Canny Edge Detection

Indian Institute of Information Technology Vadodara

Digital Image Processing

Shubham Shubhankar Sharma : 201352004

1 Introduction

The purpose of line detection algorithm is to compress the image while preserving the structural properties of the image. The Canny Edge Detector classifies a pixel as an edge if the gradient magnitude of pixel is larger than those of pixels at both sides in the direction of maximum intensity change. The main aim of John. F Canny was to develop an algorithm that is optimal with regards to the following criteria:

- **Low error rate detection:** The probability of detecting the real edge points should be maximized and the probability of detecting the false edge points should be minimized. This corresponds to maximizing the signal to noise ratio.
- **Localization:** The detected edges should be as close as possible to the real edges.
- **Single Edge point response:** One real edge should not give rise to more than one detected edges.

Edge detection is a step in process of estimating the pose (position and orientation) of the image. Before applying the canny edge detection, we have to do some preprocessing steps which include:

- Determining the ROI (Region of Interest).
- Conversion to gray-scale to limit the computational requirements.
- Histogram Stretching, so that the image has a nice contrast. This step is also necessary to balance the bad lightening condition during the time of the image capture.

2 Canny Edge Detection Algorithm

The algorithm runs in 5 steps:

- **Smoothing:** Blurring of the image to remove noise.
- **Gradient Computation:** The edge should be marked where the gradient of the image has large magnitudes.
- **Non-maximum suppression:** Only local maxima should be marked as edges.
- **Double threshold:** Potential edges should be determined using a threshold.
- **Edge tracking by hysteresis:** Final edges are determined by suppressing all the edges which are not connected with strong edges.

2.1 Smoothing

To reduce the noise from the image we use filtering. In this particular algorithm we use Gaussian filter. The standard deviation is set depending on the quality of the image. This gives the degree of smoothness to the image.

2.2 Gradient Computation

This algorithm finds the edges where the gray scale intensity of the image changes the most. These areas are found by determining gradients of the image at each pixel. And we use the Sobel-operator for this purpose. First we have to approximate the gradient in x- and the y-direction respectively.
The gradient magnitudes(edge strength) can be determined by as an Euclidean distance by applying the Pythagoras Theorem.

The image of the gradient magnitudes often indicate the edges quite clearly. However, the edges are typically very broad and we could not clearly find out where exactly the edges are. To make this possible the direction of the edges are determined and stored.

2.3 Non-maxima suppression

The problem is edges generated using gradient typically wide ridges around local maxima. We use non-maxima suppression to thin those ridges. The algorithm is for each pixel in the image:

- Round the gradient direction 0 to nearest 45, corresponding to the use of an 8-connected neighbourhood.
- Compare the edge strength of the current pixel with the edge strength of the pixel in the positive and the negative gradient direction.
- If the edge strength of the current pixel is largest; preserve the values of the edge strength. If not, suppress the values.

Each pixel is compared with the pixel above and just below.

2.4 Double Thresholding

The edge pixel remaining will probably have many true edges but some may be caused by color or noise variation due to rough surfaces for example. The simplest way to filter out the strong edges is to decide a threshold, so that only the edges stronger than a particular value will be preserved. We use double thresholding in canny edge detection algorithm. Edge pixel stronger than the high threshold are marked as strong; edge pixel weaker than the low threshold are discared and the pixels in between are called weak.

2.5 Edges tracked by Hysteresis

Strong edges are interpreted as certain edges and immediately included in the final image. But the weak edges are only included when it is connected to strong edges. Edge tracking can be implemented by the BLOB-analysis(Binary Large Object). The edge pixels are divided into connected BLOB's using 8-connected neighbourhood. BLOB's containing at least one strong edge is preserved while others are suppressed.

3 References

- dasl.mem.drexel.edu/alumni/bGreen/www.pages.drexel/can_tut.html
- https://en.wikipedia.org/wiki/Canny_edge_detector
- www.massey.ac.nz/ mjjohnso/notes/59731/presentations/img_proc.PDF
- www.cse.iitd.ernet.in/ pkalra/csl783/canny.pdf
- www.di.ubi.pt/ agomes/cvm/teoricas/05-edgedetection.pdf