中山大學
SUN YAT-SEN UNIVERSITY

《计算机视觉实验报告》

（作业三）

学院名称 ： 数据科学与计算机学院

专业班级 ： 17 级软件工程 5 班

学生姓名 ： 张淇

学　　号 ： 17343153

时　　间 ： 2019 年 10 月 14 日

成绩：

Exp3 ：Edge and Hough Tranformation

一、 实验内容

1. 给定不同角度的A4纸，输出图像边缘、直线方程、边缘图直线、边缘点、四个角点。
2. 给定有若干硬币或交通指示牌的图片，输出图像的边缘、将其拟合成圆、输出硬币/指示牌的数量。

二、 实验原理

霍夫变换：霍夫变换是一种特征检测(feature extraction)，被广泛应用在图像分析（image analysis）、计算机视觉(computer vision)以及数位影像处理(digital image processing)。霍夫变换是用来辨别找出物件中的特征，例如：线条。他的算法流程大致如下，给定一个物件、要辨别的形状的种类，算法会在参数空间(parameter space)中执行投票来决定物体的形状，而这是由累加空间(accumulator space)里的局部最大值(local maximum)来决定。

三、 实验器材

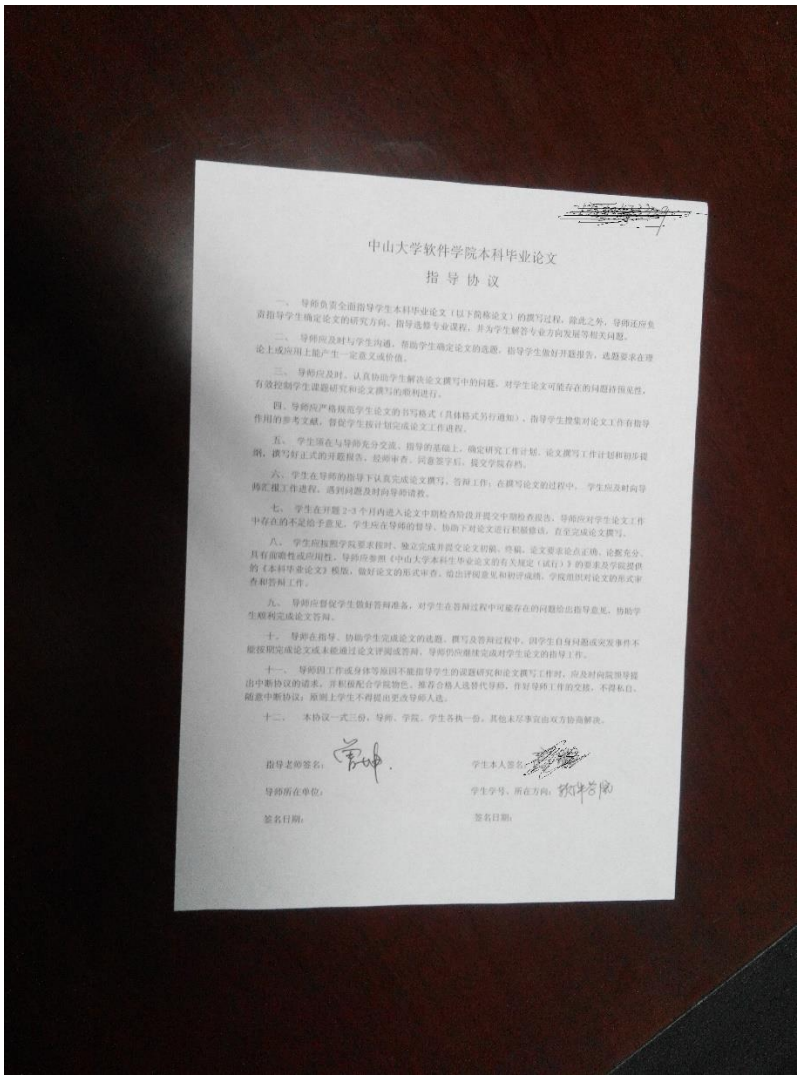
Ubuntu 18.04 Terminal， g++ 7.4.0， CImg

四、 实验过程与结果

(一) 直线边缘检测

以下说明只采用一个实验样例的结果作为代表（其他结果可以在对应文件夹下的 ans_img 文件中查看）：

源图像：



中山大学软件学院本科毕业论文

指 导 协 议

一、 导师负责全面指导学生本科毕业论文（以下简称论文）的撰写过程，除此之外，导师还应负责指导学生确定论文的研究方向、指导选修专业课程，并为学生解答专业方向发展等相关问题。

