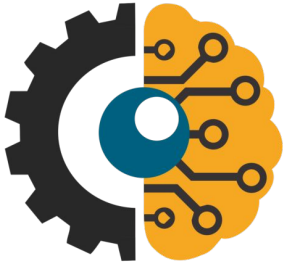
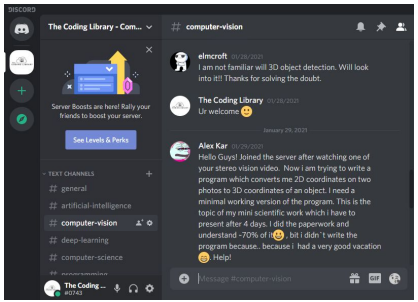


THE CODING LIBRARY

Computer Vision

Hough Transform - Circle and Line Detection



Discord Link in Description

Hough Transform Overview

THE CODING LIBRARY

- Hough Transform Lines
- Hough Transform Circles
- Hough Generalised

- Direct transformation from image space to probability of the existence of some feature
    - Lines, Circles and Generalised shapes

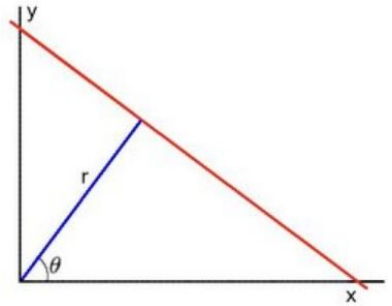
Hough Transform - Line Detection

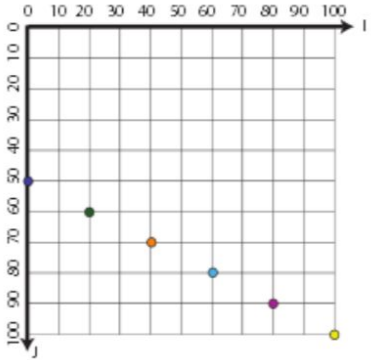
THE CODING LIBRARY

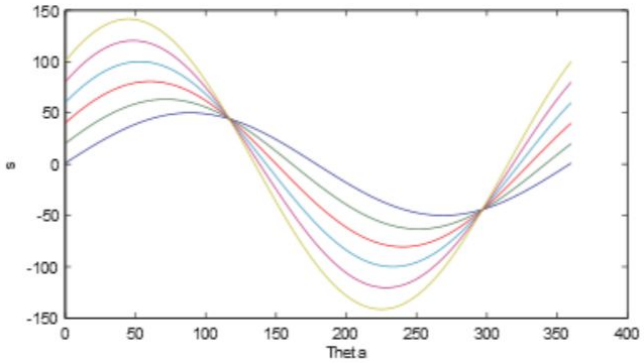
- Equations that represents lines
  - Lines in Hough space
  - Sinusoidal curves
  - Cartesian or polar coordinates

a. In the **Cartesian coordinate system:** Parameters: (m, b).
b. In the **Polar coordinate system:** Parameters: (r, θ)

