



廣東科技學院



GUANGDONG UNIVERSITY OF SCIENCE & TECHNOLOGY

机器视觉实验指导书

专 业:	_____ 机器人工程 _____
课 程:	_____ 机器人视觉技术 _____
制 订 人:	_____ 陈高工 _____
审 核 人:	_____ 李 磊 _____
制订时间:	_____ 2021 年 12 月 _____

目 录

1 视觉设备简介.....	3
2 图片读取.....	5
3 图像采集.....	7
4 二值化.....	14
5 图像增强.....	17
6 图像滤波.....	20
7 灰度变换.....	23
8 小球计数.....	24
9 条码识别.....	29
10 二维码识别.....	31
11 电池尺寸测量.....	33
12 齿轮检测.....	37
13 电缆颜色检测.....	41
14 识别定位和计数.....	44
15 开关管脚宽度测量.....	47
16 模板匹配.....	49

1 视觉设备简介

根据应用型大学《机器人视觉技术》课程教学的要求，构建了机器视觉实验设备，采用多家公司的光电设备和视觉处理软件，兼容多种型号的工业相机装置。将机器视觉中的常用设备和算法摆在了学生面前，通过实验可以达到深化理论，增加实践和动手能力的目的。本装置可以用于授课教材的例行实验，也可以用于学期末的课程设计及毕业设计，能减轻教师的教学难度，提高学生的实践能力，把学生从抽象的学习中解放出来。

本实验设备可以实现课程主要知识点学习、硬件实验功能，完成教学的实训任务。

表1 实验设备明细表

序 号	设 备 名 称	规 格 / 型 号	数 量	备 注
1	彩色工业相机	OPT-CC500-GL-04	10	2592 x 1944像素
2	黑白工业相机	OPT-CM500-GL-04	10	2592 x 1944像素
3	FA镜头/C接口	焦距：12、16、25、50	各5	
4	环形无影光源-白光	P-RL-120-90-W	10	
5	环形无影光源-白光	P-RL-100-60-W	5	
6	环形无影光源-白光	P-RL-100-30-W	5	
7	模拟光源控制器	PM-A-24W-2T	18	
8	数字光源控制器	PM-D-24W-4TE	2	
9	圆顶光源	P-DL-175-W	1	
10	同轴光源	P-COL-120-W	1	
11	方形背无影光源	P-HFL-100-100-W	1	
12	AOI光源	P-SAI-120-RGB	1	
13	彩色环形无影光源	P-RL-100-30-R/G/B	RGB各1	
14	条形无影光源	P-BL2-147-28-W	2	
15	标定板等	GB050-1.875/GB100-5	各5	
16	圆盘式实验台	GK320	1	
17	三维实验台	GK3D-120	1	
18	视觉控制器	海康VC4000	1	
19	视觉软件	Halcon-18.11	1	
20	照度计	DT-1330A	2	
21	6.5X变倍镜头/物镜	VP-LZ-63100/66105	1	
22	相机支架	900	20	
23	钢板尺	60cm	2	
24	卡尺	15cm	2	

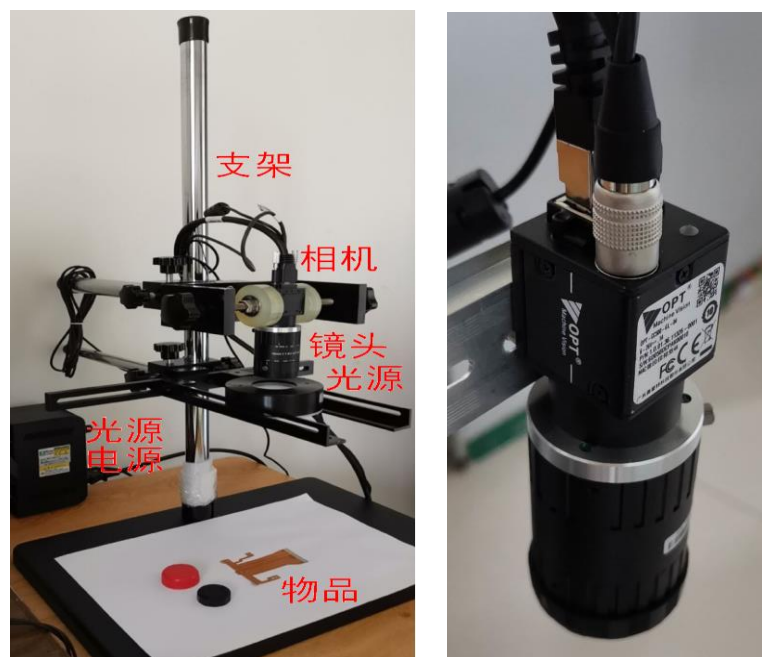


图 1 机器视觉基本实验设备

2 图片读取

一、实验目的：熟悉图片文件爆乳方法，掌握文件命名技巧。

二、实验要求：按照指定命名方式，导入图像文件。

三、实验内容：通过在某文件夹导入一张或多张图片，实现显示和后续处理。

四、实验设备：微机或视觉控制器。

五、方法步骤

1、单张图片读取 read_image 函数，比如

```
read_image (Image, 'bin_switch.png')
```

```
get_image_size (Image, Width, Height)
```

```
dev_open_window(0,0,Width,Height,'black',WindowHandle)
```

```
dev_display (Image)
```

写完之后按 F2 快捷键，再按 F5 快捷键就可完成读图和显示。

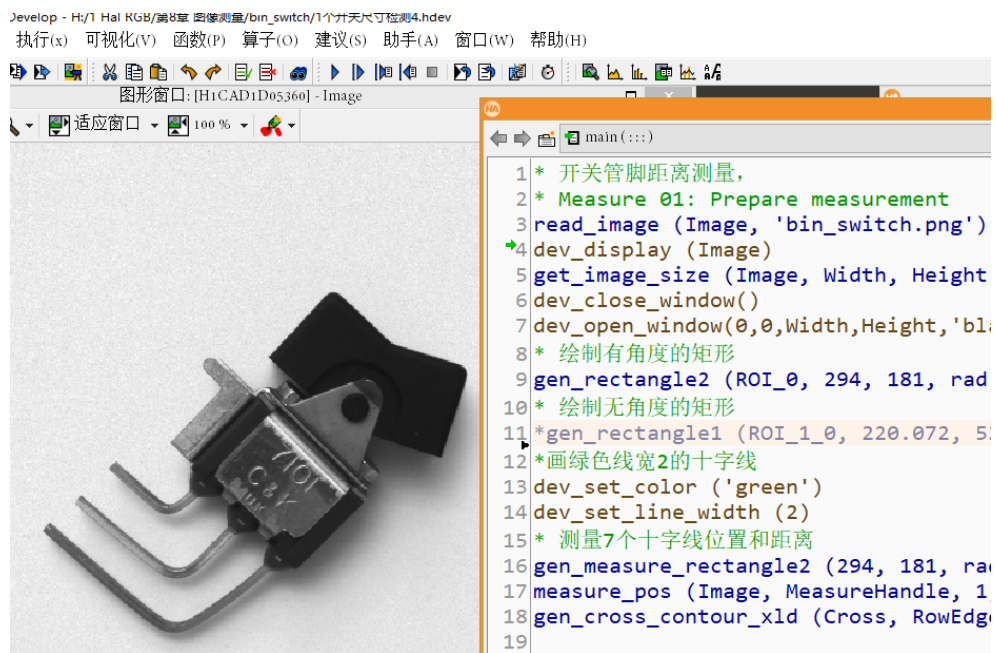


图 2 读单张图片

2、读序号连续的多张图片

```
dev_update_off ()
```

```
dev_close_window ()
```

```
read_image (Image, 'smd/smd_backlight_01')
```

```
dev_open_window_fit_image (Image, 0, 0, -1, -1, WindowHandle)
dev_set_line_width (2)
* 读多张 (Num) 图, 2 位文件序号, 如 smd_backlight_15,
for Index := 1 to Num by 1
    read_image (Image, 'smd/smd_backlight_' + Index$'02')
endfor
```

3 图像采集

- 一、实验目的：熟悉相机连接方法，掌握图片采集和存储方法。
- 二、实验内容：掌握相机连接设置，学会图像采集方法。
- 三、实验设备：OPT-CC500-GL-04 GigE 工业相机。
- 四、实验原理：采用网线连接 GigE 相机，图像采集后存储到计算机中，再进行处理。



图 1 相机连线图

五、方法步骤

- 1、打开桌面的 ”OPT Camera”软件。



图 1 设备安装图

- 2、连接相机并调整相机参数。

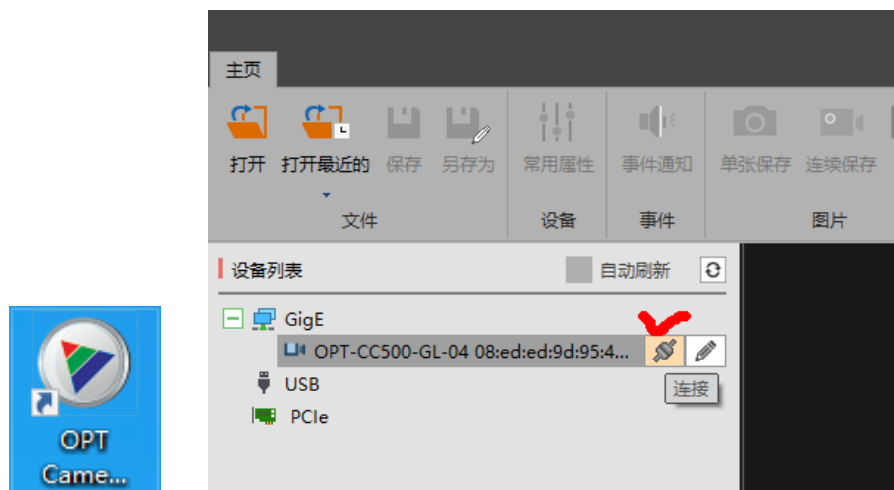


图 1 连接相机

3、修改参数：点击“修改”，可以修改相机的网址。



图 1 相机 IP 地址

设备信息	
MAC	08:ed:ed:9d:95:4a
IP Address	192.168.1.10
Subnet Mask	255.255.255.0
Default Gateway	0.0.0.0
Vendor	OPT Machine Vision
Model	OPT-CC500-GL-04
ManufactureInfo	OPT Machine Vision
Version	V1.000.00.0.R(20180727,5...
Serial Number	6D058DCPAK00010

