的最大偏差不能超过 1mm。如果最大偏差超过 1mm, 则可能是由于基准点示教有误, 或是示教不当、照明条件不好, 从而导致校准目标的位置不一致。
Avg X error, Avg Y error	校准期间产生的平均误差。
X tilt, Y tilt	表示相机的相对倾斜。 倾斜方向是相机在图像缓冲区坐标中获取的方向。即, X 倾斜时, 正值代表向右边倾斜, 负值代表向左边倾斜。Y 倾斜时, 正值代表向下方倾斜, 负值代表向上方倾斜。 Vision Guide 7.0 具有相机倾斜补偿功能, 但是请确保倾斜值小于 1.0。倾斜值小于 1.0 可以提高重复测量精度, 准确的检测到工件。 当倾斜值过高时, 往往是由于未能正确示教基准点导致的。例如, 进行独立相机校准时, 需要输入校准目标的坐标, 如果输入的坐标值不正确, 则可能导致倾斜值过高。
FOV	相机视野的宽度和高度(单位: 毫米)。

## 7.6 校准程序

本节包含用来校准每种相机安装方式的步骤说明。

### 7.6.1 校准程序：可移动式相机

通过安装在机器人上的相机搜索对象，然后读取对象在机器人坐标系中的位置。

#### 准备1：将相机安装至机器人

- 将相机安装至机器人。

您可以在任何旋转角度上安装相机。相机必须与您将使用的本地坐标系的 Z 轴垂直对准。

#### 准备2：选定基准点类型

设定基准点有以下三种选项：

自动基准

使用已手动示教点

使用固定向上相机找到的点

要使用自动基准，将 `AutoReference` 属性设为 `True`。将 `AutoRefMode` 设为 `Fine` 可获得最佳精度。

**注意：** 通过使用自动基准，执行校准时自动移动机器人并获取基准点坐标。注意机器人与外围装置的相互干扰。使用 `Fine` 模式进行 SCARA 机器人第 2 轴安装的可移动式相机校准时需特别注意，因为校准过程中机器人方向会大幅移动（切换左终端执行器和右终端执行器的方向）。此外，为防止自动基准功能处理过程中发生错误，应避免使用各轴延伸的奇异点附近的方向。

为达到最大的准确性，您应使用一台固定向上相机来寻找基准目标。

具体程序请参见本章后面的 **7.6.3 校准程序：固定向上相机**。

如果没有使用自动基准或固定向上相机寻找基准点，则使用下面的方法之一来示教基准点：

- 使用一个安装在 U 轴并穿过轴心的连杆。

- 如果用于拾取零件的工具可以同校准基准点对准，则使用该工具。

如果您(使用 `TLSet` 命令)确定一个工具用于拾取工具的校准杆，则您在校准过程中将不必示教零和  $180^\circ$  基准点。

### 准备3: 创建寻找点网格图案的视觉序列(执行失真校正时)

- (1) 创建校准板。
- (2) 参见以下章节创建视觉序列。  
*7.4.3 用于失真校正的视觉序列*

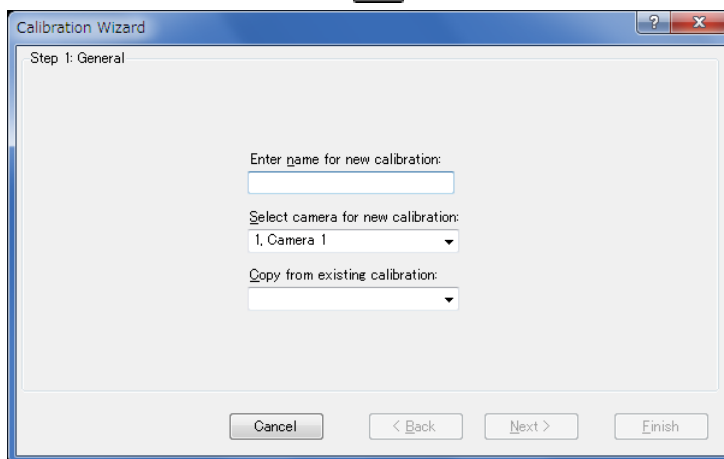
### 准备4: 创建视觉序列来寻找校准基准目标

- (1) 参见以下章节创建视觉序列。  
*7.4.1 检测一个目标的视觉序列*

完成准备后，运行校准向导配置所需设定。

### 第1步: 运行校准向导

点击Vision Guide工具栏上的 <New Calibration>按钮运行校准向导。

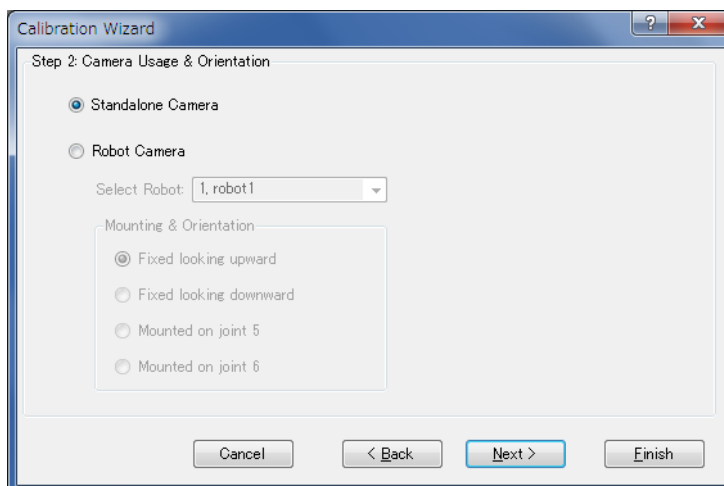


在 [Enter name for new calibration] 中输入校准的名称。

在 [Select camera for new calibration] 中选择校准用相机。

可以通过在 [Copy from existing calibration] 中选择源校准数据复制设定。

点击<Next>按钮进入步骤2。



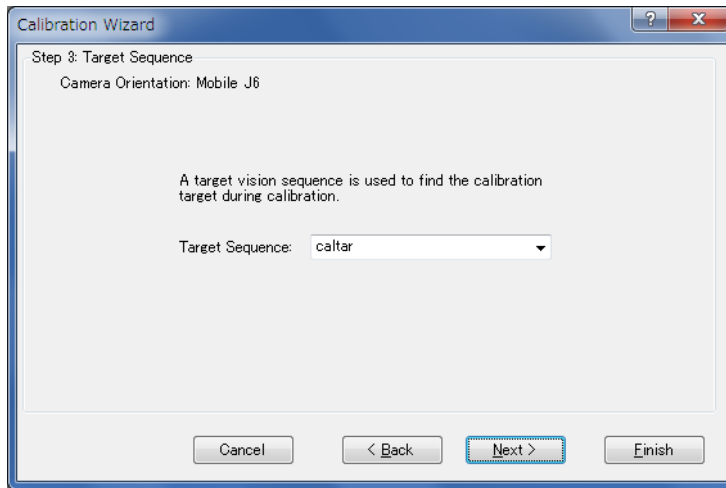




### 第3步：指定目标序列

指定目标序列。

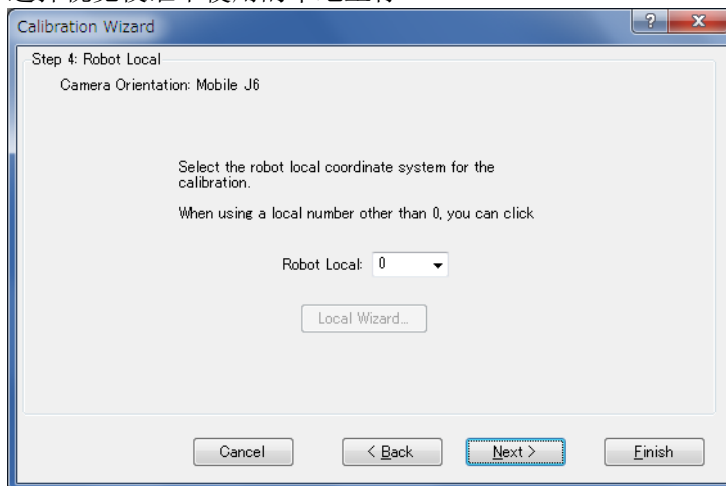
从列表中选择在 *准备4：创建视觉序列来寻找校准基准目标* 中创建的视觉序列。在该步骤中，列表中仅显示步骤1中指定相机的视觉序列。



选择目标序列后，点击<Next>按钮。

### 第4步：设定本地

选择视觉校准中使用的本地坐标。

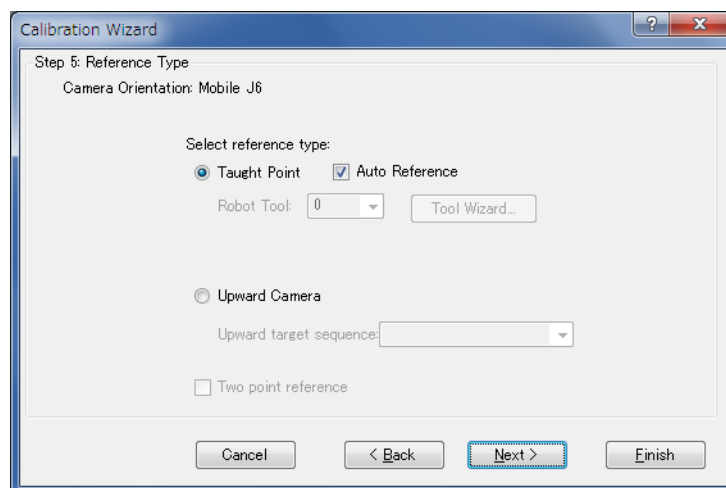


在该对话框中，可以运行本地向导将本地坐标系定义至本地编号。要运行向导，选择除“0”以外的本地编号，然后点击 <Local Wizard...>按钮。使用相机的本地设定详细内容在 *7.7 使用相机的本地设定* 中说明。

选择本地坐标后，点击<Next>按钮。

### 第5步：设定基准点类型

设定校准中使用的基准点类型。



用于校准的基准点是通过机器人坐标指定的点。

选择<Taught Point>时：

如果勾选Auto Reference，将自动寻找基准。否则需要步进机器人并手动示教基准点。

必须指定作为机器人机械臂末端安装的终端执行器基准点的工具编号和机械臂编号。(追加机械臂设定仅SCARA机器人可用。)

勾选[Auto Reference]复选框可省略第2、4或6中任意一个轴上安装的相机的工具和机械臂设定。如果跳过，将自动执行机械臂和工具计算，并自动指定基准点。

在该步骤中，也可以通过选择除“0”以外的工具编号后点击<Tool Wizard...>按钮，运行工具向导并定义工具。

此外，可以通过选择除“0”以外的机械臂编号后点击<Arm Wizard...>按钮，运行机械臂向导并定义相机的机械臂。

如何定义相机工具和机械臂。

参见：7.8 检测可移动式相机安装位置

提示：执行校准时，如果通过Auto Reference功能省略基准点示教后无法获得所需校准精度，不使用Auto Reference功能执行校准可能可以提高精度。

选择<Upward Camera>时：

可以使用已校准的固定向上相机精确检测基准点。

如果基准点类型选择<固定向上相机>，还要在[Upward target sequence]中选择用于检测的视觉序列。

列表中仅显示指定了所选相机的视觉序列。

选择[Two point reference]时：

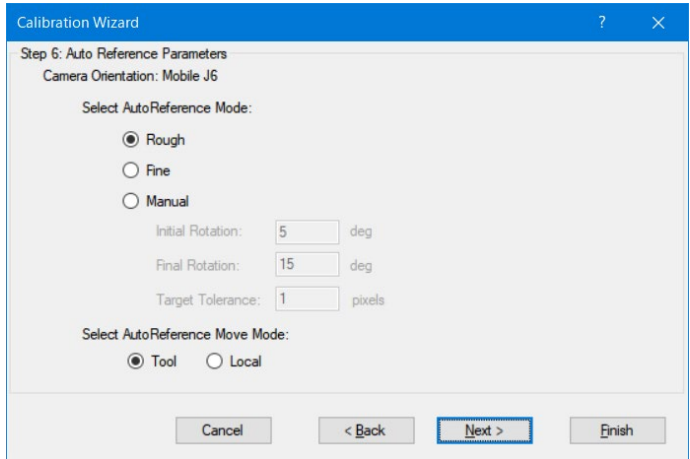
将U旋转180°的第二个基准点将用于校准。

使用工具时，可以省略该设定。

设定基准点类型后，点击<Next>按钮。

**步骤6: AutoReference参数设定**

设定自动寻找基准点时使用的参数。如果在步骤 5 中勾选[Auto Reference]复选框, 将显示该步骤。



设定自动寻找基准点时使用的模式。请注意，机器人移动因相机安装位置而异。

**选择<Rough>时:**

- 用于机器人移动较小的粗略定位。
- 对于MobileJ2相机，该设定将以较小增量移动机械臂。
- 对于MobileJ4或J6相机，该设定将以较小增量旋转工具。

**选择<Fine>时:**

- 用于机器人移动较大的精确定位。
- 对于MobileJ2相机，机械臂将随机器人右/左方向变化移动。
- 对于MobileJ4或J6相机，该设定将以较大增量旋转工具。

**选择<Manual>时:**

- 通过该设定用户可以手动输入机器人移动角度和目标容差。即使选择了<Rough>仍感觉机器人移动距离过大时，可选择<Manual>减小机器人移动。机械臂不随右/左方向变化移动。

使用MobileJ6时，可以设置自动寻找参考点时的动作模式。

**选择<Tool>时:**

- 机器人在工具0坐标系的XY平面上动作。安装相机时，应使光轴与工具0坐标系的Z轴方向(垂直于第6关节法兰面)大致平行。

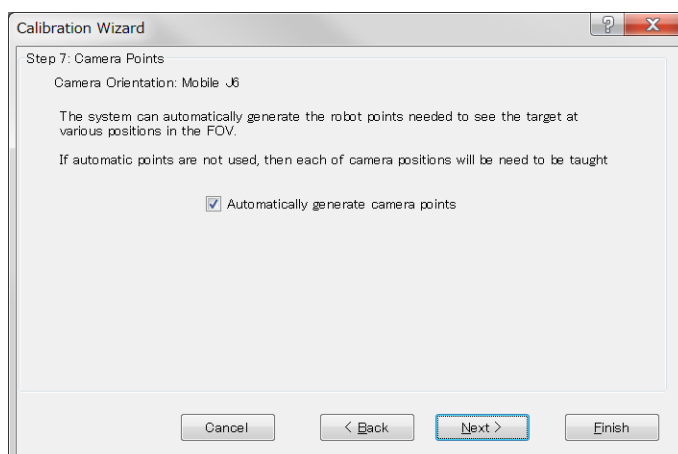
**选择<Local>时:**

- 机器人在Sptep4中指定的本地坐标系的XY平面上动作。与选择Tool时不用的是，相机可以安装在任意角度。但是需要制定本地坐标系，本地坐标系的XY平面应与相机的图像平面大致平行。

设定完成后，点击<Next>按钮。

### 步骤7：相机点设定

指定执行校准时是否自动生成相机点。



选择[Automatically generate camera points]时：

通过在相机FOV中检测目标对象的同时自动操作机器人，自动生成多个相机点。对各相机点检测FOV中目标对象的位置。使用自动生成时，必须示教目标对象放置在FOV中心附近的一点。

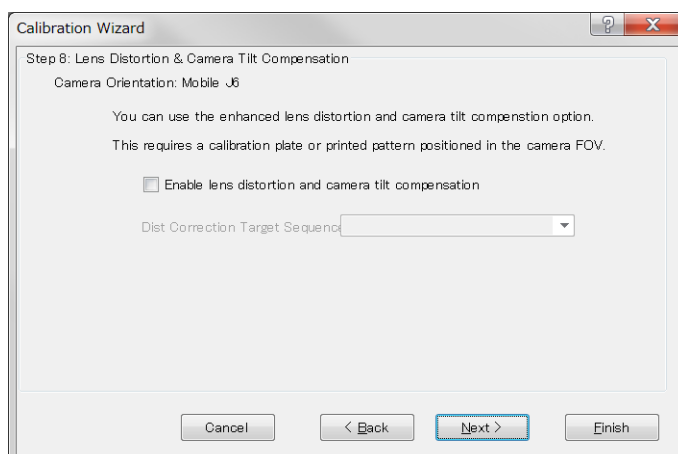
未选择[Automatically generate camera points]时：

执行校准前必须步进机器人手动示教所需数量的相机点。

设定自动相机点生成后，点击<Next>按钮。

### 步骤8：镜头失真和相机安装歪斜校正设定

指定是否校正镜头失真和相机安装歪斜。

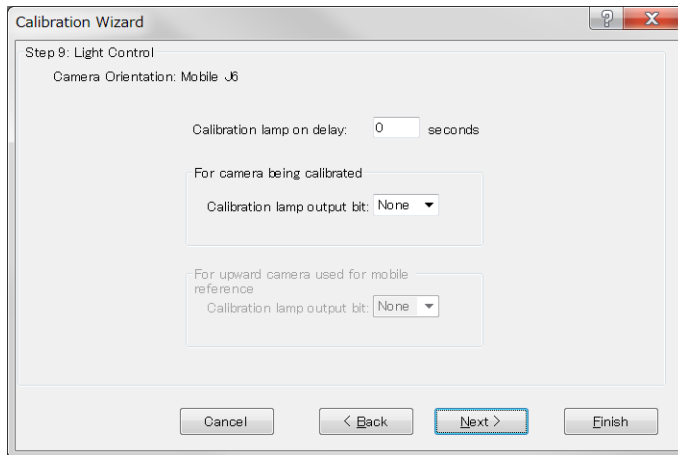


勾选复选框启用校正。要校正失真，必须预先创建失真校正用目标序列并在此步骤指定。

设定失真校正后，点击<Next>按钮。

### 步骤9：照明控制设定

设定用于校准的照明控制。如果无需照明控制，则无需更改设定。



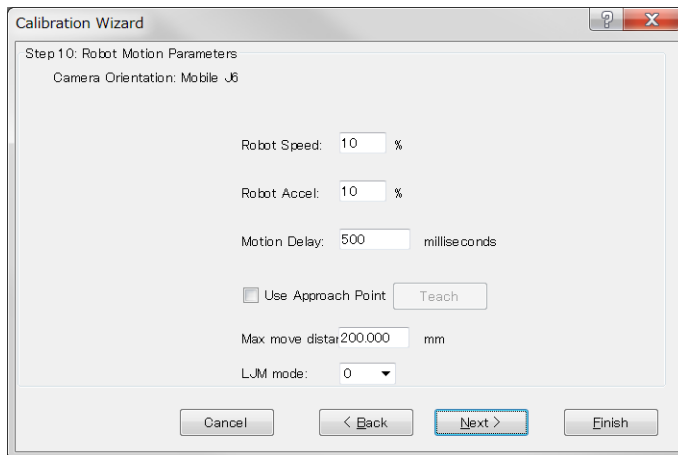
如果为相机使用照明，以毫秒为单位指定照明开启前的等待时间。此外，指定开启照明的输出位。

如果步骤5中基准点类型选择了<Upward Camera>，还可以指定开启固定向上相机照明的输出位。

设定照明控制后，点击<Next>按钮。

### 步骤10：机器人动作设定

配置机器人动作设定。



设定速度和加速度以及机器人动作后的调整时间(拍摄图像前的等待时间)。为执行精密校准，设定较慢速度和加速度并确保足够的调整时间。

也可以指定接近点。

如果指定了接近点，机器人始终从指定的接近点移动至校准点。

这样可以使机器人从固定方向接近校准点，机器人位置稳定。

如何配置接近点：

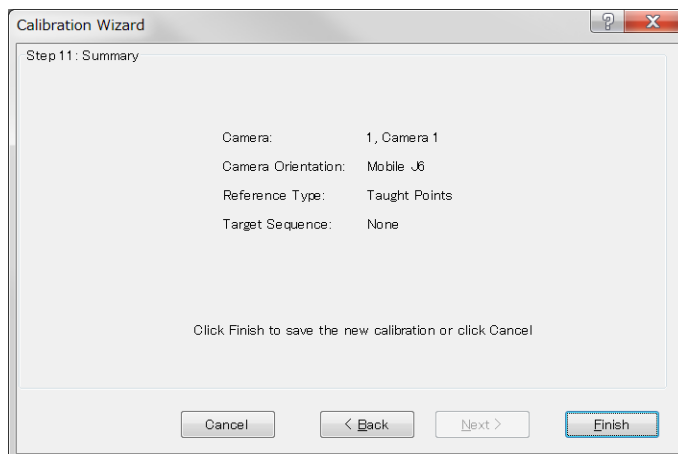
勾选[Use Approach Point]复选框，然后点击<Teach>按钮。

在显示的点示教对话框中示教接近点。

设定接近点后，点击<Next>按钮。

**步骤11：确认设定**

显示已配置的项目。确认设定。



点击<Finish>按钮完成向导。

**检测失真校正(失真校正启用时)**

- (1) 在序列或校准目录树中选择已创建的校准。
- (2) 在工件平面放置点网格图案。
- (3) 选择属性列表中 DistCorrect 属性下的 Cal 属性，执行点网格图案检测。

通过在Calibration属性(校准目标定位用序列)中选择已创建的校准方案，可以确认已校正镜头失真和相机倾斜的图像。即使省略该设定，执行校准时将自动校正失真。

**示教点**

- (1) 点击<Teach Points>按钮。  
出现[Teach Calibration Points]对话框。
- (2) 遵循显示在对话框底部的消息框中的指示。  
所需相机点数量和基准点示教的详细内容因相机安装和基准点类型以及其他设定而异。

- (3) 示教相机点。

如果启用自动相机点生成，示教一个相机位置。

如果禁用自动相机点生成，示教九个相机位置。

相机位置表示九个机器人位置。示教第一个点使目标到达左上角，然后示教第二个点使目标到达图像显示区域中心。在视野中相应位置示教余下各点。为获得最佳结果，示教点时使其分散于整个视野范围。

使用垂直6-轴机器人时，通常相机点的V坐标将为零，W坐标将为零或180，具体取决于本地坐标系的方向。示教校准点时无需更改V坐标和W坐标。这是为将相机位置设定在相对于相机点本地的一条直线上。

- (4) 基准点类型选择了“Teach Points”时。

示教基准点。

将基准点示教至机器人，使拾取工件用工具到达工件正上方。(这与作为机器人动作目标的示教点相同。)

### 校准

点击<Calibrate>按钮开始校准周期。

机器人移动到每个相机位置并执行校准目标视觉序列。

移动到全部九个位置后，系统确定校准参数并重复周期以收集统计数据。

点击<Cancel>按钮停止校准。

### 7.6.2 校准程序：固定向下相机

以机器人的工作区域为基准，镜头朝下安装相机，使固定相机可以捕捉到被检测对象，并能读取到对象在机器人坐标系中的位置。本章节说明在工作平面上放置校准板的程序。有关使用终端执行器安装的工具上的目标作为基准点时的校准程序详细内容，参见“7.6.3 校准程序：固定向上相机”。

#### 准备1：安装相机

- (1) 相对于机器人的工作区域，镜头朝下安装相机。  
确保机器人不接触相机。

#### 准备2：失真校正准备 (执行失真校正时)

- (1) 创建失真检测用的图案。  
参见：7.2 镜头失真和相机倾斜校正  
也可以使用选装部件校准板。
- (2) 创建一个视觉序列。  
参见：7.4.3 用于失真校正的视觉序列

