Hough Transform for Line and Circle detection

Daniel Amirtharaj
damirtha@buffalo.edu

1 Objective

To detect lines and circles (bonus task) in the given image using the Hough transform.

2 Analysis

Some of the concepts that were used to solve this task are covered here.

2.1 Edge Detection

Edges are any discontinuity in the intensity of an image. They are detected by the use of 2 kinds of operators, the first order derivative or the gradient operator (sobel or prewitt) or the second order derivative or the laplacian operator. Derivatives tend to be noisier than the input image and hence these operators incorporate some element of smoothing to combat this issue.

2.2 Hough Transform

The Hough transform is a useful algorithm to find lines, circles and other shapes that can be represented mathematically. It is thus very useful in image analysis, where the properties of an image's constituent objects can be understood.

Hough line transform Hough transform works by mapping each point in the image plane to a sinusoid in the hough space. The intersection of multiple sinusoids in the Hough space soorespond to a line in the image plane.

The equation of a line in image plane is taken to be r = xcosθ + ysinθ. Here, r is a weighted sum of cosθ and sinθ and since cosθ + sinθ results in a periodic oscillation (since time period does not change), this results in a sinusoid as well (A sinusoid is defined as a periodic oscillation with a smooth curve). It can also be observed that as θ is defined to range from -90 to 90 degrees and image points have only positive locations, this results in a positive weighted sum of cosθ and sinθ for each θ, which gives rise to a sinusoidal wave in the Hough space for the span of θ.

Since each image point (x,y) changes the relative weights of the sine and cosine terms, their amplitudes and phases change due to the relative weights of x and y applied. This is shown below. The amplitude of the sinusoid is given by r = $\sqrt{(\mathrm{x}^2 + y^2)}$and phase = $\arctan(y/x)$. Intersection of different sinusoids imply a constant r,θ. On applying this to the line equation described above, it can be found that it forms a line in the image plane.

The following steps illustrate the steps in algorithm.

1. Edges in the image are detected and thresholded to help the algorithm perform better.
2. An accumulator for different values of θ and r is laid out and initialized to zero.

3. For each point in the image, r and θ values which satisfy the line equation are recorded and the accumulator is incremented for that value of r and θ.

4. Accumulator cells with larger entries correspond to good lines. A threshold to get good lines is chosen and applied on the accumulator to get r and θ values for such cells.

5. The r and θ obtained in the previous step correspond to the lines detected and can be used to generate lines in the image plane using the line equation for analysis.

Hough transform for circles Hough transform is applied to circles pretty much the same was as lines except the equations defining the circle will be different and thus the dimensions of the hough space. The equation of a circle is taken to be x = a + rcosθ, y = b + rsinθ for θ between 0 and 360 degrees. A circle in the image plane will map to a point in the Hough space, where the hough space is defined by the radius and the coordinates of the centre of a circle (r, a, b). Other than the shape of the accumulator and the equation of the circle the algorithm is nearly the same.

3 Method

The following steps were applied to detect lines in the image.

1. Edge detection was performed on the image using a laplacian operator. (Same operator used for point detection). Good edges were separated with a threshold, and a binary image was populated.

2. On the binary image, Hough transform was applied using the algorithm described previously and cells with at least an entry of 120 hits were selected as the lines detected.

3. The lines detected in the form of r, θ were sampled and added to slightly larger bins to avoid classifying the same line as different lines (one line in x,y was detected as part of multiple r,$theta$ due to various approximations and rounding operations performed in the algorithm).

3. These lines were then differentiated based on their θ values, whether they were vertical lines or diagonal lines and converted to the x,y coordinates to display the detected vertical and diagonal lines separately.

The following steps were applied to detect circles in the image.

1. Edge detection was performed on the image using a laplacian operator. (Same operator used for point detection). Good edges were separated with a threshold, and a binary image was populated.

2. Since circles were hard to detect in the edge detected image, some operations were performed on the image before Hough transform was applied so circles could be detected better.

    1. Lines detected were removed from the binary image to give an image only with circles.

    2. Closing of the image obtained from the previous step was performed after which erosion of the resultant image was obtained. These 2 images were subtracted to get the boundary of the closing the image from step 1. Since a lot of edges were present inside every coin, the algorithm was unable to detect the circle properly.

3. On the binary image, Hough transform was applied using the algorithm described previously and cells with at least an entry of 200 hits were selected as the circles detected.

4. The circles were then added to larger accumulator bins (to avoid detecting the small variations of the same circle multiple times) and converted to the x,y coordinates and displayed on the given image.

4 Results

4.1 Line detection

The following binary image was obtained by applying a laplacian operator to the given image and thresholding the image to take values greater than 0.1 of the max intensity in the resulting image.

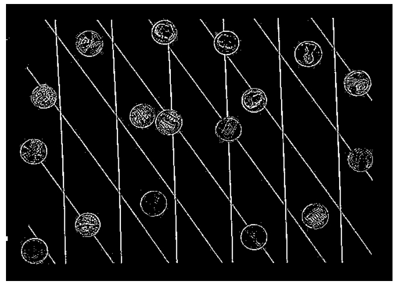
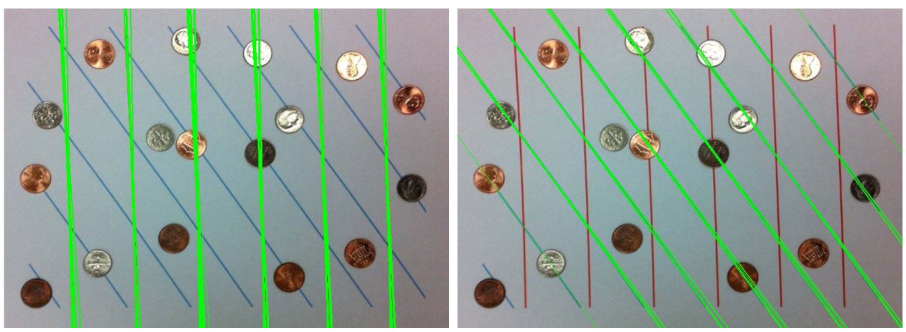


Figure 1: Edges detected in the given image using a 5x5 laplacian operator.

