**B659 Assignment 3: Edge and region-based recognition**

**Term:** Spring 2015

**Date:** Friday March 27, 2015

**Team Members:**

1. Gautam Poornachandra – gpoornac
2. Anirudh Ramesh - anramesh

**GitHub repository:** anramesh-gpoornac-p3

**Language used for the project:** C++

**Contents**

1. Introduction
2. Detecting edges and circles
   1. Detection of edges
   2. Detection of circles
3. Finding circular regions
   1. Segmentation algorithm
   2. Improvements on segmentation
   3. Evaluate circular region
4. Coin counting
5. Conclusion
6. **Introduction**

This project is an implementation of object detection, in particular circular object detection in Image Processing. This will guide through exploring the Hough space, segmentation and understanding some of its power in the Image domain.

This gives a clear understanding of efficient ways of edge recognition and region based recognition.

1. **Detecting edges and circles**

**2.1 Detecting edges**

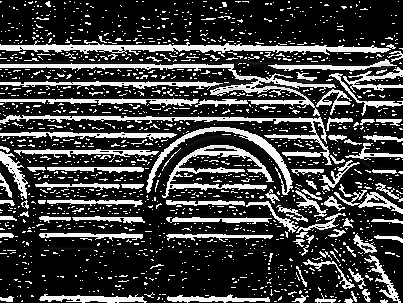
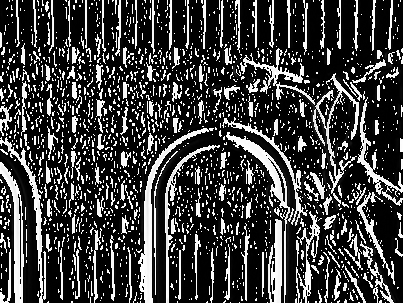
*Assumptions and approach:*

* First estimated partial derivatives in the x and y directions (Ix and Iy) using the Sobel operator.
* Then computed the magnitude of the gradient Im(x, y) = Ix(x, y) 2 + Iy(x, y) 2 at each pixel, and then threshold Im.
* The image is then converted to produce a binary image.

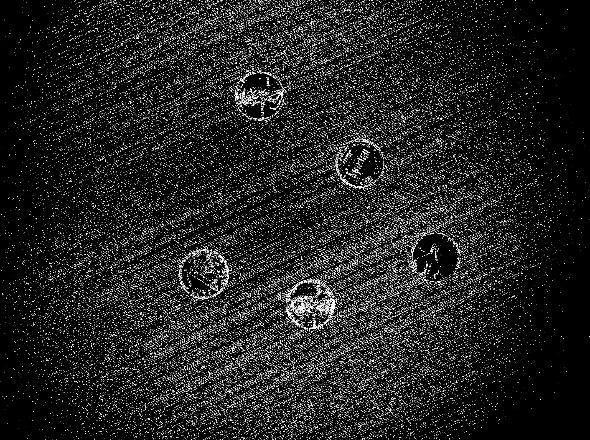
Input Image Output Image

The input image has intermediate x-component and y-component.

x-component y-component

Other examples:

Input Image Output Image

*Observation:* The output image is clearly noticeable with the edges inside it.

**2.2 Detecting circles using Hough transform**

*Assumptions and approach:*

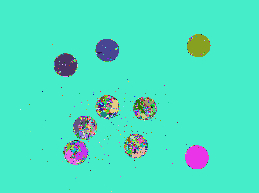
* . A circle can be specified in terms of thre parameters: the row and column coordinates of its center and its radius.
* First we create our accumulator space which is made up of a cell for each pixel, initially each of these will be set to 0.
* For each(edge point in image(i, j)): Increment all cells in the accumulator space according to the equation of a circle and satisfies it.
* The maximum radius is calculated in the range assumed. This maximum value is taken as the radius value to avoid the concentric pattern of circles.
* The local maxima cells, these are any cells whose value is greater than every other cell in its neighborhood. These cells are the one with the highest probability of being the location of the circle(s) we are trying to locate.
* Calculate the local maxima and then take that as the center to draw the circle.

1. **Finding circular regions**

**3.1 Segmentation algorithm**

*Assumptions and approach:*

* Three different classes are created.
* Vertex\_class : contains red, blue, green, ID and coordinates of a pixel.
* Edge\_class: contains edge weight and pointers to vertex\_class
* Graph: contains vectors of pointers to the above two classes
* Edges are initially sorted. Vertices are stored in a 2D vector and combined into components based on threshold where
* Threshold value chosen is 20
* Problems Faced: Takes longer time for a bigger image.

Input Image Output Image

Since, the size of the image is high, the image is resized.

For 3.2, Gaussian filter with sigma = 1 is applied and K = 3.

However, the results were not better.

1. **Coin counting**

*Assumptions and Approach: Mark Image*

* The sample test images are used for testing the counting of coins.
* The implementation of detection of edges and circles is used again to calculate the number of coins in the given image.
* The input image is fed to Sobel operator to get the image with the edge detection.
* This image is then taken as input to calculate the circles using the Hough transform algorithm.
* The local maxima is calculated in the Hough transform. The number of coins in the image, i.e. the count is imagined to be the number of circles detected in the image.
* When calculating the local maxima, we have a counter which calculates the number of centers within the given radii range. Number of centers will give the number of circles in the image approximately.

1. **Conclusion:**

* The edge detection and circle detection of the images works well with Sobel operator and the Hough transform respectively. The application of the Hough transform was clearly witnessed in this project as we used it to detect the number of circles in the given image.
* The segmentation algorithm using vector of vectors proved a costly operation and we were unable to use it for the Images other than very small size or we had to resize it. This computational dependency proved costly in the implementation of the 3.1 for the segmentation. The data structures play a critical role and optimization with the dynamic programming techniques would have generated outputs for larger images within the given time.