二、 导师应及时与学生沟通，帮助学生确定论文的选题，指导学生做好开题报告，选题要求在理论上或应用上能产生一定意义或价值。

三、 导师应及时、认真协助学生解决论文撰写中的问题，对学生论文可能存在的问题持预见性，有效控制学生课题研究和论文撰写的顺利进行。

四、 导师应严格按范学生论文的书写格式（具体格式另行通知），指导学生搜集对论文工作有指导作用的参考文献，督促学生按计划完成论文工作进程。

五、 学生原在与导师充分交流、指导的基础上，确定研究工作计划、论文撰写工作计划和初步提纲，撰写好正式的开题报告，经师审查、同意签字后，提交学院存档。

六、 学生在导师的指导下认真完成论文撰写、答辩工作，在撰写论文的过程中，学生应及时向导师汇报工作进程，遇到问题及时向导师请教。

七、 学生在开题 2-3 个月内进入论文中期检查阶段并提交中期检查报告，导师应对学生论文工作中存在的不足给予意见，学生应在导师的督导、协助下对论文进行积极修改，直至完成论文撰写。

八、 学生应按照学院要求按时、独立完成并提交论文初稿、终稿，论文要求论点正确、论据充分、具有前瞻性或应用性，导师应参照《中山大学本科生毕业论文的有关规定（试行）》的要求及学院提供的《本科毕业论文》模版，做好论文的形式审查，给出评阅意见和初评成绩，学院组织对论文的形式审查和答辩工作。

九、 导师应督促学生做好答辩准备，对学生在答辩过程中可能存在的问题给出指导意见，协助学生顺利完成论文答辩。

十、 导师在指导、协助学生完成论文的选题、撰写及答辩过程中，因学生自身问题或突发事件不能按原定完成论文或未能通过论文评阅或答辩，导师仍应继续完成对学生论文的指导工作。

十一、 导师因工作或身体等原因不能指导学生的课题研究和论文撰写工作时，应及时向院领导提出中断协议的请求，并积极配合学院物色、推荐合格人选替代导师，作好导师工作的交接，不得私自、随意中断协议；原则上学生不得提出更改导师人选。

