horizontal line

Part II: Coins Detection Using Hough Transform

**3rd May 2021**

| **Name** | **ID** |
| --- | --- |
| **Jamal Eldin Ahmed Khalaf** | **15** |
| **Yomna Gamal El-Din Mahmoud** | **63** |

# OVERVIEW

The Hough transform can be used to determine the parameters of a circle when a number of points that fall on the perimeter are known. A circle with radius R and center (a, b) can be described with the following equations: (x 􀀀 a)2 + (x 􀀀 b)2 = R2 . Where the locus of (a, b) points in the parameter space fall on a circle of radius R centered at (x, y). The true center point will be common to all parameter circles, and can be found with a Hough accumulation array.

# GOALS

The objective of this part of the assignment is the detection of circular coins in an image using Hough Transform.

# DATASET

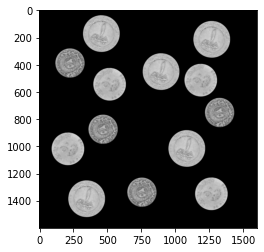
It is called Egyptian Coins. It has been preprocessed and the noise has been removed and we can assume the following.

* Images include the following coins only: one Egyptian pound (135 pixels radius), 50 piastres (120 pixels radius) and 25 piastres (107 pixels radius).
* There are at most 20 coins per image.
* Each image has at least one coin of each type.
* For simplicity, all the coins of the same type have the same radius.

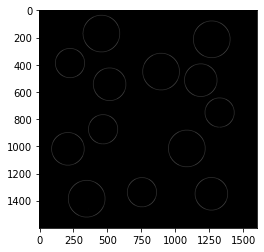
# IMPLEMENTATION DETAILS

## Smoothing the image

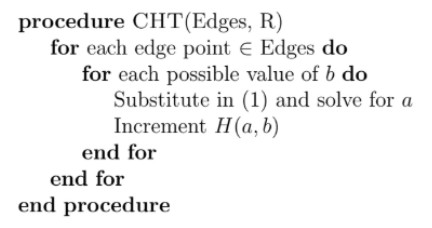
First we read the image then make 2 copies the first 1 is an RGB image and the second one is gray scale image. A median filter is applied on the gray scale image with kernel size = 7 to remove the noise in the image.

The result after applying the filter:  


## Edge Detection

After smoothing the image, Canny's algorithm has to be used for edge detection. We used relatively high values for thresholding to remove most of the noise.  


## Accumulation into (a, b)-space using circular Hough transform



**NOTE:**

* The possible values for b are the number of rows in the image.
* Solving for a means solve the circle equation for a
* H(a,b) is the matrix that contains the centers of possible circles.

## Rening Coordinates and CHT Post-Processing

After obtaining the possible centers we need to find the local maxima in the matrix of houghman.

Our approach is just to threshold the resulting houghman image by mean value 127 and max value 255. Then take the pixels that are not equal to zero and take a point that has a distance lower than 50 from its neighbors. If the distance is bigger than 50 that means we entered another area and we have to take a point from the new area and so on.

Thresholding the image  
Find the index of pixels that are bigger than zero (centers)  
Let previous → centers[0]  
Let peeks → empty list  
For each c in centers:  
 If distance(previous, c) > 50:  
 peeks.append(c)  
 pervious → c  
Return peeks

# Sample Tests

