Analysis of Edge Detection Techniques: Application to Document Scanner

Aman Kumar
201911018
Dhirubhai Ambani Institute of Information Technology
Gandhinagar, India
201911018@daiict.ac.in

Harsh Vardhan Singh
201911022
Dhirubhai Ambani Institute of Information Technology
Gandhinagar, India
201911022@daiict.ac.in

***Abstract*—Image segmentation is the first pre-processing step for feature extraction in images. Edge Detection is an image segmentation technique has been widely used for a variety of applications such as Object detection and Recognition, Video Retrieval, Image Enhancement. This paper reviews the use of different type of edge detection techniques for an application of Document scanner . This paper demonstrates the experimental result of edge detection algorithms compares all of them and shows how the Canny edge detection is best among all of them. With the base of Canny, Sobel operator and the improvement, this paper discusses various aspects of edge detection for real-time application.**

***Index Terms*—Edge Detection, Laplacian, Sobel, Canny, Prewitt**

I. Introduction

Image Processing has been evolved basically for two purposes, first is for better interpretation of images by human and second is for data storage, analysis and transfer. There have been significant advances in the field of Image Processing. The applications of image processing have emerged in different fields like medical, security and surveillance, retrieval and much more. Various applications such as Feature extraction, Object detection and recognition, Image segmentation is the process of dividing the image into regions or segments which are dissimilar in certain aspects or features such as colour, texture or gray level. Image Segmentation can be done in many ways. This paper discusses the edge based image segmentation

II. Edge Detection

Edge detection is a type of image segmentation technique where edges are detected in the image. Edges are introduced as a set of connected points lie on the boundary between two regions. The edges represent object boundaries and therefore can be used in image segmentation to subdivide an image into its basic regions or objects. The image is simply a combination of different intensities, noise, etc. The local change of intensity in an image is edge. Edges divide the image in different segments or regions. Edges are boundary between segments or regions. Edge detection means extraction of information about the image. Examples are location, shape, size of object in image, image sharpening and enhancement. Extracted features can be used for recognition as a higher-level computer vision algorithm. As edge detection is a fundamental step in computer vision, it is necessary to point out the true edges to get the best results from the matching process. That is why it is important to choose edge detectors that fit best for the application. There are many edge detection algorithms such as canny edge detection, Prewitt, Sobel, Laplacian.
Out of these algorithms, efficiency of Canny and Sobel edge detection algorithms is higher than other algorithms.

III. Sobel Edge Detection

The Sobel operator measures a 2-Dimensioanal spatial gradient on an image and gives more attention on regions of high spatial gradient corresponding to edges . It is used for finding gradient magnitude at each point in a gray scale image. There are various pairs of Sobel operator such as 3×3, 5x5 convolution kernels.
The major difference is that in sobel operator the coefficients of masks are not fixed and they can be adjusted according to our requirement unless they do not violate any property of derivative masks.

TABLE I
SOBEL MASKS

Gx			Gy		
1	0	-1	-1	-2	-1
2	0	-2	0	0	0
1	0	-1	1	2	1

The gradient magnitude is given by:

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{1}$$

And the approximate value of the Magnitude is calculated by:

$$|G| = |Gx| + |Gy| \tag{2}$$

Input to the Sobel is a sample image. The expected output is efficient detection of object boundaries

A. Pseudo code algorithm

The pseudo code algorithm as discussed as follows:
Step1: Accept the sample input image.

Step2: Perform masking on given image.
Step3: Apply algorithm and the gradient.
Step4: On the input image perform mask manipulation in both the directions.
Step5: Find the absolute magnitude of the gradient. The absolute magnitude is the output edge.

B. Advantages

The advantages as seen are:
1. Random noise of the image affected by smoothing effect of average factor.
2. Because of X-directional and Y-directional difference of the image improves element edge on both sides, therefore edge looks thick and bright.

IV. Prewitt Edge Detection

Prewitt operator is used for edge detection in an image. It detects two types of edges

- Horizontal edges
- Vertical Edges

Edges are calculated by using difference between corresponding pixel intensities of an image. All the masks that are used for edge detection are also known as derivative masks. Because as we have stated many times before in this series of tutorials that image is also a signal so changes in a signal can only be calculated using differentiation. So that's why these operators are also called as derivative operators or derivative masks.
All the derivative masks should have the following properties:

- Opposite sign should be present in the mask.
- Sum of mask should be equal to zero.
- More weight means more edge detection.

Prewitt operator provides us two masks one for detecting edges in horizontal direction and another for detecting edges in an vertical direction.