图 1 相机信息

4、调整参数：点击“属性”后，出现参数调整对话框，调节“滑竿”的位置，实现相机参数调整。如先做调整“伽马”，图像颜色变白，向右调整颜色变红。调整 R、G、B 增益可以改变色调，如变红，变绿、变蓝等。还可调整亮度、增益、曝光时间等。调整适当后，即可退出此软件。

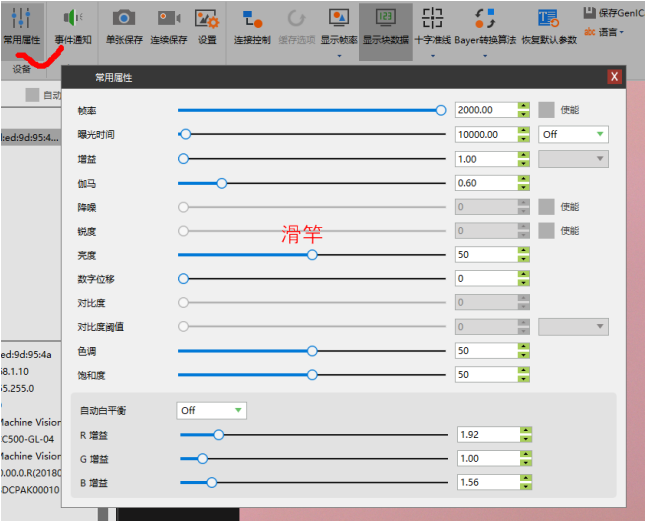


图 1 常用属性调整框

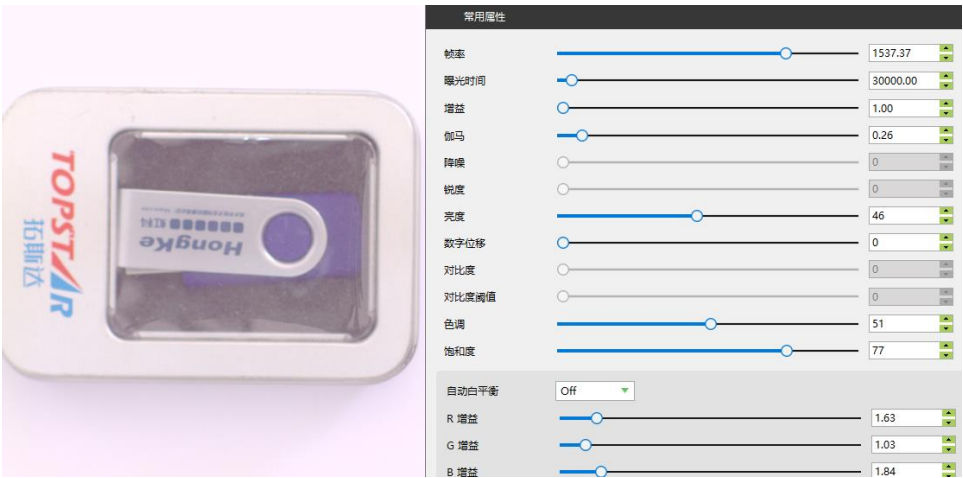


图 1 参数调整

5、打开 Halcon 软件。

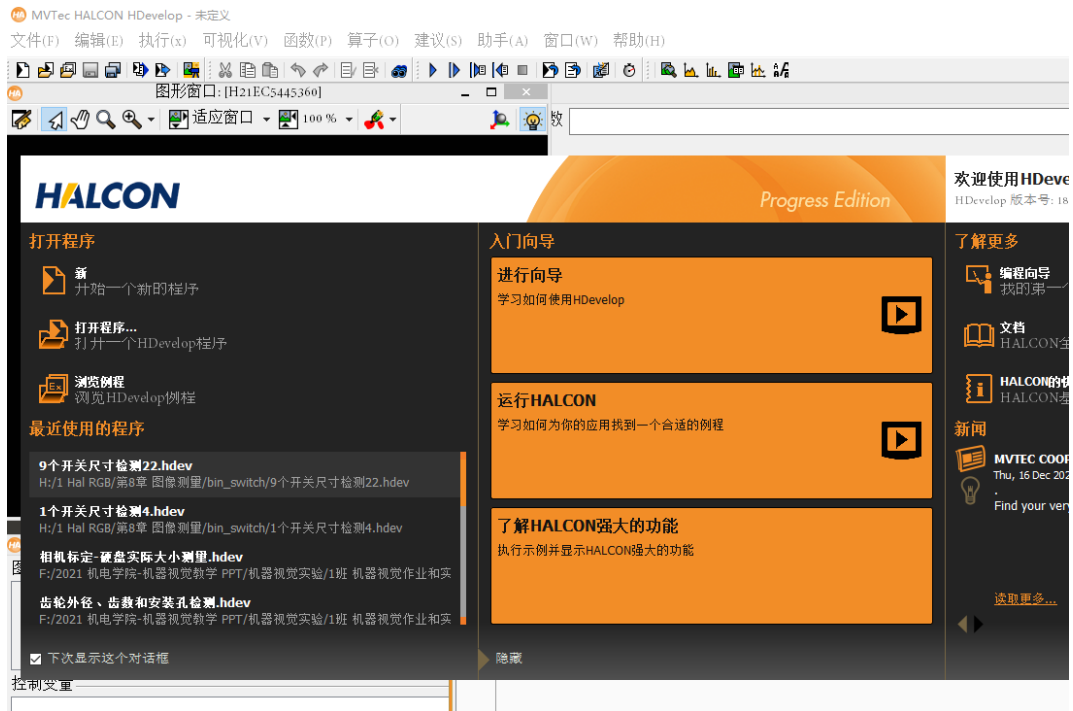


图 1 Halcon 软件界面

再点击图像采集助手。

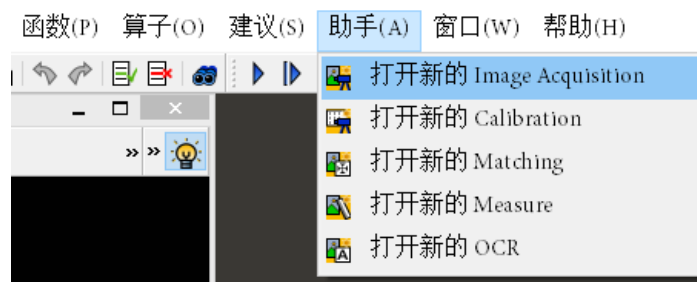


图 2 打开采集助手



图 2 查找相机



图 2 连接相机

再点击“采集”或“实时”按钮，可以采集一张图像或连续显示图像。



图 2 采集图像

6、用采集助手生成代码。



图 插入采集代码

```

1 * Image Acquisition 01: Code generated by Image Acquisition 01
2 open_framegrabber ('GigEVision2', 0, 0, 0, 0, 0, 0, 'progressiv
3 grab_image_start (AcqHandle, -1)
4 while (true)
5     grab_image_async (Image, AcqHandle, -1)
6     * Image Acquisition 01: Do something
7 endwhile
8 close_framegrabber (AcqHandle)

```

图 新插入的采集代码

7、修改采集代码。

```

* Image Acquisition 01: Code generated by Image Acquisition 01
* 采集1张或连续序号的多张图像,打开相机,获取相机句柄AcqHandle
open_framegrabber ('GigEVision2', 0, 0, 0, 0, 0, 0, 'progressive',
grab_image_start (AcqHandle, -1) // 抓取图像开始
for Index := 1 to 10 by 1 // 间隔0.6秒,连续采集10张图像
*实时读取并显示图像
    grab_image_async (Image, AcqHandle, -1) //抓取图像
    dev_display (Image)
*存储单张图像到当前文件夹
    *write_image (Image, 'png', 0, 'TX')
*连续保存2位序号的不同格式图像到本文件夹中
    write_image (Image, 'png', 0, 'TXpng'+Index$.2') //png格式
    write_image (Image, 'bmp', 0, 'TXbmp'+Index$.2') //bmp格式
    write_image (Image, 'jpg', 0, 'TXjpg'+Index$.2') //jpg格式
    wait_seconds (0.6) //延时0.6秒
endfor
close_framegrabber (AcqHandle) // 关闭相机

```

图 修改的采集代码

修改的采集代码:

* Image Acquisition 01: Code generated by Image Acquisition 01

* 采集 1 张或连续序号的多张图像,打开相机,获取相机句柄 AcqHandle

```

open_framegrabber ('GigEVision2', 0, 0, 0, 0, 0, 0, 'progressive', -1, 'default', -1, 'false', 'default',
    '08eded9d954a_OPTMachineVision_OPTCC500GL04', 0, -1, AcqHandle)

```

```

grab_image_start (AcqHandle, -1) // 抓取图像开始

```

```

for Index := 1 to 10 by 1 // 间隔 0.6 秒,连续采集 10 张图像

```

*实时读取并显示图像

```

grab_image_async (Image, AcqHandle, -1) //抓取图像

```

```

dev_display (Image)

```

*存储单张图像到当前文件夹

```

*write_image (Image, 'png', 0, 'TX')
*连续保存 2 位序号的不同格式图像到本文件夹中
write_image (Image, 'png', 0, 'TXpng'+Index$.2')    //png 格式
write_image (Image, 'bmp', 0, 'TXbmp'+Index$.2')    //bmp 格式
write_image (Image, 'jpg', 0, 'TXjpg'+Index$.2')    //jpg 格式
wait_seconds (0.6)    //延时 0.6 秒
endfor
close_framegrabber (AcqHandle) // 关闭相机

```

8、采集结果



图 采集的其中 6 张图像

4 二值化

一、实验目的与要求

- 1、熟悉二值化的理论知识；
- 2、掌握创建矩形，显示图像；
- 3、要求：打光均匀、图像清晰、识别准确、通讯可靠。

二、实验条件

1、硬件

表 4.1 设备表

序号	设备名称	型号	数量	用途
1	相机	OPT-CC500-GL-04	1	获取图片
2	光源	P-RL-120-90-W	1	打光
3	牙膏盒			要识别的对象

2、软件

HALCON 是德国 MVtec 公司开发的一套完善的标准的机器视觉算法包，拥有应用广泛的机器视觉集成开发环境。它节约了产品成本，缩短了软件开发周期——HALCON 灵活的架构便于机器视觉，医学图像和图像分析应用的快速开发。

