Image Processing Tutorial - Edge Detection

Seongin Na

March 2021

1 Introduction

Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly helping to extract features in an image. Among many methods, sobel edge detection and canny edge detection are the two most representative techniques in edge detection. In this tutorial, we will write a MATLAB script for sobel and canny edge detection from scratch without image processing toolbox.

2 Background Knowledge

2.1 Sobel Edge Detection

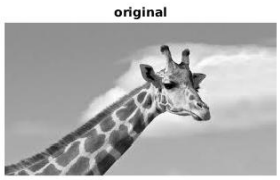
Sobel Edge detection was proposed by Irwin Sobel and Gary Feldman, colleagues at the Stanford Artificial Intelligence Laboratory (SAIL) in 1968. It extracts gradients in the images by convolving image with a small-size filter in the horizontal and vertical directions. As the edges in the image have high pixel value difference comparing to its neighbours, i.e. high gradients, we can detect edge by computing gradients of the image. Due to its simplicity, the computation cost of Sobel edge detection is inexpensive. The gradient in horizontal and vertical direction, G_x, G_y, can be computed by convolving the image and the 3×3 filter in horizontal and vertical direction as described below.

$$\mathbf{G}_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * \mathbf{I}, \quad \mathbf{G}_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * \mathbf{I} \tag{1}$$

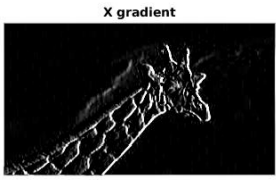
, where $\mathbf{I}$ is the image matrix. By combining $\mathbf{G}_x$ and $\mathbf{G}_y$, the gradient of the image. $\mathbf{G}$, can be computed as below:

$$\mathbf{G} = \sqrt{\mathbf{G}_x^2 + \mathbf{G}_y^2} \tag{2}$$

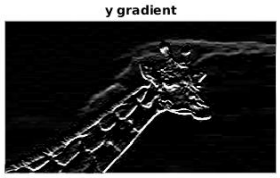
Figure 1 shows how the gradient of the image is obtained step-by-step. Figure 1a shows the original image, Figure 1b shows the gradient in horizontal direction, Figure 1c shows the gradient in vertical direction, and Figure 1d shows the



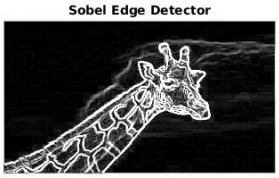
(a) Original Image



(b) X gradient



(c) Y gradient



(d) Result

Figure 1: Edge detection with Sobel edge detection.

resultant gradient of the image. The awesome giraffe image is downloaded from the website shown in the footnote [1].

2.2 Canny Edge Detection

Canny edge detection is a multi-stage edge detection algorithm devised by John F. Canny in 1986. It is more advanced technique than Sobel edge detection; moreover, one of the stages of the Canny edge detection is Sobel Edge detection. The stages of Canny edge detection is listed below.

1. Apply Gaussian filter to smooth the image for noise reduction
2. Find the intensity gradients of the image (Sobel edge detection)
3. Apply gradient magnitude thresholding or lower bound cut-off suppression to get rid of unwanted response to edge detection
4. Apply double threshold to determine potential edges
5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

[1] https://animals.sandiegozoo.org/animals/giraffe

2.2.1 Applying Gaussian Filter

Gaussian filter is a famous noise removing technique in image processing. By convolving an image with Gaussian kernel, smoothing effect is applied into the target image. The element value at (i, j) 2D Gaussian Kernel with size of $(2k+1) \times (2k+1)$, H_{ij}, is assigned as follows:

$$H_{ij} = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(i-(k+1))^2 + (j-(k+1))^2}{2\sigma^2}\right); 1 \leq i, j \leq (2k+1) \quad (3)$$

where σ is a standard deviation of the Gaussian kernel.

2.2.2 Find the intensity gradients of the image

This part is similar to Section 2.1, but angle for gradient values are required for next step. The angle for gradient values are computed as follows:

$$\Theta = atan2(\mathbf{G_y}, \mathbf{G_x}) \quad (4)$$

2.2.3 Gradient Magnitude Thresholding

This step is to find the locations with the sharpest change of intensity value. In Figure 1d, we can see that the resultant image of Sobel edge detection has rather thick edges. For Canny edge detection, by applying gradient magnitude thresholding, only the most intense parts of the edge remain.

This can be done by the following process.

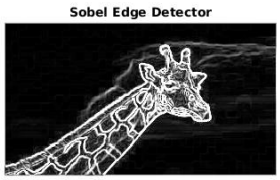
1. Convert the continuous gradient values, Θ, into one of the four angles, 0deg, 45deg, 90deg, and 135deg.
2. Compare the pixel with the neighbor pixels along with the gradient. For example, if a pixel has 0deg of gradient, it has to compare with horizontally left and right pixels.
3. If the selected pixel has the higher intensity than the two neighbour pixels, select that pixel only as an edge. Otherwise, ignore that pixel.
4. Save the edge information into another matrix.

2.2.4 Double Thresholding

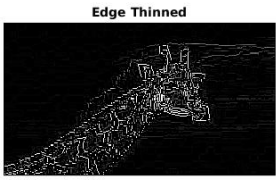
After applying gradient magnitude thresholding, the filtered image will have more accurate edge information. However, inaccurate edge pixels might remain in the filtered image due to noise or color variation, etc. To suppress such pixels, double thresholding can be used. Double thresholding can be applied for the filtered image using two threshold values, T_h, T_l. If the pixel of the filtered image has value higher than T_h, it can be classified as strong edge, thereby remaining pixel value. If the pixel value of the filter image is between T_h and T_l, it can be classified as weak edge so lower value can be assigned to the pixel value. If the

(a) Original Image

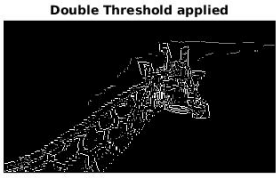
(b) Gaussian filter applied



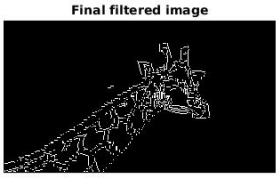
(c) Sobel operator applied



(d) Edge Thinning applied



(e) Double thresholding applied



(f) Hysteresis thresholding applied

Figure 2: Edge detection with Canny edge detection.

pixel value of the filter image is lower than T_l, the pixel value can be suppressed to 0 so that it is no longer classified as edge.

2.2.5 Hysteresis Thresholding

Through the last step, the inaccurate edge data has been removed. we would like to further deal with the weak edge values. To distinguish the weak edge values if they are real edges or not, the neighbour pixels need to be checked. If one of the 8 neighbour pixels is true edge, i.e. strong edge values, we consider the weak edge value as true edge. Otherwise we dismiss the weak edge value.

Figure 2 shows the process of Canny edge detection is applied.

3 Question

This tutorial asks you to make two edge detection filter: Sobel and Canny. With the guide of the MATLAB script you are asked to write separate script to construct two filters.

Please check the given MATLAB script (sobel_edge_detection_student.m and canny_edge_detection_student.m) and add your own code to build the two filters by using given background information about Sobel and Canny filter. You can use your own image for this tutorial.