EDGE DETECTOR

I use Canny Edge Detector on this assignment. The Canny Edge Detector was created by John F. Canny in 1986 as an edge detection operator. This detector detects a wide variety of edges in pictures using a multi-stage technique. The existence of a Gaussian filter allows any noise in a picture to be removed. The signal may be improved in terms of noise ratio using the non-maxima suppression approach, which produces one-pixel wide ridges as an output. The thresholding approach is used to detect the edges in a noisy condition. Parameters can be used to change the efficacy. It has good localization and responsiveness, and it is noise-resistant. I also set my parameters as min threshold equals 100 and max threshold equals 200 for the edge detector.

I will share the edge detector examples with the hough circle transform results below.

HOUGH TRANSFORM STEPS

Firstly we find input image size and specify R and Theta ranges.

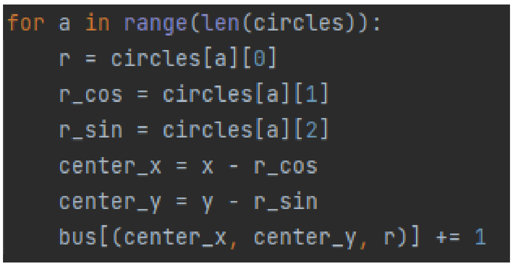
Evaluate and retain the candidate circles dx and dy for various delta radiuses on hand.

x = x_center + r * cos(t) and y = y_center + r * sin(t).

I use defaultdic instead of standard dict as this will initialize for key, I found it while researching.

Then we extend as pixels the image you can think as x and y coordinates height and width.

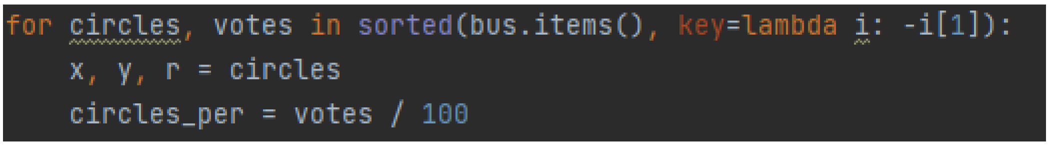
Then found an edge pixel to find and vote for circle from the candidate circles passing through this pixel.



```
for a in range(len(circles)):
    r = circles[a][0]
    r_cos = circles[a][1]
    r_sin = circles[a][2]
    center_x = x - r_cos
    center_y = y - r_sin
    bus[(center_x, center_y, r)] += 1
```

We take our circle candidate through specify x, y and r.

Sort the accumulator based on the votes for the candidate circles.

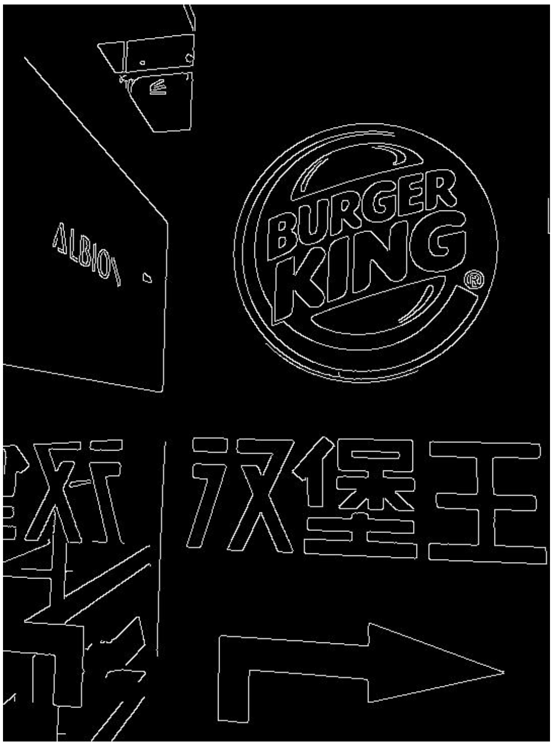


```
for circles, votes in sorted(bus.items(), key=lambda i: -i[1]):
    x, y, r = circles
    circles_per = votes / 100
```

