

HOUGH CIRCLE TRANSFORM

Sheela Ahmed

WHAT IS HOUGH CIRCLE TRANSFORM?

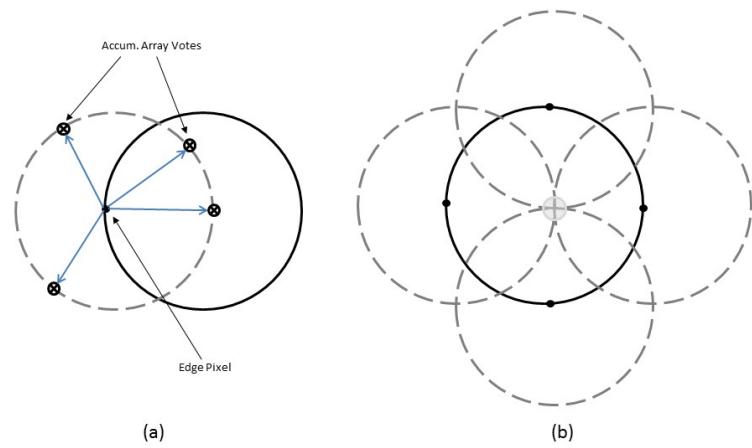
Detect circles in an image

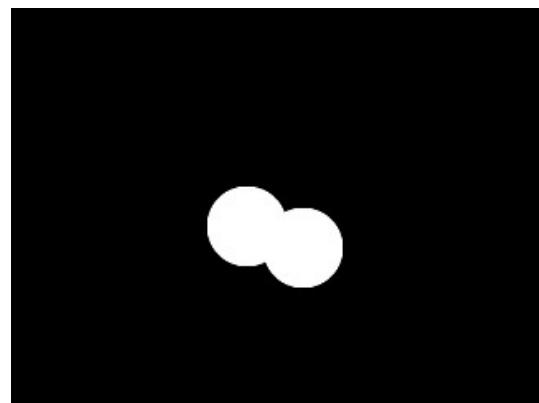
If a circle can be defined as $(x - a)^2 + (y - b)^2 = r^2$,

Then an arbitrary edge point (x_i, y_i) can be transformed into a right circular cone in the (a, b, r) parameter space.

If all image points lie on a circle, then the cones will intersect at a single point in the (a, b, r) corresponding to the parameters of the circle

Paper: “Circular Hough Transform” by Simon Pederson

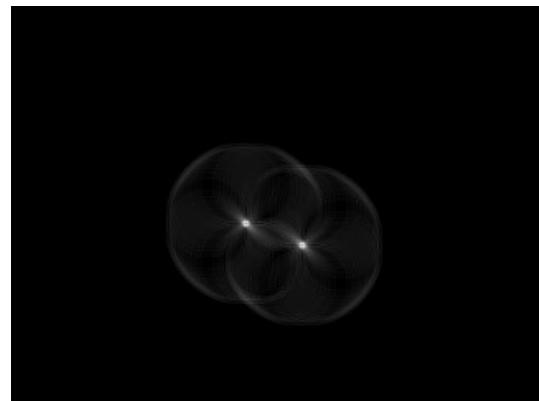




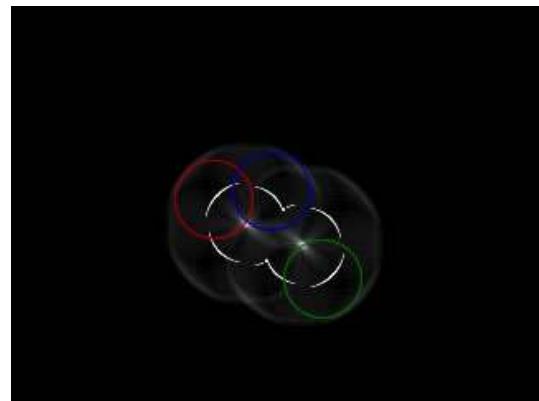
<http://www.aishack.in/tutorials/circle-hough-transform/>