十二、 本协议一式三份，导师、学院、学生各执一份，其他未尽事宜由双方协商解决。

指导老师签名：

导师所在单位：

签名日期：

学生本人签名：

学生学号、所在方向：

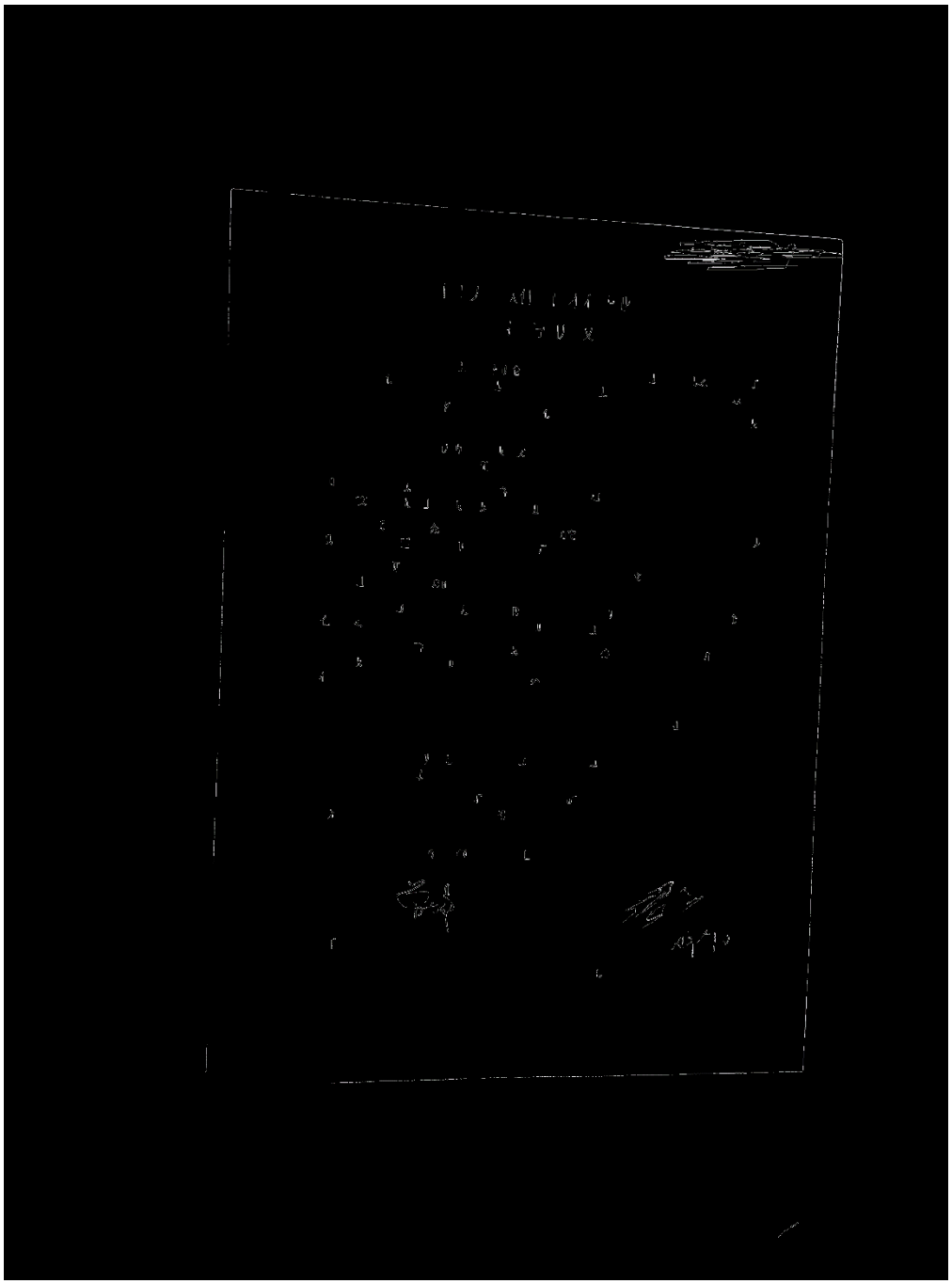
签名日期：

1. 输出图像边缘：这个功能可以使用 Ex2 中的 Canny 算法输出。

   程序中的函数为：

```
Canny::Canny(string filename)
```