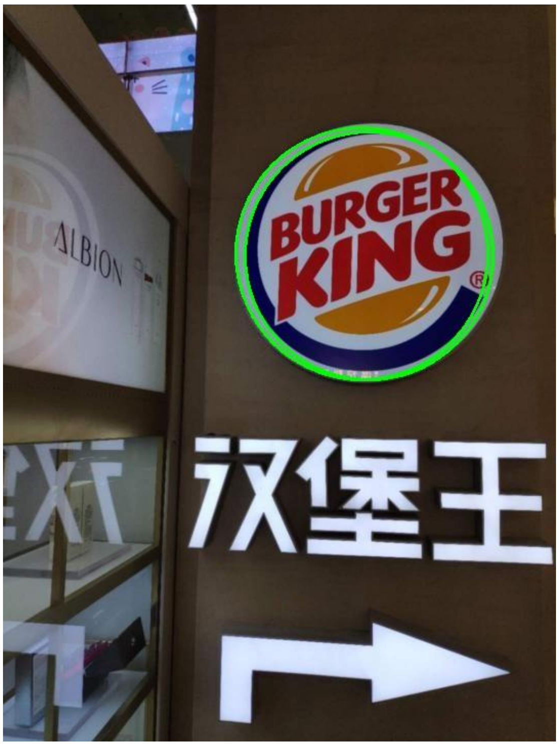
The findings will be post-processed. Circles that are too close together are excluded. Based on pixel threshold, remove duplicate circles in the vicinity. On the result image, draw shortlisted circles.

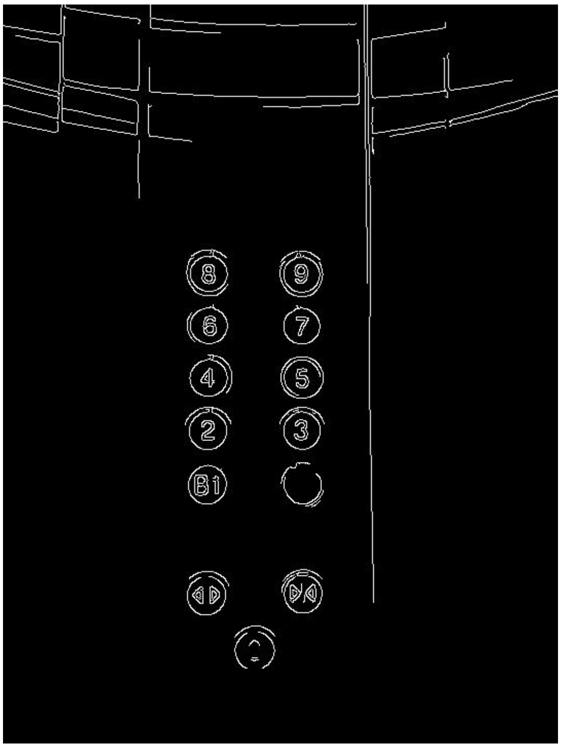
EDGE DETECTOR AND HOUGH CIRCLE TRANSFORM EXAMPLES