A. Vertical direction

TABLE II
Prewitt Vertical Operator

-1	0	1
-1	0	1
-1	0	1

Above mask will find the edges in vertical direction and it is because the zeros column in the vertical direction. When you will convolve this mask on an image, it will give you the vertical edges in an image.

1) How it works: When we apply this mask on the image it prominent vertical edges. It simply works like as first order derivate and calculates the difference of pixel intensities in a edge region. As the center column is of zero so it does not include the original values of an image but rather it calculates the difference of right and left pixel values around that edge. This increase the edge intensity and it become enhanced comparatively to the original image.

B. Horizontal direction

TABLE III
Prewitt Vertical Operator

-1	-1	-1
0	0	0
1	1	1

Above mask will find edges in horizontal direction and it is because that zeros column is in horizontal direction. When you will convolve this mask onto an image it would prominent horizontal edges in the image.

1) How it works: This mask will prominent the horizontal edges in an image. It also works on the principle of above mask and calculates difference among the pixel intensities of a particular edge. As the center row of mask is consist of zeros so it does not include the original values of edge in the image but rather it calculate the difference of above and below pixel intensities of the particular edge. Thus increasing the sudden change of intensities and making the edge more visible. Both the above masks follow the principle of derivate mask. Both masks have opposite sign in them and both masks sum equals to zero. The third condition will not be applicable in this operator as both the above masks are standardize and we can't change the value in them.

V. Laplacian Edge Detection

Laplacian Operator is also a derivative operator which is used to find edges in an image. The major difference between Laplacian and other operators like Prewitt, Sobel, Robinson and Kirsch is that these all are first order derivative masks but Laplacian is a second order derivative mask. In this mask we have two further classifications one is Positive Laplacian Operator and other is Negative Laplacian Operator.
Another difference between Laplacian and other operators is that unlike other operators Laplacian didn't take out edges in any particular direction but it take out edges in following classification.
• Inward Edges
• Outward Edges
Let's see that how Laplacian operator works.

A. Positive Laplacian Operator

In Positive Laplacian we have standard mask in which center element of the mask should be negative and corner elements of mask should be zero.

TABLE IV
Laplacian Positive Operator

0	1	0
1	-4	1
0	1	0

Positive Laplacian Operator is use to take out outward edges in an image.

B. Negative Laplacian Operator

In negative Laplacian operator we also have a standard mask, in which center element should be positive. All the elements in the corner should be zero and rest of all the elements in the mask should be -1.

TABLE V
LAPLACIAN NEGATIVE OPERATOR

0	-1	0
-1	4	-1
0	-1	0

Negative Laplacian operator is use to take out inward edges in an image

C. How it works

Laplacian is a derivative operator; its uses highlight gray level discontinuities in an image and try to deemphasize regions with slowly varying gray levels. This operation in result produces such images which have grayish edge lines and other discontinuities on a dark background. This produces inward and outward edges in an image.

The important thing is how to apply these filters onto image. Remember we can't apply both the positive and negative Laplacian operator on the same image. we have to apply just one but the thing to remember is that if we apply positive Laplacian operator on the image then we subtract the resultant image from the original image to get the sharpened image. Similarly if we apply negative Laplacian operator then we have to add the resultant image onto original image to get the sharpened image.

VI. CANNY EDGE DETECTION

The canny edge detection algorithm was proposed to enhance the edge detection process. Three important criteria were taken into consideration for this purpose. The first and most important criterion was to detect all the important edges in the source image. This means the goal was to lower the error rate. The second criterion was that the edge points to be detected as close as possible to the true edge, also called as localization. A third criterion was not to have more than one response to a single edge.

It first smooths the image to eliminate noise. Then the image gradients are calculated to point out those regions where the gradient difference is maximum, which have high spatial differences. Finally, it then tracks along these regions and discards any pixel that weakly defines an edge (nonmaxima suppression) in order to make the edges thinner. To further reduce the gradient array, it performs hysteresis which tracks along the remaining pixels that have minimum gray level values but have not been suppressed. The equation below is used.

$$G(x,y) = \frac{1}{2\pi\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3)$$

A. Working of Canny Edge algorithm

The algorithm runs in 5 separate steps:

1) Smoothing: The input image must be converted into gray scale by adjusting contrast and brightness, so that the image is blurred to remove noise. Thus, the first step is to filter out the noise in the original image to make the location and detection edges efficient. Generally a Gaussian filter is used for noise removal.

2) Finding gradients: Edge pixels are those where there is a sharp change in gray level values, these are identified by computing the gradient of the image. The gradient is a unit vector which points in the direction of maximum intensity change. In this step first the vertical and horizontal components of the gradient are computed and then the magnitude and direction of the gradient is computed.