#### 准备3：制作一个校准板

- (1) 制作一块带有九个孔或九个目标的校准板，该九个孔或目标应跨越相机的整个视野。

#### 准备4：创建视觉序列来寻找校准基准目标

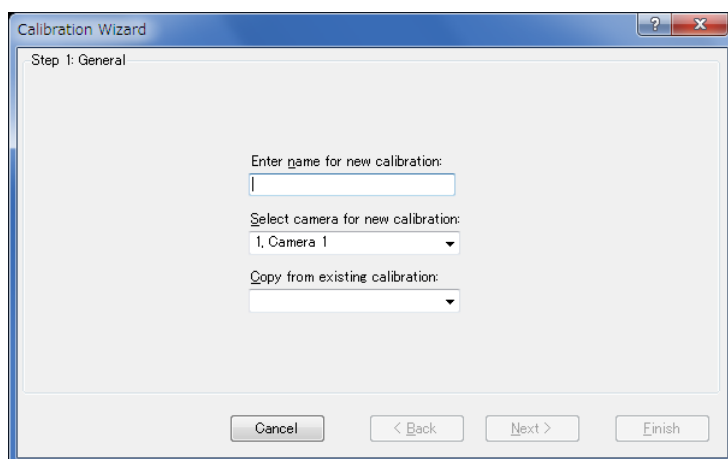
- (1) 创建一个视觉序列。  
参见：7.4.1 检测一个目标的视觉序列

完成准备后，运行校准向导配置所需设定。



### 第1步：启动校准向导

(1) 点击 Vision Guide 工具栏上的  <New Calibration>按钮运行校准向导。



在[Enter name for new calibration]中输入校准的名称。

在[Select camera for new calibration]中选择校准用相机。

可以通过在[Copy from existing calibration]中选择源校准数据复制设定。

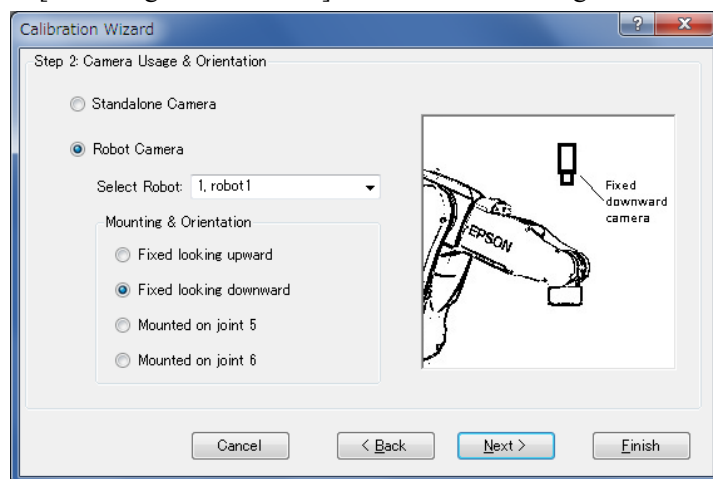
点击<Next>按钮进入步骤2。

### 第2步：校准类型和相机方向设定

点击<Robot Camera>。

在[Select Robot]中选择安装了相机的机器人。

在[Mounting & Orientation]中选择<Fixed looking downward>。

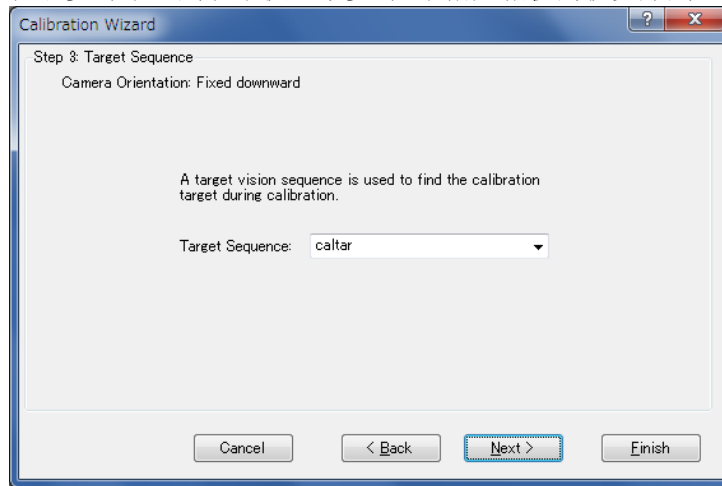


选择相机安装类型后，点击<Next>按钮。

### 第3步：指定目标序列

指定目标序列。

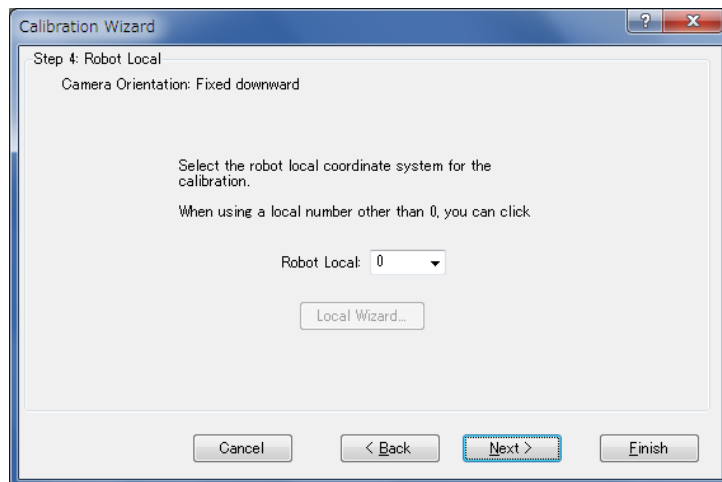
从列表中选择在 *准备4：创建视觉序列来寻找校准基准目标* 中创建的视觉序列。在该步骤中，列表中仅显示步骤1中指定相机的视觉序列。



选择目标序列后，点击<Next>按钮。

### 第4步：设定本地

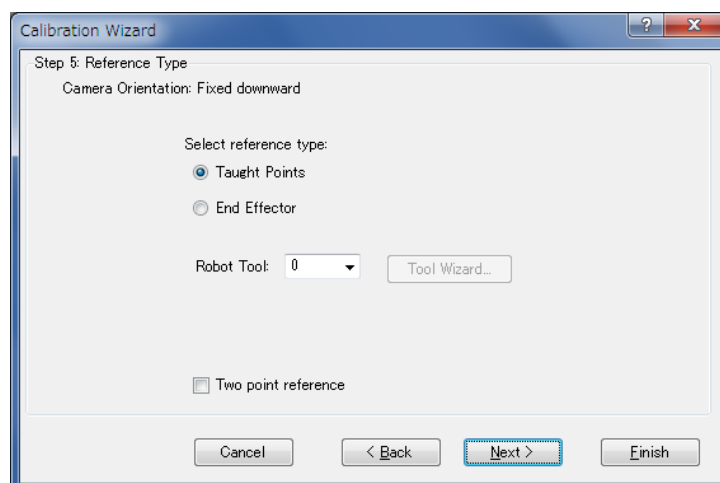
选择视觉校准中使用的本地坐标。



在该对话框中，可以运行本地向导将本地坐标系定义至本地编号。要运行向导，选择除“0”以外的本地编号，然后点击 <Local Wizard...>按钮。使用相机的本地设定详细内容在 *7.7 使用相机的本地设定* 中说明。选择本地后，点击<Next>按钮。

### 第5步：设定基准点类型

设定校准中使用的基准点类型。



用于校准的基准点是通过机器人坐标指定的点。

选择<Taught Points>。

需要步进机器人并手动示教基准点。

必须指定作为机器人机械臂末端安装的终端执行器基准点的工具编号和机械臂编号。(追加机械臂设定仅SCARA机器人可用。)

在该步骤中，也可以通过选择除“0”以外的工具编号后点击<Tool Wizard...>按钮，运行工具向导并定义工具。

此外，可以通过选择除“0”以外的机械臂编号后点击<Arm Wizard...>按钮，运行机械臂向导并定义相机的机械臂。

相机的工具和机械臂设定：

参见：7.8 检测可移动式相机安装位置

选择< End Effector>时：

参见：7.6.3 校准程序：固定向上相机

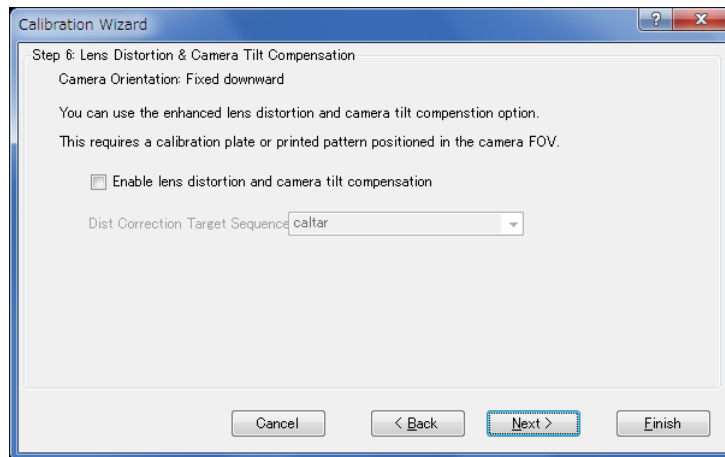
勾选[Two point reference]复选框时：

校准将使用两个基准点。

设定基准点类型后，点击<Next>按钮。

### 第6步：镜头失真和相机安装歪斜校正设定

指定是否校正镜头失真和相机安装歪斜。

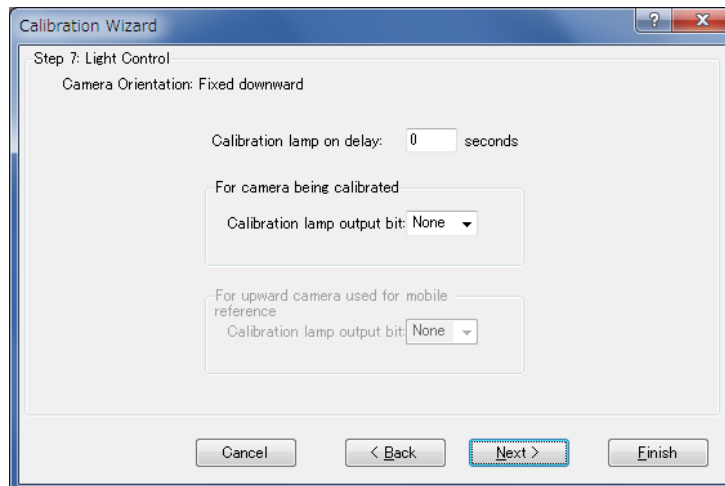


勾选复选框启用校正。要校正失真，必须预先创建失真校正用目标序列并在此步骤指定。

设定失真校正后，点击<Next>按钮。

### 第7步：照明控制设定

设定用于校准的照明控制。如果无需照明控制，则无需更改设定。

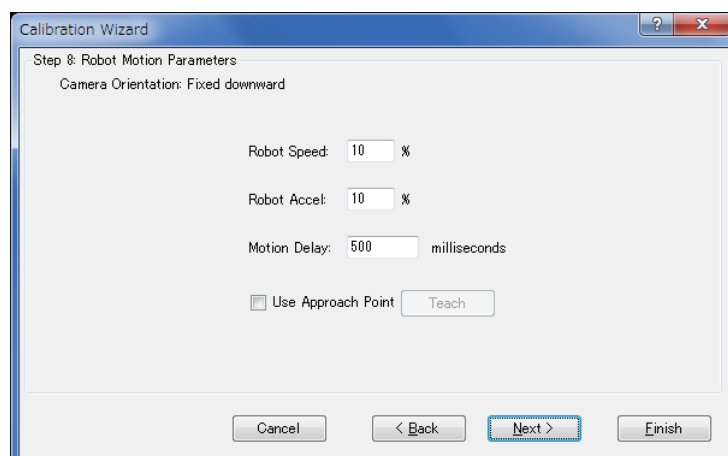


如果使用照明控制，以毫秒为单位指定照明开启前的等待时间。此外，指定开启照明的输出位。

设定照明控制后，点击<Next>按钮。

## 第8步：机器人动作设定

配置机器人动作设定。



设定速度和加速度以及机器人动作后的调整时间(动作延迟)。为执行精密校准，设定较慢速度和加速度并确保足够的调整时间。

也可以指定接近点。

如果指定了接近点，机器人始终从指定的接近点移动至相机点。这样可以使机器人从固定方向接近相机点，机器人位置稳定。

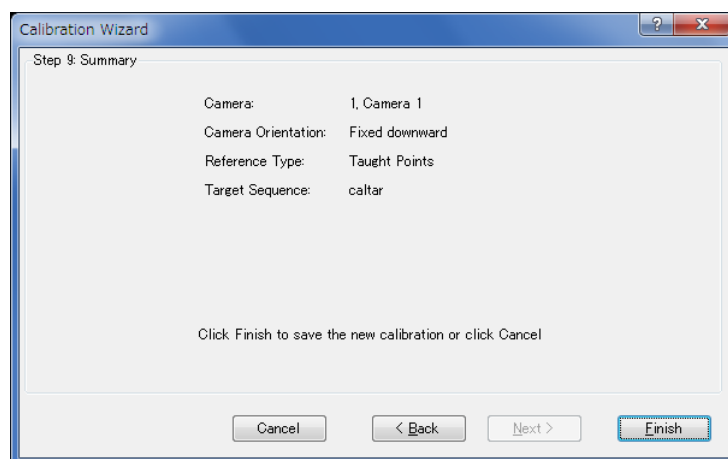
如何配置接近点：

勾选[Use Approach Point]复选框，然后点击<Teach Points>按钮。在显示的点示教对话框中示教接近点。

设定接近点后，点击<Next>按钮。

## 第9步：确认设定

显示已配置的项目。确认设定。



点击<Finish>按钮完成向导。

### 检测失真校正(失真校正启用时)

- (1) 在序列或校准目录树中选择已创建的校准。
- (2) 在工件平面放置点网格图案。
- (3) 选择属性列表中 DistCorrect 属性下的 Cal 属性，执行点网格图案检测。

通过在Calibration属性(校准目标定位用序列)中选择已创建的校准方案，可以确认已校正镜头失真和相机倾斜的图像。即使省略该设定，执行校准时将自动校正失真。

### 示教点

- (1) 点击<Teach Points>按钮。  
出现[Teach Calibration Points]对话框。
- (2) 遵循显示在对话框底部的消息框中的指示示教基准点。  
将会提示您示教该点，如果TwoRefPoints设为True，则必须是第4轴旋转180°来再次示教该点。使用工具时，可以跳过该步骤。要跳过该步骤，点击<Next>按钮移至下一步骤。

### 校准

点击<Calibrate>按钮开始校准周期。

校准软件定位九个目标，然后再次定位目标后确定校准参数以收集统计数据。

点击<Cancel>按钮停止校准。

### 7.6.3 校准程序：固定向上相机

以机器人的工作区域为基准，镜头朝上安装相机，使固定相机可以捕捉到被检测对象，并能读取到对象在机器人坐标系中的位置。

固定向上相机可以通过将基准点类型设为“EndEffector”进行同样的校准。

#### 准备1：安装相机

- (1) 相对于机器人的工作区域，镜头朝上安装相机。

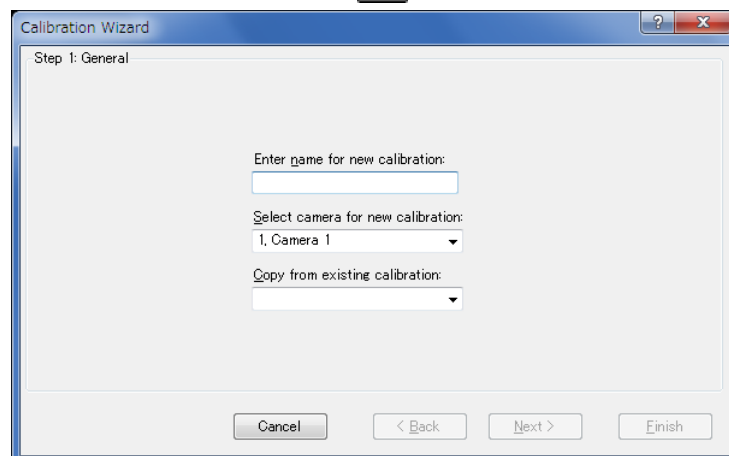
#### 准备2：创建视觉序列来寻找校准终端执行器目标

- (1) 创建一个序列来定位终端执行器上的目标。在序列中创建一个或多个对象来定位目标。校准软件将利用序列中的最后一步来读取目标的位置。X 和 Y 的最后一步结果应该是目标的中心。
- (2) 在校准过程中，终端执行器将移动至相机视野中九个不同的点并搜索目标。在每个位置，校准软件将 U 轴旋转 180° 并再次搜索目标。这使得软件为每个点确定 U 轴的中心。为获得最好结果，请使用圆形的目标。

完成准备后，运行校准向导配置所需设定。

### 第1步：运行校准向导

点击Vision Guide工具栏上的 <New Calibration>按钮运行校准向导。



在[Enter name for new calibration]中输入校准的名称。

在[Select camera for new calibration]中选择校准用相机。

可以通过在[Copy from existing calibration]中选择源校准数据复制设定。

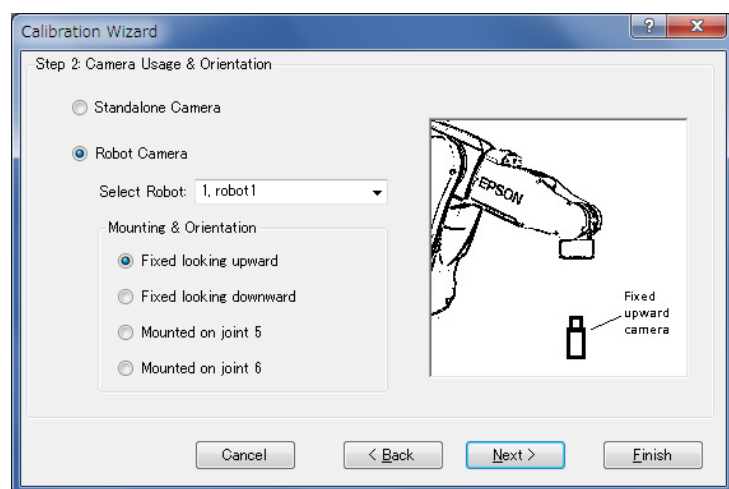
点击<Next>按钮进入步骤2。

### 第2步：校准类型和相机方向设定

点击<Robot Camera>。

在[Select Camera]中选择安装了相机的机器人。

在[Mounting & Orientation]中选择<Fixed looking upward>或<Fixed looking downward>。



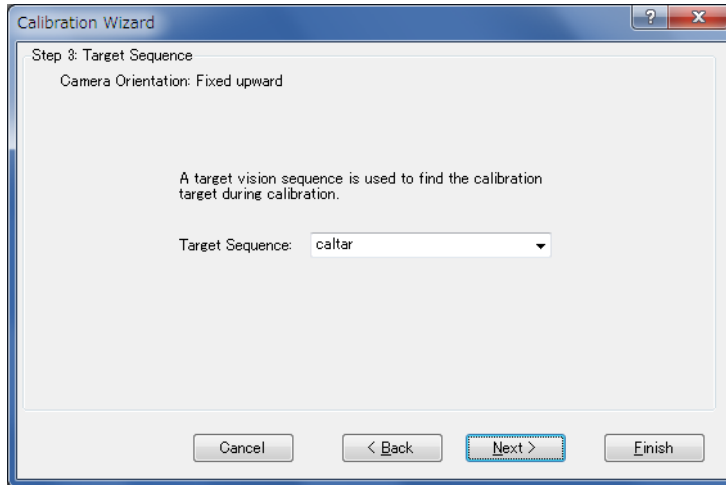
选择Mounting & Orientation后，点击<Next>按钮。

### 第3步：指定目标序列

指定目标序列。

从列表中选择在 *准备2：创建视觉序列来寻找校准终端执行器目标* 中创建的视觉序列。

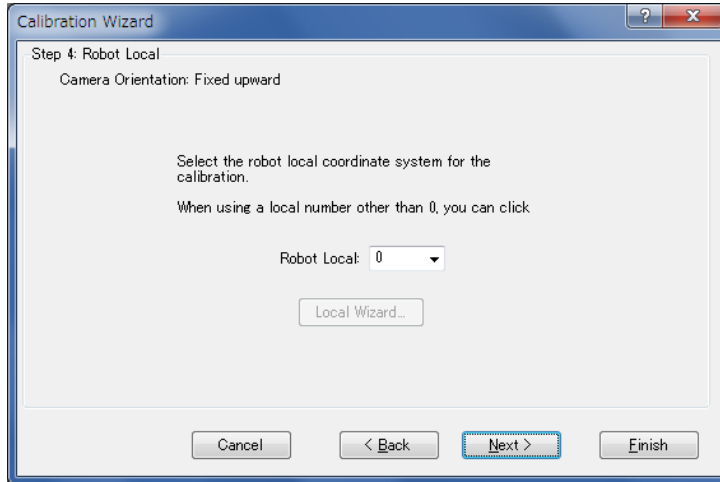
在该步骤中，列表中仅显示步骤1中指定相机的视觉序列。



选择目标序列后，点击<Next>按钮。

### 第4步：设定本地

选择视觉校准中使用的本地坐标。



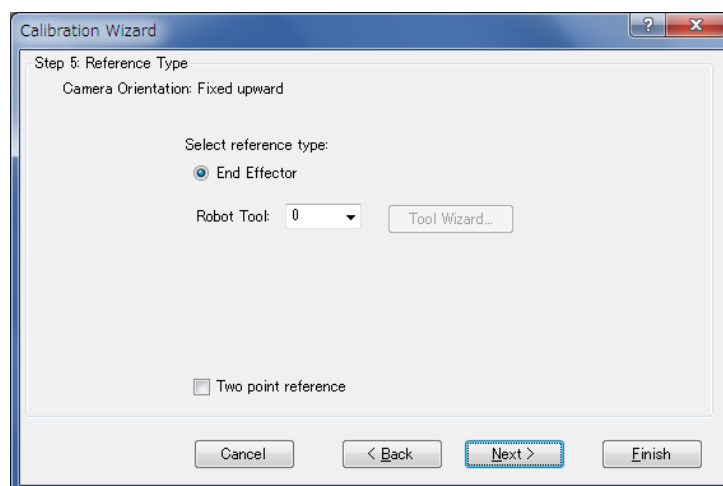
在该对话框中，可以运行本地向导将本地坐标系定义至本地编号。  
要运行向导，选择除“0”以外的本地编号，然后点击 <Local Wizard...>按钮。  
使用相机的本地设定详细内容在 *7.7 使用相机的本地设定* 中说明。

选择本地后，点击<Next>按钮。



### 第5步：设定基准点类型

设定校准中使用的基准点类型。



固定向上相机的基准点类型为<End Effector>。

基准点是机器人终端执行器安装的工具上的目标。

需要能从相机中看到目标。

(此步骤中固定向上相机可以选择<Reference type>，但此处选择<End Effector>)

在该步骤中，也可以通过选择除“0”以外的工具编号后点击<Tool Wizard...>按钮，运行工具向导并定义工具。

此外，可以通过选择除“0”以外的机械臂编号后点击<Arm Wizard...>按钮，运行机械臂向导并定义相机的机械臂。

(追加机械臂设定仅SCARA机器人可用。)

相机的工具和机械臂设定：

参见：7.8 检测可移动式相机安装位置

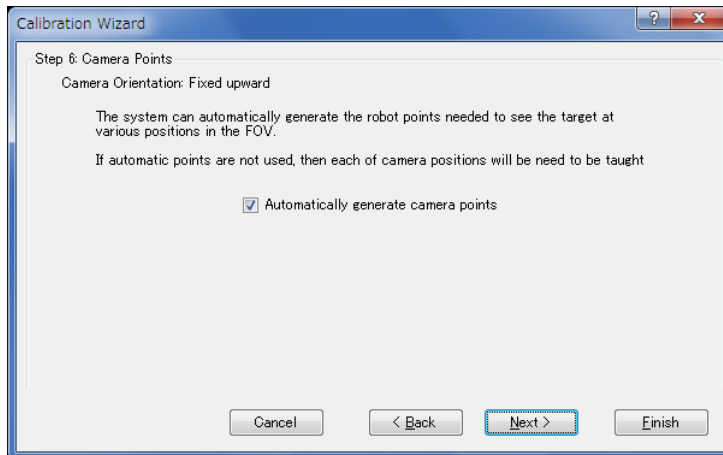
勾选[Two point reference]复选框时：

校准将使用两个基准点。

设定基准点类型后，点击<Next>按钮。

### 第6步：相机点设定

指定执行校准时是否自动生成相机点。



勾选复选框时：

通过在相机FOV中检测目标对象的同时自动操作机器人，自动生成多个相机点。对各相机点检测FOV中目标对象的位置。无需通过步进机器人示教其他相机点。

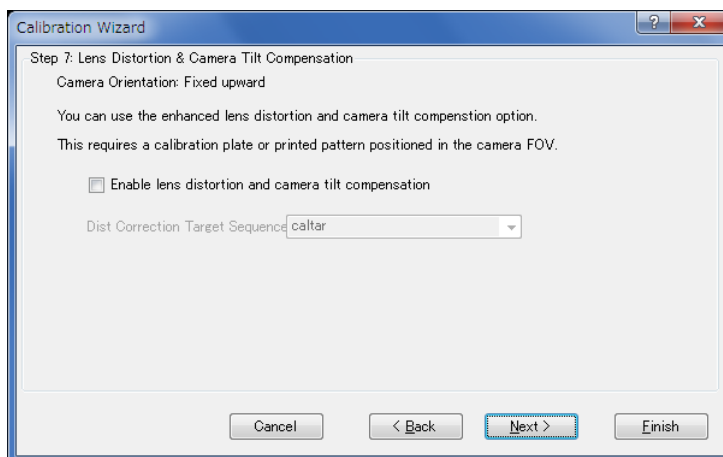
未勾选复选框时：

执行校准前必须步进机器人手动示教所需数量的相机点。

设定自动相机点生成后，点击<Next>按钮。

### 第7步：镜头失真和相机安装歪斜校正设定

指定是否校正镜头失真和相机安装歪斜。

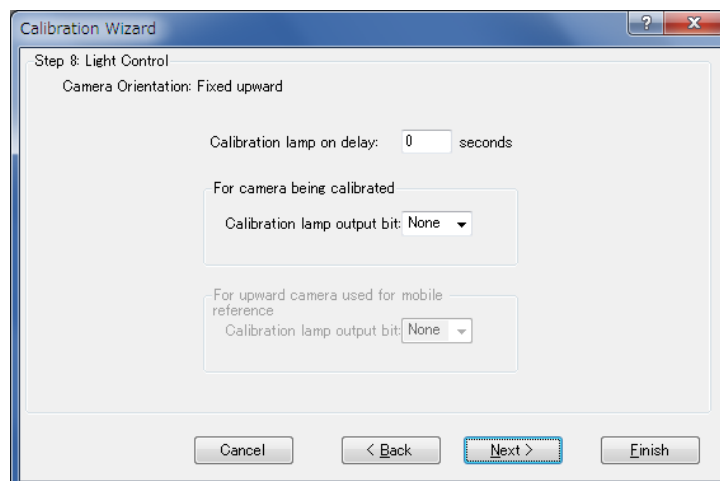


勾选复选框启用校正。要校正失真，必须预先创建失真校正用目标序列并在此步骤指定。

设定失真校正后，点击<Next>按钮。

### 第8步：照明控制设定

设定用于校准的照明控制。如果无需照明控制，则无需更改设定。

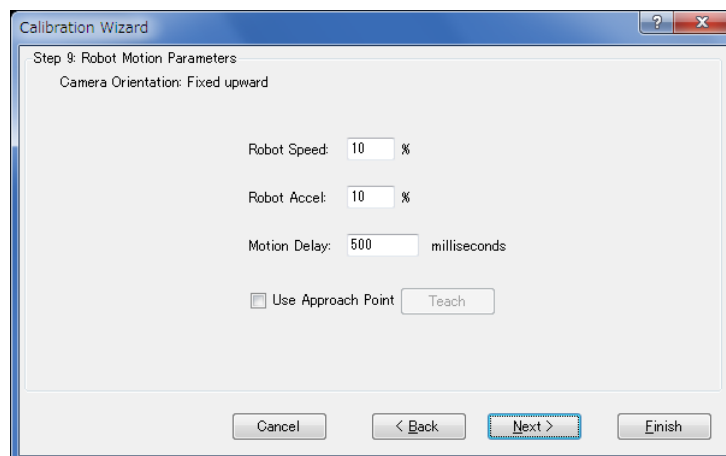


如果使用照明控制，以秒为单位指定照明开启前的等待时间。此外，指定开启照明的输出位。

设定照明控制后，点击<Next>按钮。

### 第9步：机器人动作设定

配置机器人动作设定。



设定速度和加速度以及机器人动作后的调整时间(动作延迟)。为执行精密校准，设定较慢速度和加速度并确保足够的调整时间。

也可以指定接近点。

如果指定了接近点，机器人始终从指定的接近点移动至校准点。这样可以使机器人从固定方向接近校准点，机器人位置稳定。

如何配置接近点：

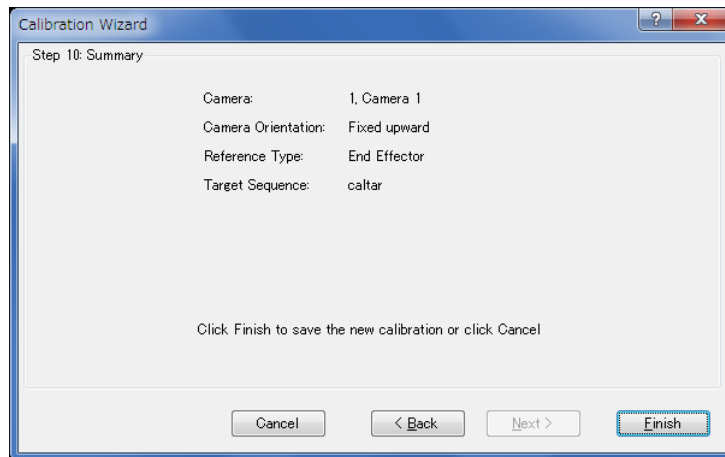
勾选[Use Approach Point]复选框，然后点击<Teach>按钮。