三、实验原理和程序

1、实验原理

将 256 个亮度等级的灰度图像通过适当的阈值选取而获得仍然可以反映图像整体和局部特征的二值化图像。所有灰度大于或等于阈值的像素被判定为属于特定物体，其灰度值为 255 表示，否则这些像素点被排除在物体区域以外，灰度值为 0，表示背景或者例外的物体区域。通过二值化指令筛选出牙膏盒所在位置，根据盒子位置作一个矩形，显示所需图像。

2、实验程序

```

*****读取图片
read_image (Image, 'Photo_1222_1b.jpg')
get_image_size (Image, Width, Height)
dev_close_window ()
dev_open_window (0, 0, Width * 0.3, Height * 0.3, 'black', WindowHandle)
dev_display (Image)
rgb1_to_gray (Image, GrayImage)
***二值化
threshold (GrayImage, Regions, 100, 234)
fill_up (Regions, RegionFillUp)
connection (RegionFillUp, ConnectedRegions)
select_shape (ConnectedRegions, SelectedRegions, 'area', 'and', 90000, 976511)
opening_circle (SelectedRegions, RegionOpening, 9.5)
shape_trans (RegionOpening, RegionTrans, 'rectangle2')
reduce_domain (GrayImage, RegionTrans, ImageReduced)

```

程序代码：

*****读取图片

read_image (Image, 'Photo_1222_1b.jpg')

get_image_size (Image, Width, Height)

dev_close_window ()

dev_open_window (0, 0, Width * 0.3, Height * 0.3, 'black', WindowHandle)

dev_display (Image)

rgb1_to_gray (Image, GrayImage)

***二值化

threshold (GrayImage, Regions, 100, 234)

fill_up (Regions, RegionFillUp)

connection (RegionFillUp, ConnectedRegions)

select_shape (ConnectedRegions, SelectedRegions, 'area', 'and', 90000, 976511)

opening_circle (SelectedRegions, RegionOpening, 9.5)

shape_trans (RegionOpening, RegionTrans, 'rectangle2')

reduce_domain (GrayImage, RegionTrans, ImageReduced)

四、实验内容及步骤（包含简要的实验步骤流程）

(1)连接相机

(2)读取图像



图 4.2 读取图像软件截图

(3)图像处理

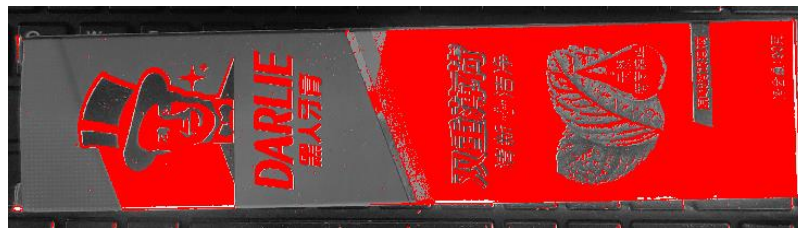


图 4.3 图像处理软件截图

五、实验结果

- 1、选择出了盒子的图像，通过调整阈值，实现了二值化处理；
- 2、思考问题：怎么利用部分图像来做出完整的图像框，再获取完整的所需图像。

5 图像增强

一、实验意义

在图像分析中，图像质量的好坏直接影响识别算法的设计与效果的精度。因此，在图像分析前，需要进行图像预处理。图像预处理的主要目的是消除图像中无关的信息、恢复有用的真实信息、增强有关信息的可检测性和最大限度地简化数据，从而改进特征提取、图像分割、匹配和识别的可靠性。进行图像预处理可以改善图像数据，抑制不需要的变形，也可增强对后续处理重要的图像特征。

二、实验目的

掌握软件 halcon，并应用 halcon 对图像进行处理，通过直方图处理、图像的锐化、图像失焦处理、增强对比度、图像的平滑与去噪处理等方法，使图像变得更清晰，图像特征更加明显。

三、实验内容

通过文件夹导入一张或多张图片，实现显示和后续处理。

四、实验方法

1、直方图均衡化

将统计学中直方图的概念引入到数字图像处理中，用来表示图像的灰度分布，称为灰度直方图。在 HALCON 图像处理中，灰度直方图是一个简单、有用的工具。它可以描述图像的概貌和质量，采用修改直方图的方法来增强图像是一种实用而有效的处理方法。

程序：

```
dev_clear_window()

get_image_size (Image, Width, Height)

dev_open_window_fit_size (0, 0, Width, Width, -1, -1, WindowHandle1)

dev_display(Image)

*直方图均衡化

rgb1_to_gray (Image, GrayImage)

equ_histo_image (GrayImage, ImageEquHisto)

dump_window (WindowHandle1, 'bmp', 'E:/halcon/A 作业/1-直方图结果')
```

五、实验结果

1、图像对比：

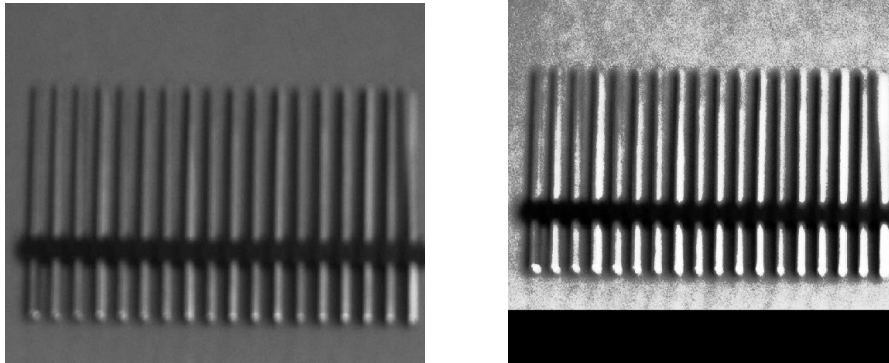


图 图像效果对比

2、增强对比度

增强图像的高频区域（边缘和拐角）的对比度，使图像看起来更清晰。

程序：

```
dev_clear_window()
get_image_size (Image1, Width, Height)
dev_open_window_fit_size (0, 0, Width, Width, -1, -1, WindowHandle1)
dev_display(Image1)
rgb1_to_gray (Image1, GrayImage)
*图像取反
invert_image (GrayImage, ImageInvert)
dev_display (ImageInvert)
*增强对比度
emphasize (GrayImage, ImageEmphasize, 10, 10, 1.5)
dev_display (ImageEmphasize)
*减小对比度
scale_image (GrayImage, ImageScaled, 0.5, 0)
dev_display (ImageScaled)
*增加亮度
scale_image (GrayImage, ImageScaled1, 1, 100)
dev_display (ImageScaled1)
*减小亮度
```

scale_image (GrayImage, ImageScaled2, 1, -50)

dev_display (ImageScaled2)

*Image: 输入的要增强的图像;ImageEmphasize: 输出的对比度增强的图像; MaskWidth,
MaskHeight: 输入的掩膜宽度、高度;

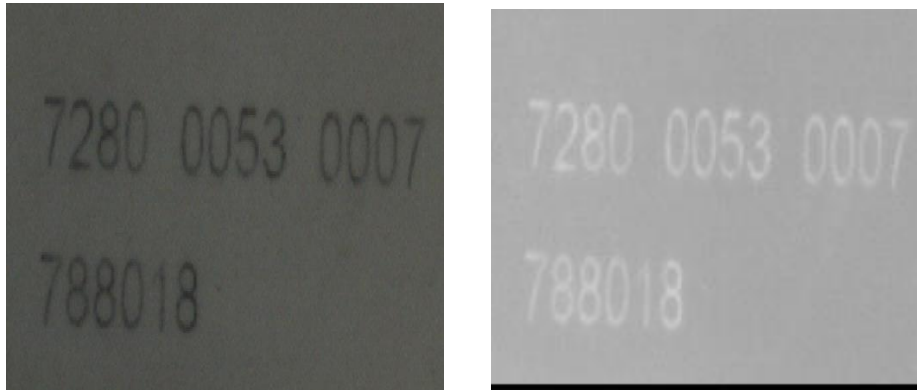
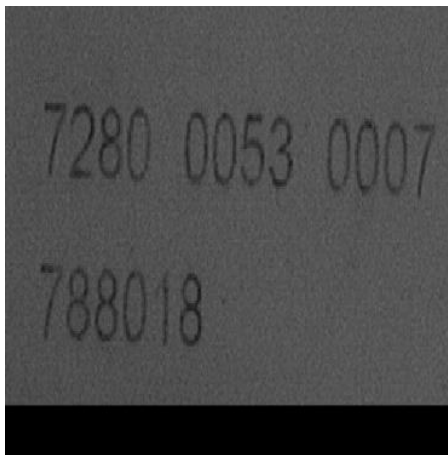
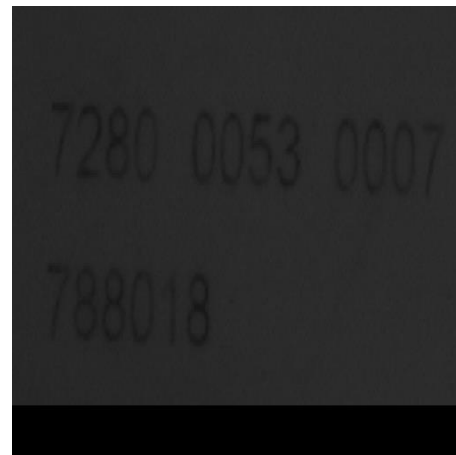


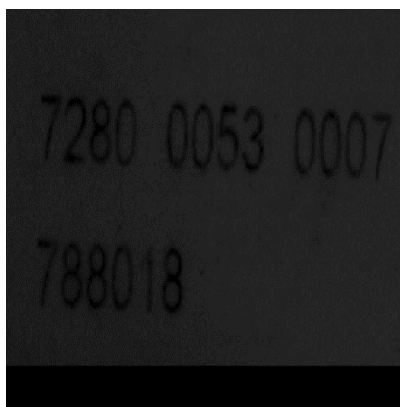
图 图像取反对比图



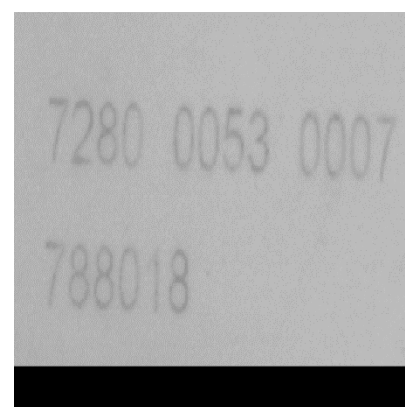
增强对比度



减小对比度



减小亮度



增加亮度

6 图像滤波

一、实验目的与要求：

- 1、熟悉HALCON的相关功能与应用；
- 2、掌握HALCON的算子的运用；
- 3、学会运用HALCON对图像进行滤波处理的方法；
- 4、要求：打光均匀、图像清晰、识别准确、通讯可靠。