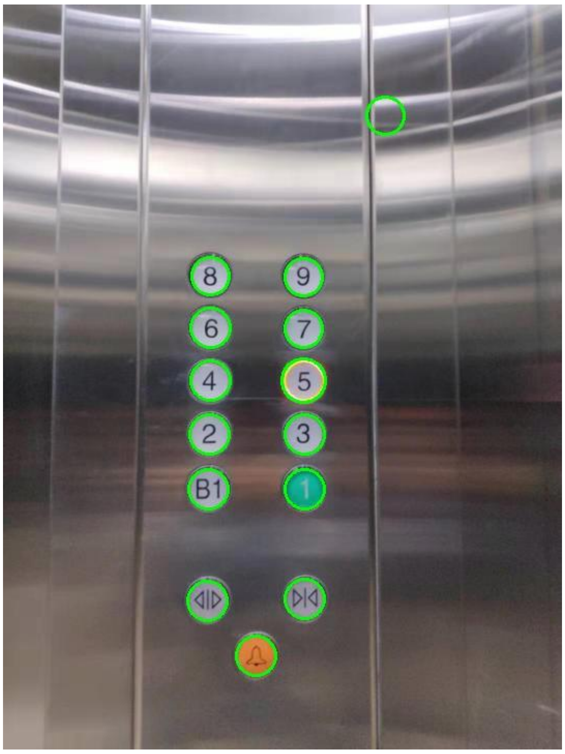
CANNY EDGE DETECTOR

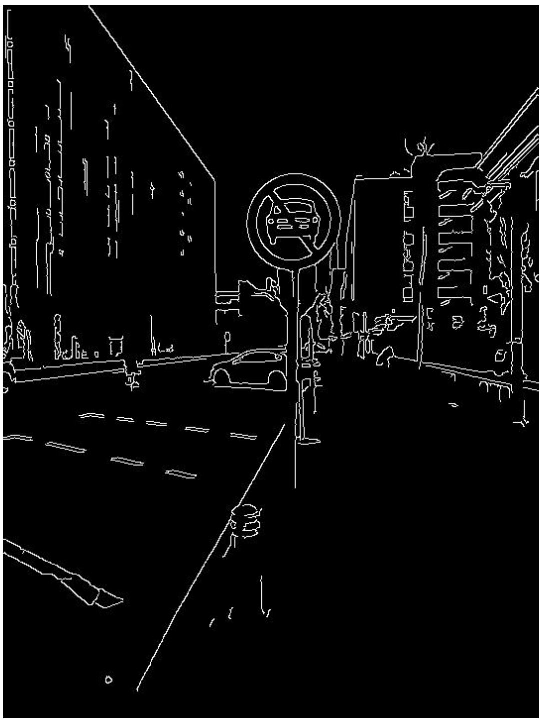


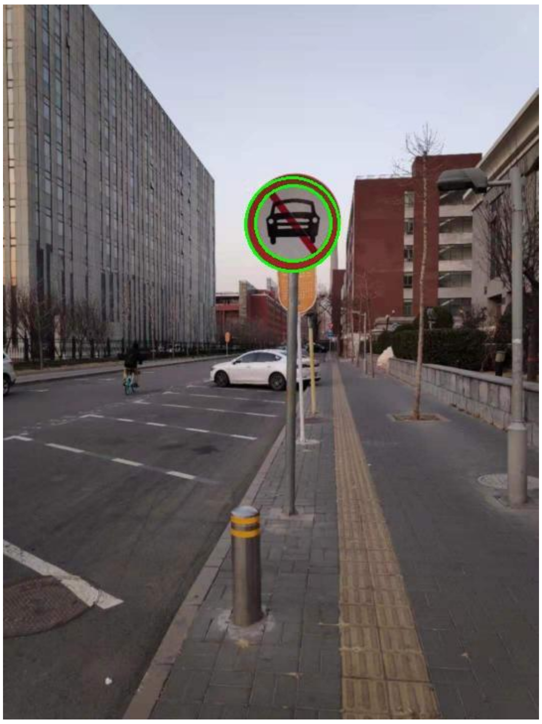
HOUGH CIRCLE TRANSFORM