在显示的点示教对话框中示教接近点。

设定接近点后，点击<Next>按钮。

### 第10步：确认设定

显示已配置的项目。确认设定。



点击<Finish>按钮完成向导。

### 示教点

- (1) 点击<Teach Points>按钮。  
出现[Teach Calibration Points]对话框。
- (2) 遵循显示在对话框底部的消息框中的指示示教基准点。  
将会提示您示教该点，如果TwoRefPoints设为True，则必须是第4轴旋转180°来再次示教该点。使用工具时，可以跳过该步骤。要跳过该步骤，点击<Next>按钮移至下一步骤。

### 校准

点击<Calibrate>按钮开始校准周期。

校准软件将把机器人移至每个相机位置，并搜索目标。TwoRefPoints设为“True”时，机器人将第4轴旋转180°并再次搜寻目标。重复校准以收集统计数据。

点击<Cancel>按钮停止校准。

### 7.6.4 校准程序：独立相机

该校准将允许您进行物理测量。

当相机被校准为独立相机时，无法用于计算机器人坐标的值。只能返回以毫米为单位，CameraX 和 CameraY 的值。

#### 第1步：安装相机

- (1) 安装相机，使其与工作平面成 40° 至 90° 角。

#### 第2步：制作点网格图案(执行失真校正时)

- (1) 创建超过 100 点的点网格图案。网格图案必须覆盖整个视野，且失真尽可能小。网格图案的精度会影响图像处理的精度。

#### 第3步：制作一个校准板

- (1) 制作一块带有九个孔或九个目标的校准板，该九个孔或目标应跨越相机的整个视野。


#### 第4步：创建寻找校准基准目标的视觉序列 (执行失真校正时)

- (1) 创建一个序列。  
参见：7.4.3 用于失真校正的视觉序列

#### 第5步：创建视觉序列来寻找校准基准目标

- (1) 创建一个视觉序列。  
参见：7.4.2 检测九个目标的视觉序列

#### 第6步：创建校准方案

- (1) 点击 Vision Guide 工具栏上的  <New Calibration>按钮运行校准向导。
- (2) 运行校准向导。  
选择校准名和相机。根据需要选择用于复制设定的源校准。
- (3) 选择独立相机。
- (4) 配置用于检测校准基准目标的视觉序列。
- (5) 指定是否启用失真校正。  
如果启用失真校正，选择检测点网格图案的视觉序列。
- (6) 通过以下步骤配置设定以完成向导。

#### 第7步：镜头失真和相机安装歪斜校正设定

- (1) 在序列或校准目录树中选择已创建的校准。

- (2) 在工件平面放置点网格图案。
- (3) 选择属性列表中 DistCorrect 属性下的 Cal 属性，执行点网格图案检测。
- (4) 通过在 Calibration 属性(校准目标定位用序列)中选择步骤 6 创建的校准方案，可以确认已校正镜头失真和相机倾斜的图像。即使省略该设定，执行校准时将自动校正失真。

### 第8步：校准

- (1) 移除点网格图案后，放置步骤3创建的校准板。
- (2) 点击<Teach Points>按钮后设定校准板上九个目标的坐标。
- (3) 点击<Calibrate>按钮开始校准周期。校准软件定位九个目标，然后再次定位目标后确定校准参数以收集统计数据。

## 7.7 使用相机的本地检测

可以通过使用6-轴机器人的相机执行本地检测。

在Robot Manager中选择[Locals]选项卡，启动Local Wizard。这些设定向导可以从校准向导显示。

### 7.7.1 在工作平面上定义本地

本章节说明如何通过使用可移动式相机检测工作平面上的校准板来定义与工作平面平行的本地坐标。使用6-轴机器人第6机械臂(J6)安装的可移动式相机时，该功能启用。

**NOTE:** 要使用相机配置工作平面上的本地，需使用选装部件校准板。步进机器人，使相机光轴和校准板大约成90°。使用相机的本地设定可能由于相机的硬件配置不可用。

参见：7.视觉校准

可以在本地向导中执行使用相机的本地坐标定义。

通过以下任意一种方式运行本地向导。

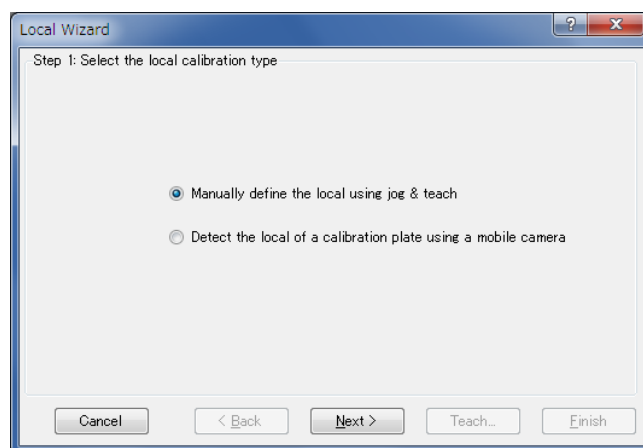
- (1) 在Robot Manager中选择[Locals]选项卡。  
出现<Local Wizard...>按钮。点击该按钮。
- (2) 在校准向导中选择机器人本地编号的步骤选择了除“0”以外的机器人本地编号时，出现<Local Wizard...>按钮。  
点击该按钮。

以下说明本地向导启动后的步骤。(向导的内容因选择项目而异)

#### 第1步：选择类型

运行本地设定向导时，出现以下对话框。

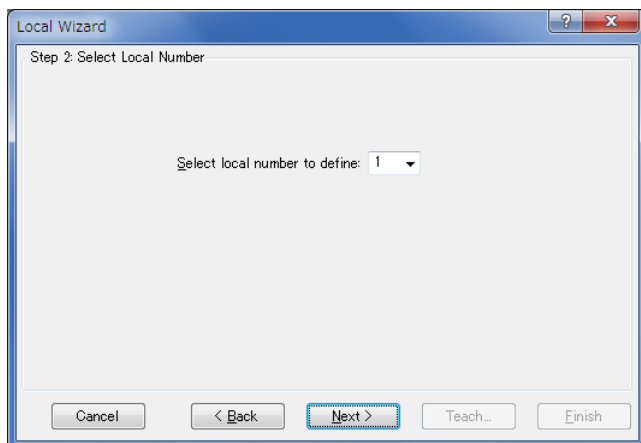
点击<Detect the local of a calibration plate using a mobile camera>按钮。



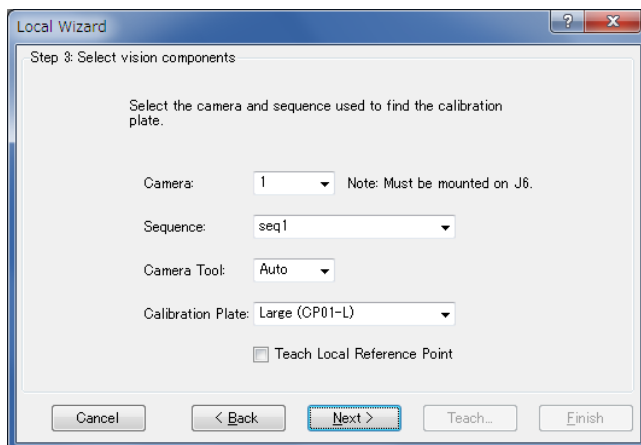
点击<Next>按钮。

**第2步：选择本地编号**

选择要配置的本地编号。



点击<Next>按钮。

**第3步：选择视觉要素**

设定以下项目。

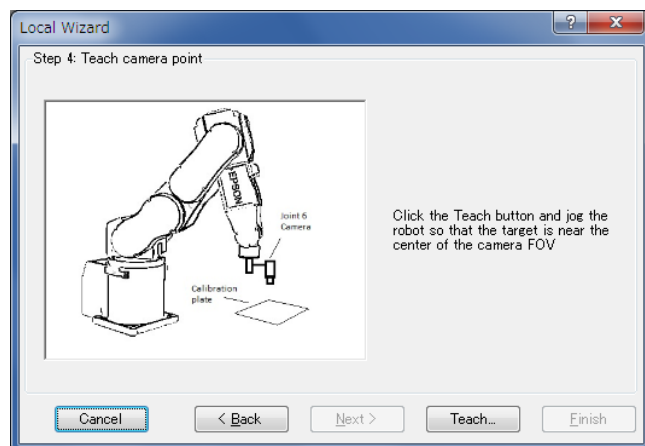
项目	说明
Camera	选择校准过程中使用的相机。 并且相机需要安装在第6机械臂(J6)上。
Sequence	选择用于检测校准板的视觉序列。
Camera Tool	指定已校准可移动式相机的工具编号。 选择“Auto”时，将自动检测相机工具。
Calibration Plate	选择校准板的类型。
Teach Local Reference Point	示教本地基准点时，检测到的本地平面将设为经过已示教点。不示教基准点时，本地平面将设为经过Tool 0。 如果该功能启用，必须步进机器人使终端执行器到达本地平面，并示教点。

点击<Next>按钮。

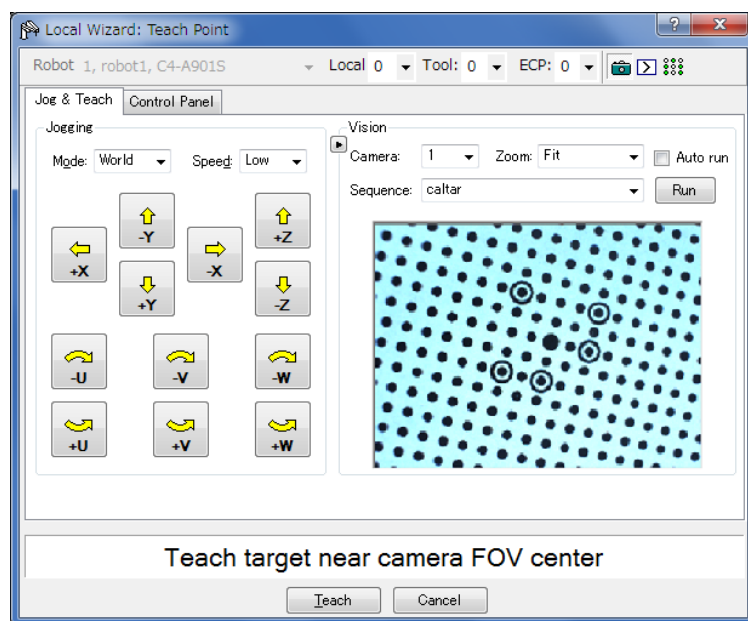


#### 第4步：示教相机点

点击<Teach...>按钮。出现步进对话框。



步进机器人，并将校准板中心移动至视野中心。



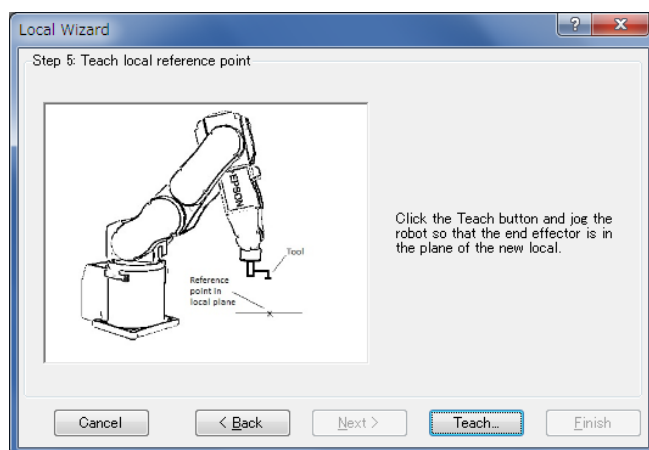
相机点确定时，点击<Teach>按钮。

进入下一步骤。

### 第5步：示教本地基准点

仅当本地基准点需要示教时显示该步骤。

点击<Teach...>按钮将出现步进对话框。

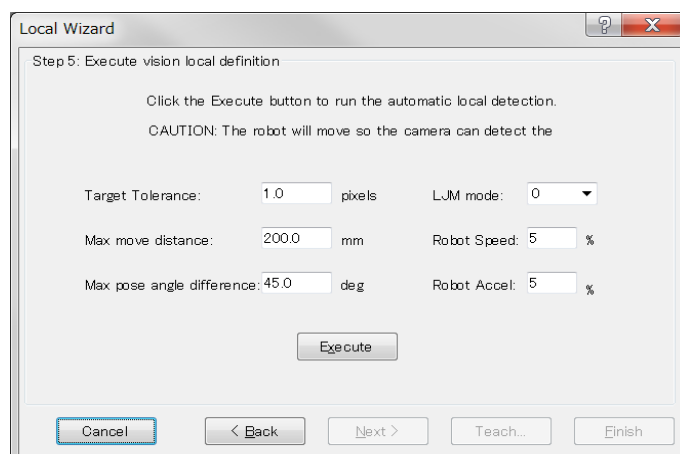


步进机器人，使终端执行器到达本地平面。

点击<Teach>按钮示教点，向导返回该步骤。

### 第6步：定义视觉本地

根据需要更改以下设定。



根据需要更改以下设定。

项目	说明
Target Tolerance	执行检测，使图像上的误差在该对话框中指定的允许范围内。
Max move distance	指定机械臂端部的移动距离限制。 如果指定“0”，则距离无限制。
Max pose angle difference	工具方向的(UVW)最大位移角。(单位：度) 如果指定“0”，则角度无限制。
LJM mode	指定SPEL+的LJM函数使用的值。 LJM模式对点数据的方向标志进行控制，防止腕关节意外旋转。 如果指定“0”，则不使用LJM
Robot Speed	设定机器人速度。 设定较低速度值以配置精确的本地设定。
Robot Acceleration	设定机器人加速度。 设定较低加速度值以配置精确的本地设定。

点击<Finish>按钮。

检测完成后，将显示结果。确认结果。

点击<Finish>按钮将结果设定至指定本地编号。

**NOTE:** 根据校准板识别和目标检测结果，机器人自动移动。

注意机器人与外围装置的相互干扰。此外，为防止本地检测过程中发生错误，应避免使用各轴延伸的奇异点附近的方向。

## 7.8 检测可移动式相机安装位置

可以检测可移动式相机的安装位置。

SCARA机器人第2机械臂(J2)安装的相机可以作为追加机械臂设定的参数配置。

SCARA机器人第4轴(J4)或垂直6-轴机器人第6轴(J6)安装的相机可以作为工具配置。

### 7.8.1 相机安装位置的工具设定

以下说明为SCARA机器人第4轴(J4)或垂直6-轴机器人第6轴(J6)安装的相机配置工具设定的步骤。

预先创建检测目标对象所需的视觉序列。

参见：7.4.4 用于本地、工具和机械臂设定的视觉序列

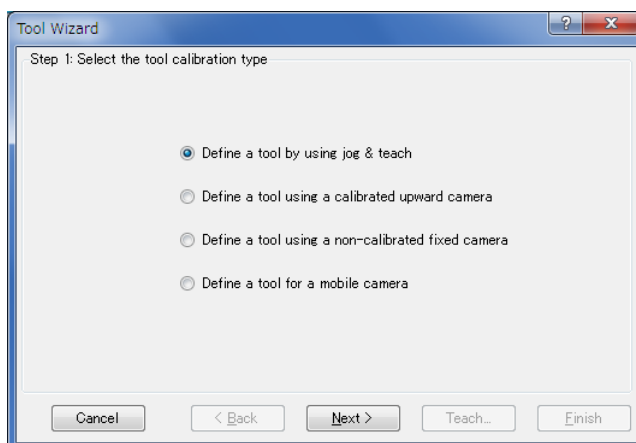
通过以下任意一种方式运行工具向导。

- (1) 在Robot Manager中选择[Tools]选项卡。  
出现<Tool Wizard...>按钮。  
点击该按钮。
- (2) 在校准向导中选择工具编号的步骤选择了除“0”以外的工具编号时，出现<Tool Wizard...>按钮。  
点击该按钮。

#### 第1步：选择类型

运行工具向导时，出现以下对话框。

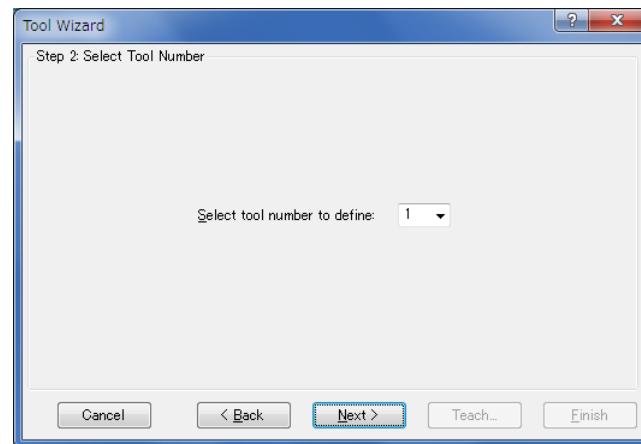
点击<Define a tool for a mobile camera>按钮。



点击<Next>按钮。

### 第2步：选择工具编号

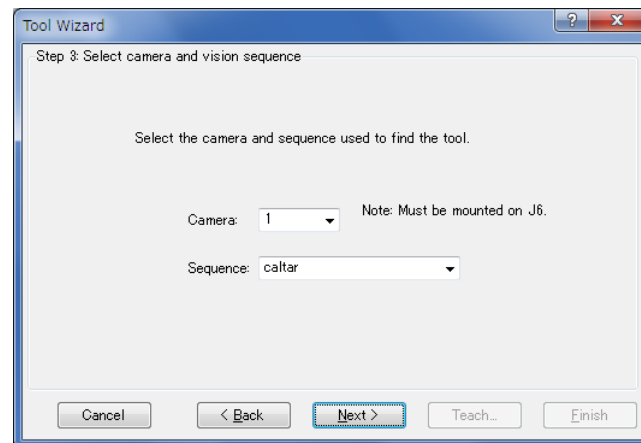
选择要配置的工具编号。



点击<Next>按钮。

### 第3步：配置视觉

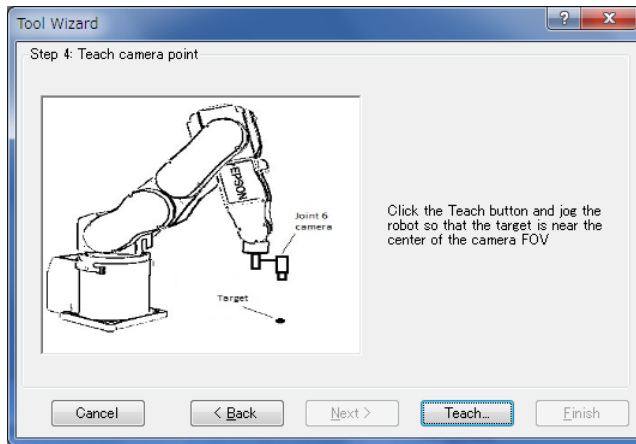
选择检测相机和目标对象的序列。



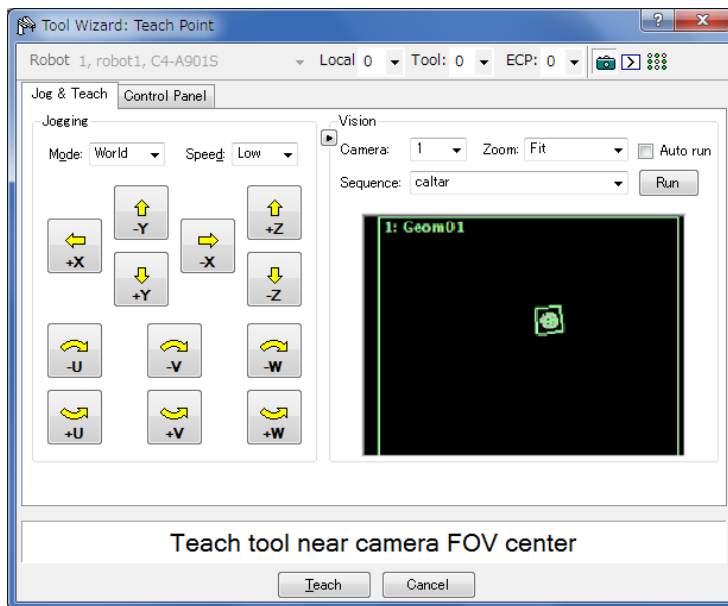
点击<Next>按钮。

## 第4步：示教相机点

点击<Teach...>按钮。出现点示教对话框。



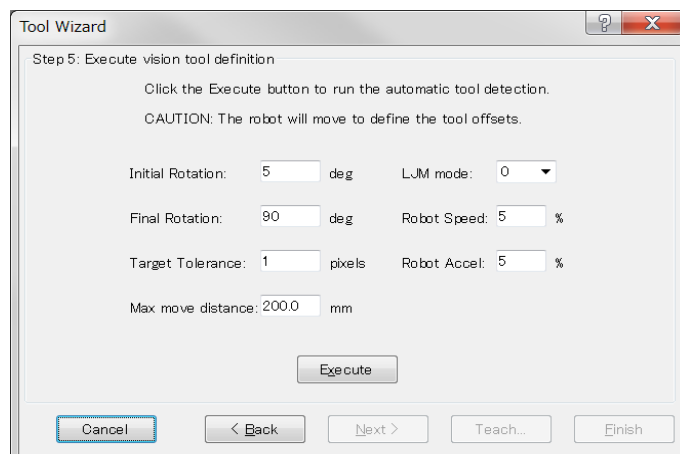
步进机器人，使目标对象在视野中心附近被检测到。



点击<Teach>按钮。

进入下一步骤。

## 第5步：执行



根据需要更改以下设定。

项目	说明
Initial Rotation	在粗略定位中工具最初的旋转角。(度)
Final Rotation	工具最终旋转角(建议大于90°)。最终旋转角越大，工具设定的精度越高。
Target Tolerance	执行检测，使图像上的误差在该对话框中指定的允许范围内。
Max move distance	指定机械臂端部的移动距离限制。 如果指定“0”，则距离无限制。
LJM mode	指定SPEL+的LJM函数使用的值。 LJM模式对点数据的方向标志进行控制，防止腕关节意外旋转。 如果指定“0”，则不使用LJM
Robot Speed	设定机器人速度。 设定较低速度值以配置精确的本地设定。
Robot Acceleration	设定机器人加速度。 设定较低加速度值以配置精确的本地设定。