二、实验（训）条件

表 6.1 设备表

序号	设备名称	型号	数量	用途
1	相机	OPT-CC500-GL-04	1	获取图片
2	光源	P-RL-120-90-W	1	打光
3	鸡蛋			要识别的对象

三、实验原理和程序

1、实验原理：

图像滤波，即在尽量保留图像细节特征的条件下对目标图像的噪声进行抑制，去噪分为时域去噪和频域去噪，是图像预处理中不可缺少的操作，其处理效果的好坏将直接影响到后续图像处理和分析的有效性和可靠性。常见的滤波方式有均值滤波、中值滤波、高斯滤波等。

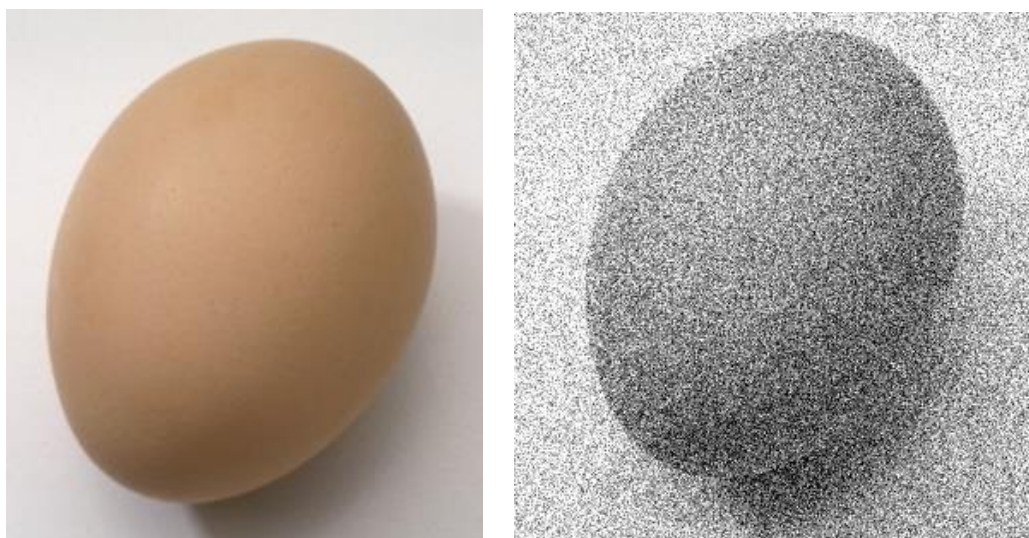


图 6-1 实验图片

2、实验程序：

*读单张图

```
read_image (Image, ' 鸡蛋.png')
```

```
get_image_size (Image, Width, Height)
```

```
dev_close_window ()
```

```
dev_open_window_fit_size (0, 0, Width, Width, -1, -1, WindowHandle1)
```

```
dev_display (Image)
```

```
rgb1_to_gray (Image, GrayImage)
```

*添加高斯噪声 Sigma 参数 20~60

```
gauss_distribution (50, Distribution) //gauss_distribution ( : 西格玛: 分布)
```

```
add_noise_distribution (GrayImage, ImageNoise, Distribution) //add_noise_distribution
```

*均值滤波 模板 3, 3 有效

```
mean_image (ImageNoise, ImageMean1, 3, 3) //3*3 的像素格
```

```
mean_image (ImageNoise, ImageMean2, 7, 7) //7*7 的像素格
```

```
mean_image (ImageNoise, ImageMean3, 9, 9) //9*9 的像素格
```

*添加椒盐噪声

```
sp_distribution (5, 5, Distribution_sp)
```

```
add_noise_distribution (GrayImage, ImageNoise_sp, Distribution_sp)
```

*中值滤波

median_image (ImageNoise_sp, ImageMedian4, 'square', 1, 'mirrored') //同等半径下, 圆形与正方形的比较

```
median_image (ImageNoise_sp, ImageMedian5, 'circle', 1, 'mirrored')
```

```
*median_image (ImageNoise_sp, ImageMedian6, 'square', 3, 'mirrored')
```

median_image (ImageNoise_sp, ImageMedian4, 'square', 1, 'mirrored') //同等类型下, 不同半径的比较

```
*median_image (ImageNoise_sp, ImageMedian5, 'circle', 1, 'mirrored')
```

```
median_image (ImageNoise_sp, ImageMedian6, 'square', 3, 'mirrored')
```

四、实验内容及步骤（包含简要的实验步骤流程）

1、连接相机

2、读取图像

3、图像处理

五、实验结果

1、完成了对图像的滤波处理，

2、学会了如何使用 HALCON 对图像进行滤波处理

7 灰度变换

一、实验目的：目的是为了改善画质，使图像的显示效果更加清晰。

二、实验要求：要求图片要减小噪声，灰度均匀。

三、实验内容：通过文件夹导入一张图片，实现显示和灰度变换。

四、实验设备：:

五、实验原理

灰度变换是图像增强的一种经典而有效的方法。灰度变换的原理将图像的每一个像素的灰度值根据某种目标条件按一定变换关系逐点改变源图像中每一个像素的灰度值。

六、方法步骤

1、连接相机

2、导入图片

3、灰度变换



图3 图片灰度变换

七、程序分析

* 读图 默认显示 512*512

```
read_image(Image, 'T001') // * 读取图的长宽，窗口变黑，显示彩图
```

```
get_image_size(Image, W, H)
```

```
dev_open_window(0, 0, W, H, 'black', WindowHandle)
```

```
dev_display(Image)
```

```
rgb1_to_gray(Image, GrayImage) // * 彩图变为灰度图
```

```
dev_display(Image)
```

8 小球计数

一、实验目的

- 1、熟悉机器视觉中有关相机的调试与使用，HALCON与计算机的连接，光源与摄像机的安装调试；
- 2、掌握HALCON的编程，对图像进行分析，在实践中对HALCON中的算子进行充分实践了解；
- 3、学会对于相机的选用需要考虑的参数，光源的考虑的参数。掌握相机光源的调节，调节适合的焦距，使得图像清晰，边界分明的方法；
- 4、要求：打光均匀、图像清晰、识别准确、通讯可靠使用合适的光源，避免出现光影部分影响识别

二、实验要求

相机与电脑HALCON软件链接。获取清晰的图像，以图像中产品颜色作为基础去识别，对产品数量、角度、圆度、行列位置、面积进行计算和显示。

三、实验条件

PC 微机一台，Halcon软件。

四、实验原理和程序

1、实验原理：将图片导入 HALCON，获取图像数据，将图像进行阈值分割处理，随后利用圆形结构元素进行开运算，获取上述阈值操作后区域中的图像数据，将通道图像数据进行连通处理，获取区域里连接的组件，之后对图像进行腐蚀和膨胀处理，处理结束后便对图像产品进行计数，接着计算产品的角度和圆度，上述都计算结束后便计算产品的行列位置和产品面积数据，最终将计算的全部结果显示在图像上。

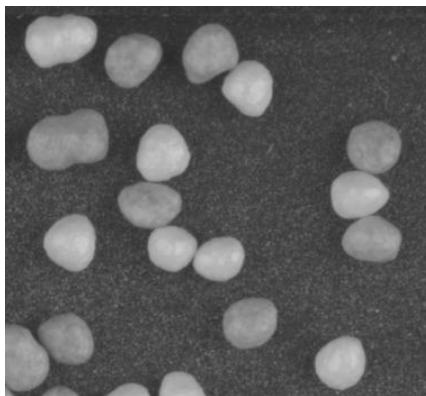


图 8-1 实验图片

2、实验程序：

dev_update_off () //将 dev_update_pc, dev_update_var 和 dev_update_窗口切换到关闭

read_image (Image, 'E:/视觉作业/小球计数.png')

*关闭活动图形窗口。

dev_close_window ()

*返回图像的大小

get_image_size (Image, Width, Height)

*打开一个新的图形窗口

dev_open_window (0, 0, Width*2, Height*2, 'black', WindowHandle)

*修改要显示的图像部分

dev_set_part (0, 0, Height-1, Width-1)

*设置字体属性

set_display_font (WindowHandle, 16, 'mono', 'true', 'false')

dev_set_colored (6) //设置多个输出颜色

dev_set_draw ('margin') //设置 region 填充模式

dev_set_line_width (3) //设置 region 轮廓线宽 (margin 模式)

*在当前图形窗口中显示图像对象

dev_display (Image)

*自动阈值分割，用二进制阈值来分割图像

binary_threshold (Image, Region, 'max_separability', 'light', UsedThreshold)

*利用圆形结构元素进行开运算

opening_circle (Region, RegionOpening, 3.5)

//将 Region1 中去掉半径小于 1 的圆形区域后，将剩下的区域保存到 RegionOpening 中

dev_display (RegionOpening)

*进行连通操作

connection (RegionOpening, ConnectedRegions)

dev_display (Image)

dev_display (ConnectedRegions)

*利用圆形结构元素腐蚀

erosion_circle (RegionOpening, RegionErosion, 7.5)

```

dev_display (Image)
dev_display (RegionErosion)
connection (RegionErosion, ConnectedRegions1)
dev_display (Image)
dev_display (ConnectedRegions1)
*利用圆形结构元素膨胀
dilation_circle (ConnectedRegions1, RegionDilation, 7.5)
*计算对象数量
count_obj (RegionDilation, Number)
*对象排序
sort_region (RegionDilation, SortedRegions, 'first_point', 'true', 'column')
*计算单个对象角度
orientation_region (SortedRegions, Phi)
*计算单个对象圆度
circularity (SortedRegions, Circularity)
dev_display (Image)
dev_display (RegionDilation)
*行列，面积
TupleR := []
TupleC := []
TupleA := []
for Index := 1 to Number by 1
    select_obj (SortedRegions, ObjectSelected, Index)
    area_center (ObjectSelected, Area, Row, Column)
    TupleR[Index] := Row
    TupleC[Index] := Column
    TupleA[Index] := Area
    disp_message (WindowHandle, (Phi[Index - 1] * 57.3)$#.1f' + '°' + ', ' +
(Circularity[Index-1])$#.1f, 'window', (Row-15) * 2, (Column-30) * 2, 'black', 'true')