<http://www.aishack.in/tutorials/circle-hough-transform/>



<http://www.aishack.in/tutorials/circle-hough-transform/>



<http://www.aishack.in/tutorials/circle-hough-transform/>

GENERAL STEPS FOR ALGORITHM

Find edges in the image

Create circles around each edge pixel

Create an accumulator list that counts when a certain pixel matches two circles

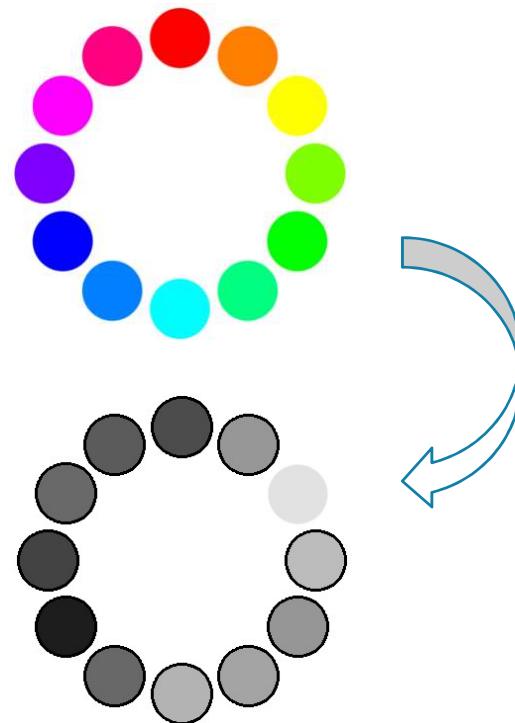
Maximum count value can be inferred as the center of a circle in the image

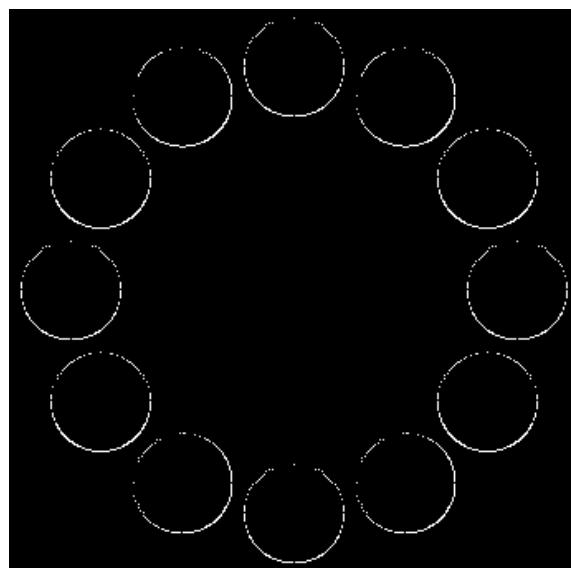
Need to determine which radius value to use to test

Create a range of radii to go through

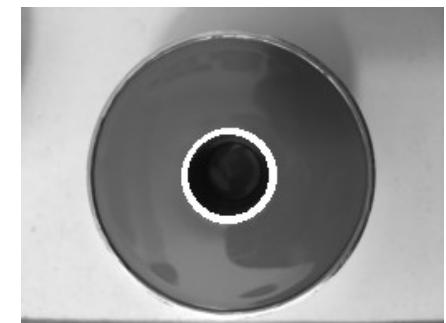
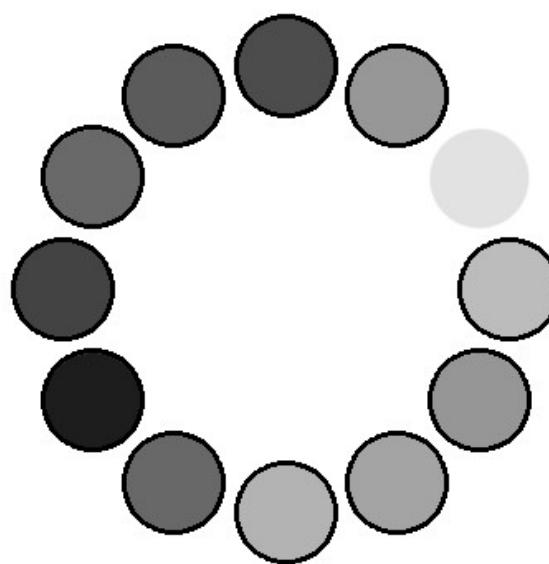
FIRST ITERATION OF CODE TOOK ~11 MINUTES...

```
sma4652@aldrin: ~/src/python/modules/1P  
sma4652@aldrin:~/src/python/modules/ipcv $  
Max acc value 87  
Max acc value 82  
Max acc value 75  
Max acc value 69  
Max acc value 66  
Max acc value 68  
Max acc value 61  
Max acc value 67  
Max acc value 60  
Max acc value 63  
Max acc value 64  
Max acc value 67  
Max acc value 67  
Max acc value 72  
Max acc value 75  
Max acc value 80  
Max acc value 82  
Max acc value 101  
Max acc value 106  
Max acc value 145
```