点击<Execute>按钮。

检测完成后，将显示结果。确认结果。

点击<Finish>按钮将结果设定至指定工具编号。

NOTE: 根据目标对象检测结果，机器人自动移动。

注意机器人与外围装置的相互干扰。此外，为防止工具设定过程中发生错误，应避免使用各轴延伸的奇异点附近的方向。

## 7.8.2 相机安装位置的机械臂设定

以下说明将SCARA机器人第2轴(J2)安装相机的安装位置作为追加机械臂参数配置的步骤。

预先创建检测目标对象所需的视觉序列。

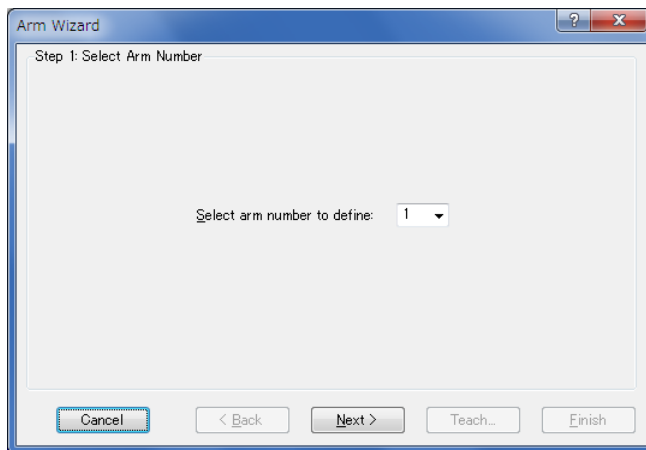
参见：7.4.4 用于本地、工具和机械臂设定的视觉序列

通过以下任意一种方式运行机械臂向导。

- (1) 在Robot Manager中选择[Arms]选项卡。  
出现<Arm Wizard...>按钮。  
点击该按钮。
- (2) 在校准向导中选择机械臂编号的步骤选择了除“0”以外的机械臂编号时，出现<Arm Wizard...>按钮。  
点击该按钮。

### 第1步：选择机械臂编号

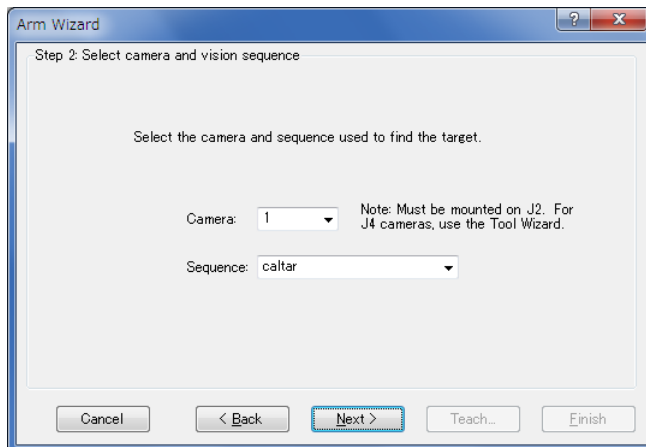
选择要配置的追加机械臂编号。



点击<Next>按钮。

### 第2步：配置视觉

选择检测相机和目标对象的序列。

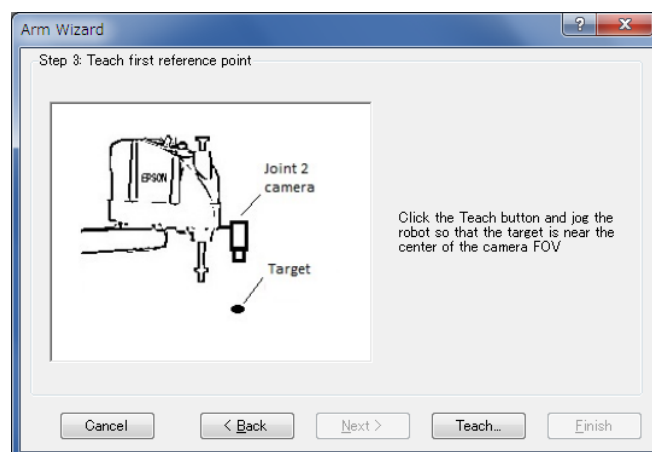


点击<Next>按钮。

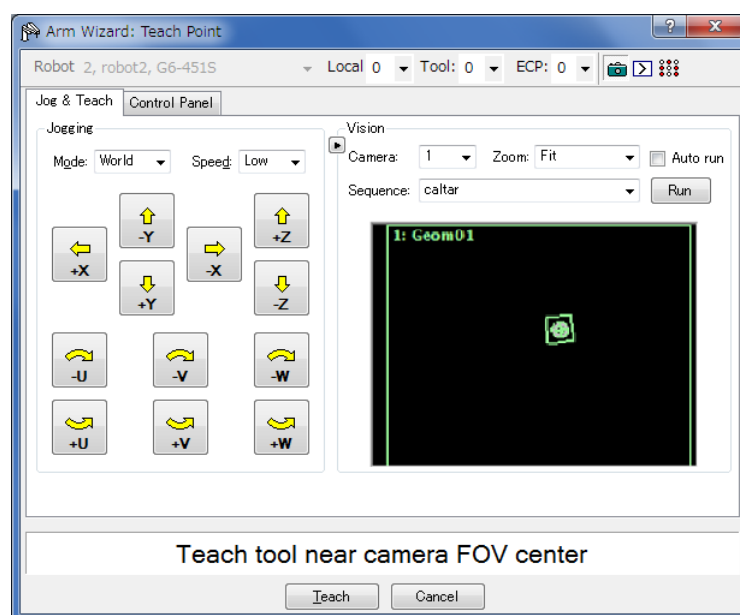


### 第3步：示教相机点

点击<Teach...>按钮。出现点示教对话框。



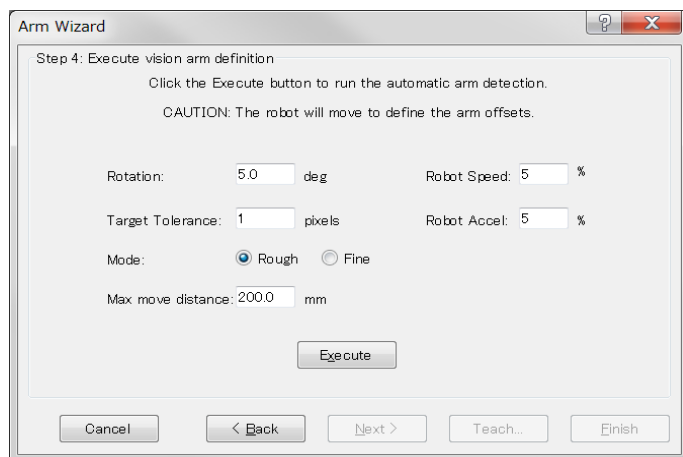
步进机器人，使目标对象在相机FOV中心附近被检测到。



点击<Teach>按钮。

进入下一步骤。

## 第4步：执行



根据需要设定以下项目。

项目	说明
Rotation	指定用于直线粗略机械臂设定的旋转角(度)。
Target Tolerance	指定视为视觉检测结果匹配目标位置时的像素距离。
Mode	<p><b>Rough:</b> 运行粗略机械臂设定的模式。 机器人动作较小。 机器人以约1 mm的精度为目标。</p> <p><b>Fine:</b> 运行精密机械臂设定的模式。 机器人移动较大，且机械臂方向改变。 机器人以高精度为目标，而非粗略精度。</p>
Max move distance	设定机械臂末端的移动距离限制。 如果设为“0”，则距离不受限。
Robot Speed	设定机器人速度。 设定较低速度值以配置精确的本地设定。
Robot Acceleration	设定机器人加速度。 设定较低加速度值以配置精确的本地设定。

点击<Execute>按钮。

检测完成后，将显示结果。确认结果。

点击<Finish>按钮将结果设定至指定追加机械臂编号。

**NOTE:** 根据目标对象检测结果，机器人自动移动。

注意机器人与外围装置的相互干扰。

## 7.9 使用相机的工具设定

可以使用固定相机检测机器人末端安装工具的工具坐标。

工具设定向导与 7.8.1 相机安装位置的工具设定中的向导步骤大致相同。

预先创建检测目标所需的视觉序列。

参见：7.4.4 用于本地、工具和机械臂设定的视觉序列

通过以下任意一种方式运行工具向导。

- (1) 在Robot Manager中选择[Tools]选项卡。  
出现<Tool Wizard...>按钮。  
点击该按钮。
- (2) 在校准向导中选择工具编号的步骤选择了除“0”以外的工具编号时，出现<Tool Wizard...>按钮。  
点击该按钮。

### 第1步：选择类型

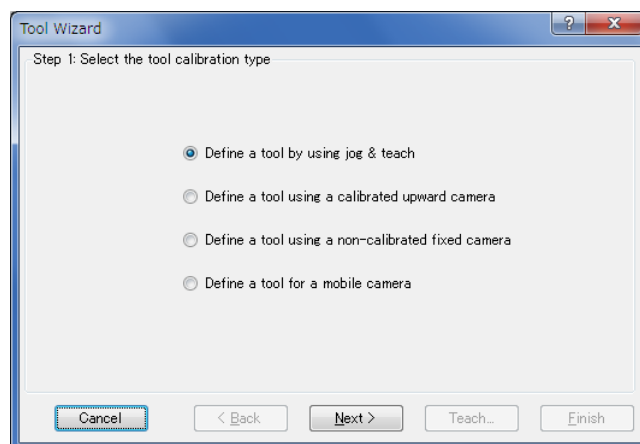
运行工具向导时，出现以下对话框。

选择以下任意一个按钮。

<Define a tool using a calibrated upward camera>

已校准的固定向上相机指定至序列的校准属性时，该功能启用。当校准固定向上相机、更改使用的工具或设置不同工具时使用。

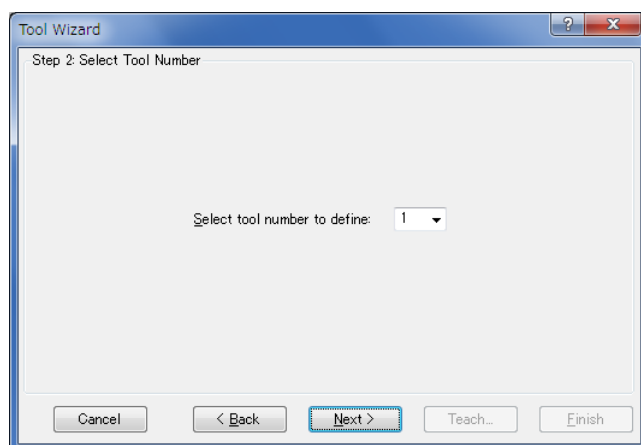
<Define a tool using a non-calibrated fixed camera>



点击<Next>按钮。

### 第2步：选择工具编号

选择要配置的工具编号。



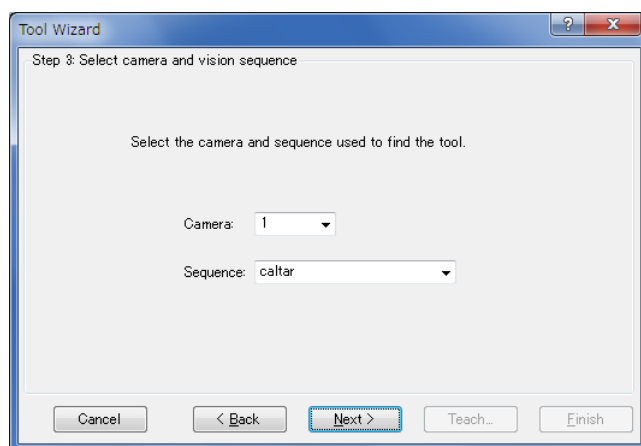
点击<Next>按钮。

### 第3步：配置视觉

选择检测相机和目标对象的序列。

使用已校准的固定向上相机时，还需选择对象。仅可选择有RobotToolXYU结果的对象。

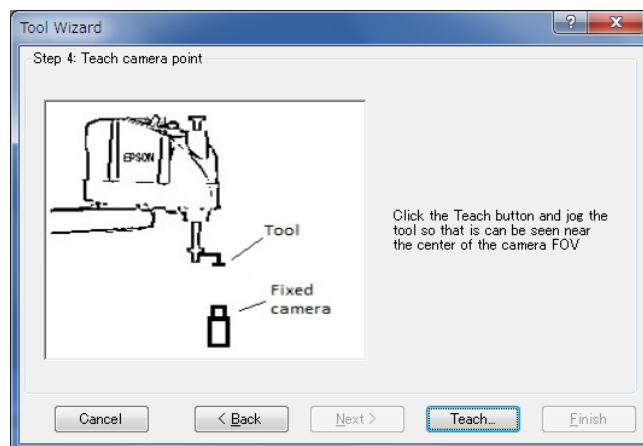
有关更多详细内容，参见 *Vision Guide 7.0 Properties and Results Reference* 的 RobotToolXYU结果。



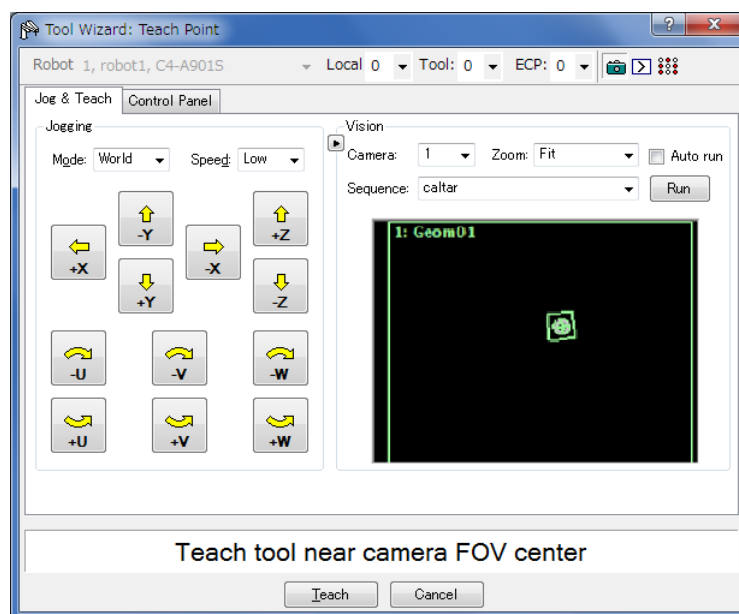
点击<Next>按钮。

## 第4步：示教相机点

点击<Teach...>按钮。出现点示教对话框。



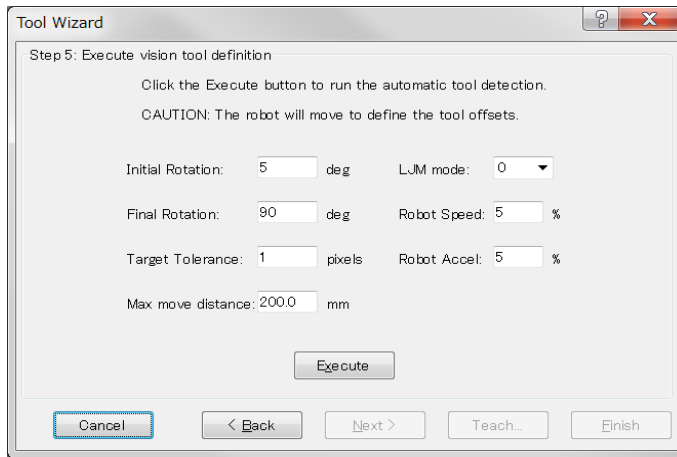
步进机器人，使目标对象在相机FOV中心附近被检测到。



点击<Teach>按钮。

进入下一步骤。

第5步：执行



使用未校准的固定相机时，配置以下项目。使用已校准的固定相机时，无需设定。

项目	说明
Initial Rotation	在粗略定位中工具最初的旋转角(度)。
Final Rotation	工具最终旋转角(建议大于90度)。最终旋转角越大，工具设定的精度越高。
Target Tolerance	执行检测，使图像上的误差在该对话框中指定的允许范围内。
Max move distance	指定机械臂端部的移动距离限制。 如果指定“0”，则距离无限制。
LJM mode	指定SPEL+的LJM函数使用的值。 LJM模式对点数据的方向标志进行控制，防止腕关节意外旋转。 如果指定“0”，则不使用LJM
Robot Speed	设定机器人速度。 设定较低速度值以配置精确的本地设定。
Robot Acceleration	设定机器人加速度。 设定较低加速度值以配置精确的本地设定。

点击<Execute>按钮。

检测完成后，将显示结果。确认结果。

点击<Finish>按钮将结果设定至指定工具编号。

NOTE：根据目标对象检测结果，机器人自动移动。

注意机器人与外围装置的相互干扰。此外，为防止工具设定过程中发生错误，应避免使用各轴延伸的奇异点附近的方向。

## 7.10 使用相机的3D工具设置

若使用6轴机器人，可使用固定相机来设置机器人夹具的3D工具 (3D位置、姿态检测)。

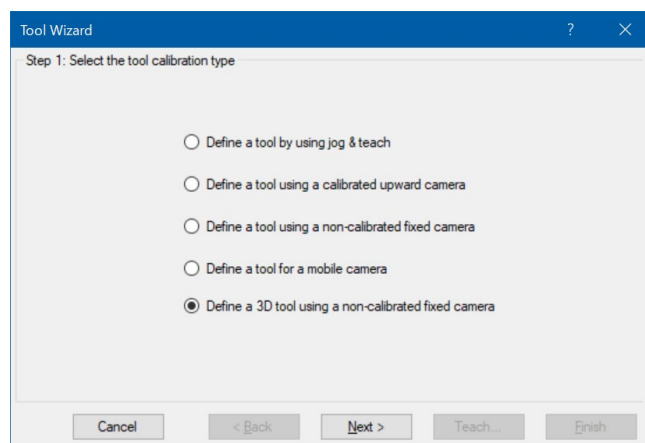
使用工具设置向导进行相机的3D工具设置。选择以下方式之一，启动工具设置向导。

- (1) 点击机器人管理器中的[Tools]选项卡。显示<Tool Wizard...>按钮。  
点击此按钮。
- (2) 选择校准向导中，除了“0”以外的工具编号时，将显示< Tool Wizard...>按钮。  
点击此按钮。

### 第1步: 选择类型

启动工具设置向导，显示以下对话框。

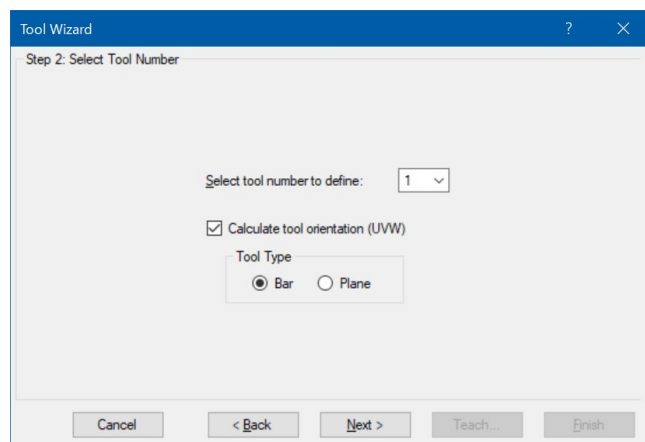
选择<Define a 3D tool using a non-calibrated fixed camera>选项。



点击<Next>按钮。

### 第2步: 选择工具编号

选择要使用的工具编号。



若要查找工具姿态，请选择<Calculate tool orientation (UVW)>。

可选择Bar(条状)或Plane(平面)的工具类型。

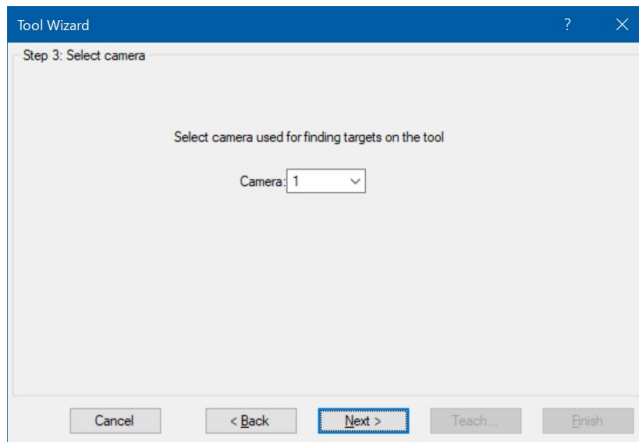
Bar: 查找如圆柱体的条状工具姿态。

Plane: 查找平面工具姿态。使用工具所夹持的工件的水平面来查找工具姿态。

点击<Next>按钮。

### 第3步: 选择相机

在工具上选择用于检测目标的相机。



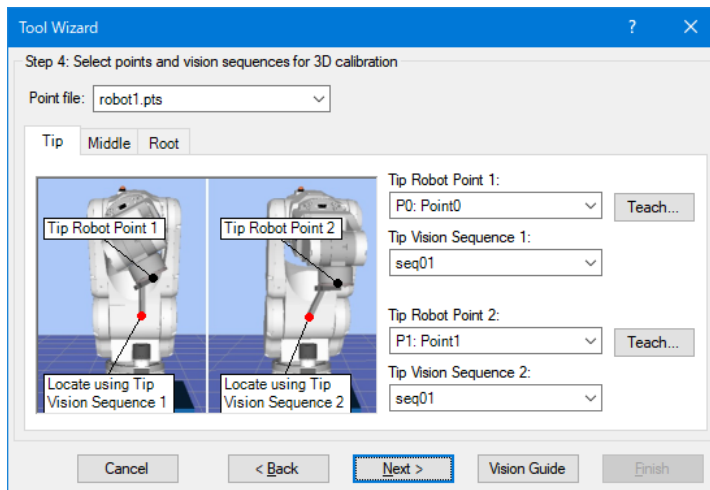
点击<Next>按钮。

### 第4步: 选择点和视觉序列

选择用具工具上目标检测的机器人点和视觉序列。

请选择目标在相机视野内的机器人点。请将机器人点1和点2设置为不同的姿态。点击<Teach...>按钮，可以在点位示教画面中示教机器人点位。

在视觉序列中选择用于检测目标的视觉序列。分别根据机器人点1和点2选择各自的视觉序列。还可以点击Vision Guide画面中的<Vision Guide>按钮创建视觉序列。

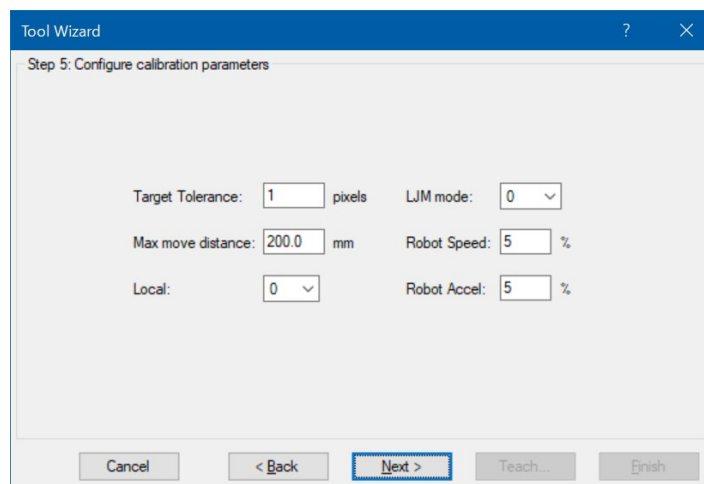


若在第2步中勾选了<Calculate tool orientation (UVW)>选项，则可切换Tip、Middle或Root选项卡，在所有选项卡中选择机器人点和视觉序列。

点击<Next>按钮。



## 第5步: 设置校准参数



请根据需要更改以下设定。

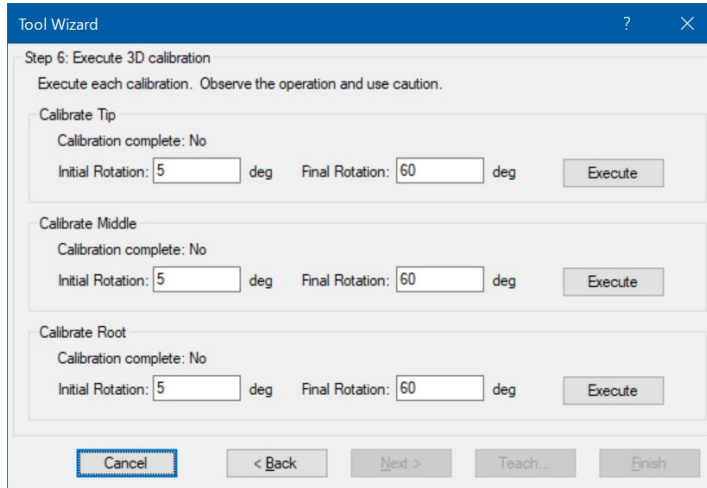
项目	说明
Target Tolerance	执行检测，使图像上的误差在该对话框中指定的允许范围内。
Max move distance	指定机械臂末端的移动距离限制。 如果指定“0”，则距离无限制。
LJM mode	指定SPEL+的LJM函数使用的值。 LJM模式对点数据的方向标志进行控制，防止腕关节意外旋转。 如果指定“0”，则不使用LJM。 如果指定“0”，则不使用LJM
Robot Speed	设定机器人速度。 设定较低速度值以配置精确的本地设定。
Robot Acceleration	设定机器人加速度。 设定较低加速度值以配置精确的本地设定。
Local	指定本地编号，编号是与相机图像面平行的XY平面。