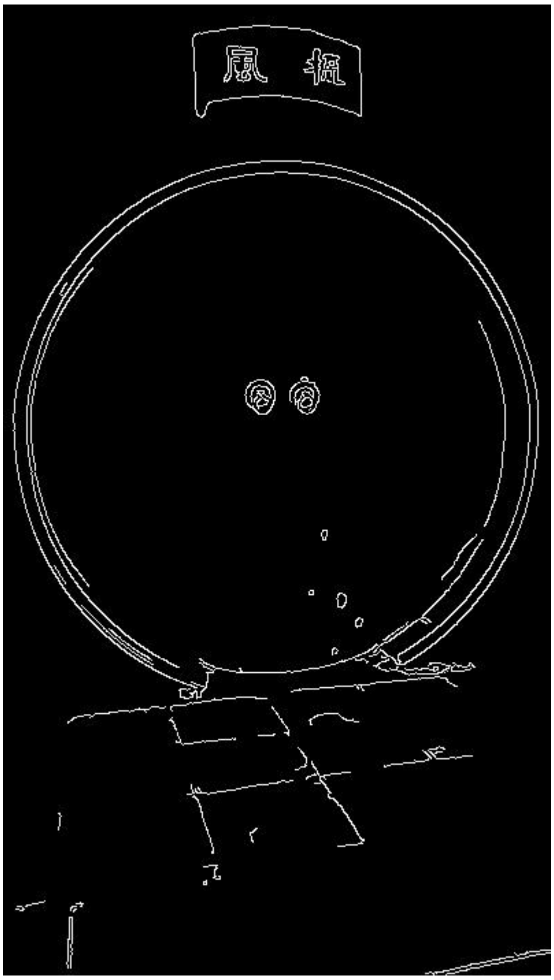



8
9
6
7
4
5
2
3
B1


8
9
6
7
4
5
2
3
B1
1

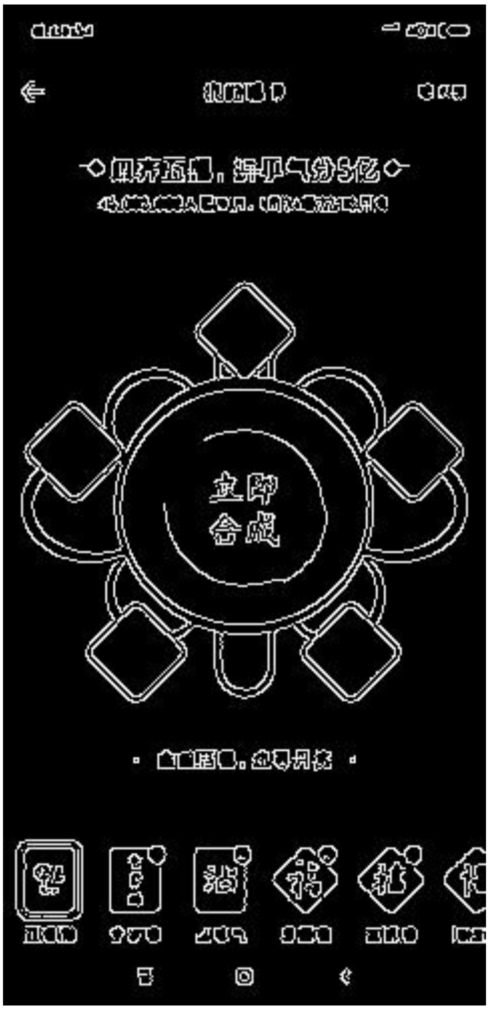


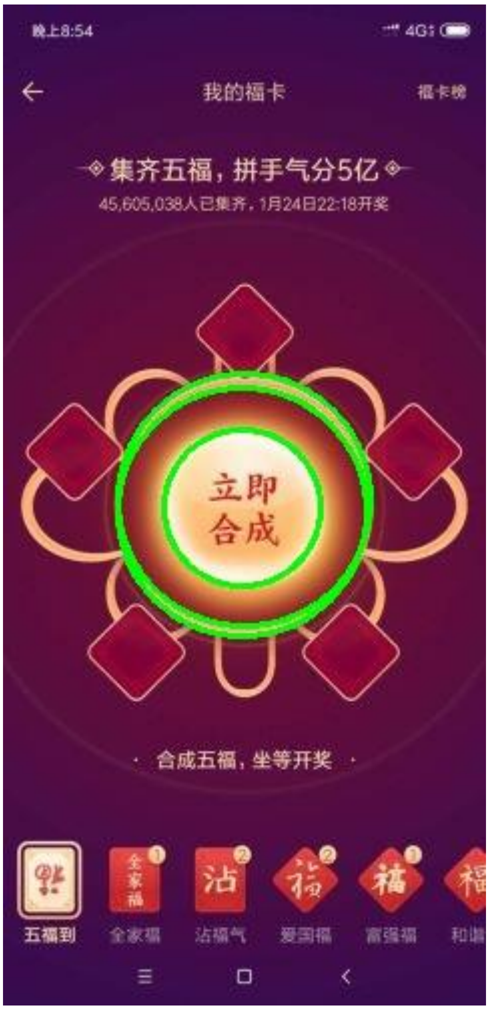