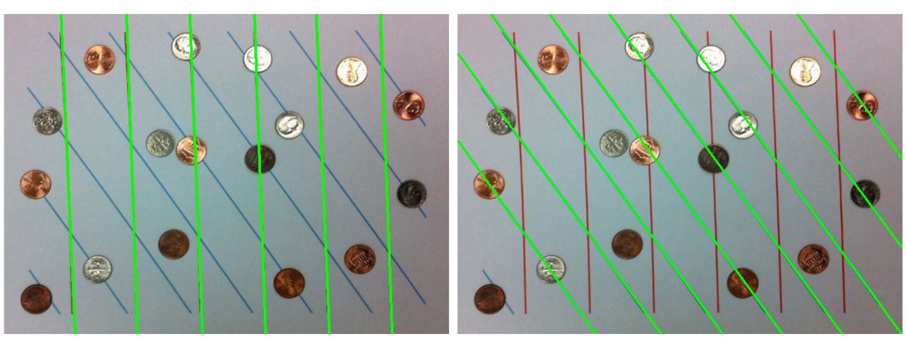
On applying Hough transform to the gradient image, all the red lines and all but one of the blue lines were detected. The following images show the detected lines before collecting the similar lines into discrete bins.



(a) Vertical/Red Lines detected in the given image.

(b) Diagonal/Blue Lines detected in the given image.

The following images show the detected lines after collecting the similar lines into discrete bins of r (+/-) 12 and θ (+/-) 1.



(a) Vertical/Red Lines detected in the given image.

(b) Diagonal/Blue Lines detected in the given image.

Figure 3: Result of applying Hough line transform on the image. (Final result)

4.2 Circle detection

The image shown in Figure 4 was obtained by subtracting the lines detected from the gradient binary image. This was done to reduce the number of points to be taken for the transform. Since the accumulator was 3 dimensional the algorithm performed slowly, and thus removing the lines already detected improved its performance.

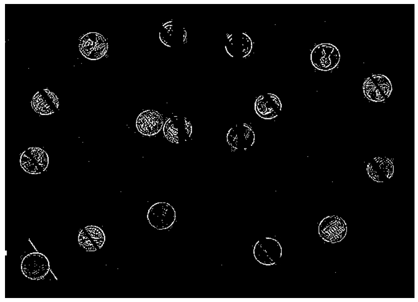
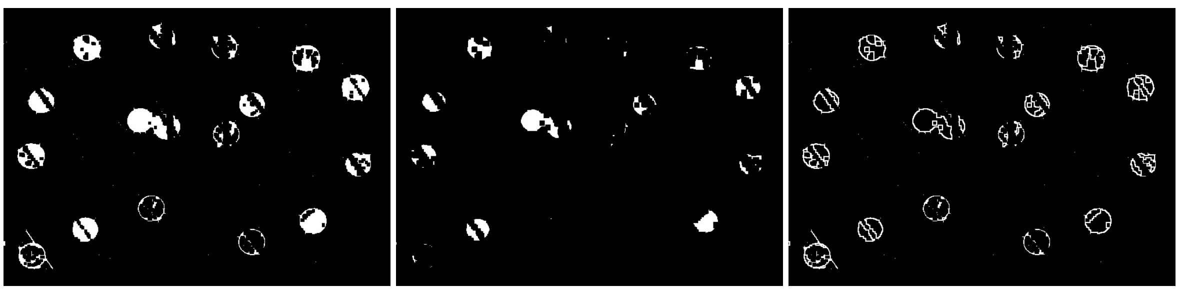


Figure 4: Edges detected in the given image after removing the lines detected.

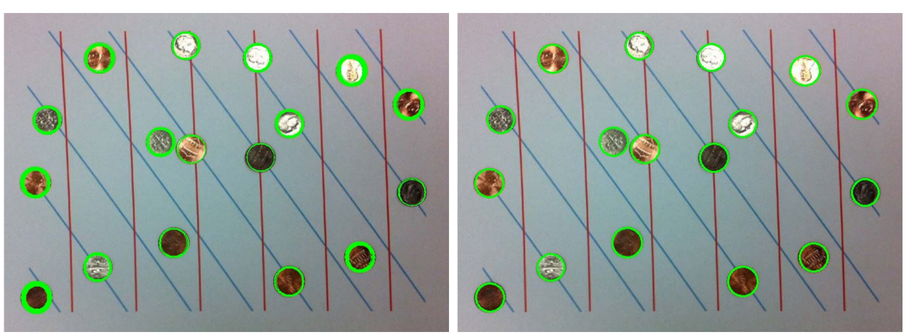
Figure 5 shows various morphological operations applied to Figure 4 in order to accentuate and to detect the circles better. A 5x5 square structuring element with origin at the centre was used.



(a) Closing of the above image. (b) Erosion of the left image. (c) Boundary of the closing (a-b)

Figure 5: Pre-processing image in Fig.4 using morphological operators, for better circle detection.

The following images show the result of the coin detection program. All the coins in the image were detected. Number of coins detected in the image : 17.



(a) Circles detected by applying Hough transform on the pre-processed image. (b) Circles detected by adding larger bins to classify similar circles as same circle.

Figure 6: Result of Hough transform for circle detection.