$$r = x \cos\theta + y \sin\theta$$

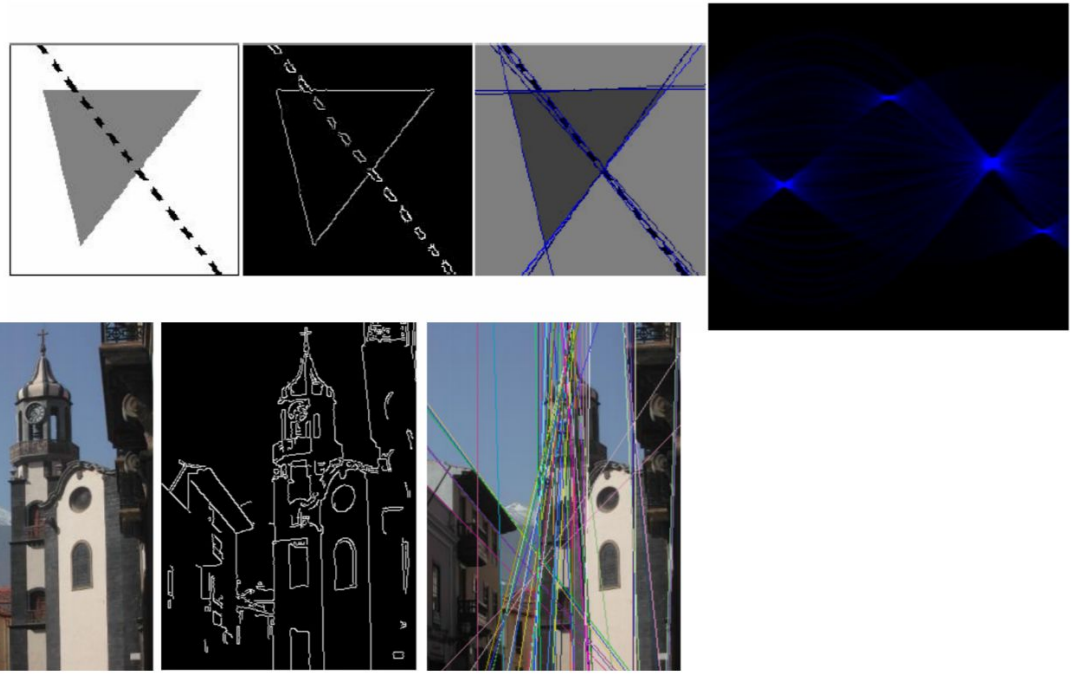






Hough Transform - Line Detection

THE CODING LIBRARY



Line Detection in OpenCV

THE CODING LIBRARY

- Hough Transform for lines:

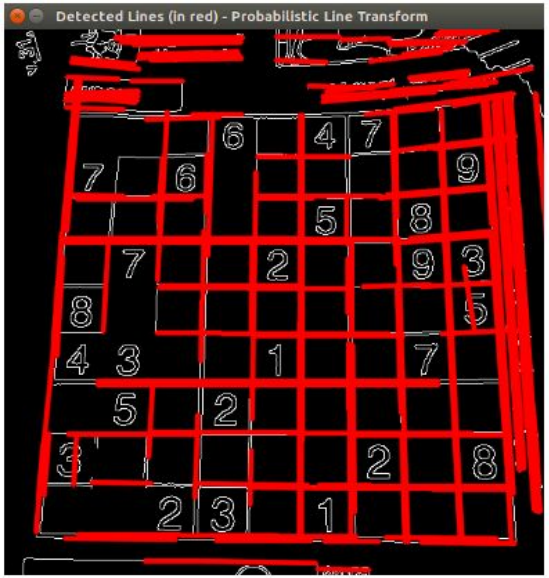
```
vector<Vec2f> hough_lines;
HoughLines( binary_edge_image, hough_lines, 1, PI/200.0, 60);
```

- Probabilistic Hough Transform for line segments

```
vector<Vec4i> line_segments;
HoughLinesP( binary_edge_image, line_segments, 1.0,
                                                    PI/
200.0, 20, 20, 5);
```

Example of Line Detection

THE CODING LIBRARY



Hough Transform - Circle Detection

THE CODING LIBRARY

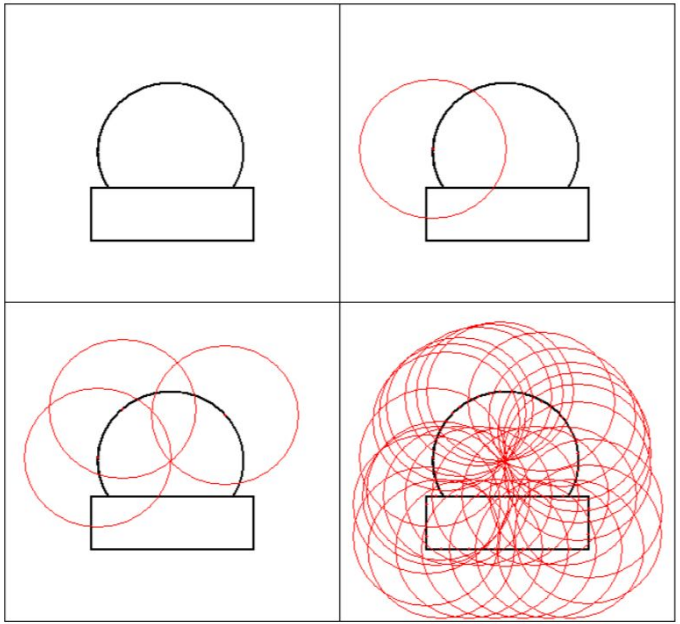
- Roughly same approach as with lines
- A line was defined by 2 parameters
- For a circle we need 3 parameters
- OpenCV uses Hough Gradient Method
  - More efficient
  - Divided in two stages
    - Edge Detection and finding possible circle centers
    - Finding the best radius for each candidate center

