

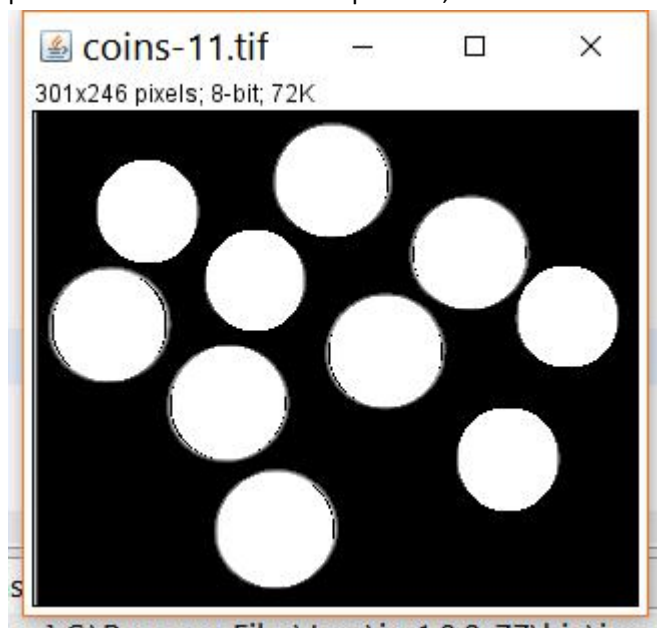
Report of homework3 in ECE545

Yu Li

PS: The image processing might be slow. Please wait until the Finish message.

1.Circular Hough

In Circular_Hough, firstly I copy the original image and reduce noise by implementing a Gaussian filter to the copied image, then I use findEdges function to find all the edges in this image and make the image binary. By traversing all the edges, I use the characteristic of a circle to detect the potential central points. If the array used to count number of intersect points of each circle is large than 9 and the radius of the circle is between 25 and 33, it is considered to be a true central points. To benefit the second question, I make all the round white. The result is shown like this.



2.Circle Flood

In Circle_Flood, there is a boolean to choose the naive way or the coherence way to label the image. In order to distinguish each circle, I make the pixel value $p = 255 - 20 * m$ where m is the label rather than $p = m + 1$. The result is like this.



3.Edge Detection

(1)

In Gradient_Magnitude, I make the image wrapped in the leftmost/rightmost line. Then compute the gradient value of x direction of processed image, then wrap the upmost and downmost line and compute the gradient value of y direction. The gradient of the image is the $\text{det} = (\text{detx}^2 + \text{dety}^2)^{0.5}$. Where detx and dety are the gradient values of processed copied image pixel value in corresponding position. Then I set the det to the original image. The result is shown like this



(2)

In Laplacian_Image ,use the direct of wrapping in all four edge lines and four corners. Then implement Laplacian filter to the wrapped image, the result is like this.



(3)In Zero_Crossing, same method of wrapping, and detect zero crossing by implementing second derivative of image. The result is shown like



(4)Combined them all together, first overlay the result of Laplacian image and zerocrossing image, then set threshold of gradient magnitude to the new image. The result is like this.



4.Pupil measure

In this problem I use the specific image Eyes1.JGP and convert it to 8bit gray image in the image>type>8-bit.Then I set proper threshold in method Hough_Circle.The result is like this.

