**ALGORITHM**

1) Reading the image to be analyzed using cv2.imread().

2) Converting the image to gray and resize the image in order to make analysis on an image of fixed size.

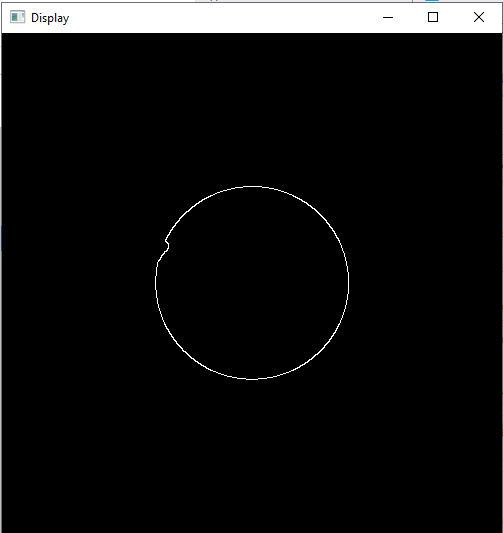
3) Perform Canny Edge detection using cv2.Canny function (). Now we will have only the edges.

|  |  |  |
| --- | --- | --- |
| Flash | Cut | Good |

***IDEA:*** *Here we can see in case of Good Images there are two perfect circles. But in case of defects at the place of circle we have imperfections. Our aim is to find these pixels and then identify the type of defect.*

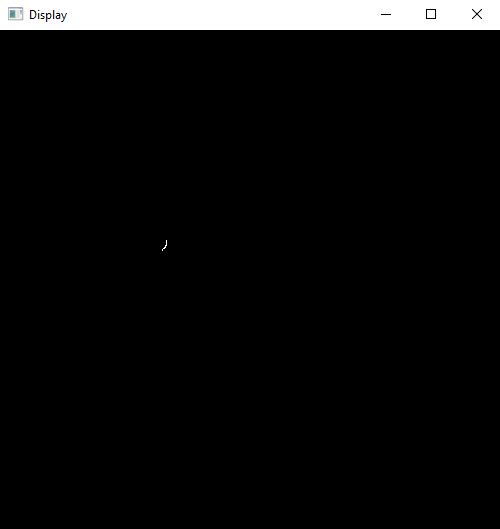
4) We can detect the circles using Hough Transform but the challenge is we cannot detect concentric circles. So, using cv2.HoughCircles we first detect the outer circle alone (parameters are adjusted to detect outermost circle).

5) Once outer circle is detected we can store the value of center and radius. Also, we can mask the outer circle with a thickness in the image by drawing a circle with obtained center and radius.

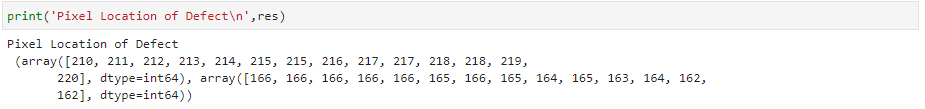


6) Now we can again use Hough circles to detect the inner circle.

7) Now if we mask the inner circle also, we will be able to see the pixels in areas of defect alone.



8) **LOCALIZING** We can now identify the pixel location of the defects



x and y coordinates of defect location

9) Flash Defect – Defect pixel location distance from center will be less than the radius of inner circle or Defect location distance from center will be greater than the radius of outer circle

Cut Defect – Defect pixel location distance from center will be greater than inner circle radius and less than outer circle radius

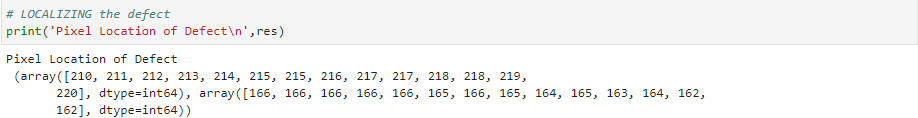
Using this technique for each point we identify the defect type.

*Note: In the given five images all the points were identified correctly based on defect*

**In order to generalize we check if there are minimum two points matching the defect type. If there is less than two points detected as defect then we classify it as good. This ensures that we do not miss indemnifying any defective sample**

**OUTPUTS**

Image – Defect1.png

Pixel Location: 

Flash Defect correctly identified

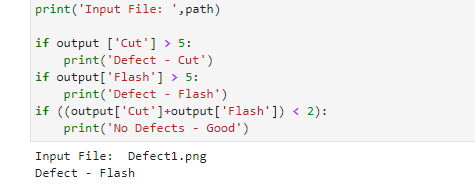
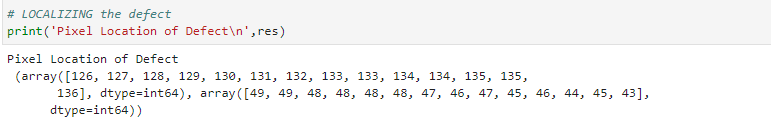


Image – Defect2.png

Pixel Location: 

Cut Defect correctly identified:

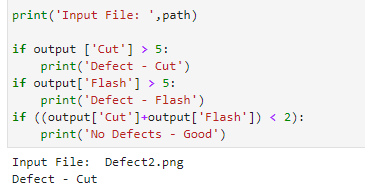
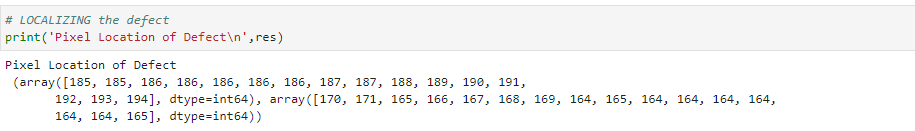


Image – Defect3.png

Pixel Location: 

Cut Defect correctly identified:

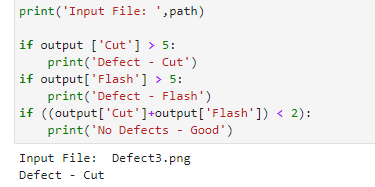
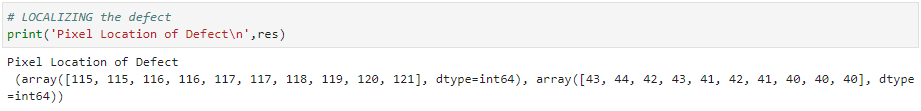


Image – Defect4.png

Pixel Location: 

Cut Defect correctly identified:

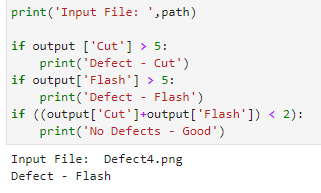
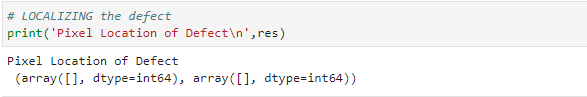
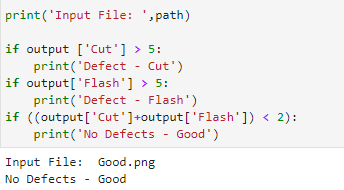


Image – Good.png

No pixels detected as defective: 



**Conclusion:**

We were able to correctly identify the type of defect with making any changes to the algorithm.