```

```
disp_message(WindowHandle, '(' + Row$#.1f' + ',' + Column$#.1f' + ') + ',' + Area$#.1f',
'window', Row * 2, (Column-30) * 2, 'black', 'true')
```

```
endfor
```

*显示最终计数结果

```
disp_message(WindowHandle, Number + 'pellets detected', 'window', 12, 500, 'black', 'true')
```

五、实验内容及步骤（包含简要的实验步骤流程）

1、连接相机

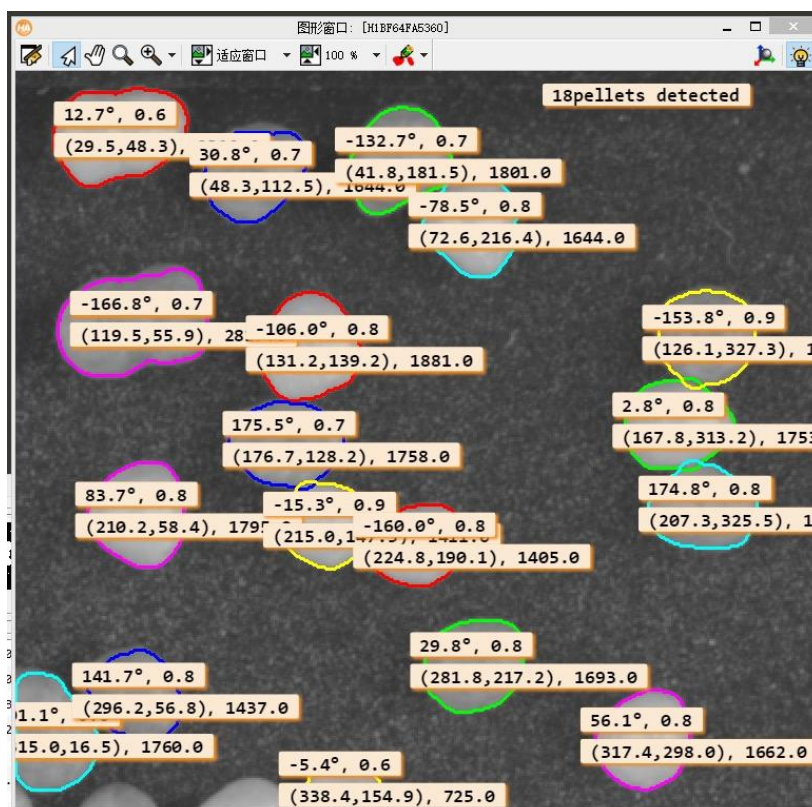


图 8-2 运行结果图片

2、读取图像

读取图片中的信息，获取产品图像数据，将图像数据显示出来，调节不同的参数设定范围，利用腐蚀和膨胀处理图像产品，随后计算产品的各个数值。

3、图像处理

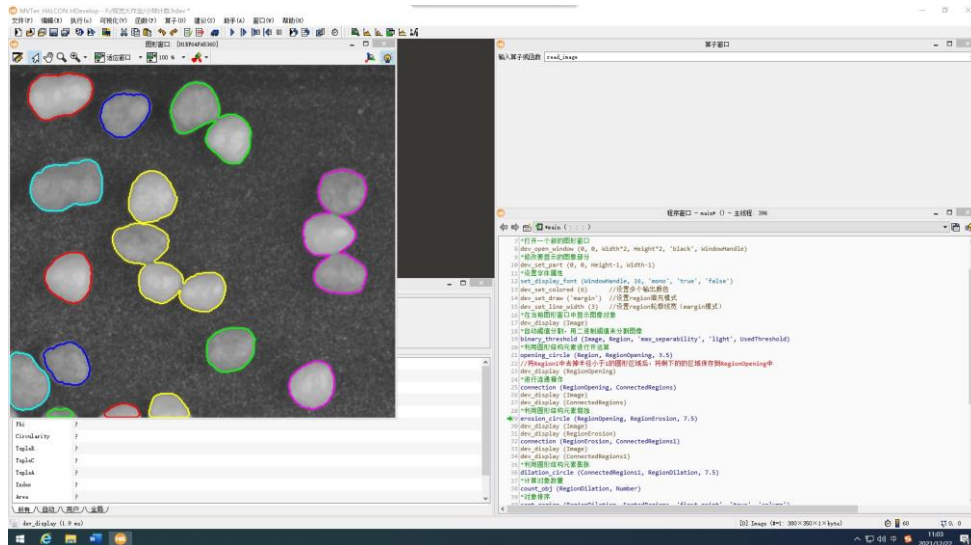


图 8-3 腐蚀设置

六、实验结果

1、完成对产品数量、角度、圆度、行列位置、面积进行计算和显示，对机器视觉机器硬件组装调试，完成摄像头的挑选调试以及光源的调试安装，了解如何避免出现光影，影响摄像头拍照清晰度和软件识别错误。

2、掌握对产品数量、角度、圆度、行列位置、面积进行计算和显示。

9 条码识别

一、实验目的

利用实验设备实现饮料瓶的条形码识别。

二、实验要求

：打光均匀、图像清晰、识别准确、通讯可靠。识别成功后，把识别成功和条形码数值显示在图片左上角。

三、实验（训）条件

1、硬件：PC 一台、相机一个、饮料一瓶、光源一个。

2、软件：Halcon。

四、实验原理和程序

1、实验原理：由于不同颜色的物体，其反射的可见光的波长不同，白色物体能反射各种波长的可见光，黑色物体则吸收各种波长的可见光。条形码就是由一组规则排列的条、空以及对应的字符组成的标记，“条”指对光线反射率较低的部分，“空”指对光线反射率较高的部分，这些条和空组成的数据可以表达一定的信息，能够用特定的设备识读，并转换成与计算机兼容的二进制和十进制信息。每一种物品通过编码唯一识别，编码通过数据库建立条形码与物品信息的对应关系，当条形码的数据传输到计算机时，计算机对数据进行操作和处理。

2、实验程序：

```
read_image (Image, 'C:/Users/Administrator/Desktop/饮料条形码.jpg') //读取图片
get_image_size (Image, W, H) //得到图片尺寸
dev_open_window (0, 0, 900, 400, 'black', WindowHandle) //打开一个黑色窗口
scale_image (Image, ImageScaled, 1, 50) //灰度值
emphasize (ImageScaled, ImageEmphasize, 50, 50, 1) //对比度增强
create_bar_code_model ([], [], BarCodeHandle) //创建一维条形码
set_bar_code_param (BarCodeHandle, 'barcode_width_min', 300) //设置一维条形码参数
find_bar_code (ImageScaled, SymbolRegions, BarCodeHandle, 'auto', DecodedDataStrings) //
检测并读取图像中的一维条形码

clear_bar_code_model (BarCodeHandle) //删除条形码模型并释放分配内存
a:=DecodedDataStrings
if(a='6949843700149') //如果 a=6949843700149，显示以下信息
```

```

disp_message (WindowHandle, '识别成功: OK', 'window', 12, 12, 'black', 'true')

disp_message (WindowHandle, '条码数值='+a, 'window', 32, 12, 'black', 'true')

else //否则, 显示'识别失败:NG'

disp_message (WindowHandle, '识别失败:NG', 'window', 12, 12, 'black', 'true')

Endif

```

五、实验内容及步骤

1、组装实验设备

安装相机、镜头、光源等。将设备与连接 PC，完成网线连接，进行调试实验设备。

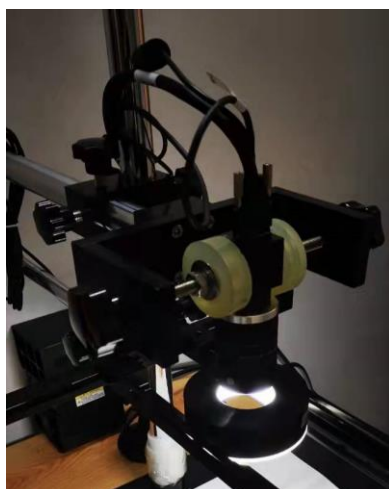


图 9-1 组装设备图片

2、读取图像

将饮料瓶放在工作台上，调节相机焦距、光源亮度以及角度，使图片清晰可见。

3、图像处理

利用 HALCON 软件进行饮料条形码识别的程序编写。

4、实验结果

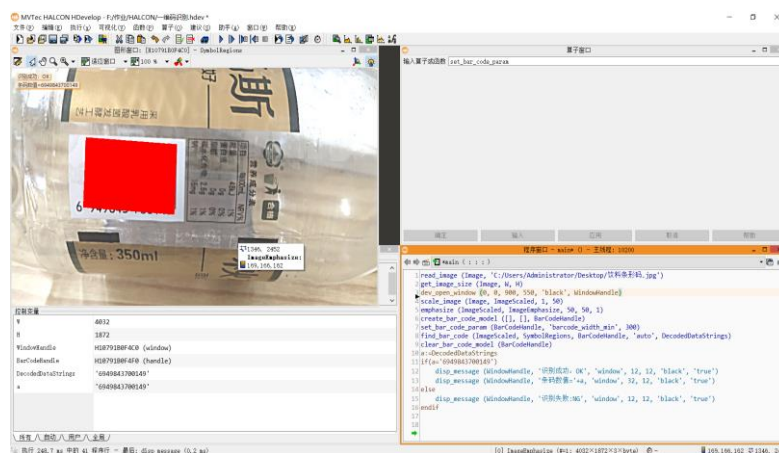


图 9-2 实验结果

10 二维码识别

一、实验目的

利用实验设备实现书籍或饮料瓶的二维码识别，掌握 HALCON 软件二维码识别方法。

二、实验要求

打光均匀、图像清晰、识别准确、通讯可靠。识别成功后，把识别成功和条形码数值显示在图片左上角。

三、实验条件

1、硬件：PC 一台、相机一个、饮料一瓶、光源一个。

2、软件：Halcon。

四、实验原理和程序

1、实验原理：二维码的机制就是用特定的几何图形在二维平面上分布的黑白相间的图形。它的实质就是计算机的逻辑基础语言，0 和 1 进行的排列组合，使用若干个与二进制相对应的几何形体来表示文字数值信息。从字面上就可以看出，二维码是一种比一维码更为先进的条码格式。

2、实验程序：

```
read_image (Image, 'q90g.jpg')
```

```
get_image_size (Image, Width, Height)
```

```
dev_open_window (0, 0, Width, Height, 'black', WindowHandle)
```

```
create_data_code_2d_model ('QR Code', [], [], DataCodeHandle)
```

