Canny Edge Detection

Theory

Canny edge Detection is a popular edge detection algorithm. It was developed by John F.Canny.

It contains five stages:

1. Grayscale Conversion
2. Noise Reduction (Gaussian Filter)
3. Determining Intensity Gradients (Vertical and Horizontal Kernels(Sobel Filter))
4. Non_Max Suppression
5. Edge Tracing by Double Thresholding Hysteresis

Grayscale Conversion:

We first convert the RGB coloured image to Grayscale image by using OpenCV methods.

Noise Reduction:

Since edge detection is susceptible to noise in the image, second step is to remove the noise by using Gaussian Filter.

Intensity Gradient:

Smoothened image is then filtered with the Nobel Kernel in both horizontal and vertical direction to get first derivative of horizontal direction (Gx) and vertical direction (Gy).

Sobel Filters:

$$G_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * I;\ G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I; \qquad (1)$$

From these two images, we find edge gradient and direction for each pixel as follows:

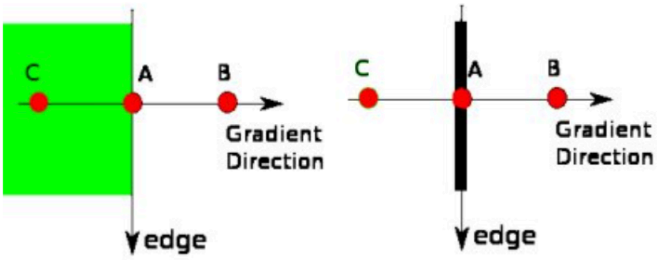
$$\text{Edge\_Gradient}\ (G) = \sqrt{G_x^2 + G_y^2}$$

$$\text{Angle}\ (\theta) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

Gradient direction is always perpendicular to edges. It is rounded to one of four angles representing vertical, horizontal and two diagonal directions.

Non Maximum Suppression:

After getting gradient magnitude and direction, a full scan of image is done to remove any unwanted pixels which may not constitute the edge. For this, at every pixel, pixel is checked if it is a local maximum in its neighbourhood in the direction of gradient.

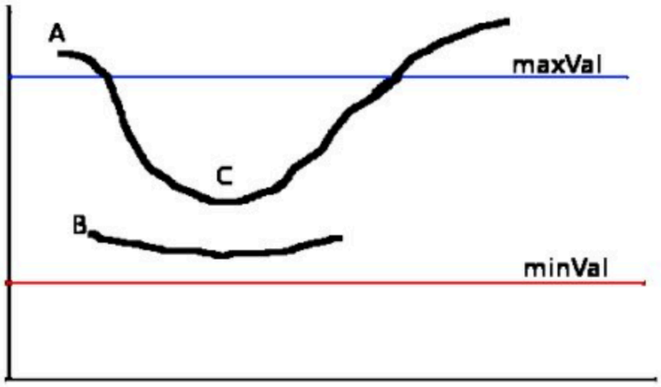


Point A is on the edge (in vertical direction). Gradient direction is normal to the edge. Point B and C are in the gradient directions. So point A is checked with point B and C to see if it forms a local maximum. If so, it is considered for next stage, otherwise, it is suppressed(put to zero).

In short, the result we get is a binary image with "thin edges".

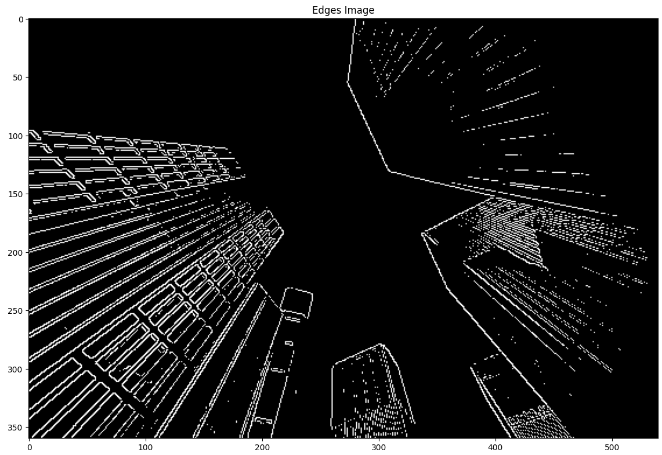
Hysteresis Thresholding:

This stage decides which are all edges are really edges and which are not. For this, we need two threshold values, minVal and maxVal. Any edge with intensity gradient more than maxVal are sure to be edges and those below minVal are sure to be non edges, so discarded. Those who lie between these two thresholds are classified edges or non edges based on their connectivity. If they are connected to "sure edge" pixels, they are considered to be part of edges. Otherwise, they are also discarded.



The edge A is above the maxVal, so considered as "sure edge". Although edge C is below the maxVal, it is connected to edge A, so that also considered as valid edge and we get the full curve. But edge B, although it is above minVal and is in same region as that of edge C, it is not connected to the "sure edge", so that is discarded. So it is very important that we have to select minVal accordingly to get the correct result.

This stage removes small pixel noises on the assumption that edges are long lines. So finally we get is strong edges in the image.


Edges Image
0
50
100
150
200
250
300
350
0
100
200
300
400
500