

计算机视觉第一次作业

姓名：王钰晴

专业：计算机科学与技术

学号：2120150425

1. 问题描述：

使用同时包含三个物体的图像作为样本，利用物体二值图像的几何性质和/或拓扑性质作为识别特征，将单个物体的图像作为测试图像，识别签字笔、校园一卡通和硬币三种物体。

前提假设：

- 待识别物体颜色较深，背景颜色较浅
- 光照均匀
- 物体内部没有或只有较少孔洞

2. 主要内容：

➤ 函数介绍

`binary.m`：对图像进行二值化。

`corrode.m`：腐蚀操作，对于图像中的每个点，如果其八邻域中有一个点为 0，则该点为 0。

`inflate.m`：膨胀操作，对于图像中的每个点，如果其八邻域中有一个点为 1，则该点为 1。

`connected.m`：计算连通域，输入为二值图像，输出为标记后的连通域和每个域的面积。

`Boundary.m`：8 连通的连通域边界跟踪算法，输入为标记好连通域的图像，输出为每个连通域的周长。

`train.m`：读取包含三个物体的图像，调用上述函数获取物体的周长和面积，计算不同物体紧密度。

`test.m`：读取包含单个物体的图像，计算物体紧密度，从而判别物体类型。

➤ 算法流程：

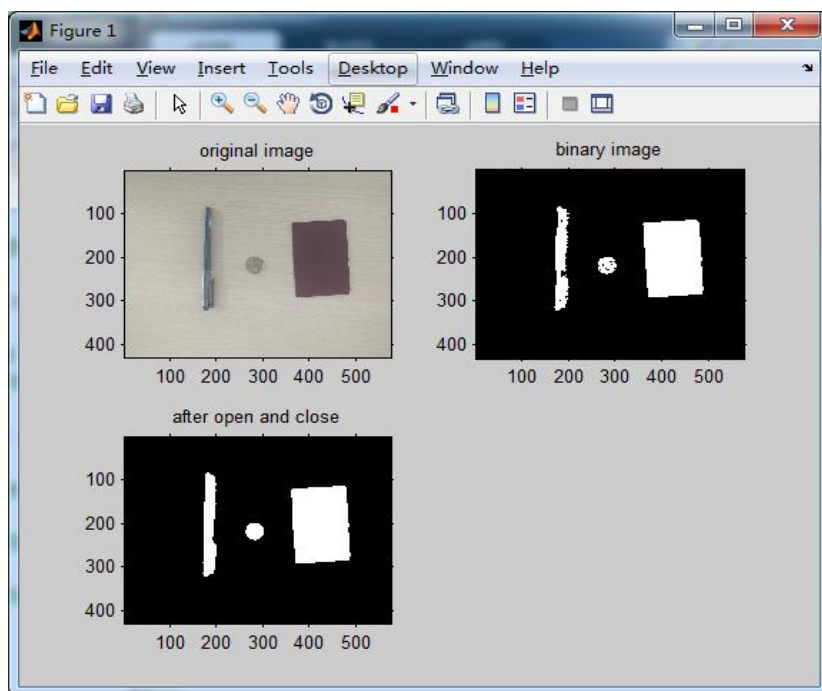
- 1) 获取图像，选取 RGB 图像的单通道作为灰度图像，并对其进行二值化。
- 2) 对图像进行腐蚀膨胀，去除噪点，填充孔洞。
- 3) 对处理后图像进行连通域标记。
- 4) 对每个连通域计算其面积和周长。
- 5) 将周长的平方除以面积作为紧密度的参数，根据不同物体的紧密度不同进行分类。

3. 处理结果：

首先运行 `train.m`，输入图像为 `1.jpg, 2.jpg, 3.jpg` 时的情况如下：

从前到后依次为：原图，二值化图像，腐蚀膨胀后图像，面积向量，周长向量，紧密度：

图像 1.jpg:

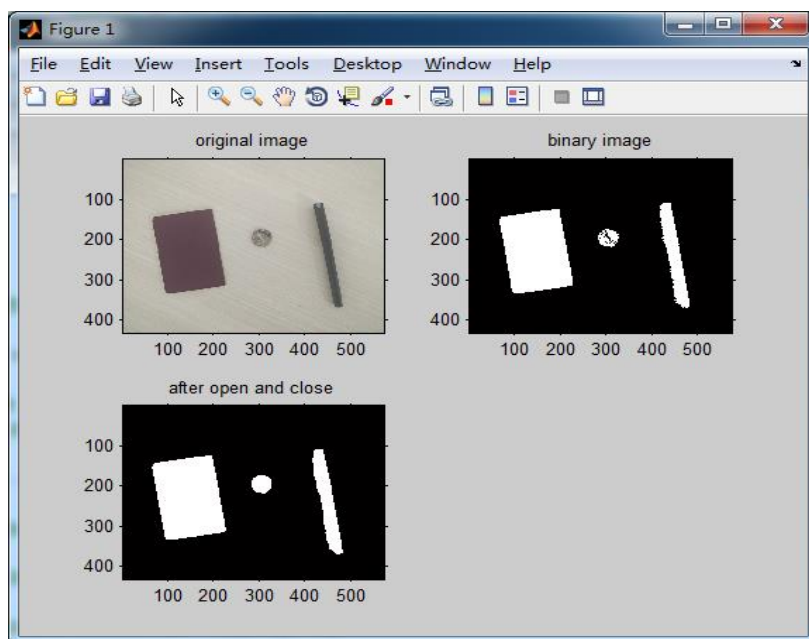


area <3x1 double>		
	1	2
1	5356	
2	20577	
3	1254	
4		

perimeter <3x1 double>		
	1	2
1	509	
2	564	
3	113	
4		

ratio <3x1 double>		
	1	2
1	48.3721	
2	15.4588	
3	10.1826	
4		

图像 2.jpg:

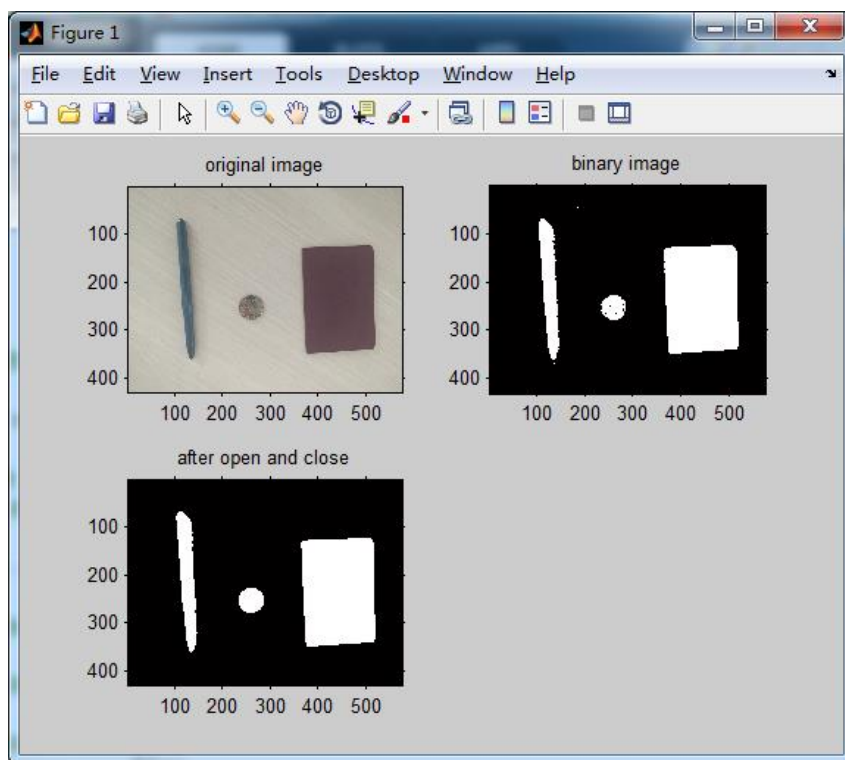


area <3x1 double>		
	1	2
1	7534	
2	26229	
3	1613	
4		

perimeter <3x1 double>		
	1	2
1	559	
2	632	
3	133	
4		

ratio <3x1 double>		
	1	2
1	41.4761	
2	15.2283	
3	10.9665	
4		

3. jpg:



area <3x1 double>		
	1	2
1	7675	
2	32650	
3	2210	
4		

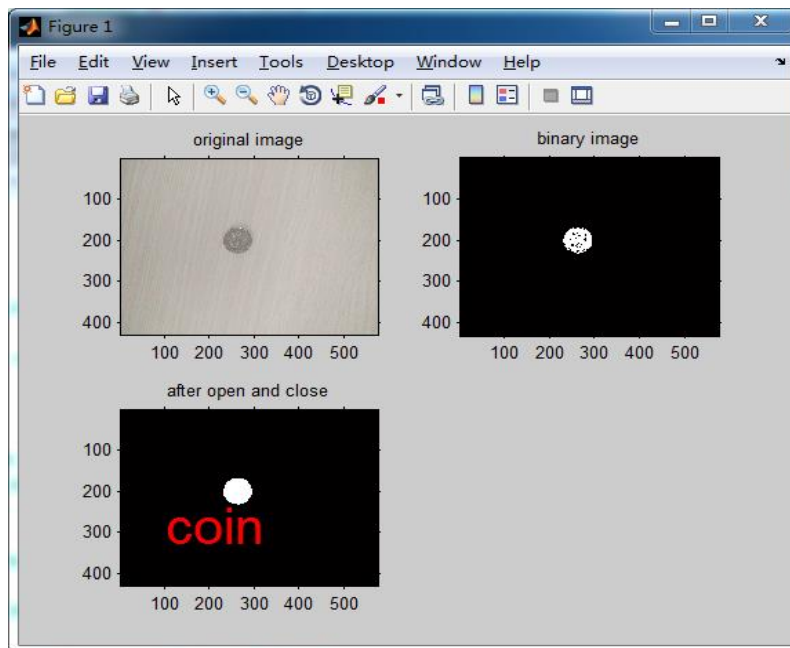
perimeter <3x1 double>		
	1	2
1	604	
2	716	
3	148	
4		

ratio <3x1 double>		
	1	2
1	47.5330	
2	15.7016	
3	9.9113	
4		

由以上训练样本数据不难发现，笔的紧密度最大，在 30 以上，卡片的紧密度在 30~12 之间，硬币的紧密度在 12 以下，我们基于此对三类物体进行分类，得到下列测试结果。

运行 test.m:

4.jpg:

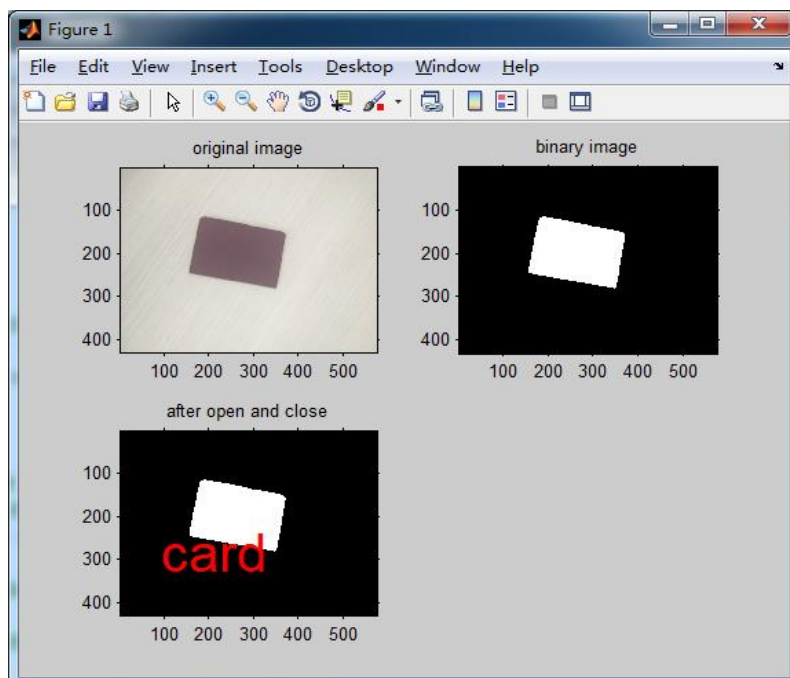


area <1x1 double>		
	1	2
1	3228	

perimeter <1x1 double>		
	1	2
1	182	

ratio <1x1 double>		
	1	2
1	10.2615	

5.jpg

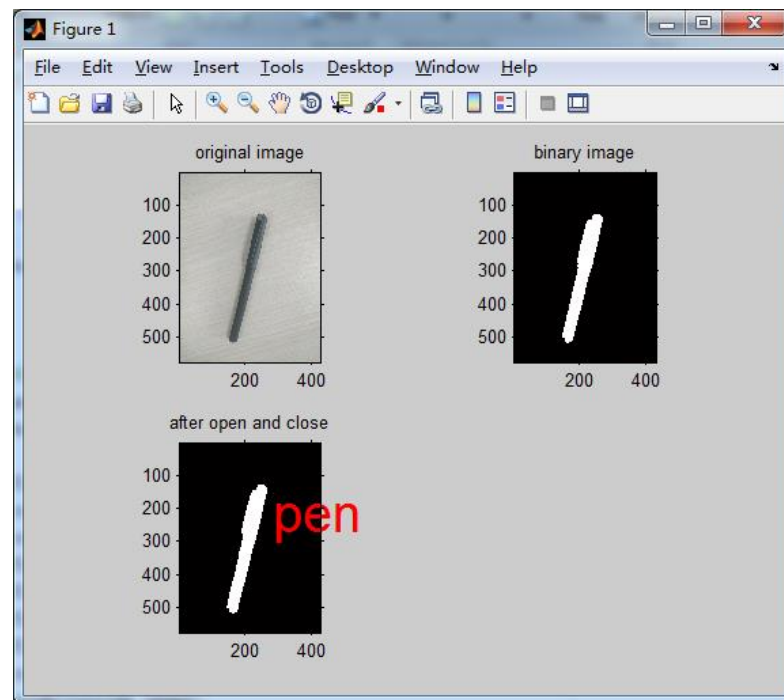


area <1x1 double>		
	1	2
1	26482	
2		

perimeter <1x1 double>		
	1	2
1	633	
2		

ratio <1x1 double>		
	1	2
1	15.1306	
2		

6.jpg:



area <1x1 double>		
	1	2
1	15896	
2		

perimeter <1x1 double>		
	1	2
1	826	
2		

ratio <1x1 double>		
	1	2
1	42.9212	
2		

4. 分析:

- 由以上结果可知，对于满足假设条件：光照均匀，目标与背景灰度差距大，目标内部没有很多孔洞的情况下，该算法的检测结果比较准确。
- 图像的预处理对后期结果的影响很大，图像的开闭运算可以很好的消除噪点和孔洞。如果前期未能对其进行很好的处理，则会在后期出现很多小连通域，干扰结果的分析。如果图像中存在单条毛刺的话，会导致边界跟踪算法无法找到周围的点而陷入循环，遍历失败。
- 对于目标情况更为复杂的情况，则需要更加自适应的算法或者更多的特征进行训练。
- 编程的过程是对书面的算法进行实际情况的实现的过程，需要考虑各种边界和限制条件。如在边界寻找算法中，对于某点的周围点的顺时针查找，会随着所在位置的不同导致遍历起始角度的不同，则顺时针遍历矩阵也需进行实时的旋转，即每次迭代中都需进行更新。