*参数默认 增强识别

```
set_data_code_2d_param (DataCodeHandle, 'default_parameters', 'enhanced_recognition')
```

*500ms 识别有效

```
set_data_code_2d_param (DataCodeHandle, 'timeout', 500)
```

*白底黑字

```
set_data_code_2d_param (DataCodeHandle, 'polarity', 'dark_on_light')
```

```
find_data_code_2d (Image, SymbolXLDs, DataCodeHandle, 'stop_after_result_num', 3,  
ResultHandles, DecodedDataStrings)
```

```
clear_data_code_2d_model (DataCodeHandle)
```

```
a:=DecodedDataStrings
```

```
if(a='4008535118906')
```

```

disp_message(WindowHandle,'识别成功: OK','window',12,12,'black','true')
disp_message(WindowHandle,'条码数值='+a,'window',32,12,'black','true')
else
    disp_message(WindowHandle,'识别失败: NO','window',12,12,'black','true')
    disp_message(WindowHandle,'条码数值='+a,'window',32,12,'black','true')
Endif

```

五、实验内容及步骤

1、组装实验设备

安装相机、镜头、光源等，将相机与连接 PC，进行调试实验设备。

2、相机调节

调节相机焦距、光源亮度以及角度，使图片清晰可见。

3、图像处理

利用 HALCON 软件进行饮料条形码识别的程序编写。

4、实验结果



图 10-1 实验结果

11 电池尺寸测量

一、实验目的

- 1、利用 HALCON 进行手机电池尺寸测量的实验。
- 2、掌握 HALCON 中 `smallest_rectangle2` 算子的应用。

二、实验要求：

打光均匀、图像清晰、识别准确、通讯可靠。

三、实验（训）条件

- 1、硬件：PC 微机一台。
- 2、软件：Halcon。

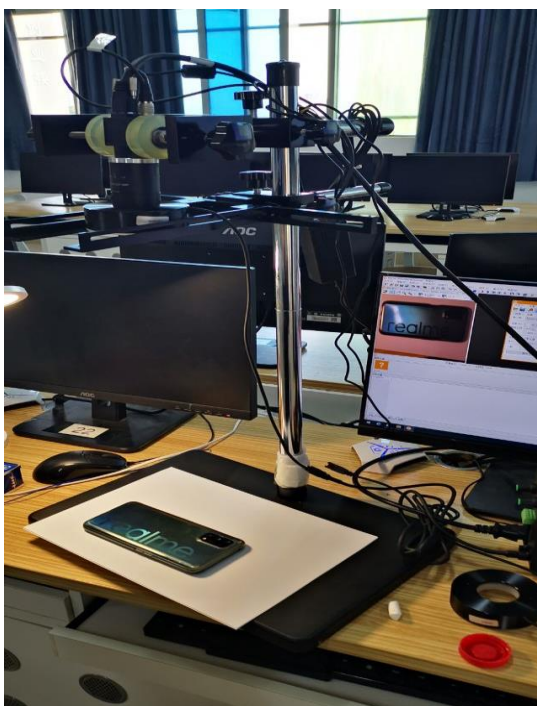


图 11-1 实验设备图片

四、实验原理和程序

1、实验原理：利用 HALCON 中的 `smallest_rectangle2` 算子将图中需测量的手机电池的矩形提取出来，再进行计算测量。

2、实验程序：

```
dev_close_window()
```

```
read_image (Image, 'E:/2018148149/2018148149/电池.jpg')
```

```
get_image_size(Image,Width,Height)
```

```

dev_open_window(0,0,Width/2,Height/2,'black',WindowHandle)
*使用动态阈值分割图像
mean_image(Image,ImageMean,101,101)
dyn_threshold(Image,ImageMean,Region,1,'dark')
*合并像素相连区域
connection(Region,ConnectedRegions)
*选择具有特定特征的区域
select_shape(ConnectedRegions,SelectedRegions,'height','or',500,2000)
*选定最小外接矩形。
smallest_rectangle2(SelectedRegions,Row,Column,Phi,Length1,Length2)
tuple_max(Length1,MaxLength1)
for Index := 0 to |Length1|-1 by 1
    if(Length1[Index]==MaxLength1)
        MaxLength2:=Length2[Index]
        MaxRow:=Row[Index]
        MaxColumn:=Column[Index]
        MaxPhi:=Phi[Index]
    endif
endfor
dev_set_color('pink')
dev_set_line_width(5)
dev_set_draw('margin')
dev_set_color('red')
gen_rectangle2(Rectangle,MaxRow,MaxColumn,MaxPhi,MaxLength1,MaxLength2)
disp_message (WindowHandle, '宽度为: '+MaxLength2*2, 'window', 0, 0, 'black', 'true')
disp_message (WindowHandle, '长度为: '+MaxLength1*2, 'window', 0, 100, 'black', 'true')
disp_message (WindowHandle, '角度为: '+MaxPhi, 'window', 0, 200, 'black', 'true')

```

五、实验内容及步骤

1、连接相机

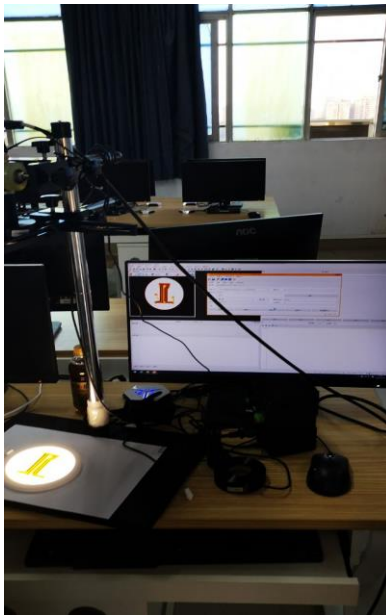


图 1 相机连接

2、读取图像



图 2 电池

3、图像处理



图 3 阈值分割

六、实验结果



图4 测量结果

- 1.考虑到照片采集时光线不均匀的问题，采用阈值分割和灰度值得方法让图片变得相对清晰。
- 2.如需要测量电池的实际尺寸，则要考虑相机的 dpi 问题。

12 齿轮检测

一、实验目的

目的：利用实验设备实现齿轮齿数、外径、安装孔检测。

二、实验要求

：打光均匀、图像清晰、识别准确、通讯可靠。检测成功后，把检测成功的齿轮齿数、外径、安装孔的数据显示在屏幕上

三、实验条件

1、硬件：PC 一台、相机一个、齿轮图像一张、光源一个。

2、软件：Halcon。

四、实验原理和程序

1、实验原理：将图像进行阈值分割获取齿轮区域，对齿轮区域进行开运算减少图像的细节毛刺和平滑图像的边缘。获取齿轮区域的最小外接圆求得齿轮外径，把最小外接圆减去稍微膨胀的最大内接圆就可以求得齿轮的齿所在的区域并求得齿所在位置和齿的数目等信息，求取图像的 XLD 轮廓信息根据圆度、面积等形状特征信息可以求得齿轮的安装孔信息。

2、实验程序：

```
read_image (Image, '齿轮 1.jpg')
dev_close_window ()
get_image_size (Image, Width, Height)
dev_open_window (0, 0, Width, Height, 'black', WindowHandle)
dev_set_draw ('margin')
dev_set_line_width (2)
dev_display (Image)
rgb1_to_gray (Image, GrayImage)

for Index := 0 to 1 by 1
  for Index2 := 0 to 1 by 1
    gen_rectangle1 (Rectangle, 0 + Index*Width/2, 0 + Index2*Height/2 + 10, 202 +
Index*Width/2, 209 + Index2*Height/2)
    reduce_domain (GrayImage, Rectangle, ImageReduced)           //将图像分为四个区域分别
检测
```

threshold (ImageReduced, Regions, 0, 242)

connection (Regions, ConnectedRegions)

select_shape (ConnectedRegions, SelectedRegions, 'area', 'and', 3000, 150000)

opening_circle (SelectedRegions, RegionOpening, 3.5)

fill_up (RegionOpening, RegionFillUp)

*****检测外径*****

shape_trans (RegionFillUp, RegionTrans, 'outer_circle') //将区域转换为最小外接圆

shape_trans (RegionFillUp, RegionTrans1, 'inner_circle') //将区域转换为最大内接圆

dilation_circle (RegionTrans1, RegionDilation, 3.5) //膨胀最大内接圆

area_center (RegionTrans, Area, Row, Column) //求得最小外接圆的面积,

根据面积可以知道齿轮外径

*****计算齿轮齿数*****

difference (RegionFillUp, RegionDilation, RegionDifference)//最小外接圆减去膨胀最大内接圆获得齿区域

connection (RegionDifference, ConnectedRegions1) //计算连通区域

area_center (ConnectedRegions1, Area1, Row1, Column1) //获取齿的位置信息

count_obj (ConnectedRegions1, Number) //计算齿数

*****检测安装孔*****

edges_sub_pix (ImageReduced, Edges, 'canny', 1, 20, 40) //获取图像的 XLD 轮廓

union_adjacent_contours_xld (Edges, UnionContours, 10, 1, 'attr_keep') //合并相邻的 XLD 轮廓

select_shape_xld (UnionContours, SelectedXLD, 'circularity', 'and', 0.9, 1) //根据形状特征选取区域

select_shape_xld (SelectedXLD, SelectedXLD, 'area', 'and', 600, 1500)

area_center_xld (SelectedXLD, AreaXLD, Row2, Column2, PointOrder)

//获得安装孔的面积和位置信息

dev_display (ImageReduced)

dev_display (RegionTrans)

dev_display (SelectedXLD)

```

disp_cross (WindowHandle, Row1, Column1, 6, 0)
disp_message (WindowHandle, '齿轮齿数为: ' + Number + ', 齿轮外径为: ' + sqrt(Area /
    3.1415926)$'0.2f' + ', 安装孔直径为: ' + sqrt(AreaXLD / 3.14159)$'0.2f', 'window',
    Width / 2 - 3, Height / 2 - 150, 'black', 'true')
stop()
endfor
endifor
endifor