3) Non-maxima suppression: In this step the detector converts the thick edges in the image, based on the gradient magnitudes, to approximately thin and sharp edges which can be further used for recognition purpose. Mainly edge thinning is performed in non-maxima suppression. In this process the image is scanned along the edge direction and discards any pixel value that is not considered to be an edge which will result in thin line in the output image.

4) Double Thresholding: The threshold value consists of 2 characters, T1 = High Threshold, T2 = Low Threshold. The pixels having values of gray scale level higher than T1 are strong edge pixels, and the result is edge region. The pixels having values of gray scale level less than T2 are weak edge pixels, and the result is non-edge region. If the pixels have values of gray scale level between T1 and T2, the result is depending on the neighbouring pixels.

5) Edge tracking by hysteresis: Edges that do not connect to a very certain (strong) edge are discarded in the final output image. Strong edges are interpreted as "certain edges" and are included in the final edge image. Weak edges that are linked with strong edges are included in the output image.

RESULT

B. Sobel Edge Detector

Sobel edge detector gives thicker and disturbed edges.

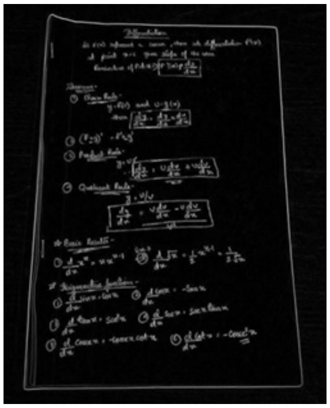

Fig. 1. Sobel Edge Detector.

C. Laplacian Edge Detector

Greyish edge lines and other discontinuities on a dark background. This produces inward and outward edges in an image.

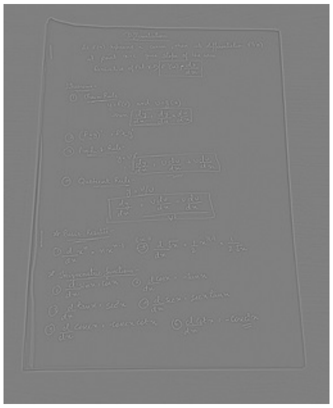

Fig. 2. Laplacian Edge Detector.

D. Canny Edge Detector

Canny edge detector gives the best crisp edges.

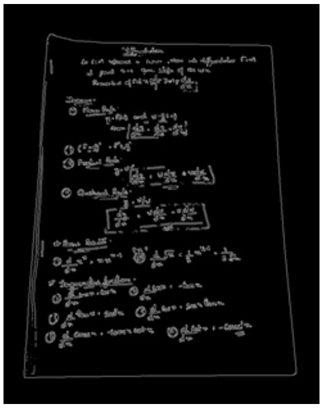

Fig. 3. Canny Edge Detector.

VII. Steps for Creating the Document Scanner

A. Edge Detection

Edge detection is finding points where there are drastic changes in intensity. These points gives us edges of objects on the screen. Most of the shape information of an image is enclosed in edges. So first we detect these edges in an image and by using these filters and then by enhancing those areas of image which contains edges, sharpness of the image will increase and image will become clearer.

Canny edge detection is a multi-step algorithm that can detect edges with noise supressed at the same time. Smooth the image with a Gaussian filter to reduce noise and unwanted details and textures.

1. We take the input image from the user

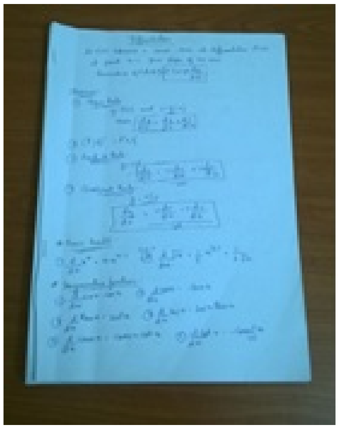

Fig. 4. Original Image.

2. We apply the Gray-Scale transformation to reduce the high intensities and complexity for processing purposes. We also apply Gaussian blur to remove the unwanted edges

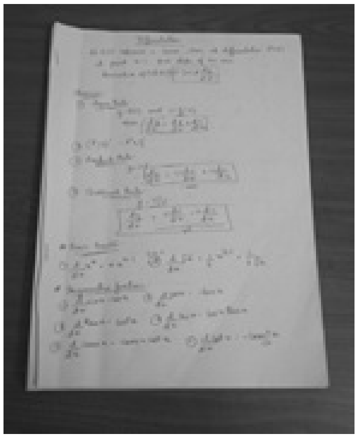

Fig. 5. Gray-Scale Image.