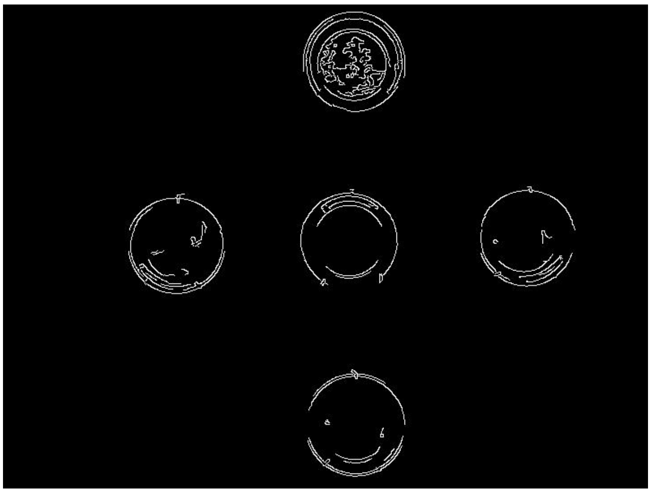


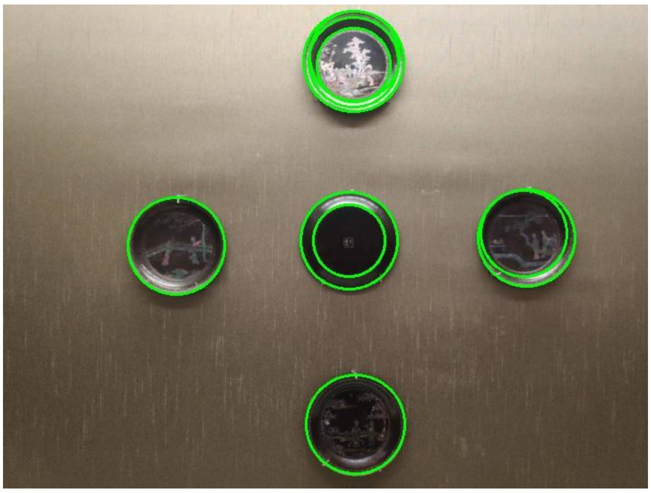
風 流

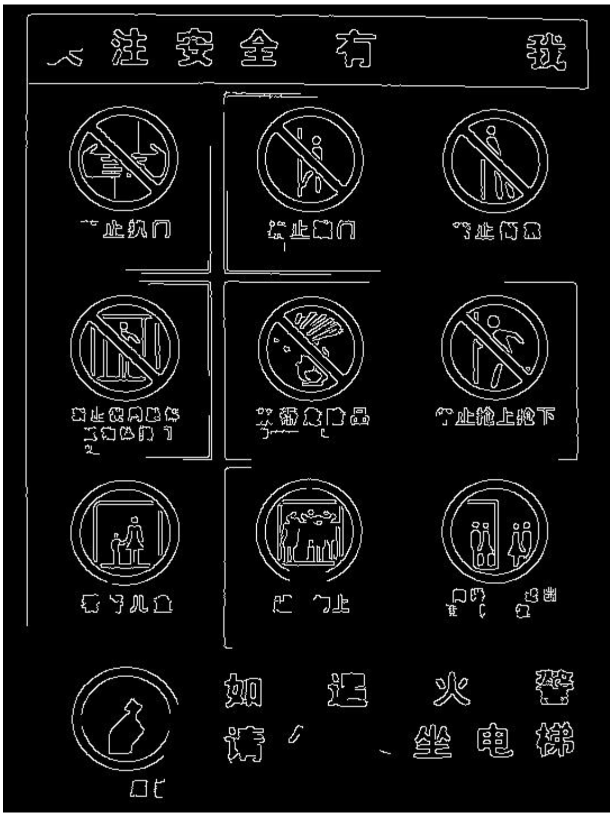
風 流

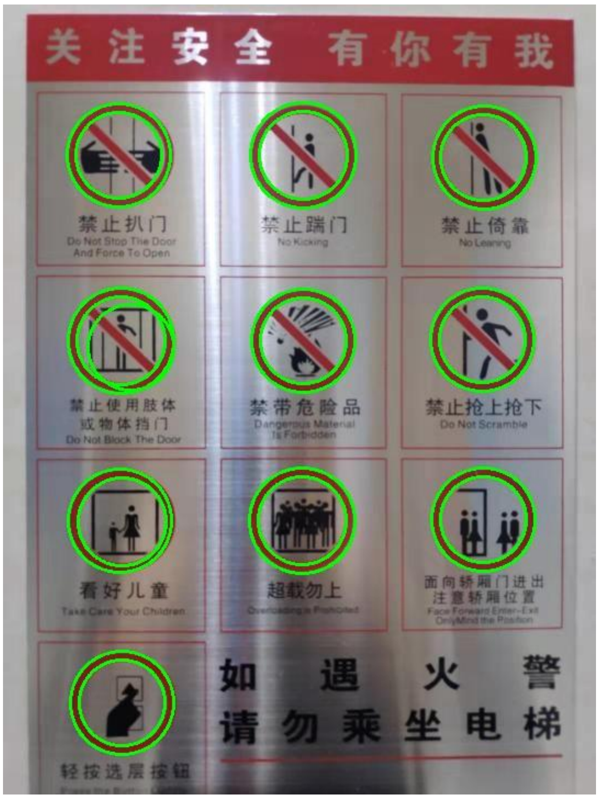