```

五、实验步骤

1、组装实验设备。

安装相机、镜头、光源等。将设备与连接 PC，完成串口连接，进行调试实验设备。

2、读取图像。

将齿轮放在工作台上，调节相机焦距、光源亮度以及角度，使图片清晰可见。

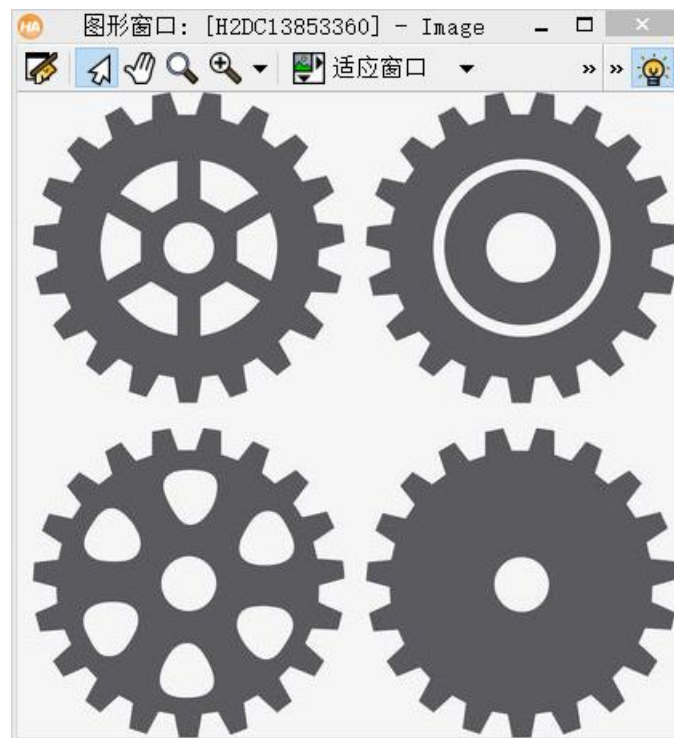


图 12-1 读取图像图片

(3) 图像处理

利用 HALCON 软件进行饮料条形码识别的程序编写。

六、实验结果

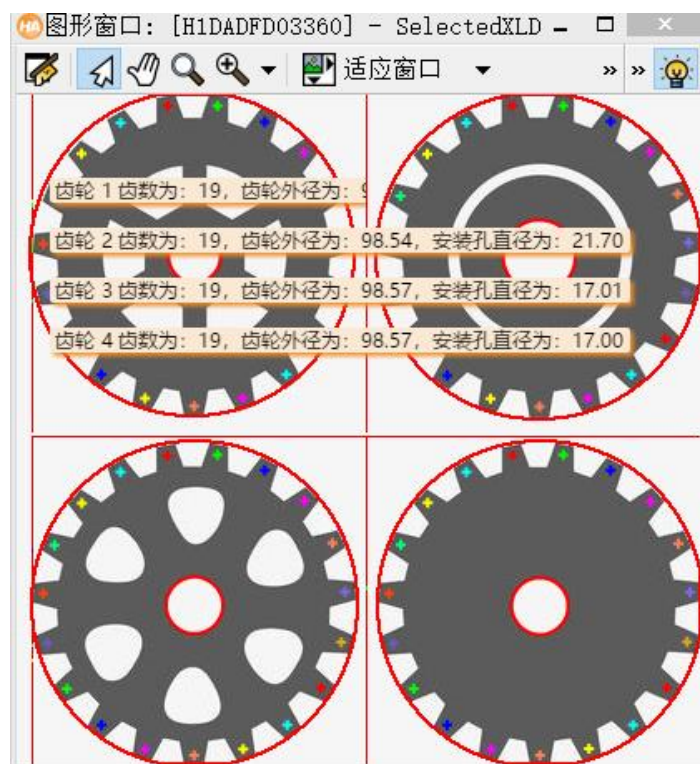


图 12-2 实验结果

13 电缆颜色检测

一、实验目的

1、熟悉机器视觉中有关摄像机的调试与使用，HALCON 与计算机的连接，光源与摄像机的安装调试；

2、掌握 HALCON 的编程，对图像进行分析，在实践中对 HALCON 中的算子进行充分实践了解；

3、学会对于相机的选用需要考虑的参数，光源的考虑的参数。掌握相机光源的调节，调节适合的焦距，使得图像清晰，边界分明的方法；

二、实验要求

：在摄像机于电脑 HALCON 软件链接。获取清晰的图像，以图像中产品颜色作为基础去识别，对后面产品颜色进行识别和显示。

三、实验条件

1、硬件：PC 微机一台。

2、软件：Halcon。

四、实验原理和程序

1、实验原理：将图片导入 HALCON，获取一个三通道图像里的每个通道的图像数据，将 RGB 三通道数据转化为 HSV 色彩空间的三通道图像数据，对 HSV 图像中的饱和度通道进行阈值操作，不同颜色调节不同的阈值范围。获取上述阈值操作后区域中的色调通道图像数据，将区域分割。对上述色调通道图像数据进行阈值处理，获取区域里连接的组件，保留 ConnectedRegions 里的最大的区域，对 SelectedRegions 进行闭运算操作，将 red 区域里的图像剪切出来，显示出对应颜色产品。



图 13-1 产品图片

2、实验程序

*关闭已经打开的窗口

```
dev_close_window ()
```

*打开新窗口

```
dev_open_window (0, 0, 640, 480, 'black', WindowHandle)
```

```
for i := 1 to 2 by 1
```

*打开一张图像

```
read_image (Image, 'cable' + i)
```

*获取一个三通道图像里的每个通道的图像数据

```
decompose3 (Image, Red, Green, Blue)
```

*将 RGB 三通道数据转化为 HSV 色彩空间的三通道图像数据

```
trans_from_rgb (Red, Green, Blue, Hue, Saturation, Intensity, 'hsv')
```

*对 HSV 图像中的饱和度通道进行阈值操作

```
threshold (Saturation, HighSaturation, 100, 255)
```

*获取上述阈值操作后区域中的色调通道图像数据

```
reduce_domain (Hue, HighSaturation, HueHighSaturation)
```

*对上述色调通道图像数据进行阈值处理

```
*threshold (HueHighSaturation, Yellow, 20, 50)
```

```
threshold (HueHighSaturation, red, 220, 255)
```

*获取区域里连接的组件

```
connection (red, ConnectedRegions)
```

*保留 ConnectedRegions 里的最大的区域

```
select_shape_std (ConnectedRegions, SelectedRegions, 'max_area', 0)
```

*对 SelectedRegions 进行闭运算操作

```
closing_circle (SelectedRegions, Red, 3.5)
```

*将 red 区域里的图像剪切出来

```
reduce_domain (Image, red, ImageReduced)
```

*显示图像

```
dev_clear_window()
```

```
dev_set_color ('white')
```

```
dev_set_draw ('margin')
```

*显示黄色的电线

```
dev_display (Image)
```

```
dev_display (Red)
stop ()
endfor
```

五、实验步骤

1、连接相机

进行产品拍照，获取产品信息

2、读取图像

读取图片中的信息，获取一个三通道图像里的每个通道的图像数据，将 RGB 三通道数据转化为 HSV 色彩空间的三通道图像数据，调节不同的参数设定范围，把符合要求的颜色进行框选出来。

3、图像处理

六、实验结果

1、完成对产品颜色拍照识别，对机器视觉机器硬件组装调试，完成摄像头的挑选调试以及光源的调试安装，了解如何避免出现光影，影响摄像头拍照清晰度和软件识别错误。

七、思考问题

1、将一种颜色识别进行举一反三，可以做出对应许多种颜色的识别。

2、思考进行拓展，可以识别不同产品。

3、综合多种实验做出更多的产品特征识别。

14 识别定位和计数

一、实验目的

- 1、熟悉机器视觉在机器人抓取过程中物品识别、位置测量的应用。
- 2、掌握 HALCON 的计数、方位算子的应用；

二、实验要求

相机与电脑连接，获取物品的清晰图像，软件对产品进行识别和显示。

三、实验条件

- 1、硬件：PC 微机一台。
- 2、软件：Halcon。

四、实验原理和程序

1、实验原理：将物品图像读入软件，根据物品的模板或特征参数，获取图像中的每个物品的种类、序号、位置、大小、方位角、计数等。

2、实验程序

* 大枣坐标方位计算程序

```
read_image(Image,'大枣 9.jpg')
```

```
get_image_size(Image,W,H)
```

```
dev_close_window()
```

```
dev_open_window(0,0,W,H,'black',WindowHandle)
```

```
rgb1_to_gray(Image,GrayImage)
```

```
threshold(GrayImage,Regions1,2,90)
```

```
fill_up (Regions1, RegionFillUp)
```

```
connection(RegionFillUp,Region2)
```

```
select_shape(Region2, SelectedRegions, 'area', 'and', 15000, 99999)
```

```
sort_region (SelectedRegions, SortedRegions, 'first_point', 'true', 'column')
```

* 行坐标、列坐标、方位角 的数组

```
TupleR:=[]
```

```
TupleC:=[]
```

```
TuplePhi:=[]
```

* 九次循环，求个物体的参数

```

for Index := 1 to 9 by 1
select_obj (SortedRegions, ObjectSelected, Index)
area_center (ObjectSelected, AreaObjectSelected, RowObjectSelected, ColumnObjectSelected)
* 方位角计算和显示, $#.1f' 显示小数点 1 位精度
orientation_region (ObjectSelected, Phi)
set_tposition(WindowHandle, RowObjectSelected+60, ColumnObjectSelected+100)
write_string(WindowHandle,(57.3*Phi)$#.1f' )
TuplePhi[Index]:=Phi*57.3
* 序号显示
set_tposition(WindowHandle, RowObjectSelected+0, ColumnObjectSelected+100)
write_string(WindowHandle,Index )
set_tposition(WindowHandle, RowObjectSelected+20, ColumnObjectSelected+100)
* $#.1f' 显示小数点 1 位精度, 中心行坐标
write_string(WindowHandle,RowObjectSelected$#.1f' )
TupleR[Index]:=RowObjectSelected
* $#.1f' 显示小数点 1 位精度, 中心列坐标
set_tposition(WindowHandle, RowObjectSelected+40, ColumnObjectSelected+100)
write_string(WindowHandle,ColumnObjectSelected$#.1f' )
TupleC[Index]:=ColumnObjectSelected
endfor