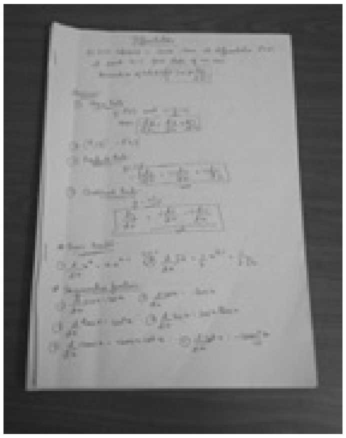

Fig. 6. Gaussian Blurred Image.

3. Apply the canny edge detection.

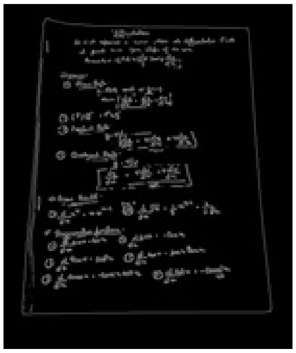

Fig. 7. Canny Edge Detected Image.

B. Finding Contour

We then use edged picture from above to draw the contour around receipt image with green border as shown below. This contour is crucial to understand orientation and 4 important points (top left (tl), top right(top right), bottom right (br), and bottom left (bl)

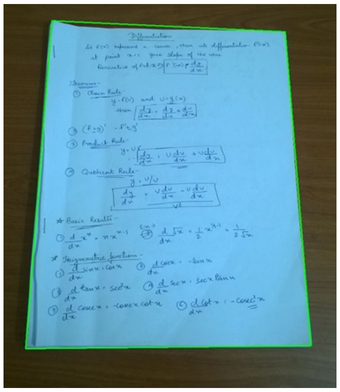

Fig. 8. Four Point Edge Detected Image.

C. Perspective Transform

Once we establish tl, tr, bt and bl all we need to do is to rotate and scale these points to get birds eye view of the image by using perspective transform.

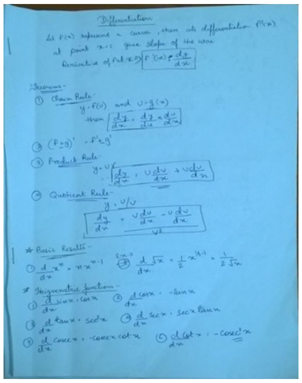

Fig. 9. Wrapped to Perspetive Image.

D. Contrast Enhancement

At last we enhance the contrast of the image for clearer and better results.

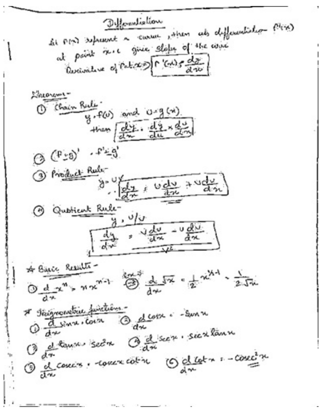

Fig. 10. Contrast Enhanced Image.

LIMITATIONS

- The amount of parameters leads to infinitely tweaking for getting just that little better result.
- Also due to the Gaussian smoothing: the location of the edges might be off, depending on the size of the Gaussian kernel.
- Fails in detecting edges when image brightness is lower.
- The method has problems with corners and junctions:
  - The Gaussian smoothing blurs them out, making them harder to detect (same goes for the edges them self)
  - The corner pixels look in the wrong directions for their neighbours, leaving open ended edges, and missing junctions

CONCLUSION

Edge detection is very useful in image processing. This paper discussed and analyzed the strengths and weaknesses of a few edge detection methods. Canny edge detectors have many favorable features such as smoothing effect to remove noise, and improving signal to noise ratio through a process known as non-maximal suppression. Complex algorithms used in Canny method makes it time consuming and difficult to implement to reach real time response speeds.
Hence for applications like document scanner, Canny edge detectors performs the best.

REFERENCES

[1] Zolqernine Othman, Habibollah Haron and Mohammed Rafiq Abdul Kadir, “Comparison of Canny and Sobel Edge Detection in MRI Images”.

[2] https://medium.com/@nikatsanka/comparing- edge-detection-methods-638a2919476e

[3] https://www.researchgate.net/publication/297736749_Comparison_of_Edge_Detection_Techniques

[4] https://www.researchgate.net/publication/321776293_A_Comprehensive_Analysis_of_Image_Edge_Detection_Techniques/link/5b3efdec0f7e9b0df5fe74ee/download

[5] http://www.tech-quantum.com/contour-detection-in-an-image/