点击<Next>按钮。

**第6步: 执行3D工具设置**

请根据需要设置初始旋转和最终旋转。

点击<Execute>按钮开始3D工具设置。当正常完成时，校准完成会显示“**Yes**”。



**注意:** 机器人会根据目标的检测结果自动运作。请注意机器人和周边设备之间的干涉。为避免在工具设置中报错，请尽量不要使用各关节的奇异点附近的位置。

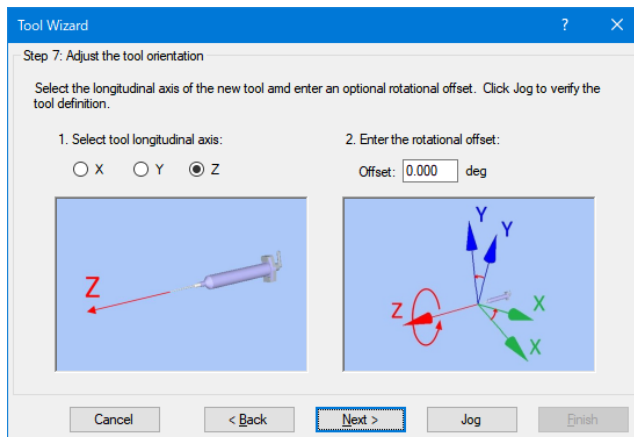
若在第2步中勾选了<Calculate tool orientation (UVW)>选项，请点击Tip、Middle和Root选项卡中的<Execute>按钮，校准完成后会显示“**Yes**”。

点击<Next>按钮。

若在第2步中勾选了<Calculate tool orientation (UVW)>选项，请执行第7步。若未勾选该选项，请执行第8步。

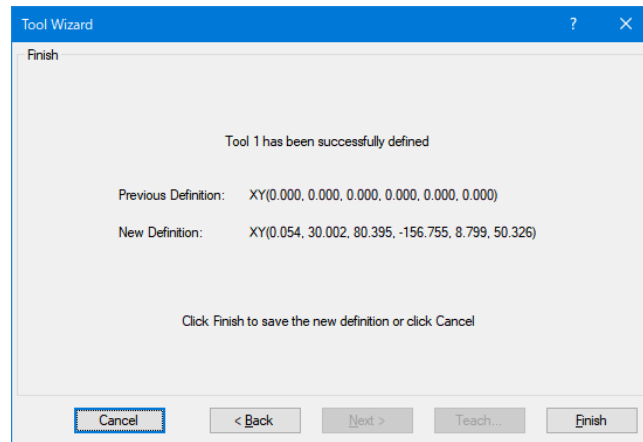
**第7步: 调整工具姿态**

选择工具的纵轴。请根据需要设置纵向的旋转偏移。点击<Jog>按钮打开示教画面，在工具坐标系中确认机器人动作。



### 第8步: 完成3D工具设置

显示结果。请确认结果。



点击<Finish>按钮，将结果设置为指定的工具编号。

## 8. 直方图工具



在使用机器视觉系统时，直方图是一种强大的功能。它们能够使您以图标格式查看数据，以便您更容易地看出问题所在，例如为何没有正确地找出某个零件。此外，由于显示了基于直方图信息计算出来的统计值，用户还能掌握图形的特征。



它们也对一些问题给出提示，例如当深灰色和浅灰色二进制对象同样能够从搜索窗口中找到时，如何寻找中灰色的二进制对象。


### 8.1 利用直方图

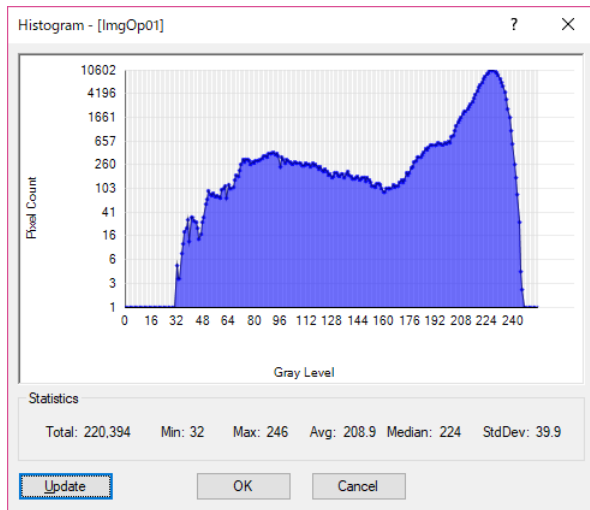
要打开[Histogram]对话框，请首先选择要查看直方图的视觉对象。(可以从流程图或序列目录树中选择)

直方图可用于以下对象。

Blob, Contour, DefectFinder, Correlation, Geometric, Polar, OCR, CodeReader, ImageOp

然后运行视觉对象或视觉序列，以确保您所选择的视觉对象的结果有效。

接下来点击  <Histogram> 工具栏按钮。如果视觉对象是 Correlation，将会出现如 8.2.1 使用相关对象的直方图所示的[Histogram]对话框；如果视觉对象是一个 Blob 对象，将会出现如 8.2.2 使用 Blob 对象的直方图所示的直方图对话框。



在[Histogram]对话框中，横轴代表照明等级，纵轴代表像素值。用户可以很直观的在当前图像中查看每个照明等级显示了多少像素。在直方图的图表中，越靠左(或照明等级越低，越接近 0 的数值)，代表像素的照明等级越暗(越黑)。在直方图的图表中，越靠右(或照明等级越高，越接近 255 的数值)，代表像素的照明等级越亮(越白)。

使用[Histogram]对话框中显示的图像结果，计算以下统计值。

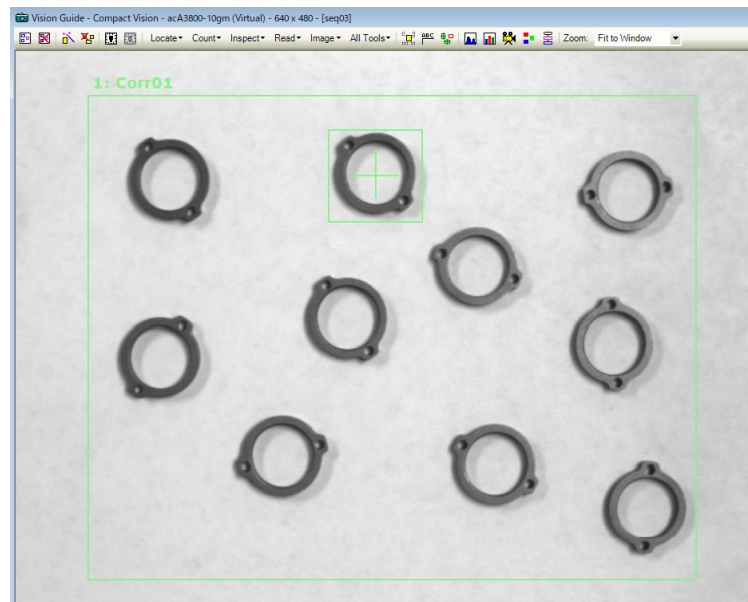
Statistics (统计值)	说明
Total (总像素值)	整个搜索窗口中的像素总和。
Min (最小)	整个搜索窗口中的最低照明等级。
Max (最大)	整个搜索窗口中的最高照明等级。
Avg (平均)	整个搜索窗口中照明等级的平均值。
Median (中间值)	整个搜索窗口中照明等级的中间值。
Std Dev (标准偏差)	整个搜索窗口中，计算照明等级获得的标准偏差

如果是Correlation, Geometric, Polar, OCR, CodeReader, ImageOp (Operation: Binarize 除外)对象，显示的[Histogram]对话框则如8.2.1 *Correlation*对象使用的直方图中所示。

如果是Blob, Contour, DefectFinder, ImageOp (Operation: Binarize时)对象，显示的[Histogram]对话框则如8.2.2 *Blob*对象使用的直方图中所示。

## 8.2 直方图举例

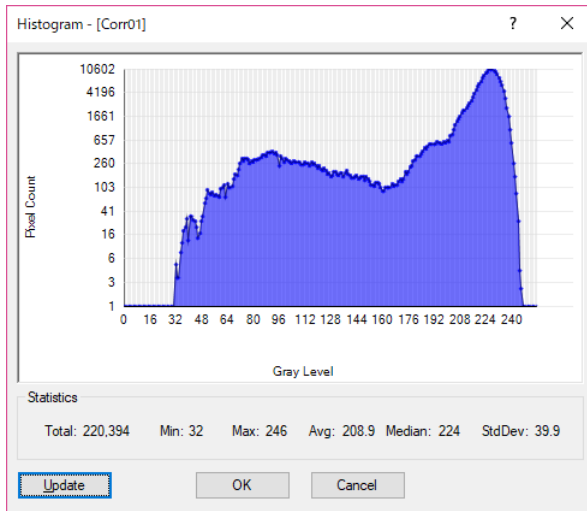
下图显示了浅灰色背景上的 10 个深色圆环。下面将通过举例 Correlation 对象或 Blob 对象检测到这些工件时情况，来说明直方图的使用方法。



10个使用Corr01对象的圆环的图像举例

### 8.2.1 Correlation对象使用的直方图

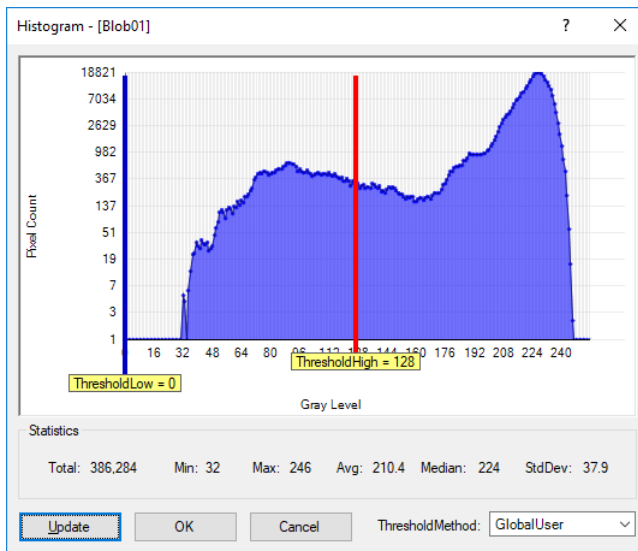
下图展示了为 Corr01 对象生成的[Histogram]对话框。图中的图表显示了像素在各种照明级别上的情况良好，因此我们能看到图像如何分布的。并且可以看出浅色像素多余深色像素(因为 8.2 直方图举例中的图像中，灰色背景的面积比深色圆环大很多)。



Correlation对象“Corr01”的直方图举例

### 8.2.2 Blob对象使用的直方图

下图显示了在Blob对象中的直方图，与Correlation对象的直方图有些许差异。注意图表中有2条垂直线，底部附有各自的值，分别是ThresholdLow和ThresholdHigh。这些是Blob对象的属性，用于指定包含了被检测出二进制部分的照明等级，以及作为背景部分的照明等级。

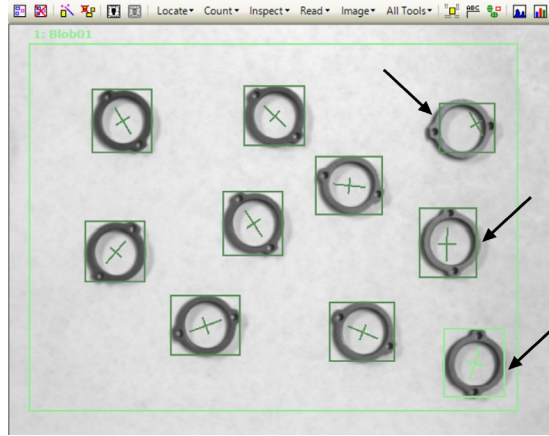


Blob对象“Blob01”的直方图举例(亮底暗图)

ThresholdLow和ThresholdHigh滑动条之间的区域代表我们想要定义为深色还是白色像素的照明等级的像素组合，我们可以利用ThresholdColor属性对此加以设置。Polarity属性决定了我们是在浅色背景上查找深色对象，还是在深色背景上查找浅色对象。

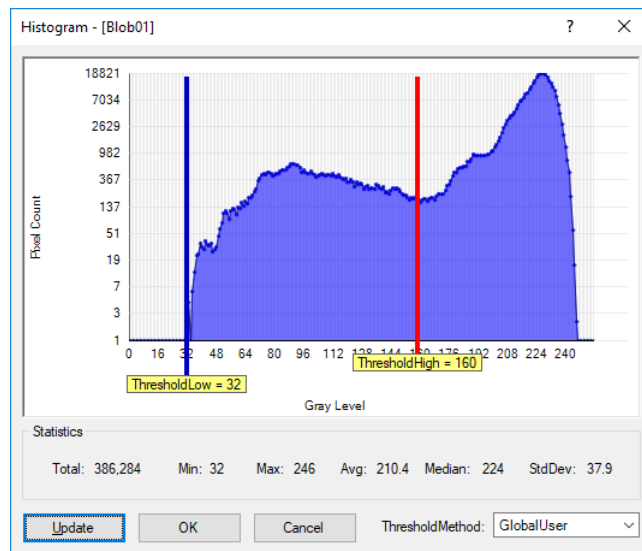
### 调整ThresholdLow和ThresholdHigh属性

再次查看上图。注意，ThresholdLow的属性设置为0，而ThresholdHigh的属性设置为128，它们是这些Blob对象的默认值。当我们首次从Rings图像上运行Blob对象时(NumberToFind设为10)，我们会得到下图所示的结果。注意，许多被发现的二进制对象的Extrema没有围绕在圆环的外部，某些情况下，零件的一部分作为一个二进制对象或1个零件作为2个二进制对象被发现(参见下图中的箭头，这些箭头指出了问题区域)。这是由于尚未根据直方图的结果调整ThresholdLow和ThresholdHigh属性。



阈值设置不佳的10个圆环的图像

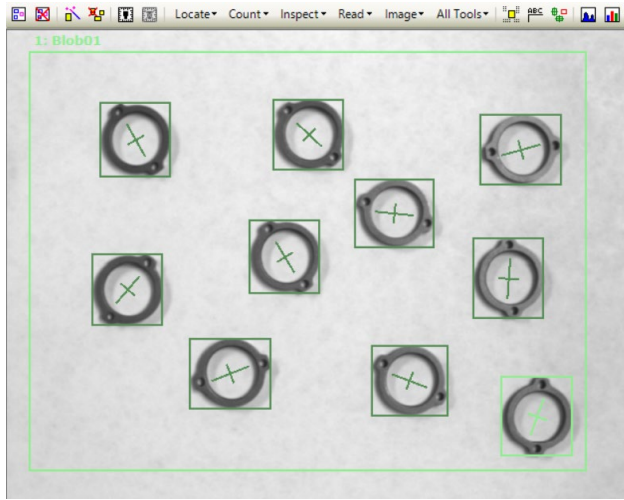
如果我们使用[Histogram]对话框来检查圆环图像的直方图，我们会看到从32灰度级开始的很大一片灰色分布。然后另一个大的高峰从170左右开始。由于圆环图像中最大的灰色分布是浅色背景，则很容易看到170以上的像素分布在背景上。同样，由于直方图中的另一个高峰是从大约32开始到170结束，这显示了图像的深色部分的分布，这就是那些圆环(我们感兴趣的部分)。可以调整ThresholdLow和ThresholdHigh，以便我们找到的每个二进制对象的方框正好环绕在圆环的外缘。通过点击[Histogram]对话框中的ThresholdLow和ThresholdHigh条，并将它们拖到下图所示的位置来完成这一操作。



阈值设置较好的圆环图像柱形图

## 8. 直方图工具

在用了新的 ThresholdLow 和 ThresholdHigh 设置并且运行了 Blob 对象之后来查看圆环图像，我们可以发现，返回的结果更加符合我们的预期。现在发现，对于每个二进制对象，每个圆环都具有正确的极值。



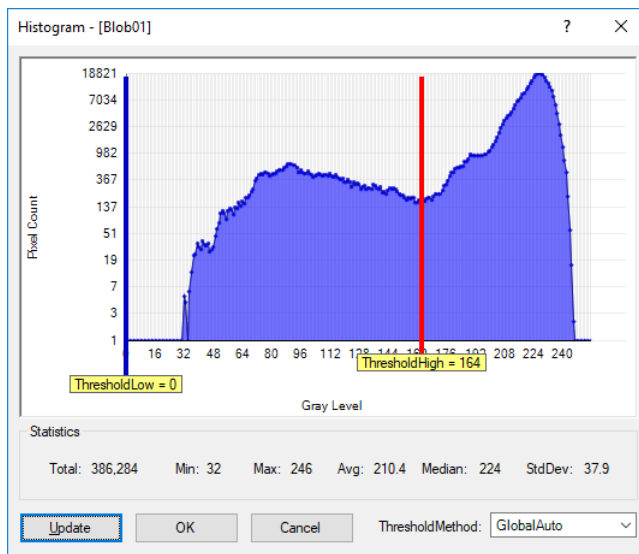
阈值设置改进以后的10个圆环的图像

此外，ThresholdAuto 复选框位于对话框的右下角。勾选该复选框，可以设定从搜索窗口认为恰当的 ThresholdLow 和 ThresholdHigh 值。

在上图的情况下，ThresholdLow 的属性值设为 0，而 ThresholdHigh 的属性值设为 164。ThresholdAuto 复选框的状态与 ThresholdAuto 属性值关联。如果 ThresholdAuto 属性值为“True”，每次执行视觉序列时自动计算阈值。这样，即使在照明系统改变时，二进制对象检测功能也能正常工作。

### NOTE:

如果ThresholdAuto设为“True”，即使在未能捕获目标工件等图像均匀(全部黑色或白色)时，设定值会降至可以检测二进制对象的阈值(至少能检测一个二进制对象)。





## 9. 使用 Vision Guide 统计



Vision Guide 7.0 有一个统计的内置功能，非常实用。

用户每次在 Vision Guide 窗口、Run 窗口或运行窗口中运行视觉对象时，Vision Guide 7.0 都自动保存每个视觉对象的各种结果数据的历史纪录，以供之后用作统计参考。

也就是说，每次运行视觉对象时，无论如何运行都会保存运行的结果数据，以用作该对象结果的统计计算。

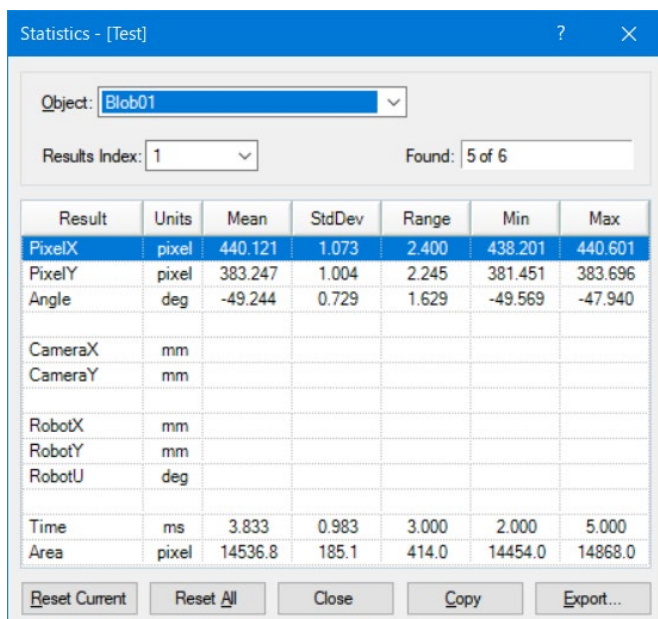
大多是的视觉应用都可以用不同的方法运行，因此用户可以尝试多种方式并比较结果。在这种情况下，统计功能是非常有效的。您能够对视觉对象进行可靠性测试，从而确认哪种视觉最适合您的应用。

您还可以通过 SPEL+语言，或无需编程直接通过 Vision Guide 页面运行这些测试。

点击<Statistics>按钮，即可显示统计对话框。

## 9.1 对话框选项/信息

选项	说明
Object	下拉菜单，用于选择要显示统计结果的对象。
Results Index	下拉菜单，用于选择要显示统计结果的结果编号。
Found	显示对象被找到的次数及运行的次数。例如，5/6表示对象运行了6次，被找到5次。
Reset	清除所选对象的统计信息。 所有数据将被删除。
ResetAll	清除序列中所有视觉对象的统计信息。 所有数据将被删除。
Close	关闭[Statistics]对话框。 所有结果数据保持不变。
Copy	将统计值复制到剪切板。
Export	将统计值导出到CSV文件。



## 9.2 视觉对象和支持的统计

支持下列视觉对象与统计一起使用：

Blob、Correlation、Geometric、Edge、Line、Point、Polar、CodeReader、ColorMatch、BoxFinder、CornerFinder、LineFinder、LineInspector、ArcFinder、ArcInspector、DefectFinder 和 Coordinates

以下统计可从[Statistics]对话框中获得：

统计	说明
Mean	计算出的指定对象的所有找到的对象的平均值。
Standard Deviation	“Sample Standard Deviation”，主要是 $(\sum(xi - mean)^2/n - 1)$ 的平方根。只有找到的对象结果用于该计算。
Range	结果的范围(最小值——最大值)。仅为找到的对象进行这一计算。
Min	指定对象的所有找到的对象的最小结果。
Max	指定对象的所有找到的对象的最大结果。

## 9.3 支持的视觉对象结果

下列视觉对象结果显示在[Statistics]对话框中。并非为每个对象类型返回所有结果。请参见 *Vision Guide 7.0 Properties and Results Reference Manual* 中的单个结果说明，以获取更多信息。

结果	说明
PixelX	像素坐标中已找到的零件位置的X坐标位置。
PixelY	像素坐标中已找到的零件位置的Y坐标位置。
Angle	像素坐标中已找到的零件的旋转量。
Area	找到的用于二进制对象的连接像素的数目。
CameraX	相机坐标系中已找到的零件位置的X坐标位置。
CameraY	相机坐标系中已找到的零件位置的Y坐标位置。
Length	Line对象的长度，以毫米为单位。 (需要校准)
PixelLength	Line对象的长度，以像素为单位。 (不需要校准。)
RobotX	返回被检测出工件在机器人坐标系中的X坐标位置。
RobotY	返回被检测出工件在机器人坐标系中的Y坐标位置。
RobotU	返回被检测出工件在机器人坐标系中的U坐标位置。 (工件在机器人坐标系中的旋转方向)
Time	返回执行该视觉对象所花费的时间。 (找到工件所需的时间。)
Score	这是一种度量，用于衡量特定零件在多大程度上被找到。(一个Correlation对象跟其模型的匹配度，或者一个边缘对象的转变强度。)

## 9.4 通过SPEL+可用的视觉对象统计

下列视觉对象统计结果可通过 SPEL+语言获得。

视觉对象	经 SPEL+支持过的统计结果
Blob	AngleMax, AngleMean, AngleMin, AngleStdDev AreaMax, AreaMean, AreaMin, AreaStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Correlation	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Geometric	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Edge	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Polar	AngleMax, AngleMean, AngleMin, AngleStdDev

视觉对象	经 SPEL+支持过的统计结果
	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Line	AngleMax, AngleMean, AngleMin, AngleStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelLengthMin, PixelLengthStdDev
Point	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev
CodeReader	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
ColorMatch	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
BoxFinder	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
CornerFinder	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev

视觉对象	经 SPEL+支持过的统计结果
	CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
LineFinder	AngleMax, AngleMean, AngleMin, AngleStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
LineInspector	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
ArcFinder	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
ArcInspector	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev TimeMax, TimeMean, TimeMin, TimeStdDev

视觉对象	经 SPEL+支持过的统计结果
DefectFinder	AngleMax, AngleMean, AngleMin, AngleStdDev AreaMax, AreaMean, AreaMin, AreaStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Coordinates	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev

## 10. 教程

### 10.1 快速启动：Vision Guide 7.0教程

#### 10.1.1 教程概述

本章的目的是带领您领略简单的视觉应用，以便向您介绍 Vision Guide 7.0 的一些基本使用概念，并向您展示它的使用是多么简单。在许多情况下，我们将仅仅对需要遵循的步骤进行说明，但不会介绍每个步骤的细节。您可在后续的章节中了解这些细节。

本教程将不会使用真实的零件，作为代替，我们将使用对象的简单图纸，因此您可以对图纸进行复制，并将其放在相机下，以学习本教程。这样做将会保证在实践本教程时，每个人都会得到相同的结果。

本教程将向您展示如何创建一个简单的应用，以利用视觉系统寻找零件，并使机器人向零件移动。就本教程而言，假设您的机器人为 SCARA 型，并且在机器人第二个臂杆的端部安装有相机。

本章将包括以下几个小节：

- 本教程所需的物品。
- 相机镜头配置。
- 创建一个新的 EPSON RC+ 7.0 项目。
- 新建一个视觉序列。
- 利用 Blob 对象来寻找零件。
- 编写一个 SPEL+程序来与视觉序列交互。
- 用相机来校准机器人。
- 利用视觉系统指示机器人向零件移动。
- 找到并移动到多个相似零件。

在本章后面的 *本教程所需的物品* 中，您将看到有两个页面，上面印有本视觉教程当中将使用的目标。这些页面的使用将贯穿整个教程。

跟在目标页面后面的是标题为 *启动EPSON RC+ 7.0并创建一个新项目* 的章节。我们将从此开始实际的教程。



### 10.1.2 本教程所需的物品

本教程将通过 EPSON 机器人与 Vision Guide 7.0 的联用来对您进行指导。我们假定您能够欣然接受 EPSON RC+ 7.0 与 EPSON 机器人的联用。如果您还不确定如何使用 EPSON RC+ 7.0，您可能想在开始本教程之前花一点时间进行回顾。完成本教程将需要以下物品：