   结果如下：



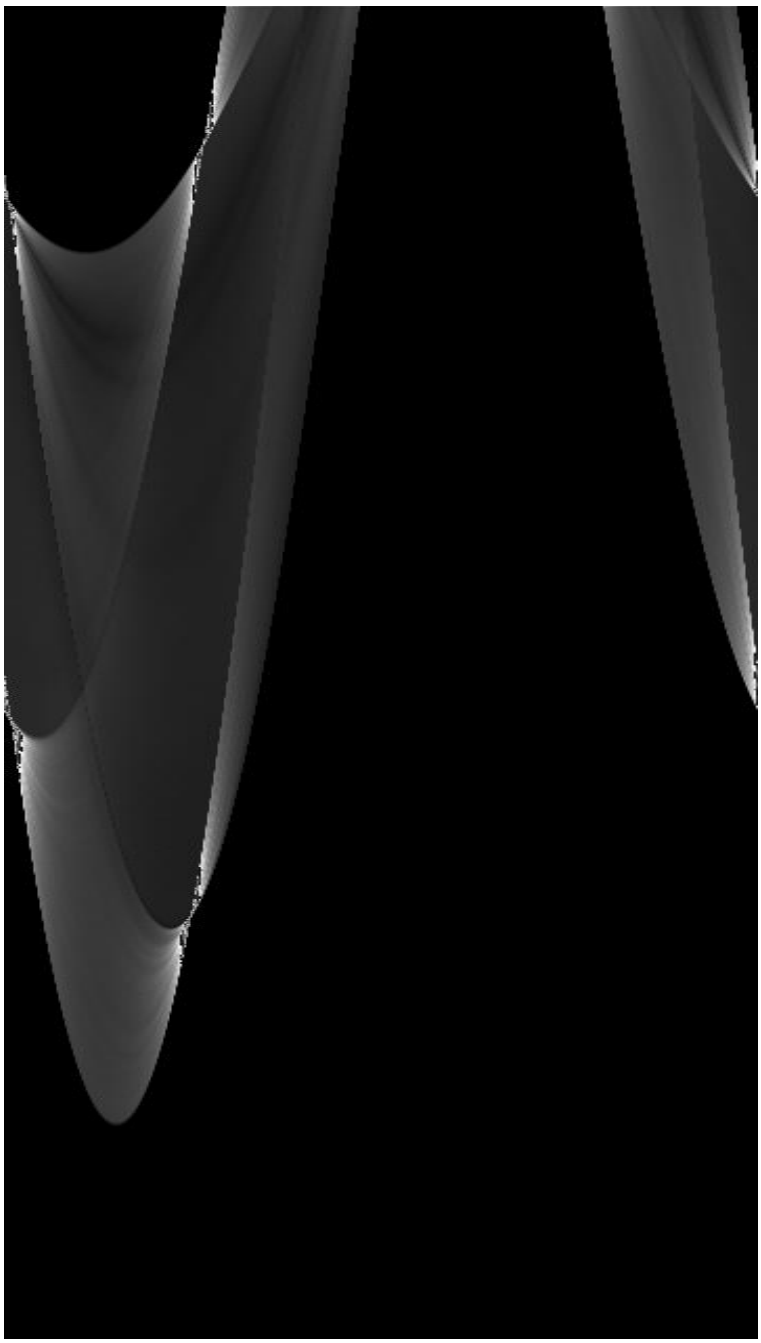
2. 输出图像边缘：

   步骤一中我们首先将图像转化为灰度图，然后使用高斯核对图像出律，再

进行 Sobel 算子对图像进行滤波.将上述得到的图片转换到极坐标（x-y 坐标系的斜率需要分情况讨论比较复杂，而且表示效果没有极坐标好）的霍夫空间中，程序中对应函数：

```
CImg<float> initHoughSpace();
```