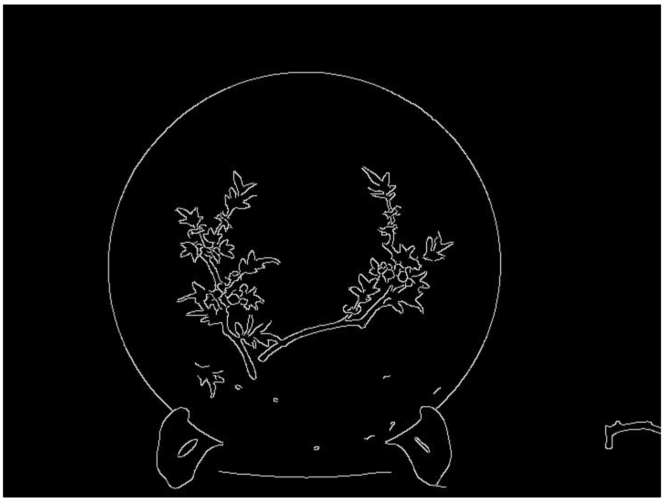

晚上8:54
4G
我的福卡
福卡榜
集齐五福，拼手气分5亿
45,605,038人已集齐，1月24日22:18开奖
立即
合成
合成五福，坐等开奖
全家福
沾
福
福
五福到
全家福
沾福气
爱国福
富强福
和谐