- 必须将 EPSON RC+ 7.0 安装在您的电脑上。
- 必须为 Vision Guide 7.0 安装一台相机并保证相机正常工作。
- 将 EPSON 机器人安装在底座上，该底座需有一个工作表面，并且该工作表面应连接在机器人正面，以便机器人能够使用此工作表面。
  - 如果机器人是 SCARA 机器人，应将相机连接到机器人第二个臂杆的端部。如果机器人是 6 轴机器人，则应将相机连接到机器人的第 6 机械臂法兰。可以从 EPSON 读取您所使用的特定机器人的相机安装支架。请镜头朝下安装相机。

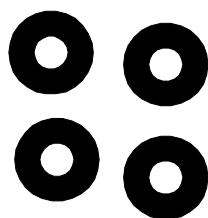
**NOTE**

连接计算机、相机、控制器和以太网交换机之间的以太网电缆时，确保计算机的主电源、控制器电源和相机电源已经关闭。未关闭电源可能造成损害。

- 呈现目标零件图纸的页面的副本应当能够放在相机下。(目标零件图纸见紧接本列表之后的小节。)
- 必须选定可以满足 40mm × 30mm 左右视野要求的相机镜头。在我们的测试中，我们一直使用带有 16mm 镜头的 Vision Guide 7.0 可移动式相机(带 NET 1044BU 相机的 CV1)，相机距离工作表面大约 210mm。然而，为得到合适的焦距，需要做一些调整。我们将指导您完成这些调整。
- 在本教程中我们不会拾取任何零件。但是我们会将机器人移动到我们画在靶纸上的零件位置。您需要将一个夹具或连杆连接到 Z 轴上。直接连接在 Z 轴中心下方的连杆将会很好地完成工作。在校准过程中，以及在向目标页面上目标零件移动的过程中，将用到该夹具或连杆。



教程所用的单个目标特征(图纸上画的是一个垫圈)





教程所用的多个目标特征(多个垫圈)

### 10.1.3 启动EPSON RC+ 7.0并创建一个新项目

- (1) 开启您的控制器。
- (2) 双击 Windows 桌面上的<EPSON RC+7.0>图标来启动 EPSON RC+7.0。
- (3) 点击 EPSON RC+7.0 菜单栏上的 Project 菜单。
- (4) 点击 Project 项目菜单的 New Project 菜单条目，会显示 New Project 对话框。
- (5) 键入新项目的名称。我们推荐您使用自己的名字，并在后面附上字母"tut"。以便试用本教程的每个人都能够为其项目建立一个独特的名称。我们此处的项目将会用到名称“vgtut”。您在名称字段键入了名称之后，点击<OK>按钮。至此，您已创建了一个名叫 xxxxtut 的新项目。

### 10.1.4 新建一个视觉序列

- (1) 一旦您创建了新的 EPSON RC+7.0 项目，您将会注意到许多的工具栏图标都启用了(不再呈灰色)。找到  <Vision>工具栏按钮。点击视觉工具栏按钮以打开 Vision Guide 窗口。
- (2) 我们在使用 Vision Guide 7.0 窗口之前，必须首先创建一个视觉序列。通过点击  <New Sequence>工具栏按钮来完成新视觉序列的创建。(该工具栏按钮位于 Vision Guide 窗口工具栏上，而不是在 EPSON RC+7.0 主工具栏上。)此时将出现 [New Sequence]的对话框。
- (3) 输入新视觉序列的名称“blobtut”并点击 OK 键。之后我们在 EPSON RC+7.0 的代码中将使用名称“blobtut”，因此须确保按照以上的拼写准确键入该名称，不带引号。现在我们已经创建了一个名为“blobtut”的新视觉序列，并且从现在起我们将围绕该视觉序列进行操作。

### 10.1.5 教程的相机镜头配置

在上述阐述中，教程里的目标视野约为 40mm × 30mm 左右，WD(工作距离)约为 210mm。可以得出结论，16mm 镜头和 1mm 伸缩管匹配的焦点 WD 为 167mm 到 240mm 之间。因此，本教程中使用 16mm 镜头和 1mm 伸缩管。

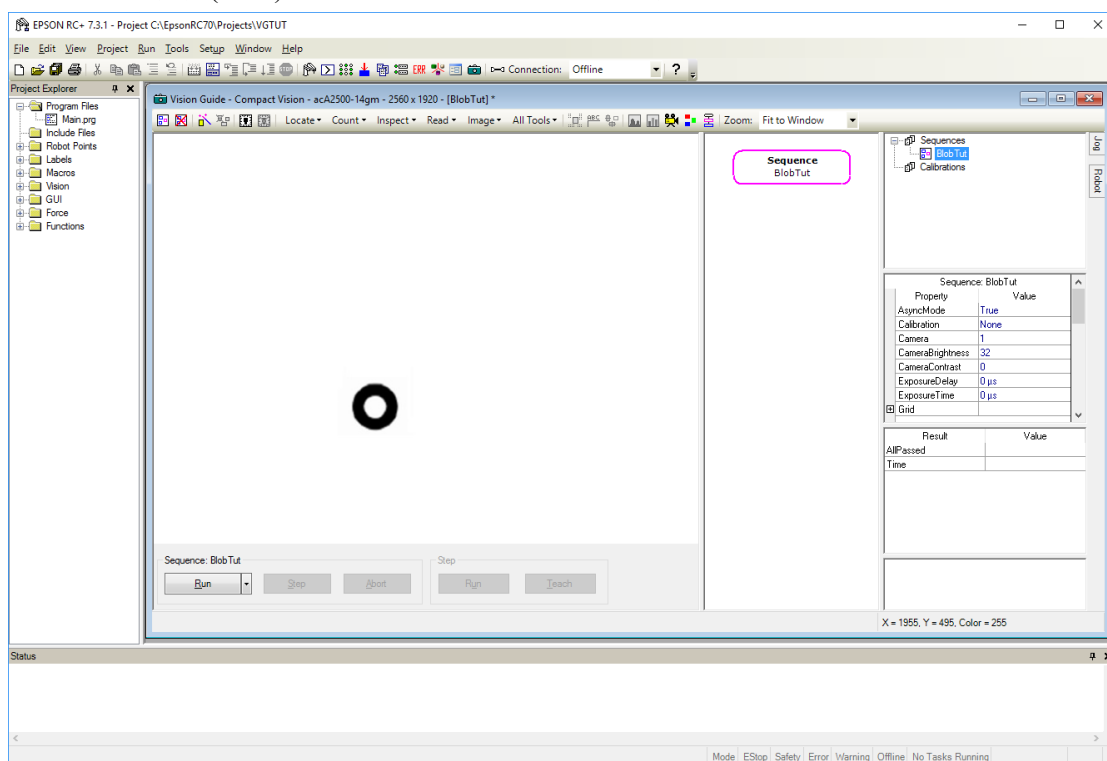
如果您没有为您的相机安装镜头，并且尝试给镜头对焦，此时这样做不失为一个好办法。在安装章节中有关于如何确定使用哪个镜头和如何对焦的说明。如果您不熟悉如何选择相机镜头以及如何相机和您的零件间对焦，在进行下一步之前您可能想回顾一下本信息。

为了检查您的对焦情况，很明显您需要通过 Vision Guide 7.0 来观察图像。由于我们已经打开了 Vision Guide 窗口，现在我们只需要确保我们能够准确看到靶纸上的目标特征。其完成步骤在下一章节中有所描述。

### 在靶纸上定位可移动式相机并调整镜头的焦距

- (1) 拿到您之前复制的靶纸。选择中心印有单独垫圈图样的靶纸并将它放在工作表面的适当位置，以便机器人能够很轻易地将相机放在上面。最好的位置通常为机器人的正前方。
- (2) 现在移动机器人，以便相机位于垫圈图片的上方。您应能手动移动机器人，而无需打开伺服电源。
- (3) 您应该能在 Vision Guide 窗口的图像显示区域看到目标特征(垫圈)。通过调整相机镜头焦距来使目标特征准确对焦。如果您不能看到目标特征或无法将其正确对焦，查看 安装 一章中名为 *正确焦距的检查和调整* 的小节。此节将详细说明相机镜头的选择和对焦。

现在，相机应位于画在靶纸上的目标特征的正上方。您应该能在 Vision Guide 窗口的图像显示区域清楚地看到目标特征。现在 Vision Guide 窗口应如下图所示。注意，目标特征(垫圈)显示在 Vision Guide 窗口的图像显示区域的中心处。



在图像显示区域中心显示目标特征的Vision Guide窗口

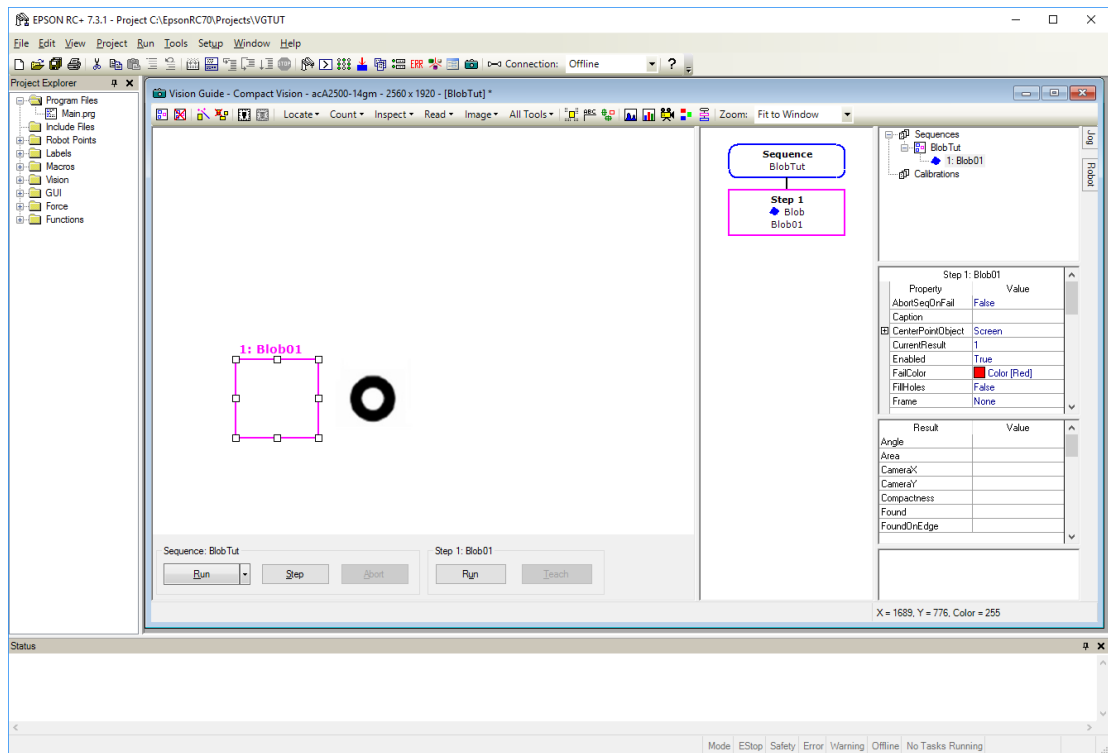
### 10.1.6 利用Blob对象来寻找零件

既然目标特征(垫圈)位于 Vision Guide 窗口图像显示区域的中心，我们便可以创建一个 Blob 对象来找到垫圈。创建 Blob 对象、对其进行配置然后通过其找到垫圈的步骤如下所示。

#### 第1步：创建一个新的Blob对象

- (1) 点击 Vision Guide 工具栏中的<All Tools>-<Blob>按钮。点击了<Blob>按钮后，请松开鼠标左键。
- (2) 现在将鼠标向下移到图像显示区域中心的垫圈图像上。离开 Vision Guide 工具栏后，鼠标指针将变为 Blob 对象图标。
- (3) 继续将鼠标移动到图像显示区域的中心，然后点击鼠标左键。以上操作将完成新 Blob 对象在图像显示区域中的定位。

此时屏幕显示如下。



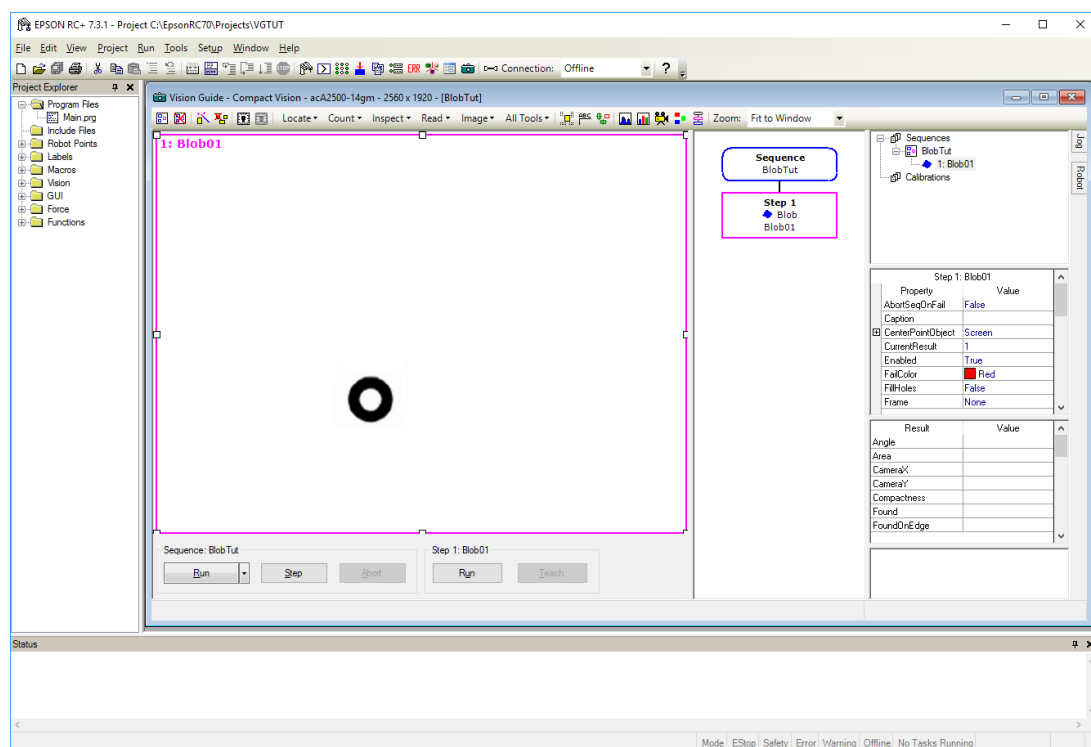
显示新Blob对象“Blob01”的Vision Guide窗口

## 第2步：确定Blob对象位置及尺寸

现在我们需要为 Blob 对象的搜索窗口进行定位并规定其尺寸。Blob 对象“Blob01”的搜索窗口为下图中垫圈左边的框。我们将搜索窗口放大，以便能够在几乎整个视野内搜索垫圈。

- (1) 把鼠标移动到 Blob 对象的名称标签上，并按下鼠标左键。在持续按住鼠标左键的同时，将 Blob 对象拖到图像显示区域的左上角，使得搜索窗口的左上角几乎接触到图像显示区域的左上角。
- (2) 将鼠标移动到 Blob01 搜索窗口的右下尺寸柄上，并按下鼠标左键。在持续按下鼠标左键的同时，将右下尺寸手柄拖动到图像显示区域的右下角。现在 Blob 对象的搜索窗口应该覆盖整个图像显示区域。这使我们能够找到所有位于相机视野以内的二进制对象。

下面的画面截图显示了重新定位和确立尺寸的 Blob 对象“Blob01”的搜索窗口，此时其覆盖整个图像显示区域。



搜索窗口放大后的Blob对象“Blob01”

### 第3步：设置属性并运行Blob对象

现在搜索窗口已经大到能够看到搜索窗口中的垫圈。现在我们准备好来测试 Blob 对象，以确保能够找到垫圈。

- (1) 点击位于 Vision Guide 窗口右上侧树中的 Blob01 对象。此举将显示 Blob01 对象的属性和结果。

The screenshot displays the 'Step 1: Blob01' configuration window. It is divided into three main sections: a properties table, a results table, and a caption field.

Property	Value
AbortSeqOnFail	False
Caption	
CenterPointObject	Screen
CurrentResult	1
FailColor	<span style="color: red;">■</span> Color [Red]
Frame	None
FrameResult	1
Graphics	All

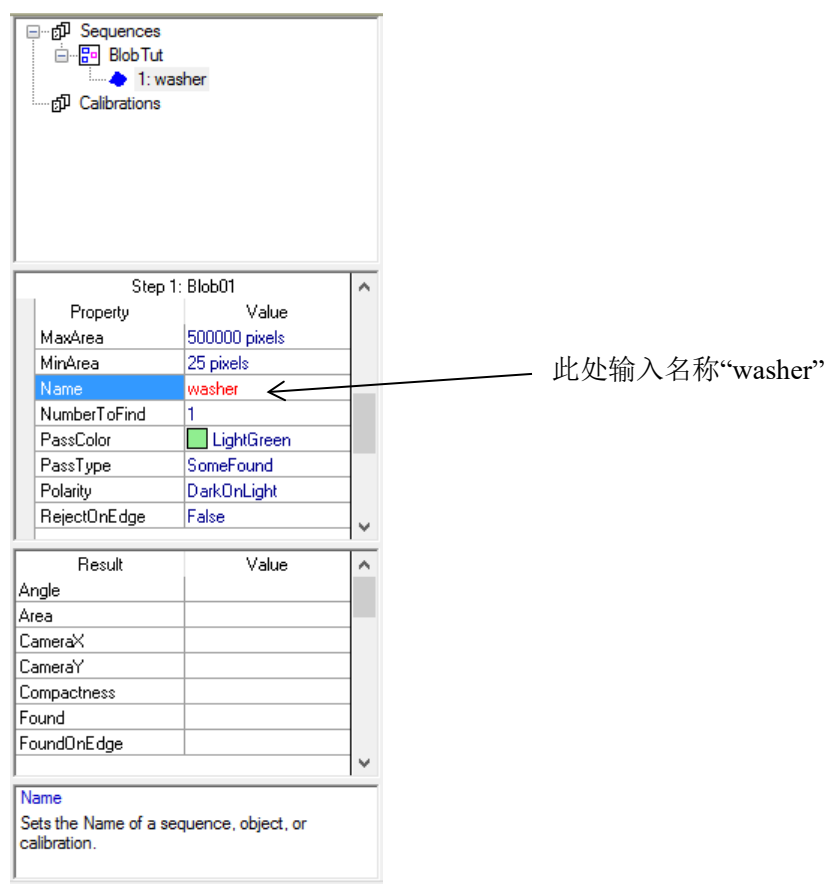
  

Result 1 of 1	Value
Angle	
Area	
CameraX	
CameraY	
Compactness	
Found	
FoundOnEdge	

**Caption**  
Optional caption for the object that replaces the default caption



- (2) 查看属性列表并查找 Name 属性。双击 Name 属性的 Value 字段以突出显示当前的名称。现在输入名称“washer”。我们就将 Blob 对象的名称改成了“washer”。查看 Name 下拉菜单中[Jog]选项卡顶部以及搜索窗口上的名称标签，您会发现两处的名称都更改了。



#### Blob对象的属性

- (3) 在您查看 Properties 列表时，您可能也想检查 Polarity 属性。由于我们刚刚创建了一个新的二进制对象，Polarity 的默认值为 DarkOnLight，表示在浅色背景上找到了一个深色二进制对象。我们也可以将该属性更改为 LightOnDark。但是由于我们希望在浅色背景上找到一个深色二进制对象(垫圈)，因此保持 Polarity 属性不变。
- (4) 现在我们准备好运行 Blob 对象“washer”。为运行该对象，点击位于运行面板右下方的 Run 按钮。此举将会执行 Blob 对象“washer”，并且在这种情况下，Blob 对象将会找到一个看起来像垫圈的二进制对象。您可在运行了 Blob 对象之后，通过检查搜索窗口的颜色，确定已经找到了 Blob。如果找到了 Blob，则搜索窗口呈绿色；如果没有找到则呈红色。(您也可以通过检查 Results 列表来查看是否找到了 Blob 接下来有更多内容)
- (5) 现在试着轻微移动印有目标垫圈的纸张，然后再次点击 Run 按钮。确保垫圈保持在您创建的搜索窗口中。您可以看见二进制对象的新位置已经找到并在图像显示窗口中用绿色加亮。(如果您移动纸张使得垫圈位于搜索窗口边界以外，则 Blob 对象将找不到垫圈，您会发现这一情况是因为在没有找到对象时，搜索窗口将变成红色。)

#### 第4步：检验结果

既然您已经运行了名为“washer”的 Blob，您可以检验该对象的返回结果。结果显示在 Results 列表中，正好处于属性列表之下。

- (1) 在 Results 列表中找出名为 Found 的结果。这时，Found 结果的值应为 True，因为我们刚刚找到了二进制对象。如果没有找到二进制对象，Found 结果将显示为 False。
- (2) 您也可以参见 Results 列表顶部的 Area 结果。此处显示了找到的二进制对象的面积。
- (3) 使用滚动条将结果列表移动到底部。在 Blob 结果列表的底部您可以看到 Time 结果。该结果告诉我们找到此二进制对象花了多少时间。

Step 1: Blob01	
Property	Value
MaxArea	500000 pixels
MinArea	25 pixels
Name	washer
NumberToFind	1
PassColor	<span style="color: lightgreen;">█</span> LightGreen
PassType	SomeFound
Polarity	DarkOnLight
RejectOnEdge	False

Result 1 of 1	Value
RobotX	(no cal) mm
RobotY	(no cal) mm
RobotU	(no cal) deg
Roughness	1.573
ShowAllResults	Click to show
Time	24 ms
TotalArea	27398.0 pixels

Time  
Indicates the amount of execution time in milliseconds.