得到的结果如下（为了更加直观，在 word 中对本图进行了拉伸）：



然后使用“投票”的方法，并且设置下阈值，找出霍夫空间中最“亮”的

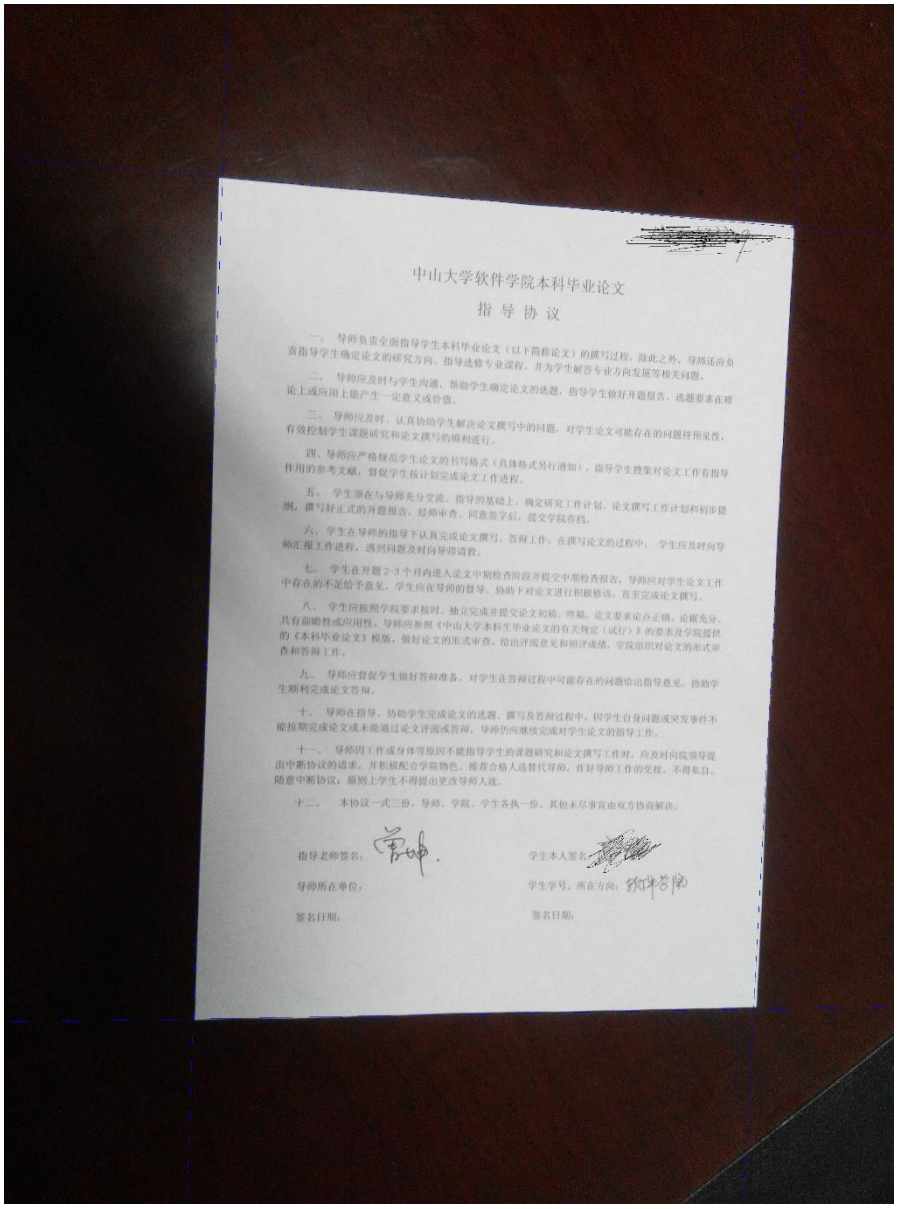
点（也即直线经过最多的点）。这里需要从一堆数据中找到前 topK 个数，并获取其在数组里的坐标。因为图片经过前面步骤的处理之后，最“亮”的点并不一定在边上。程序中对应的函数为：

```
void Hough::findPeaks()
```