```

五、实验步骤

1、连接相机

产品拍照，获取产品位置图像；

2、读取图像并分析处理显示。

六、实验结果



图 14-2 检测品原图

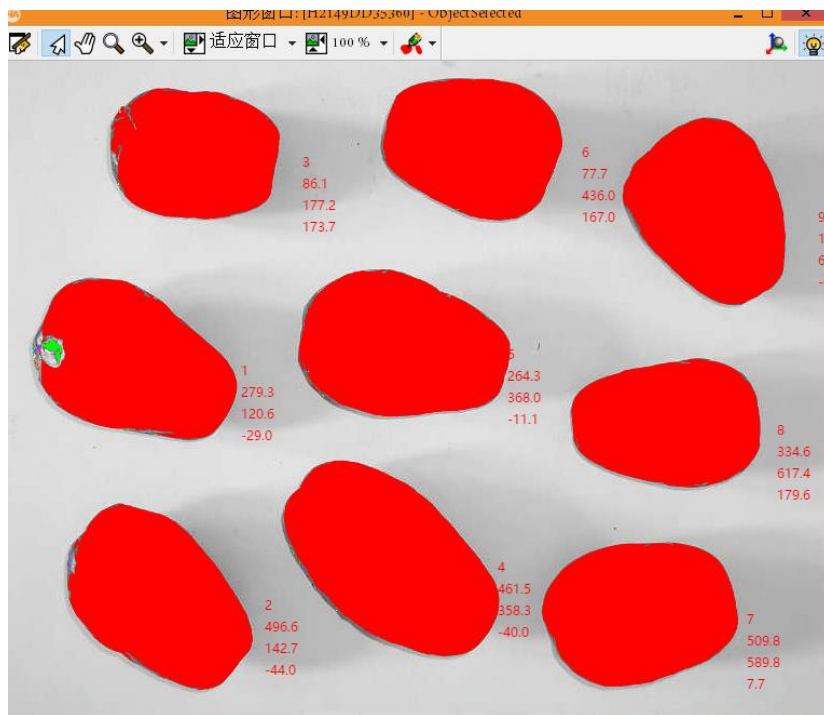


图 14-2 定位识别显示

七、思考问题

- 1、将多种产品识别，算出其中心位置、方位角、长宽和数量；
- 2、如何将定位参数传送给工位上的机器人，实现抓取动作。

15 开关管脚宽度测量

一、实验目的

熟悉产品尺寸测量算子的应用，掌握多尺寸测量的方法。

二、实验要求

获取物品的清晰产品图像，软件对产品进行识别和显示。

三、实验条件

1、硬件：PC 微机一台。2、软件：Halcon。

四、实验原理和程序

1、实验原理：将产品图像读入软件，根据物品的像素和方向，获取每个尺寸的数值。

2、实验程序

* 开关管脚距离测量，

* Measure 01: Prepare measurement

```
read_image (Image, 'bin_switch.png')
```

```
dev_display (Image)
```

```
get_image_size (Image, Width, Height)
```

```
dev_close_window()
```

```
dev_open_window(0,0,Width,Height,'black',WindowHandle)
```

* 绘制有角度的矩形

```
gen_rectangle2 (ROI_0, 294, 181, rad(47), 54, 18)
```

*画绿色线宽 2 的十字线

```
dev_set_color ('green')
```

```
dev_set_line_width (2)
```

* 测量 7 个十字线位置和距离

```
gen_measure_rectangle2 (294, 181, rad(47), 54, 18, Width, Height, 'nearest_neighbor',  
MeasureHandle)
```

```
measure_pos (Image, MeasureHandle, 1, 20, 'all', 'all', RowEdge, ColumnEdge, Amplitude,  
Distance)
```

```
gen_cross_contour_xld (Cross, RowEdge, ColumnEdge, 10, 0.785398)
```

```
dev_display (Image)
```

```
***** 6 次循环，显示 6 个距离的参数
```

```
for Index := 0 to 5 by 1
```

```
  * '$#.1f' 或 '$0.1f' 显示小数点 1 位精度，行坐标 ColumnEdge，中心列坐标  
  ColumnEdge+Index*20+30,白字显示距离
```

```
  * 白字显示
```

```
  dev_set_color ('red')
```

```
  set_tposition(WindowHandle, RowEdge[Index], ColumnEdge[Index]+Index*20+30)
```

```
  write_string(WindowHandle,Distance[Index]$.1f )
```

```
  disp_message (WindowHandle,'距离='+ (Distance[Index])$.2f, 'window', 60+20*Index, 20,  
'black', 'true')
```

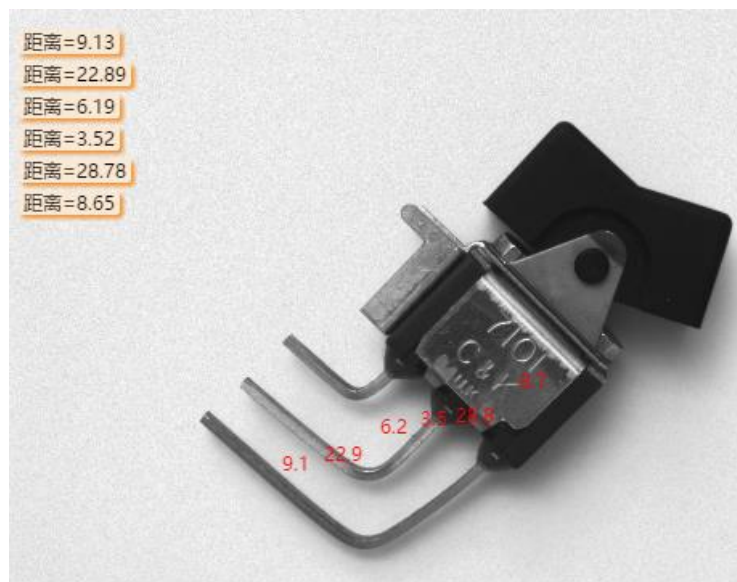
```
endfor
```

五、实验步骤

1、连接相机，产品拍照，获取产品图像；

2、读取图像并分析处理显示。

六、实验结果



七、思考问题

1、考虑如何在生产线上连续测量出多个开关的管脚尺寸；

2、如何判断管脚是否合格。

16 模板匹配

一、实验目的

熟悉模板匹配算子的应用，学会寻找相似物品的方法。

二、实验要求

获取物品的清晰产品图像，软件对产品进行识别和显示。

三、实验条件

1、硬件：PC 微机一台。2、软件：Halcon。

四、实验原理和程序

1、实验原理：将产品图像读入软件，根据物品的形状和相似度，获取每个物品的位置和方向。

2、实验程序

*模板匹配：寻找缩放和旋转的形状模型

*find scaled and rotated shape models.

dev_update_pc ('off')

dev_update_window ('off')

dev_update_var ('off')

* 读取单个图像图

read_image (Image, 'clip 单个')

get_image_size (Image, Width, Height)

dev_close_window ()

dev_open_window (0, 0, Width, Height, 'black', WindowHandle)

dev_set_color ('red')

dev_display (Image)

threshold (Image, Region, 0, 128)

connection (Region, ConnectedRegions)

* 选择面积大小

select_shape (ConnectedRegions, SelectedRegions, 'area', 'and', 6500, 8000)

* 填充和膨胀

```

fill_up (SelectedRegions, RegionFillUp)
dilation_circle (RegionFillUp, RegionDilation, 5.5)
* 获得特定区域 Region 位置的图像
reduce_domain (Image, RegionDilation, ImageReduced)
create_scaled_shape_model (ImageReduced, 5, rad(0), rad(360), 'auto', 0.8, 1.1, 'auto', 'none',
'ignore_global_polarity', 40, 10, ModelID)
*get_shape_model_contours 出现在图像窗口的 (0, 0) 处
get_shape_model_contours (Model, ModelID, 1)
* 求区域中心坐标
area_center (RegionFillUp, Area, RowRef, ColumnRef)
*从点和角度计算刚性仿射变换。
vector_angle_to_rigid (0, 0, 0, RowRef, ColumnRef, 0, HomMat2D)
*对 XLD 轮廓进行任意仿射 2D 变换。
affine_trans_contour_xld (Model, ModelTrans, HomMat2D)
dev_display (Image)
dev_display (ModelTrans)
* 读取多个图像图
read_image (ImageSearch, 'clip')
dev_display (ImageSearch)
*寻找缩放形状模板， rad(0), rad(360)好使；
find_scaled_shape_model (ImageSearch, ModelID, rad(0), rad(360), 0.8, 1.0, 0.5, 0, 0.5,
'least_squares', 5, 0.8, Row, Column, Angle, Scale, Score)
for I := 0 to |Score| - 1 by 1
    hom_mat2d_identity (HomMat2DIdentity)
    hom_mat2d_translate (HomMat2DIdentity, Row[I], Column[I], HomMat2DTranslate)
    hom_mat2d_rotate    (HomMat2DTranslate,    Angle[I],    Row[I],    Column[I],
HomMat2DRotate)
    hom_mat2d_scale    (HomMat2DRotate,    Scale[I],    Scale[I],    Row[I],    Column[I],
HomMat2DScale)
    affine_trans_contour_xld (Model, ModelTrans, HomMat2DScale)

```

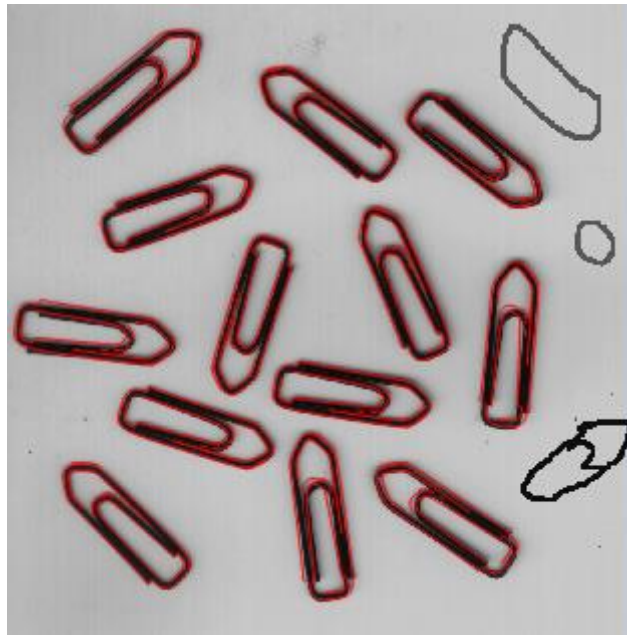
```
dev_display (ModelTrans)
```

```
endfor
```

五、实验步骤

- 1、连接相机，产品拍照，获取产品图像；
- 2、读取图像并分析处理显示。

六、实验结果



七、思考问题

- 1、考虑如何在生产线上连续识别出多个相同的位置和尺寸；
- 2、如何判断产品是否满足指标要求。