点击此处的滚动条（或使用滑块）  
来向下滚动至 Results 列表

结果列表向下滚动以显示Time结果




注意

- 周围照明和外部设备噪音可能会影响视觉序列的图像和结果。当受到周围照明和外部设备噪音的影响，拍摄的图像可能会损坏，并且检测到的位置可能是对象搜索区域中的任何位置。请尽量缩小搜索区域，并在创建图像处理序列时，考虑到检测错误的情况。

#### 第5步：保存您的视觉序列

此时我们可能需要保存我们所做的工作。同所有应用一样，经常定期地保存您的工作是一个不错的办法。EPSON RC+ 7.0 的 Project Management 功能使与该项目相关的每个事项都能立刻进行保存。可以通过下面的一个步骤轻松完成：

点击  <Save> 工具栏按钮。此按钮位于 EPSON RC+ 7.0 主工具栏的左侧。若您的项目发生任何变更，<Save> 工具栏按钮将以蓝色显示。


现在我们已经成功创建了新的 Blob 对象，正确设置了搜索窗口及其尺寸，并且运行 Blob 对象序列来寻找二进制对象。让我们进入下一步，并编写一个简单程序，以通过 SPEL+ 语言来运行视觉序列，并对我们的应用中将用到的一些结果进行检索。

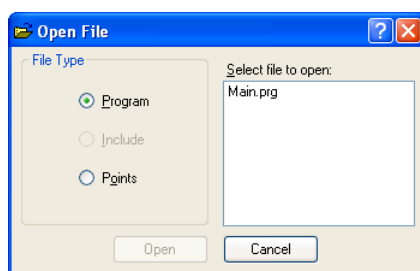
### 10.1.7 编写一个SPEL+程序来与视觉序列联用

Vision Guide 7.0 最有力的功能之一是通过任何通过点击建立的视觉序列都可以通过 SPEL+语言进行应用。这表明视觉序列为 EPSON RC+视觉应用的核心部分，而不仅仅是之后必须用 SPEL+进行重写的原型设计工具。视觉序列和 SPEL+语言集成在一起，为您提供两者的最佳组合：视觉开发点击环境的轻松使用，以及一种语言所提供的力量和灵活性。让我们快速编写一个程序来了解是否找到了二进制对象，检查二进制对象的面积，然后打印出提供此信息的那条消息。

#### 打开名为MAIN.PRG的程序文件

因为您已熟悉了 EPSON RC+ 7.0 的环境，所以您应该已经知道如何打开 main.prg 程序文件。然而，我们还是列入了以下基本步骤，以防您不知道如何打开文件：

- (1) 点击 EPSON RC+ 7.0 工具栏上的<Open File>工具栏按钮。此举将打开包含有名为 main.prg 的程序的[Open File]对话框，该对话框在您首次创建新项目时自动创建。您将会看到以下[Open File]对话框：



EPSON RC+ Open File对话框

- (2) 正如您在[Open File]对话框中看见的，因为您还没有创建任何其他项目，所以 main.prg 程序文件为突出显示状态。继续并点击对话框左下的<Open>按钮以便打开 main.prg 程序文件。

#### 创建一个SPEL+程序来与视觉序列联用

以下为示例程序，该程序将运行视觉序列“blobtut”，并检查其中一些与名为“washer”的 Blob 对象相关的属性。例如我们会检查是否找到了二进制对象。如果找到了，我们将显示一条“The washer was found!”的消息以及二进制对象的面积。如果没有找到，我们将显示一条“The washer was not found!”的消息。

现在您应看到在标题栏上显示有 MAIN.PRG 的编辑窗口。光标将位于编辑窗口中第一行的开始部分。(开始键入程序的最佳位置。)继续操作，将以下程序输入编辑窗口。不用担心字母的大小写。编辑器将会自动把所有关键词切换为大写。


```
Function main
  Real area
  Boolean found

  VRun blobtut
  VGet blobtut.washer.Found, found

  If found = True Then
    VGet blobtut.washer.Area, area
    Print "The washer was found!"
    Print "The washer area is: ", area, " pixels"
  Else
    Print "The washer was not found!"
  EndIf
Fend
```

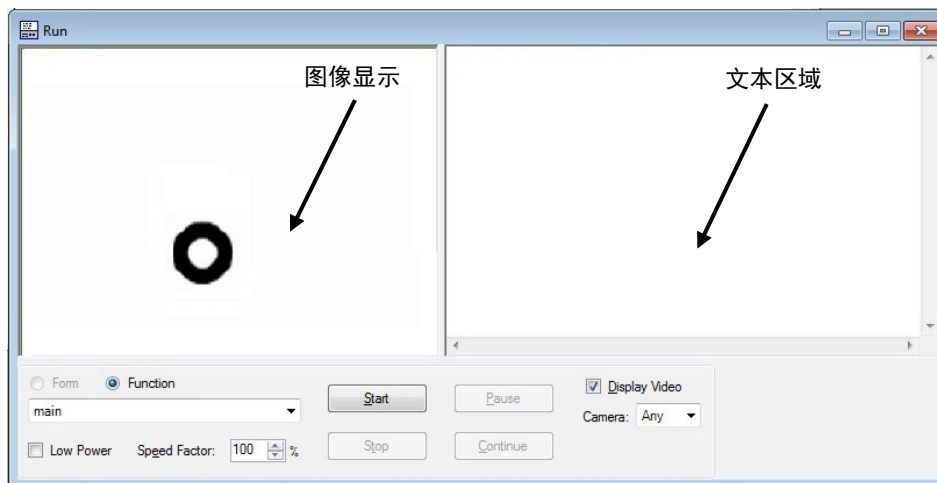
### 运行主要功能

您应熟悉 EPSON RC+ 7.0 的 Run 窗口。我们将使用 Run 窗口来运行前一节中创建的示例程序。完成这一任务的步骤如下所示：

- (1) 点击 EPSON RC+ 7.0 工具栏上的  <Run> 工具栏按钮。随后将出现 Run 窗口。注意下图中，Run 窗口分为两部分。Run 窗口的左半部分为图像显示区域(注意中间显示的垫圈)，Run 窗口的右半部分为文本区域，用于显示文本消息。

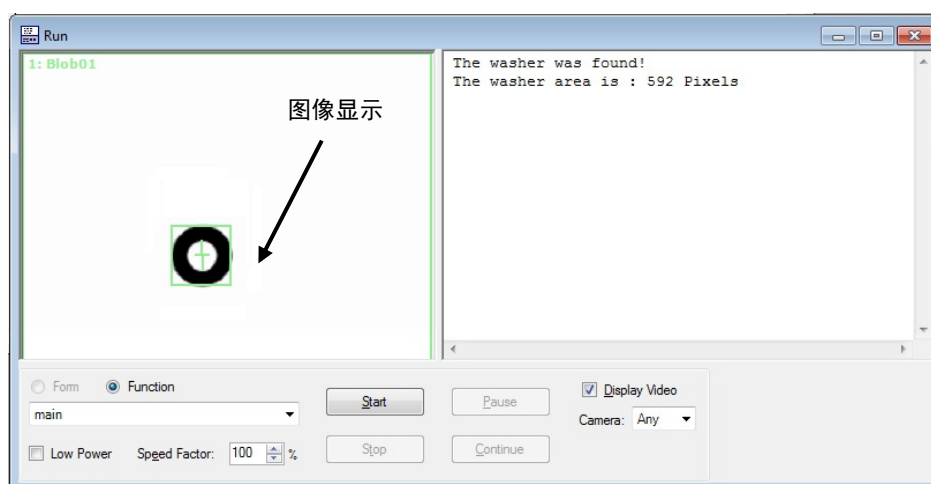


如果 Run 窗口只显示了文本区域，而没有同时出现图像显示区域和文本区域，则点击 Run 窗口上的 Display Video 检查框。



有图像显示区域和文本区域的Run窗口

- (2) 点击位于 Run 窗口左下角的 Start 按钮。此举会使名为“main”的功能运行。
- (3) 通过运行“main”功能，您可以在步骤(4)的图中看到示例结果。注意，找到了二进制对象以后，二进制对象在 Run 窗口左侧的图像显示区域中用绿色突出显示。Run 窗口右侧的文本说明二进制对象已被找到，同时还说明二进制对象的面积。
- (4) 双击位于 Run 窗口左上角的 Run 窗口 Control 菜单框。随后将关闭 Run 窗口。



运行“main”之后的Run窗口示例


### 10.1.8 校准机器人相机

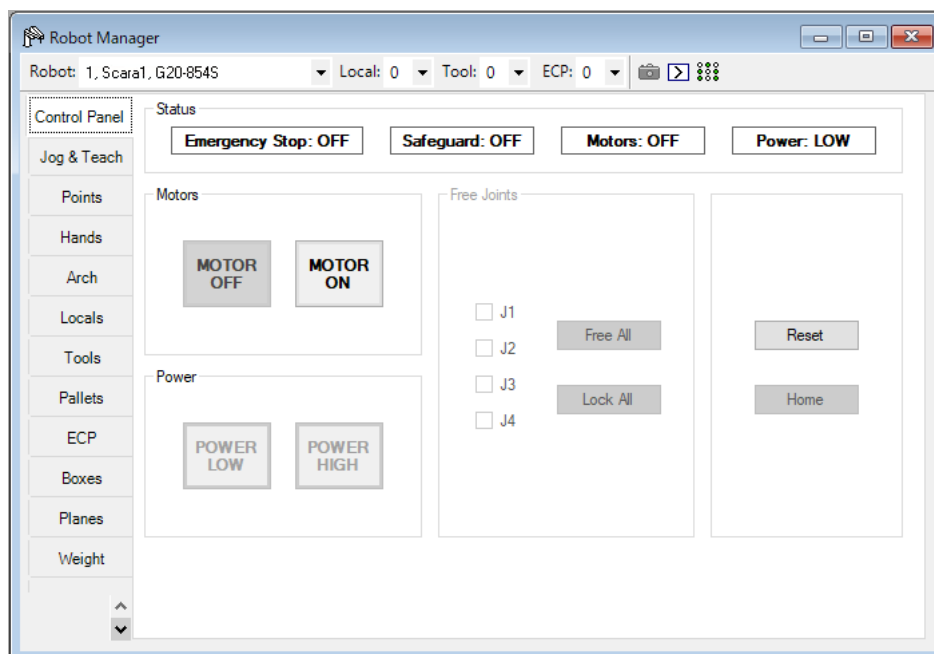
本节将介绍，如何通过将可移动式相机，安装在水平多关节机器人的第 2 机械臂上，来校准机器人。

进行校准后，视觉系统可以自动计算机器人坐标系和相机坐标系，并执行坐标转换。完成校准后，可以使用视觉系统，检测出机器人将取放的工件的位置。

在使用可移动式相机，进行机器人校准之前，需要先做好机器人的准备工作，如打开点击和设置工具的值等。

#### 第1步：开启电机

- (1) 点击 EPSON RC+ 7.0 主工具栏上的  <Robot Manager> 工具栏按钮。您会看到出现以下对话框。



机器人控制面板

- (2) 点击机器人控制面板[Motors]组中的<MOTOR ON>按钮。
- (3) 消息框上会出现询问您是否“准备好打开机器人电机”的消息。点击<Yes>按钮。开启机器人的电机。

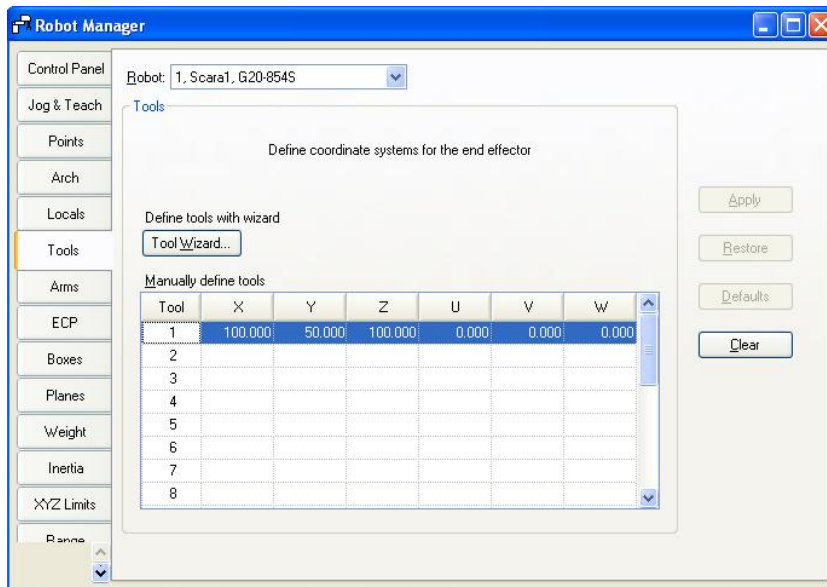
## 第2步：利用Tool Setting来创建一个新工具

为了将带有视觉系统的机器人用于指导应用，需要精确地测量工具或安装在臂端的夹具的位置。一些工具偏离于 Z-轴法兰的中心，而其它工具的安装则与 Z-轴的中心对准。在安装机器人工装时，不论我们多么仔细，也几乎可以肯定有偏离 Z-轴的情况存在。因此我们必须使用 SPEL+的 Tools 功能来进行补偿。

要使用 Vision Guide 来引导机器人，必须对工具有深入的理解。我们将使用 Tool Wizard(工具向导)功能来创建一个工具，为我们的校准杆(夹具)所用。

- (1) 点击[Tools]选项卡。
- (2) 点击<Tool Wizard>按钮。
- (3) 对于 Tool Wizard 中的第一点，步进机器人至目标位置(纸上的垫圈位置)，以便杆或夹具准确位于垫圈的中心。您需要向下步进机器人，以便您能够十分靠近靶纸，但又没有接触到靶纸。靶纸上方 5-10mm 空间内的任何位置都可以。您只需按要求尽可能地靠近靶纸，以便能够很好地看到杆(或夹具)，然后步进至垫圈的中心。
- (4) 对于 Tool Wizard 中的第二点，将 U 轴步进大约 180°，然后步进杆(或夹具)直到其居于垫圈的中心。
- (5) 点击<Finish>按钮以确定新工具并退出 Tool Wizard。至此，我们已经确定了 Tool1 的参数。

教程中 Tool1 的新工具值将如下图所示。



[Tools]选项卡(用于定义工具)



### 第3步：测试新工具

选择[Jog&Teach]选项。由于此时还没有步进机器人，所以夹具应该位于目标位置(垫圈中心的上方)。

- (1) 为防止产生干涉，先将夹具远离工作表面。点击(+Z)的步进按钮，将第 3 轴(J3)沿着 Z 方向步进 10~15mm。
- (2) 在[Jog&Teach]窗口的右上角，点击工具下拉列表的向下箭头。
- (3) 点击“1”，选择工具 1。
- (4) 点击(+U)或(-U)的步进按钮，确认 U axis 第 4 轴(J4)旋转时，夹具也不会偏移。(这种情况经常出现在第 3 轴(J3)中心偏移工具中。如果夹具从第 3 轴(J3)垂直向下，可能会很难察觉到这个现象。)
- (5) 当第 4 轴(J4)移动时，会看见夹具从目标位置稍微移动了一些，但当完成步进执行后，夹具会回到目标位置。
- (6) 如果发现工具并没有正常动作，请返回步骤 2 重新调试。
- (7) 关闭机器人管理器。完成定义并测试工具。

### 第4步：启动相机校准程序

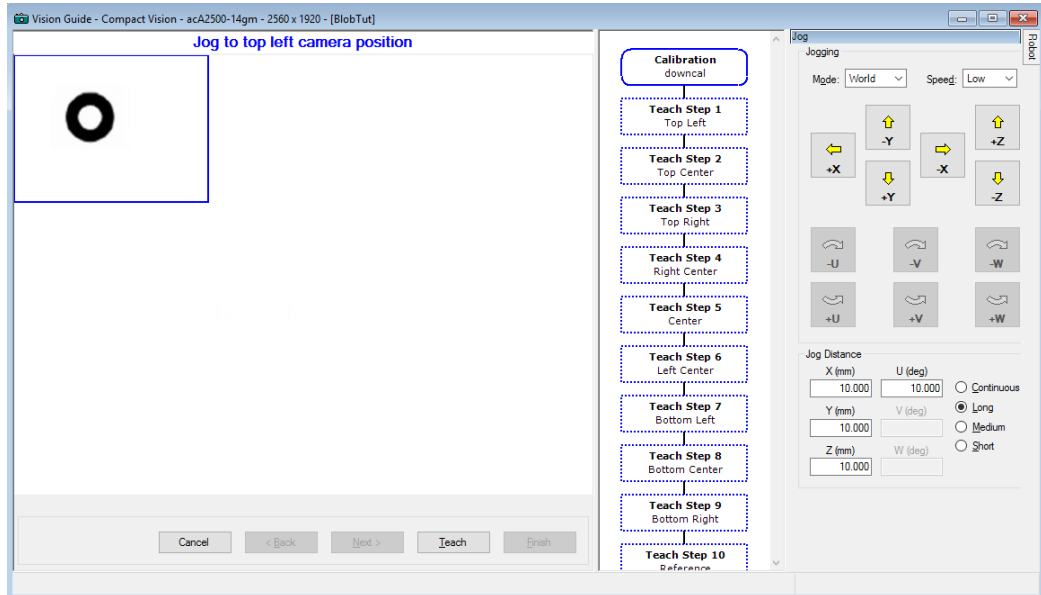
现在我们已准备好用可移动式相机来对机器人进行校准。

- (1) 点击 EPSON RC+ 7.0 工具栏上的  <Vision>工具栏按钮，使视觉开发窗口回到屏幕前方。
- (2) 点击 Vision Guide 窗口中的  <New calibration>工具栏按钮。将打开[New Calibration]对话框。
- (3) 输入名称“downcal”，并点击 OK 按钮。
- (4) 在[Calibration]窗口的属性列表中将 CameraOrientation 属性设为“Mobile J2”。
- (5) 将 RobotTool 属性更改为 1，这样一来便将在示教校准基准点时选用 Tool 1。
- (6) 将 TargetSequence 属性设置为 blobtut。该序列将用于在校准过程中寻找零件。



### 第5步：示教校准点

- (1) 点击位于视频显示下方的<Teach Points>按钮。
- (2) 如下图所示，Vision Guide 窗口切换到 Teach Points 模式。请注意，在 Vision Guide 窗口顶部将看见一则消息，要求您“Jog to top left camera position”，如下图所示。这表示将机器人(和相机)步进至能在 Vision Guide 窗口图像显示区域左上角看见垫圈的位置。下图显示了通过步进方式使机器人到达的大致位置。如下图所示，此为需要校准的 9 个相机位置中的第一个位置。



#### 示教需校准的相机位置

- (3) 示教第一个相机位置：步进机器人，以便使相机位于能够在 Vision Guide 窗口图像显示区域左上角看见垫圈的位置。
- (4) 点击 Vision Guide 窗口上的 Teach 按钮。
- (5) 示教第二个相机位置：步进机器人至第一个相机位置的右边，以便使相机位于能够在 Vision Guide 窗口图像显示区域顶部中间看见垫圈的位置。
- (6) 点击 Vision Guide 窗口上的 Teach 按钮。
- (7) 示教第三个相机位置：步进机器人至第二个相机位置的右边，以便使相机位于能够在 Vision Guide 窗口图像显示区域顶部右侧看见垫圈的位置。
- (8) 点击 Vision Guide 窗口上的 Teach 按钮
- (9) 示教第四个相机位置：从第三个相机位置向下步进机器人，以便使相机位于能够在 Vision Guide 窗口图像显示区域中间偏右处看见垫圈的位置。当您按顺序参考所有的相机位置时，该位置将在第 3 个相机位置之下，这个位置为 Z 字型图案的起点。
- (10) 点击 Vision Guide 窗口上的 Teach 按钮。



- (11)示教第五至第九个相机位置：继续根据 Vision Guide 窗口底部的指示，针对第五到第九个相机位置，步进机器人和示教点。Vision Guide 窗口将会显示针对每个相机位置需要移动的位置。相机位置 5 到 9 描述如下：

```
5 - center
6 - left center
7 - bottom left
8 - bottom center
9 - bottom right
```

- (12)接下来窗口顶部显示的消息为“Jog to fixed reference”。这表示要步进机器人，以便夹具(杆)居于校准目标零件(这里指垫圈)的中心位置。此时您可以忽略视频显示，因为我们正在将机器人夹具同校准目标对准。继续并步进机器人，以便夹具位于垫圈的中心。其定位十分重要，因为您需要将机器人向下步进至靠近垫圈的位置，以实现良好对准。
- (13)一旦您将机器人夹具定位在了垫圈的中心，点击位于 Vision Guide 窗口底部的 <Teach>按钮。
- (14)将 Z 轴向上步进至足够高的位置，使夹具在校准过程中远离工作表面，并排除任何因机器人夹具太低而撞击到其它对象的可能性。在我们示教 9 个校准点中每个校准点时，均使用此 Z 高度。
- (15)在示教了最后一个相机位置以后，会出现一个对话框，说明“所有校准点已经完成了示教”。点击 Finish 按钮继续。

校准所需的点已经完成示教。

### 第6步：运行相机校准

以下显示了完成相机校准程序的最后步骤：

- (1) 点击 Vision Guide 窗口[Calibration]选项卡底部的<Calibrate>按钮。
- (2) 将出现一个消息框，显示“注意，机器人在校准过程中将以速度 10 和加速度 10 移动，是否继续？” 点击<Yes>按钮以继续校准。在校准过程中，机器人将通过全部 9 个校准点，共两遍。如果在机器人正在移动时您需要中止校准，点击 [Calibration Cycle]对话框底部的<Abort>按钮。
- (3) 校准周期完成以后，出现[Calibration Complete]对话框并显示校准结果数据。检查数据然后点击<OK>按钮。
- (4) 注意，校准结果显示在[Calibration]窗口的 Results 列表中。

**第7步：将“downcal”校准分配至blobtut序列**

既然已经创建了“downcal”校准，我们就需要将该校准分配到我们的视觉序列(blobtut)中。这将使 blobtut 序列能够计算机器人坐标值或相机坐标值中的结果。

- (1) 从[Vision Guide]窗口的序列目录树中点击视觉序列：“blobtut”，将[Sequence]窗口置于前面。
- (2) 通过点击 Calibration 属性的值字段，将 Calibration 属性设置为“downcal”，然后点击向下箭头，最后点击所显示的名为“downcal”的校准。此为我们刚刚完成创建的校准。
- (3) 点击序列和校准目录树右侧显示的<Jog>按钮显示[Jog]选项卡。
- (4) 使用步进按钮在垫圈上方定位相机，以便您可以在图像显示区域内看到垫圈。
- (5) 从序列目录树中点击“washer”对象，将[Object]窗口置于前面。
- (6) 点击<Run Object>按钮以运行“washer”对象。当找到零件时，查看 Result 列表。您将看到 CameraX、CameraY、RobotX、RobotY 和 RobotU 的结果不再显示“nocal”结果。现在您可以看见关于机器人和相机坐标系的坐标位置数据。

**10.1.9 示教在视觉引导中使用的点**

现在我们必须示教一些点来确定垫圈拾取位置的 Z 高度、相机将为垫圈拍照的位置、以及本教程中作为起点位置的一个安全位置。

**第1步：确定“camshot”位置**

机器人必须定位在“camshot”位置，这样一来，在零件上方定位的相机才能使视觉系统看到零件。机器人必须朝零件移动，直到屏幕上图像显示区域中能够看到垫圈。有一个好方法，即定位机器人，以便垫圈位于图像显示区域的中间位置并且不靠近搜索窗口的任何一侧。由于我们刚刚完成了垫圈对象的运行，所以相机应处于获取呈现垫圈的图像的最佳位置。

- (1) 点击 EPSON RC+ 7.0 主工具栏上的<Tool> - <Robot Manager>按钮，然后点击[Jog & Teach]选项卡。
- (2) 由于我们想利用工具 1 来示教相机拍摄位置，所以需检查标为 Tool 的下拉列表框确保其设置为 1。如果没有设置为 1，继续点击下拉列表框上的箭头，从而将 Tool 设置为 1。
- (3) 验证当前 Point #字段中的点是否为 P0。我们想将 camshot 位置作为点 P0 进行示教。如果不是点 P0，则在点#字段中选定为 P0。
- (4) 点击[Jog & Teach]窗口上的<Teach>按钮。您将得到一个标签提示。输入名称“camshot”。随后将示教“camshot”位置。

### 第2步：确定一个远离垫圈的安全位置

我们将需要一个远离垫圈的已示教点，将其位置作为安全位置，并在程序开始时移动到该位置。

- (1) 在点字段中选择 P1。
- (2) 将机器人的 Z 轴向上步进，然后沿 X 和 Y 方向步进，从而将机器人定位在一个安全的位置。这就好比您的程序的起始位置。机器人在移动到垫圈之前总是先移动到这一位置。
- (3) 点击[Jog & Teach]页面上的<Teach>按钮。为标签输入“safep1”。随后将示教“safep1”位置。

### 第3步：计算垫圈拾取位置的Z高度

如果我們是在實際情況下拾取真實的墊圈，而非移向墊圈的草圖，我們將需要設置墊圈拾取位置的 Z 高度。讓我們為墊圈位置尋找一個合適的 Z 高度。在此，假設夾爪被安裝在第 3 軸(J3)上。



注意


- 確保仔細地設置了機器人動作的Z高度。  
如果計算出的Z高度不正確，則可能導致系統故障和/或安全問題。

- (1) 点击 EPSON RC+ 7.0 主工具栏上的<Tool> - <Robot Manager>按钮。然后点击[Jog & Teach]页面。
- (2) 使用步进按钮来将机器人的夹具固定在垫圈上方 5-10mm 的位置。
- (3) 当机器人夹具位于垫圈正上方时，记下当前的 Z 坐标值。在我们的程序中，随后将用到该 Z 坐标，目的是将机器人移动到这一高度。请小心不要使夹爪撞到垫圈。
- (4) 在点字段中选择 P2。
- (5) 点击[Jog & Teach]窗口上的<Teach>按钮。输入标签名称“washpos”。此举将示教我们最初的“washpos”位置。(然而，视觉系统将用于计算新的 X 和 Y 位置，然后移动到该点。根据当前的 Z 坐标位置，我们也将要在程序中设置一个固定的 Z 高度。)

### 10.1.10 使用视觉系统让机器人移向零件

现在只剩下修改我们的程序来使视觉系统和机器人联用，以便找到垫圈的位置并移动到该位置。

#### 第1步：修改SPEL+程序

- (1) 点击 EPSON RC+ 7.0 工具栏上的  <Open File> 工具栏按钮。
- (2) 因为您还没有创建任何其它程序，所以 MAIN.PRG 程序文件已经为突出显示状态。点击对话框左下方的 <Open> 按钮以便打开 MAIN.PRG 程序文件。您将会看到我们先前在本教程中运行的程序，如下所示。

```
Function main
  Real area
  Boolean found

  VRun blobtut
  VGet blobtut.washer.found, found

  If found = True Then
    VGet blobtut.washer.area, area
    Print "The washer was found!"
    Print "The washer area is:", area, "Pixels"
  Else
    Print "The washer was not found!"
  EndIf
Fend
```

- (3) 现在修改程序，使其如下一页所示。



请记住，SPEL+语言使用单引号 ' 来表示注解。单引号后面的所有字符均视为注解，并不是需要程序来执行的。(这表示您可以忽略单引号后面直至单引号所在行末尾的所有字符。)

```

Function main

'*****
' 以下是重要声明: *
' 请使用在“示教在视觉引导中使用的点”中, *
' 第3步中获得的Z坐标值。 *
' 使用Z坐标值替换下面程序中“XX”(负数)的部分。 *
'*****
#define ZHeight -xx

Real area, x, y, u
Boolean found
Integer answer
String msg$, answer$

Power Low           '低速运行机器人并加速
Tool 1              '使用工具1进行定位
Jump safept        '移动机器人至安全起始位置

Do                  '在传送带停止前循环执行
  Jump camshot      '移动机器人并拍照
  VRun blobtut      '运行视觉序列blobtut
  VGet blobtut.washer.RobotXYU, found, x, y, u

  If found = True Then
    VGet blobtut.washer.area, area
    Print "The washer was found!"
    Print "The washer area is: ", area, "Pixels"
    washpos = XY(x, y, ZHeight, u) 'Set pos to move to
    Jump washpos
    msg$ = "The washer was found!"
  Else
    msg$ = "The washer was not found!"
  EndIf
  msg$ = msg$ + CRLF + "Run another cycle (Y/N)?"
  Print msg$
  Input answer$
  If Ucase$(answer$) <> "Y" Then
    Exit Do
  EndIf
Loop
Fend