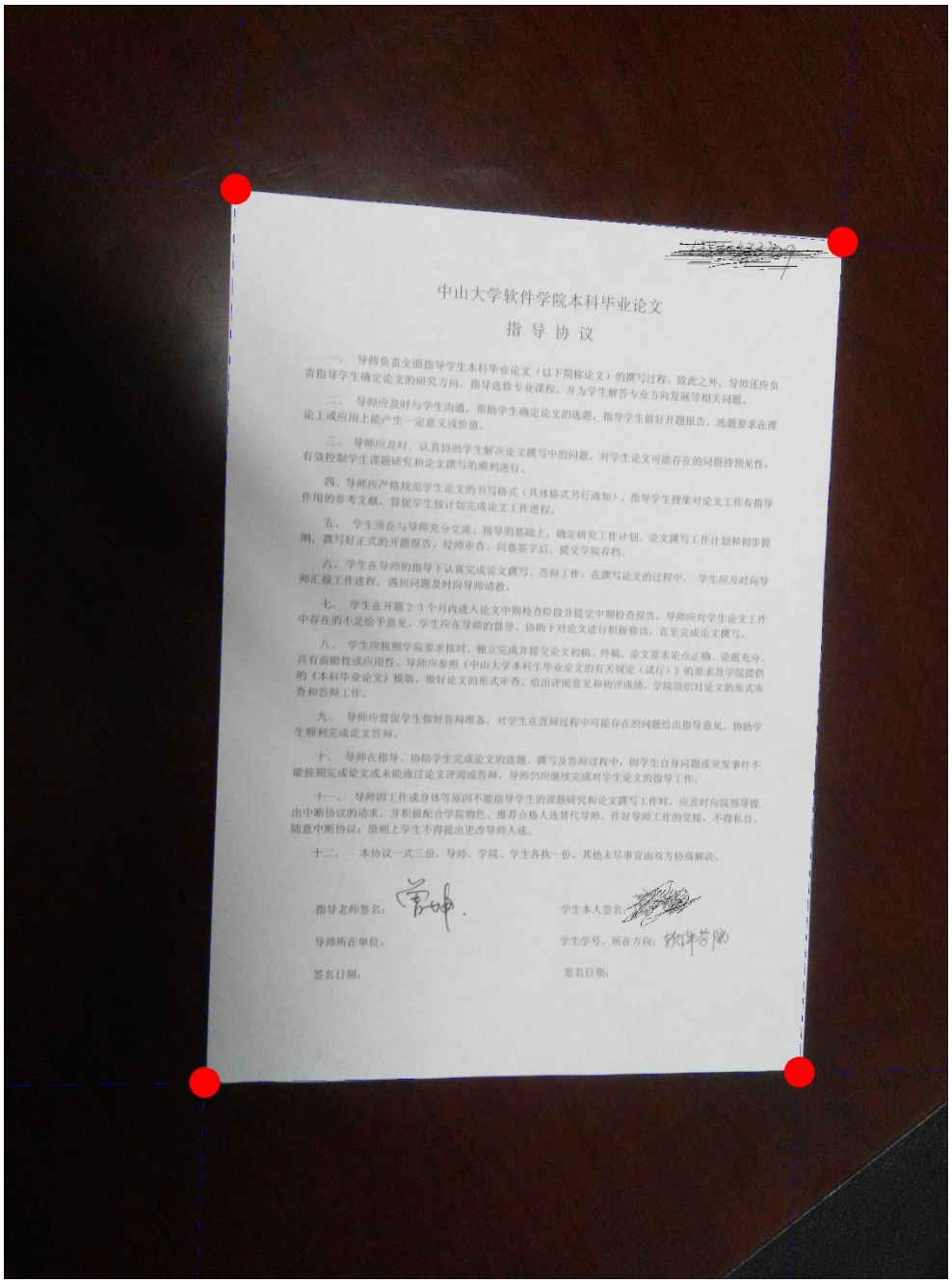
找出之后经过调参就可以大概求出 A4 纸的边缘以及对应的直线方程，以及不同直线之间的交点，最后在交点处画个圆，程序中对应的函数为：

```
void drawLines(); // 寻找并画出直线
void drawIntersections(); // 寻找并画出直线交点
```

