



## 计算机视觉 课程实验报告

学 号 : 201600181073	姓名: 唐超	班级: 智能 16
实验题目: 图像结构 2		
<p>实验内容:</p> <p><b>1. Canny 边缘检测</b></p> <p>了解 OpenCV 中 canny 边缘检测函数的用法, 并选取图像进行测试, 观察阈值对结果的影响。</p> <p><b>2. 霍夫变换</b></p> <p>实现基于霍夫变换的图像圆检测 (边缘检测可以用 opencv 的 canny 函数)。</p>		
<p>实验过程中遇到和解决的问题:</p> <p><b>1. Canny 函数的用法</b></p> <p><code>void Canny(InputArray image, OutputArray edges, double threshold1, double threshold2, int apertureSize=3, bool L2gradient=false)</code></p> <p>前两个参数是我们的输入、输出图像 (8位灰度图)。第三、四个参数分别是我们的MinVal和MaxVal。推荐的高低阈值比在2:1到3:1之间第五个参数是aperture_size。它是用于查找图像梯度的Sobel核的大小。默认情况下是3。最后一个参数是L2gradient, 它指定了求梯度幅值的公式。如果是True, 它使用更精确的方程<math>\sqrt{G_x^2+G_y^2}</math>, 否则它使用这个函数: <math> G_x + G_y </math>。默认情况下, 它是False。</p> <p>取高低阈值分别为70, 140, 原图的灰度图及边缘检测图如下:</p> <div style="display: flex; justify-content: space-around; align-items: center;">   </div> <p><b>2. 计算各像素点的梯度</b></p> <p>用 sobel 算子算出每一点 x, y 方向的梯度, 在将两方向的梯度值放进极坐标, 得到该点梯度方向与 x 轴的角度 (注意由于极坐标的特点, 此处</p>		

的角度以顺时针方向为正方向，后续投票时需要进行变换)

// 求得x和y方向的一阶微分

```
Mat sobelx;
```

```
Mat sobely;
```

```
Sobel(src_gray, sobelx, CV_32F, 1, 0, 3);
```

```
Sobel(src_gray, sobely, CV_32F, 0, 1, 3);
```

// 求得梯度和方向

```
Mat norm;
```

```
Mat dir;
```

```
cartToPolar(sobelx, sobely, norm, dir); // 直角坐标转换为极坐标，dir 中每一像素点的角度  $0 \sim 2\pi$ 
```

### 3. 投票

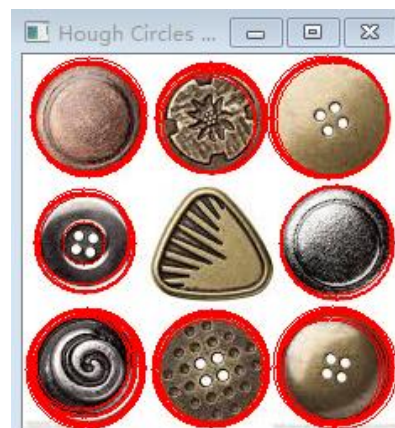
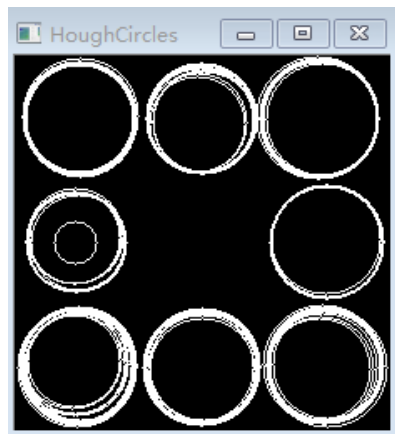
遍历 canny 检测结果中每一个像素值不为 0 的点，采用每一个可能的  $r$  (整型) 计算其可能所在圆的圆心位置，由于 2 中提到的原因，此处的  $\theta = 2\pi - \text{dir.at}\langle\text{float}\rangle(i, j)$ ，另外在处理时一律采用行列坐标，并不采用标准直角坐标，因此有以下计算公式：

$$a = i - r * \sin(\text{dir.at}\langle\text{float}\rangle(i, j));$$
$$b = j - r * \cos(\text{dir.at}\langle\text{float}\rangle(i, j));$$

接着对  $H[a][b][r]$  加 1。

### 4. 画出检测的票数高的圆

用对满足圆方程的每一点赋值的方法存在误差，因此可以对每一个票数较高的圆的参数  $a, b, r$ ，直接用 `circle` 函数画出。结果如下：



结论分析与体会：

通过这次实验熟悉了几个以前不常用的函数：`sobel`, `cartToPolar`, `circle`。对霍夫检测有了进一步的理解。