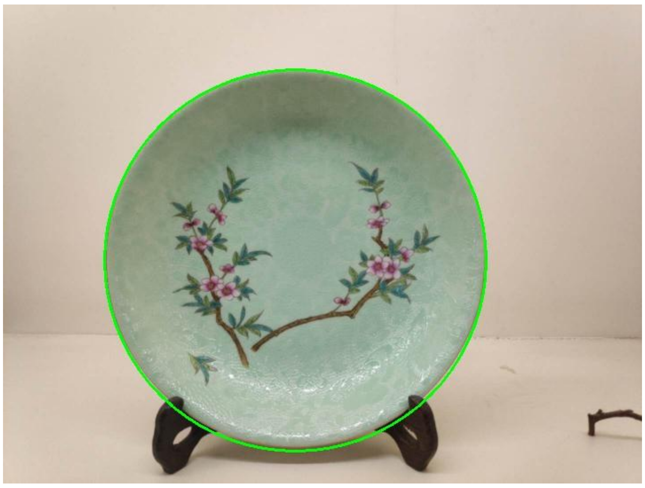


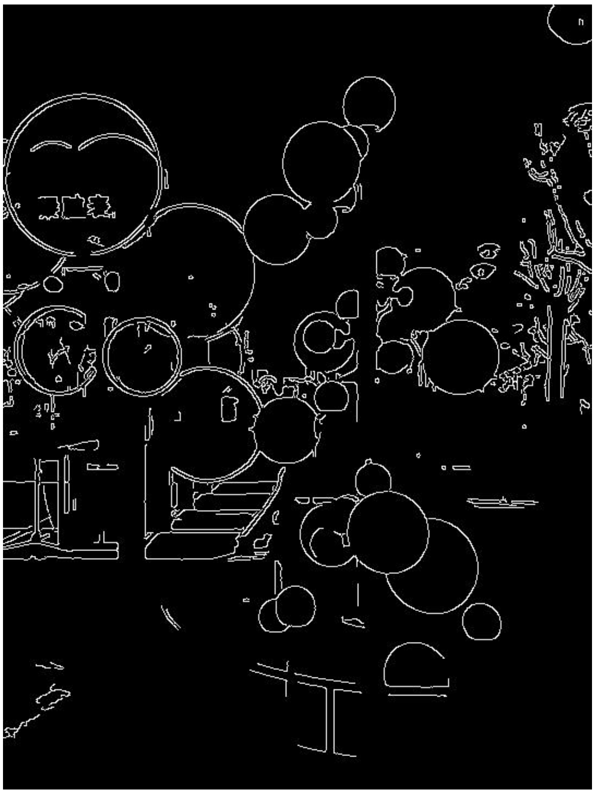


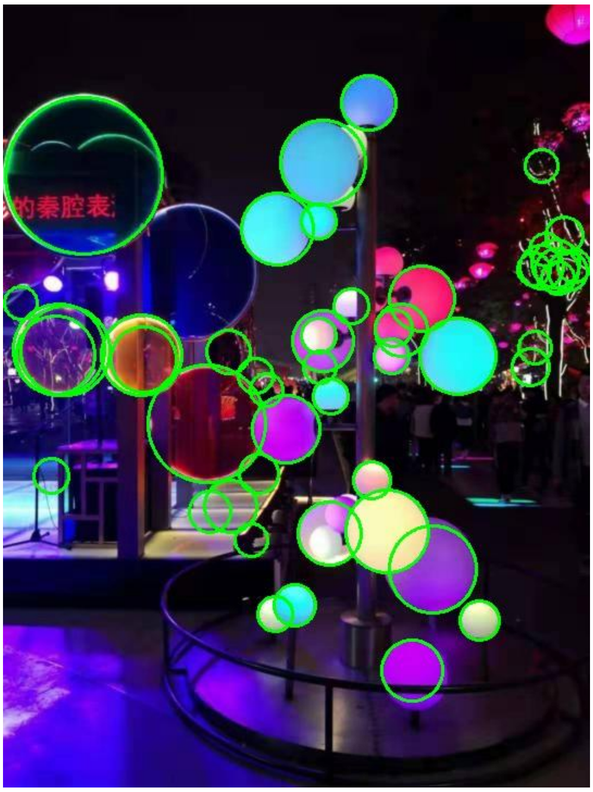



关注安全 有你有我
禁止扒门
Do Not Stop The Door And Force To Open
禁止踹门
No Kicking
禁止倚靠
No Leaning
禁止使用肢体或物体挡门
Do Not Block The Door
禁带危险品
Dangerous Material Is Forbidden
禁止抢上抢下
Do Not Scramble
看好儿童
Take Care Your Children
超载勿上
Overloading is Prohibited
面向轿厢门进出 注意轿厢位置
Face Forward Enter-Exit Only Mind the Position
轻按选层按钮
如遇火警 请勿乘坐电梯