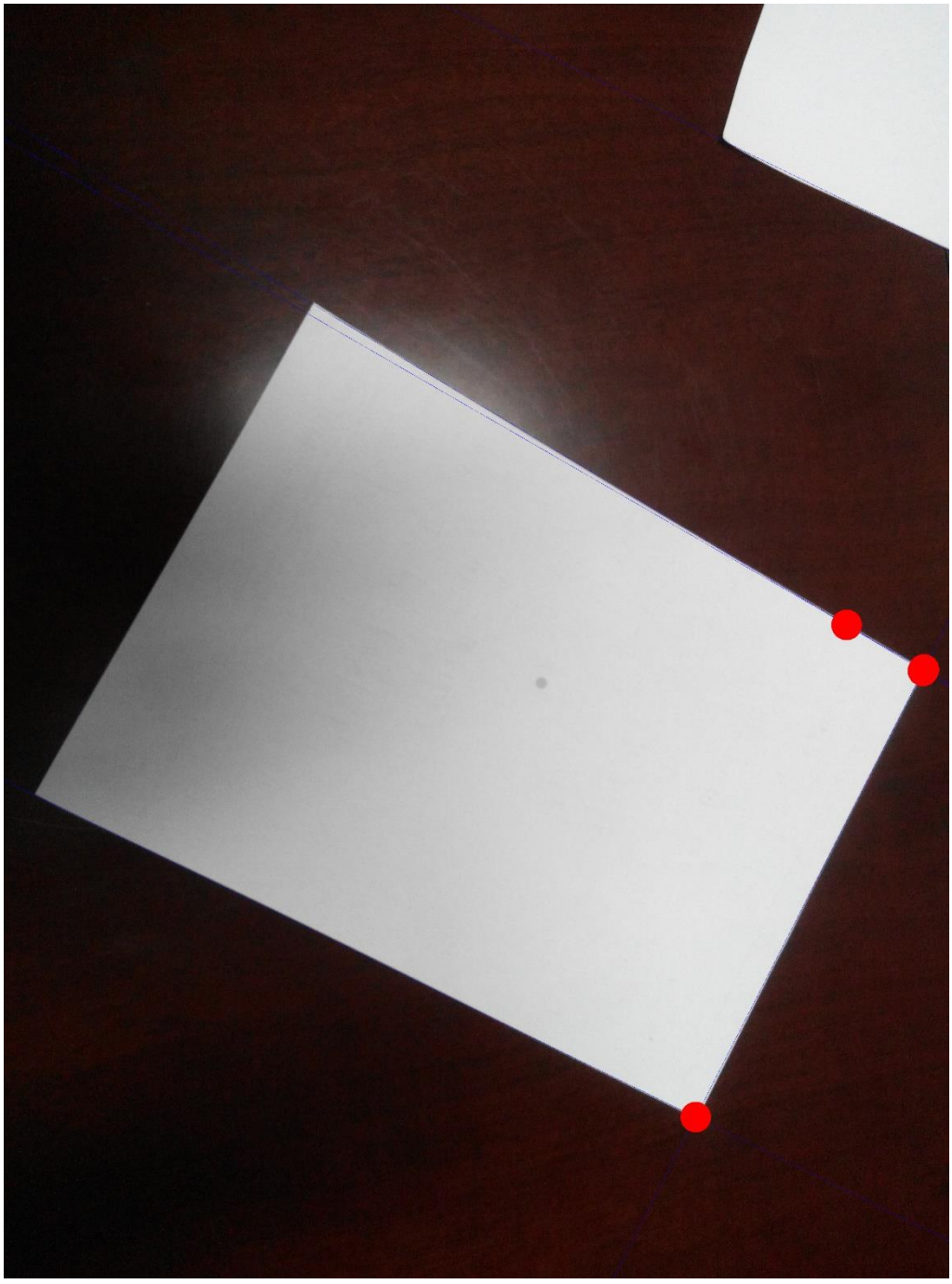
得到的结果如下：



```
yision@Marin:~/Code/hough$ ./main
Line 0: y = 0.0874887x + 541.059
Line 1: y = -28.6363x + 22550.5
Line 2: y = -19.0811x + 53672.5
Line 3: y = -0.0174551x + 3570.54
Intersection 0: x = 766.245, y = 608.097
Intersection 1: x = 2771.79, y = 783.559
Intersection 2: x = 663.198, y = 3558.97
Intersection 3: x = 2628.13, y = 3524.67
```



当然也有翻车的例子：



这个是因为左上角的整体部分都太暗了，导致在进行霍夫空间转换之前就已经丢失了很多信息，所以没有办法将其检测出来。如果在转换霍夫空间之前调一下参数，让那条边检测的准确一下，但会导致“噪声”增多，结果还是差不多。