Synthetic image



Real world image

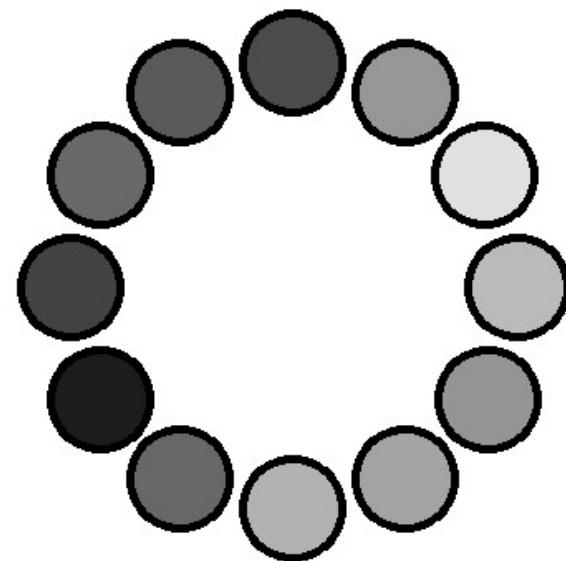
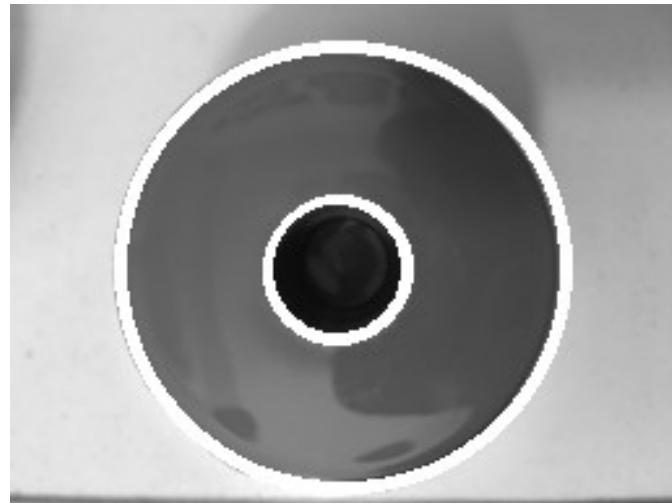
```
1 import cv2
2 import numpy as np
3
4 img = cv2.imread('opencv_logo.png',0)
5 img = cv2.medianBlur(img,5)
6 cimg = cv2.cvtColor(img,cv2.COLOR_GRAY2BGR)
7
8 circles = cv2.HoughCircles(img,cv2.HOUGH_GRADIENT,1,20,
9                             param1=50,param2=30,minRadius=0,maxRadius=0)
10
11 circles = np.uint16(np.around(circles))
12 for i in circles[0,:]:
13     # draw the outer circle
14     cv2.circle(cimg,(i[0],i[1]),i[2],(0,255,0),2)
15     # draw the center of the circle
16     cv2.circle(cimg,(i[0],i[1]),2,(0,0,255),3)
17
18 cv2.imshow('detected circles',cimg)
19 cv2.waitKey(0)
20 cv2.destroyAllWindows()
```



THERE'S AN OPENCV FUNCTION

`cv2.HoughCircles(image, method,
circles, minDist, param1,
param2, minRadius, maxRadius)`

USING OPENCV HOUGH TRANSFORM



REFERENCES

<http://www.aishack.in/tutorials/circle-hough-transform/>

https://www.mathworks.com/help/images/ref/imfc_accumarray.png

Pedersen, Simon Just Kjeldgaard. "Circular hough transform." *Aalborg University, Vision, Graphics, and Interactive Systems* 123 (2007): 123.