$$C : (x_{center}, y_{center}, r)$$

Hough Transform - Circle Detection

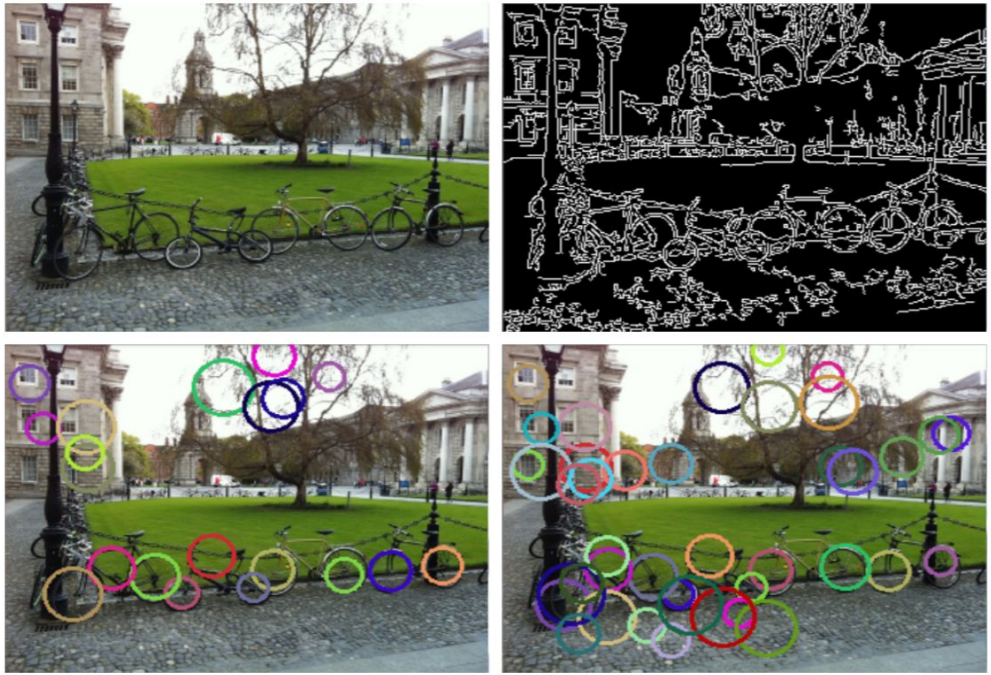
THE CODING LIBRARY

- Equation of a circle
- Transform from image space to Hough Space
- Apply the Algorithm:
  - accumulator = 0
  - for every edge point
    - increment cells in accumulator corresponding to all possible circle centers
  - Search for Maximums



Hough Transform - Circle Detection

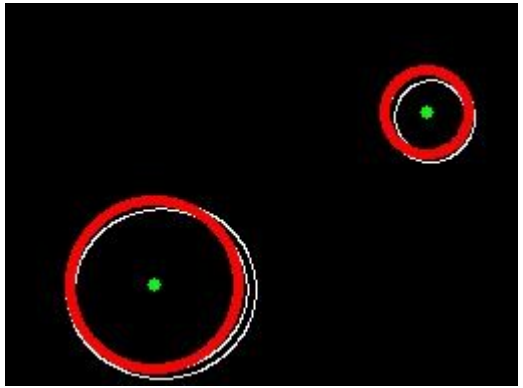
THE CODING LIBRARY



Circle Detection in OpenCV

THE CODING LIBRARY

- Hough Transform for circles



```
vector<Vec3f> circles;
HoughCircles( gray_image, circles, CV_HOUGH_GRADIENT,

2,20,300,20,5,15);
```

Examples of Hough in OpenCV

THE CODING LIBRARY