(二) 圆形检测

$$x = x_0 + r\cos\theta$$

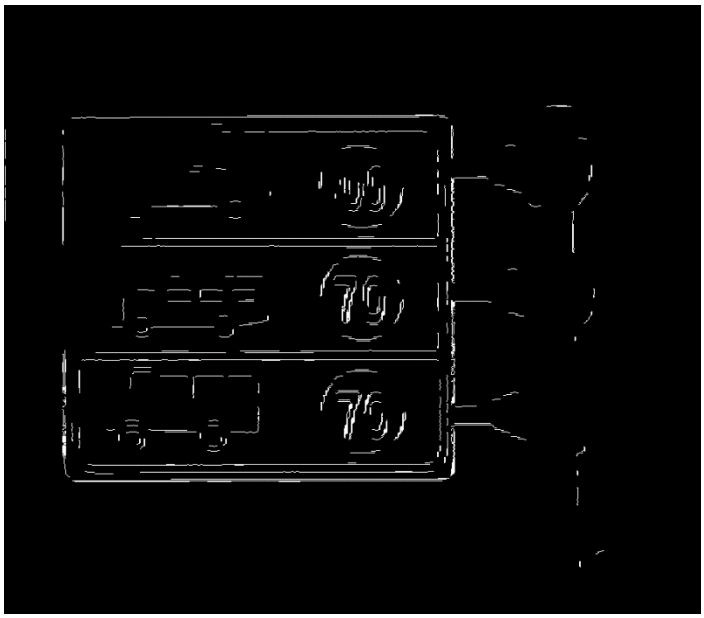
$$y = y_0 + r\sin\theta$$

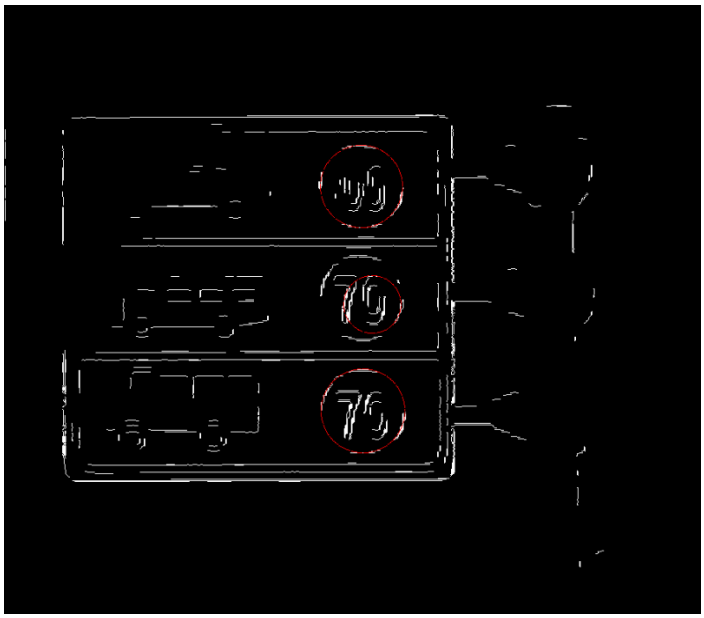
根据极坐标，圆上任意一点的坐标可以表示为如上形式。所以对于任意一个圆，假设中心像素点 p(x0, y0)像素点已知，圆半径已知，则旋转 360°由极坐标方程可以得到每个点上的坐标。同样，如果只是知道图像上像素点，圆半径，旋转 360°则中心点处的坐标值必定最强。这正是霍夫变换检测圆的数学原理。

对于圆来说，霍夫空间的参数就是圆心的坐标，然后选择不同的半径进行投票，找到每次 houghImage 里面的最大投票数，这个投票数表示当前 r 的吻合程度，然后用投票数最大的 r 作为最好的 r，票数超过阀值 rLimit 的就认为是一个圆的半径。合理的选择 r 的区间(minR, maxR)可以有效减少遍历的次数，加快处理速度。（其实这道题做出来主要还是靠慢慢调参得到的结果）

在确定了 r 之后，重新进行霍夫投票，将霍夫图像中投票数超过阀值的点对应圆心的坐标和值存入数组，这个值就代表了投票的结果，根据值进行排序，并判断检测出来的圆心坐标是否跟已检测的圆心坐标的距离，如果距离过小，默认是同个圆，最后输出结果。

直接上实验结果：





```
yision@Marin:~/Code/circle$ ./main
0
圆形个数: 3
```

也有翻车的例子:

把“排版”像圆形的其他东西也当成圆形算了进来…


40
定制：
交通标志牌
交通标志杆
聚荣网