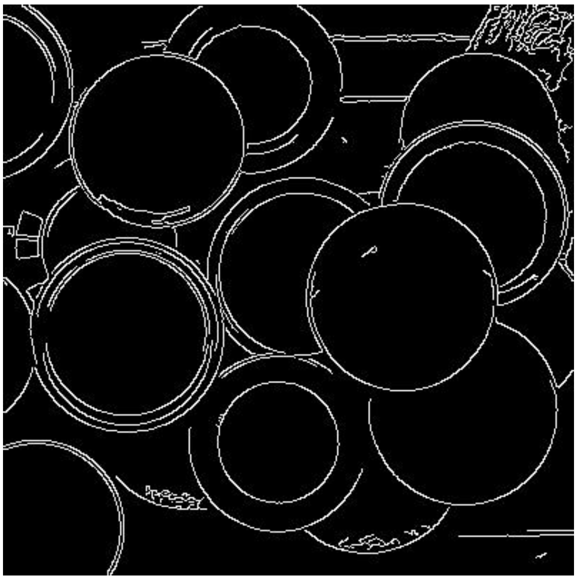


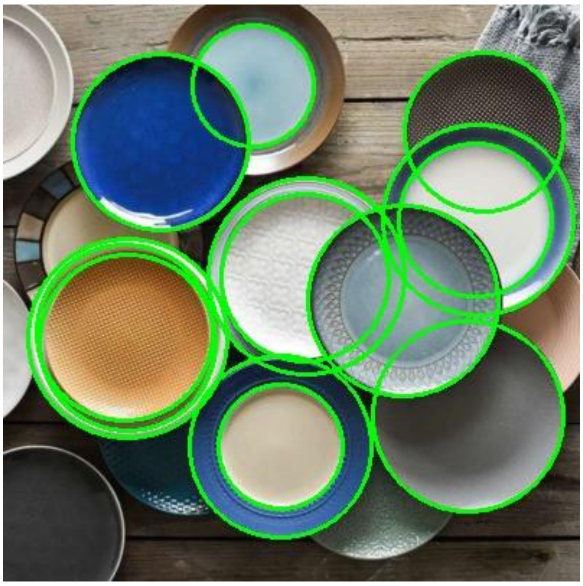




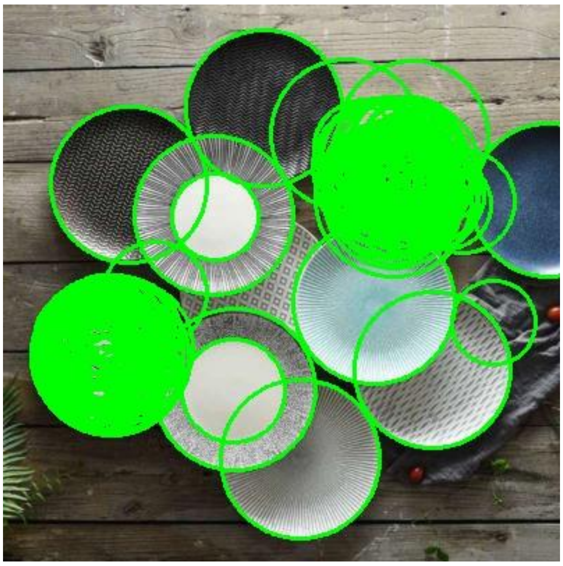


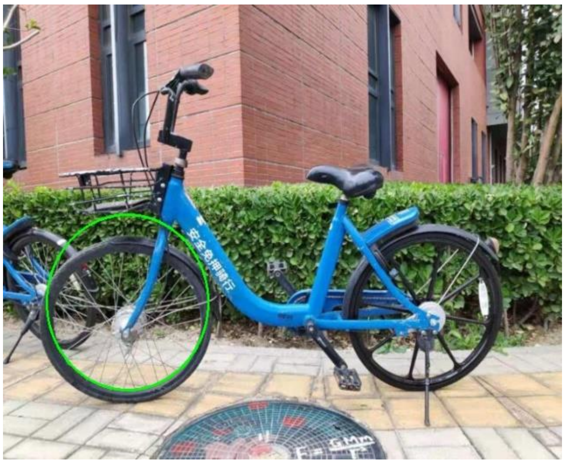
的秦腔表

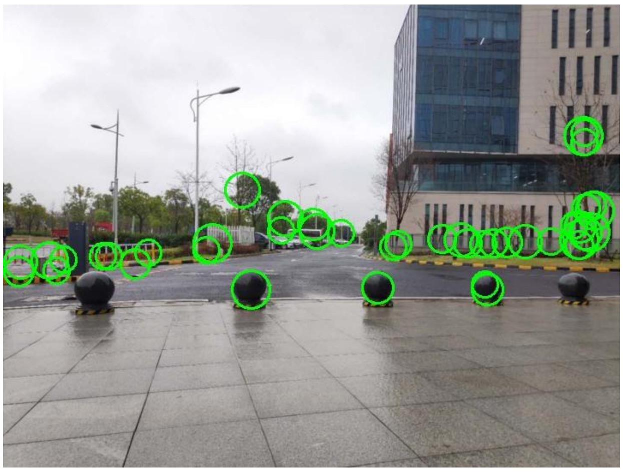


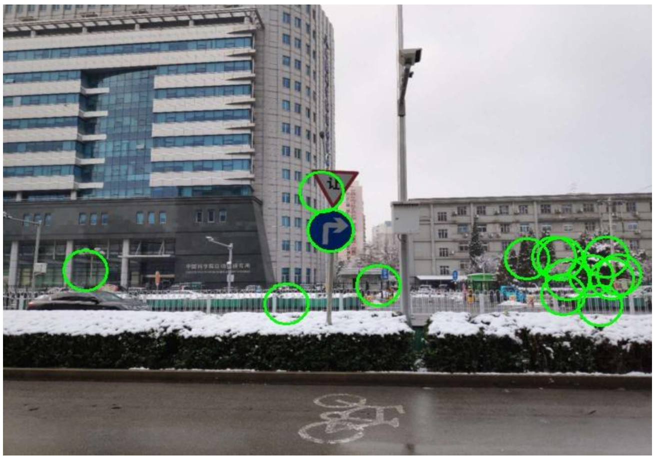


THE IMAGE EXAMPLES THAT TRANSFORM FAILED ON









Non-target shapes can produce spurious peaks in parameter space.

It can give misleading results when objects happen to be aligned by chance.

Detected lines are infinite lines described by their (m,c) values, rather than finite lines with defined end points.

The large amount of circles generated by Hough Circle Transform is caused by the low value of the threshold for center detection.

So, the minimum and maximum radius values of the failed images you see above may not be specified enough. In addition, threshold values for each photo are also important. Threshold values also need to be specified.

RESOURCES

1 http://webfiles.portal.chalmers.se/s2/undergraduate/SSY095/PDFdocuments/ForScreen/Notes/HoughTransform.pdf

2- https://web.cs.hacettepe.edu.tr/~erkut/bbm413.f17/slides/09-edges-4pp.pdf

3- https://en.wikipedia.org/wiki/Circle_Hough_Transform