```

## 第2步：运行程序以找到垫圈并移动至垫圈

- (1) 点击 EPSON RC+ 7.0 主工具栏上的<Tool> - <Run>工具栏按钮。这将导致程序进行编辑，然后开启 Run 窗口。
- (2) 点击 Run 窗口上的<Start>按钮。
- (3) 现在您的程序将找到垫圈并将机器人移动至垫圈。在您成功找到垫圈以后，试着轻微移动垫圈，然后在对话框询问您是否运行另一个周期时，点击对话框中的<Yes>按钮。如果没有找到垫圈，将会出现一个不同的对话框，询问您是否想要重试。在任何一种对话框中点击<No>将导致程序停止运行。

## 11. 在SPEL+语言中使用 Vision Guide 7.0

### 11.1 概述

Vision Guide 7.0 的设计在于尽量减少视觉编程，但是允许应用编程器通过 SPEL+程序来操纵视觉序列、对象以及校准。在大多数应用中，利用点击界面创建的视觉序列能够完成大部分的工作。通过 SPEL+运行序列并利用结果来引导机器人、检验零件等十分简单。利用这种设计，应用程序工程师无需在真正设计应用程序之前做任何编程工作便可创建序列。此举可以帮助在开展工作之前验证视野和精确度等。然后在设计的时候，便可以利用实际项目中创建的序列。

SPEL+语言中的命令对 Vision Guide 7.0 提供支持。在利用点击界面创建了视觉序列以后，您可以运行这些序列并通过 SPEL+程序从中得到结果。您也可以读取和设置视觉序列和对象的属性。

本章包含如何通过 SPEL+程序使用 Vision Guide 7.0 命令的说明。

### 11.2 Vision Guide 7.0的SPEL+命令

这是关于 SPEL+程序中可用的视觉命令的摘要。

#### 基本视觉命令

- |      |                                 |
|------|---------------------------------|
| VRun | 在当前的项目中运行一个视觉序列。                |
| VGet | 获得视觉序列或对象的结果或属性，并将其储存在一个或多个变量中。 |
| VSet | 为视觉序列或对象设置一个属性值。                |

#### **在用视觉命令编程之前**

用视觉命令进行编程之前，您需要在 EPSON RC+7.0 项目中创建一个视觉序列。您在 SPEL+程序中使用的视觉命令将会参考这些序列以及其中创建的对象。

## 11.3 通过SPEL+运行视觉序列：VRun

通过使用 VRun 命令的 SPEL+程序，您可以在当前的项目中运行任何视觉序列。  
例如：

```
Function VisionTest
    VRun seq1
End
```

以上的简单程序将会运行名为"seq1"的序列，并在 Run 窗口或 Operator 窗口中显示图像结果，这取决于程序从哪里启动。

VRun 采取的行动取决于 RuntimeAcquire 序列的属性。该属性决定在处理序列之前是否拍了照片，或者是否使用了闪光灯。下表显示了针对 RuntimeAcquire 的不同设置执行 VRun 时会发生什么情况。

RuntimeAcquire	VRun 动作	用途
0 - None	不要拍照片。使用先前的照片。 运行每一个序列步骤。	允许在相同的图片上运行多个序列。通常，在 RuntimeAcquire 设置为 Stationary 时，由第一个序列拍照。余下的序列不进行拍照。
1 - Stationary	读取新图片。 运行每一个序列步骤。	默认。这是使用 VRun 最常见的方式。拍一张照片并运行序列。
2 - Strobed	等待闪光输入。 读取新图片。 运行每一个序列步骤。	与闪光输入一同使用。VRun 将立即返回，然后系统将在拍照和运行序列之前等待闪光灯触发。

当序列属性 AsyncMode 为 True 时，VRun 会在相机曝光后返回，并在后台继续完成图像捕捉，然后处理序列。同一序列的下一视觉命令，如 VGet，将自动等待处理完成。

当 RuntimeAcquire 设为 Strobe 时，VRun 会武装触发器，然后返回。系统会在后台等待触发器启动并捕捉图像，然后处理序列。您可能需要等待完成捕捉方可从序列中检索结果。您亦可通过检查 AcquireState 序列结果进行检索。

```
Function VisionTest
    Integer state
    Boolean passed
    VRun strobedSequence
    '等待拍摄图像
Do
    VGet strobedSequence.AcquireState, state
Loop Until state = 3
    '获取结果
    VGet strobedSequence.AllPassed, passed
End
```

如果在RuntimeAcquire=Strobe时执行VRun，那么无需等待AcquireState=3即可执行第二个视觉命令，如VGet，然后第二个命令将等待接收触发并运行序列。如果接收不到触发，将中止SPEL+任务。

当您在执行多个运行视觉序列的SPEL+任务时，如果相机不同，将并行捕捉图像并处理序列。

如果多个SPEL+任务使用同一相机且RuntimeAcquire设为Strobe，那么您只能使用SyncLock和SyncUnlock每次处理一个序列。在以下实例中，序列1和序列2均使用相机1。

```
Function visTask1
  Integer state
  Do
    SyncLock 1 '锁定对相机 1 的访问
    VRun seq1
    Do
      VGet seq1.AcquireState, state
    Loop Until state = 3
    VGet <some results here>
    SyncUnlock 1 '解锁对相机 1 的访问
  Fend

Function visTask2
  Integer state
  Do
    SyncLock 1 '锁定对相机 1 的访问
    VRun seq2
    Do
      VGet seq2.AcquireState, state
    Loop Until state = 3
    VGet <some results here>
    SyncUnlock 1 '解锁对相机 1 的访问
  Fend
```

关于SyncLock和SyncUnlock的详细信息，请参见 *SPEL+ Language Reference Manual*。

**NOTE:**

在闪光模式(外部触发模式)下使用ac2500-14gm/gc GigE相机，需要使用外部闪光灯。如果未使用闪光灯，相机会以滚动快门模式工作并且无法正确识别移动对象。



## 11.4 在SPEL+中读取属性和结果：VGet、VSet

VGet 和 VSet 命令可用于读取和设置 SPEL+程序中的序列和对象属性。例如，您可以改变搜索窗口的大小和位置、一个接受参数、相机增益和最大面积。几乎所有通过 Vision Guide 7.0 点击界面可以读取的属性和结果都可以通过 SPEL+程序进行读取。也有一些特殊的属性只能通过 VSet 或 VGet 读取，因为它们设置或返回多个结果。(如：SearchWin、RobotXYU 和 ModelWin。)

VGet 和 VSet 使用相同语法。每个命令必须首先参见序列的名称。除此之外，为了读取视觉对象的属性和结果，视觉对象的名称必须遵循序列名称。利用一个句点来分隔序列、对象和属性或结果的名称。如果使用了多个结果，则通过在结果名称后面的圆括号中加上结果编号来选择特定的结果。

对于序列属性和结果，采用以下语法：

```
VGet seqName.propName, var           'put property value in variable
VSet seqName.propName, value         'set property to value
```

要读取对象的属性和结果，可使用以下语法：

```
VGet seqName.objName.resultName, var
VGet seqName.objName.propertyName, var
VSet seqName.objName.propertyName, value
```

要读取对象的多个结果，可使用以下语法：

```
VGet seqName.objName.resultName(resultnum), var
```

序列和对象名称也可以是字符串变量。参见 11.5 在序列和对象名称中使用变量。

### 11.4.1 使用VGet

VGet 检索一条属性或结果并将其置于 SPEL+变量中。您必须在程序中提供具有适当数据类型的变量，以便接受来自 VGet 的数值。

这里是在 SPEL+程序中使用 VGet 的一个例子。

```
Function Inspect
  'Run the vision sequence
  VRun InspectPart
  Integer i, numberFound
  Real area
  VGet inspPart.Part1.NumberFound, numberFound
  For i = 1 to numberFound
    ' Loop through each item that was found
    ' Get the area of the blob result
    VGet inspPart.Part1.Area(i), area
    Print "Area of result ", i, " is ", area
  Next i
Fend
```

### 11.4.2 使用VSet

VSet 设置运行时的属性值。这使得开发者能通过 SPEL+程序动态地调整属性设置。在大多数情况下，您可以从 Vision Guide 窗口中设置属性值，然后通过 SPEL+程序运行视觉序列，而不改变任何属性。然而对于那些需要动态调节的情况，可以使用 VSet SPEL+命令。

这里是在 SPEL+程序中使用 VSet 的一个例子。注意，对 VSet 的第一次调用中设置了序列属性。对 VSet 的第二次调用中设置了名为 SearchWin 的对象属性，该属性用于在运行序列之前重新定义搜索窗口的位置和大小。

```
Function findPart

  'Set camera gain for sequence "findPart"
  VSet findPart.CameraContrast, 32

  'Set search window for object "part"
  VSet findPart.part.SearchWin, 100, 100, 50, 50

  'Run the sequenced
  VRun findPart
Fend
```

## 11.5 在序列和对象名称中使用变量

字符串变量可用于 VRun、VGet 和 VSet 命令中的校准、序列和对象名称参数。下面的例子中，利用字符串变量 seq\$ 和 obj\$ 来指定与哪个视觉序列和视觉对象联用。

```
Function visTest

    #define PICKZ -100.0
    String  seq$, obj$
    Boolean found
    Real    x, y, u

    seq$ = "test"
    obj$ = "Blob01"
    VSet seq$.Camera, 1
    VSet seq$.Calibration, "CAMCAL1"
    VRun seq$
    VGet seq$.obj$.RobotXYU, found, x, y, u
    If found Then
        pick = XY(x, y, PICKZ, u)
        Jump pick
        On vacuum
        Wait .1
        Jump place
        Off vacuum
        Wait .1
    EndIf
    Jump park
Fend
```

阵列也可用于 VRun、VGet 和 VSet 命令中的序列和对象名称参数。参见以下例子。

```
Function test
    String obj$(10)
    Integer count

    obj$(0) = "corr01"
    obj$(1) = "corr02"
    obj$(2) = "blob01"
    obj$(3) = "blob02"

    For count = 0 to 3
        VRun seqname
        VGet seqname.obj$(count).Found, found
    Next count
Fend
```

## 11.6 在SPEL+中使用序列结果

在用 `VRUN` 命令运行了序列以后，有一些结果可以在 `SPEL+` 程序中进行应用。`VGet` 命令用于读取结果和属性。机器人指引最常见的一个结果是 `RobotXYU` 结果。例如：

```
Function getPart
  #define PICKZ -100.0
  Boolean found
  Real x, y, u

  VRUN findPart
  VGet findPart.corr01.RobotXYU, found, x, y, u
  If found Then
    pick = XY(x, y, PICKZ, 0)
    Jump pick
  EndIf
Fend
```

注意，在上面的例子中，将机器人移动到视觉系统所找到的位置之前，对所找到的状态进行了检查。在利用其结果来移动机器人之前，验证是否已经找到了视觉对象十分重要。

## 11.7 通过SPEL+语言读取多个结果

`Correlation` 和 `Blob` 对象等一些 `Vision Guide` 对象能够使用 `NumberToFind` 属性通过一个对象找出多个特征。如果我们在 `[Object]` 窗口中检验结果，则 `CurrentResult` 属性用于设置在结果列表中将会显示哪个结果。它也用于决定将结果返回哪个结果记录。例如，如果我们想要得到 `Blob` 对象返回的第 3 个结果的 `Area` 结果，则 `CurrentResult` 必须设置为 3。您已了解如何从 `[Object]` 窗口属性列表中完成这步。现在，让我们看看如何利用 `SPEL+` 语言读取多个结果。

通过 `SPEL+` 语言读取多个结果时，将结果视为阵列，而结果编号为将要读取的结果旁边的下标号。以下第一个示例显示的是如何获得第 3 个 `Area` 结果，以及如何将其从 `SPEL+` 语言放入名为面积的变量。

```
VGet seqname.objname.Area(3), area
```

以下第二个示例显示的是如何获得相同的第 3 个 `Area` 结果，但是这次将其指定为名叫 `area` 的阵列中第 3 个元素的值。

```
VGet seqname.objname.Area(3), area(3)
```

可使用不同的名称表示阵列中的元素，但不可以表示固定元素，如上述示例 2。注意，将名为 `var` 的变量用作 `Area` 结果的下标。

```
VGet seqname.objname.Area(var), area(var)
```

第 4 个示例假设您已采用一个视觉对象查找多个零件(假如多达 10 个)。现在，您可能需要用机器人来拾取这些零件(假设为笔)，所以您就需要将 X、Y 和 U 坐标存储到每个已找到零件坐标值的阵列变量中。以下代码利用 `RobotXYU` 结果来检索这些坐标，并将其放入 X、Y 和 Z 阵列，随后将利用这些阵列来移动机器人。

```
Function test
  Boolean found(10)
  Integer numFound, i
  Real x(10), y(10), u(10)

  Jump camshot      'move camera into position snap shot

  VRun seq01        'run the vision sequence to find the pens

  VGet seq01.blob01.NumberFound, numFound 'how many found

  For i = 1 to numFound 'get robot coords
    VGet seq01.blob01.RobotXYU(i), found(i), x(i), y(i), z(i)
  Next i

  'Add code for robot motion here.....
Fend
```

## 11.8 视觉命令和多任务处理联用

您可以在 SPEL+ 的多个任务中执行视觉序列并检索结果。SPEL+ 自动处理内部视觉命令的执行，一次一个命令。当多个任务需要用到视觉系统时，以先来后到的原则执行命令。当有一个任务正在使用视觉系统时，其余任务将暂停(等待)，直到第一个任务完成了当前的视觉命令的执行。

在一些情况下，您可以采用不同的相机在不同的任务中运行相同的序列。您将会希望在一个任务中运行序列并获得结果，而其他任务处于等待状态。使用 `SyncLock` 和 `SyncUnlock` 命令来锁定视觉系统，使其专用于一个任务，然后解锁，使其可用于其它任务。

```
Function VisionTask(camera As Integer)
  Boolean found
  Do
    SyncLock 1 'Lock vision just for this task
    VSet FindParts.Camera, camera
    VRun FindParts
    VGet FindParts.AllFound, found
    SyncUnlock 1 ' Unlock vision for other tasks
  Loop
Fend
```

## 11.9 视觉系统和机器人联用

在您使用视觉序列的结果来指导机器人时，您需要校准序列所使用的相机。想了解更多信息，请参见 7. 视觉校准。

如果您尝试从一个需要校准的序列对象读取结果，但是此序列并没有经过校准，则将出现运行时间误差。

### 11.9.1 定位结果

使用机器人校准相机的视觉序列中的对象所报告的所有定位结果均位于指定的本地坐标系中。和其他机器人/视觉系统不同，没有额外步骤来获得机器人坐标转换函数调用的像素坐标。这完全由 Vision Guide 7.0 进行处理。能够轻易检索机器人本地坐标系中零件的 X、Y 和 U 坐标。

为了准确地将终端执行器定位在视觉系统定义的位置，您需要为终端执行器定义一个工具。请参见下一节的 定义一项工具。

以下结果可用于指导机器人：

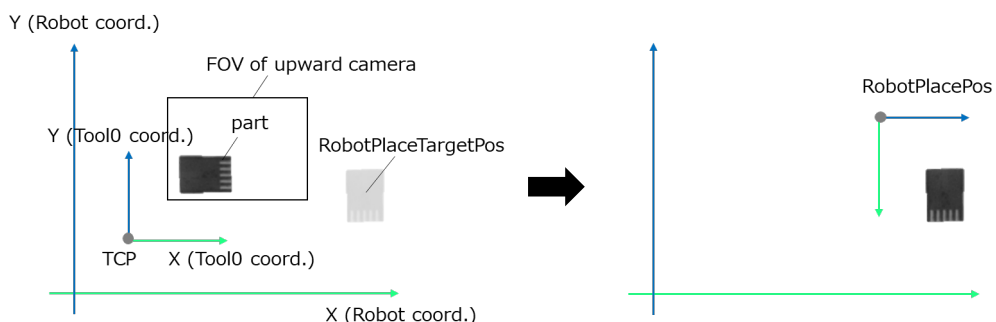
- RobotXYU**        返回 x、y 和 u 值以及找到的状态。
- RobotToolXYU**    仅限固定向上相机  
                         返回 X、Y 和 U 坐标值，以定义具有查找状态的机器人工具。
- RobotX**            返回 X 坐标。
- RobotY**            返回 Y 坐标。
- RobotU**            返回 U 坐标。
- RobotPlacePos**    仅限固定向上相机  
                         以点的形式返回机器人的位置，以便将检测到的零件放到  
                         (**RobotPlaceTargetPos**)已登录的工作安放位置上。

**NOTE:**

需要为每个对象设置**CalRobotPlacePos**。

U 坐标零度位置沿 Y 轴直直地从机器人底座向外延伸。

以下对**RobotPlacePos**进行补充说明。



如上图所示，在机器人的坐标系(工作平面)中放一个向上的相机，工件的放置位置 (**RobotPlaceTargetPos**)通过**CalRobotPlacePos**注册。在实际操作中，首先在视觉序列中检测出(**VRUN**)要拾取工件的位置。然后获取(**VGET**)放置该工件时机器人的位置 (**RobotPlacePos**)。

图中右侧为移动到RobotPlacePos后的状态。移动到RobotPlacePos，可以使工件的位置姿态和工件的放置位置相匹配。

在RobotPlacePos中获取的点，姿态Flag为默认值，所以在移动时根据需要进行修正。以下程序示例可实现，通过向上相机检测到要抓取的工件，然后以最小的关节移动距离，移动到工件的放置位置的操作。

```
Function placePart
  'Move robot into position snapshot
  Go camshot

  VRun findPart
  VGet findPart.Blob01.RobotPlacePos, P100

  '-----
  ' When the robot is SCARA
  Double diffJ4 'Angle difference between RobotPlacePos and latch
    position(J4)
  diffJ4 = PAgl(P100, 4) - PAgl(LatchPos(WithoutToolArm), 4)
  If diffJ4 > 180 Then
    Go P100 -U(360) 'Joint4 will be the shortest movement
  ElseIf diffJ4 < -180 Then
    Go P100 +U(360) 'Joint4 will be the shortest movement
  Else
    Go P100
  EndIf

  '-----
  ' When the robot is 6-axis
  Go P100 LJM 4 '4: Joint6 will be the shortest movement
Fend
```

当RobotPlacePos中获取的点超出操作范围时，可能会出现4007错误。此时，请更改工件的放置位置，或使用RobotToolXYU。

### 11.9.2 定义一项工具

为机器人终端执行器定义一项工具十分重要。此工具告知机器人终端执行器的位置，以便所有的位置信息都和终端执行器位置相关，而不是和 TOOL 0 位置相关。使用 TLSET 命令在 SPEL+中定义一项工具。以下为定义工具偏移的三种方式。

#### 用 Robot Manager Tool Wizard 来定义一项工具

您可以使用 Robot Manager 中的 Tool Wizard 来定义一项工具。

要使用 Tool Wizard，请采用以下步骤：

- (1) 打开 Robot Manager。
- (2) 点击[Tools]选项卡。
- (3) 点击<Tool Wizard>按钮。
- (4) 按照向导中的步骤来创建工具。

#### 利用固定向上相机来计算工具偏移

这是如何使用固定向上相机来计算工具偏移的示例。即使用 RobotToolXYU 结果定义工具偏移。该功能首先运行一个序列来定位工具的尖端。然后使用 VGet RobotToolXYU 检索工具偏移，并使用 TLSet 定义工具。

```
Function DefineTool
  Boolean found
  Real xTool, yTool, uTool

  VRun findTip
  VGet findTip.tip.RobotToolXYU, found, xTool, yTool, uTool
  If found Then
    TLSet 1, XY(xTool, yTool, 0, 0)
  EndIf
Fend
```

#### 手动计算工具偏移

在以下步骤中，您必须用 Jog & Teach 窗口中的 Jog 按钮来步进机器人。在按如下方式计算工具偏移时您不能释放轴(使用 SFREE)，也不能手动移动机器人。

计算工具偏移，请执行以下步骤。



NOTE

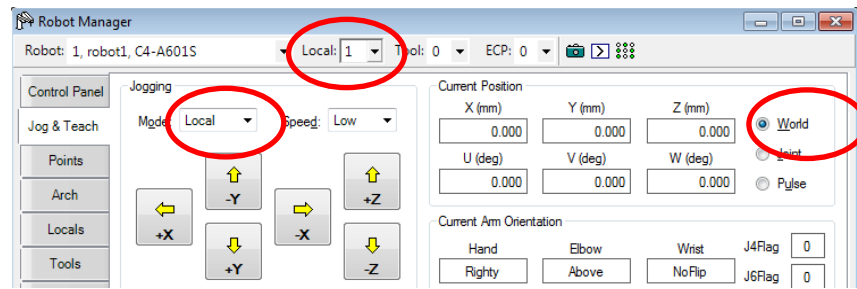
- 使用垂直 6 轴机器人时，执行接下来的 Local 命令，然后按照下一页的步骤进行。

```
> Local 1, Here
```

- 当在以下计算步骤中读取当前位置时，应使用 Local 1 中的位置。  
如果使用 Jog & Teach 窗口，将步进模式设为“Local”，当前位置的显示模式设为“World”，编号设为“1”。  
如果使用命令窗口，可执行如下命令，检查 Local 1 的位置。

```
> Print Here@1
```





- (1) 将 U 轴定位在零度位置。
- (2) 将工具设置为零(TOOL 0)。
- (3) 在基准点上步进终端执行器，然后将其仔细对准。不要改变 U 轴设置。
- (4) 记下当前的 X 和 Y 坐标，分别为 X1 和 Y1。
- (5) 步进 U 轴至 180° 位置。
- (6) 在基准点上步进终端执行器，然后将其仔细对准。不要改变 U 轴设置。
- (7) 记下当前的 X 和 Y 坐标，分别为 X2 和 Y2。
- (8) 利用以下公式来计算工具偏移：

$$xTool = (X2 - X1) / 2$$

$$yTool = (Y2 - Y1) / 2$$

- (9) 通过 Robot Manager 的 Tools 页面或通过 Command 窗口定义工具。

```
TLSET 1, XY(xTool, yTool, 0, 0)
```

- (10)测试工具设置：

将当前工具设置为您在之前的步骤中定义的工具。例如，TOOL 1。

在基准点上步进终端执行器。

现在步进 U 轴。

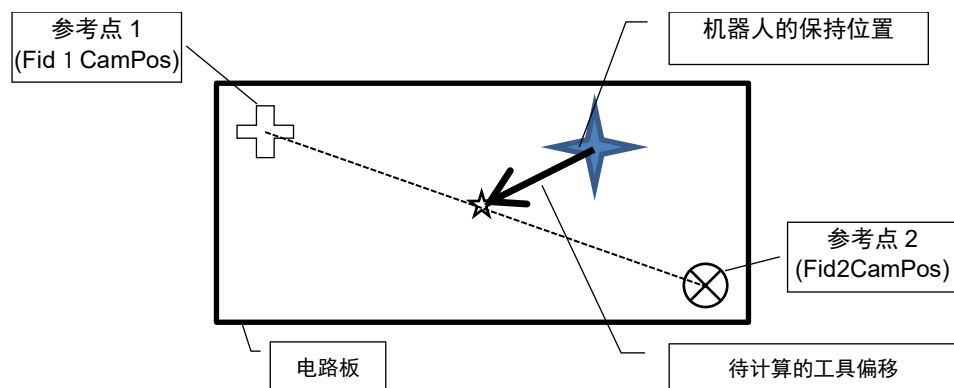
终端执行器应保持基准点之上。

### 11.9.3 机器人电路板的工具计算

本示例中，Vision Guide 7.0 用于计算机器人所配备的电路板的工具。从电路板中的两个参考点的坐标，计算工具偏移量。这需要一台固定向上相机。校准相机后，您将需要示教一次电路板的安放位置。

要示教安放位置：

- (1) 由机器人拾取电路板。
- (2) 调用示例函数 CalcBoardTool 来计算工具 1。
- (3) 转换到 Tool 1。
- (4) 步进机器人，移至电路板的安放位置。
- (5) 示教安放位置。



```
Function CalcBoardTool As Boolean
  Boolean found
  Real x, y, theta
  Real toolX1, toolY1, toolU
  Real toolX2, toolY2

  CalcBoardTool = False
  Jump Fid1CamPos 'Locate fiducial 1 over camera
  VRun SearchFid1
  VGet SearchFid1.Corr01.RobotToolXYU, found, toolX1, toolY1, toolU
  If Not found Then
    Exit Function
  EndIf

  Jump Fid2CamPos 'Locate fiducial 2 over camera
  VRun SearchFid2
  VGet SearchFid2.Corr01.RobotToolXYU, found, toolX2, toolY2, toolU
  If Not found Then
    Exit Function
  EndIf
  x = (toolX1 + toolX2) / 2
  y = (toolY1 + toolY2) / 2
  theta = Atan2(toolX1 - toolX2, toolY1 - toolY2)
  toolU = RadToDeg(theta)
  TlSet 1, XY(x, y, 0, toolU)
  CalcBoardTool = True
Fend
```

#### 11.9.4 定位相机以搜索托盘

使用安装在 J2 上的相机进行托盘搜索时，相机的成像位置很难控制。然而，通过使用 Arm 命令来确定相机的 Arm，可以使相机轻松移动到托盘上的每个成像位置。有关 Arm 命令的详细内容，请参见 *EPSON RC+7.0 SPEL+语言参考*。

**提示：**

Arm 命令仅适用于 SCARA 机器人。

