IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

An Efficient Edge Detection Approach For Better Edge Connectivity

M. Sai Sukeerthi

M.tech Scholar, DECS, ECE Department,
JNTUA College of Engineering (AUTONOMOUS) Ananthapuramu, Andhra Pradesh, India.

ABSTRACT:

Edge detection is a technique used to identify item boundaries in structural and digital images and crop out undesired areas. Edge detection is crucial for its dependability and security, which improves object recognition in computer vision and machine vision applications including face recognition, pedestrian detection, and video surveillance. Edge connectivity and edge thickness, two key constraints on edge detection that have been employed in a variety of advancements, were introduced in this study. An important problem in computer vision has been the best way to choose a threshold for efficient edge detection. Edge connectivity and edge thickness constraints are both addressed by the suggested B-edge detection method with various threshold techniques.

I.INTRODUCTION

The edges of the image show large variations in local intensity. Edges often appear in images when two different parts meet. Segmenting an image based on sudden changes in intensity is a technique called edge detection. Image edges can be used to extract important features. Edge detection is an image processing method that identifies the edges of objects in an image. It works by looking for changes in brightness. Edge detection is the most common way to identify discontinuities in gradient values. Edge detection is often considered a naive technique performed in many computer vision applications and low-level image processing. The two main methods of detecting edges in images are edge matching techniques and thresholding/enhancement techniques. Image properties enhanced by processing with operators are represented as discontinuities by the thresholding/enhancement approach. In a later edge-matching process, perfect edge pixels are matched.

Edge detection steps usually start with image smoothing/filtering, followed by image enhancement, detection, and localization.

Image smoothing and filtering: In this step, the image is filtered to reduce noise and improve the efficiency of the edge detector.

Enhancement: The basic goal of image enhancement techniques is to increase the quality of digital images. Improved methods aim to produce an image that is more suitable than the original for a given application.

Detection: Extract all edge points and select which edge pixels will remove noise during detection In general, a threshold provides a standard for edge point detection.

Localization: Some applications may require a resolution of less than 1 pixel to approximate the exact location of edges.

The basic goal of edge detection is to simplify visual data to reduce the amount of data to process. A local test discontinuity of each pixel is used to perform edge detection. Amplitude, orientation, and position of the sub-region of interest are all important aspects of the image. Edge detection allows users to examine image properties for large changes in gray levels. This texture identifies the start and end of sections in the image. Reduce the amount of data in the image while preserving structural properties.

Edge detection algorithms come in two categories: spatial domain and frequency domain. Direct manipulations, such as operator-based techniques, are performed on image pixels in the spatial domain. Operator-based methods fall into two categories 1. Gradient-based operator and 2. Gaussian-based operators. Gradient-based operators compute the first derivative of digital images, including the Sobel, Prewitt, and Robert operators. The Gaussian-based operator computes the second derivative for digital images, similar to the Canny edge detector or the Gaussian-Laplace operator. In this work, an efficient edge identification method with improved edge connectivity and edge thickness is proposed. We can perform MSE and PSNR of the proposed algorithm.

II.EXISTING METHOD

First-order and Second-order approaches are used to classify edge detection operators. The first-order method estimates the first derivative of the image gradient and roughly approximates the second derivative of the image gradient in the same way as the second-order method. The operators Sobel, Prewitt, and Robert are the first derivatives. The Laplacian of Gaussian and Canny edge operators are second derivatives.

A discrete differentiation operator called the Prewitt operator finds horizontal and vertical edges in an image. The operator determines the gradient of image intensity at each point. Each pixel's gradient vector points in the direction of potential maximum intensity, and its length reflects the rate of change in that direction. Easy to make, but very sensitive to noise.

Later, the Sobel edge detection algorithm (first derivative edge detection method) was developed. Edge detection algorithms use the discrete derivative operator to compute a gradient approximation of the image intensity function. The Sobel operator produces a vector normal or appropriate gradient vector for each pixel in the image. Computes the approximate vertical and horizontal derivatives using two 3 x 3 kernels or masks convolved with the input image. Sobel is a weighting method that locates edges based on the phenomenon that extreme values can be reached at edge points. To do this, we weigh all neighboring point pixels in each direction. Local averaging can affect Sobel edge detection, which can also be very noisy and lead to false edges.

Canny developed a famous operator in 1986 known as the Canny operator. Canny edge detection is a method to extract meaningful structural information from various objects while reducing the amount of data to process. The image is smoothed and denoised by a Canny operator using a Gaussian filter. Canny wanted to determine the most effective edge detection method. In this context, effective detection, accurate localization, and minimal response are the optimal edge detector characteristics.

Canny used the variational method. This is a technique for identifying work that best utilizes a particularly practical approach that meets the above requirements. This article describes an ideal Canny detector function that smoothes the input image by median filtering. The traditional Canny algorithm performs the following steps:

Smoothing: Blur the image using the median filter to remove noise.

$$d = \frac{\{8\times(\emptyset)-1\}}{2} \qquad (1)$$

Find Gradients:

Mark the edges of the image where the gradients are very large. After calculating the magnitude of the gradient, non-maximal suppression was used.

Gauss Kernel=$\frac{1}{\sqrt{2\pi\times\emptyset}}.\exp\left[\frac{-(E^2)}{2\times(\emptyset^2)}\right]$ (2)

Non-maximal suppression: edges should only be local maxima. Double Threshold: The threshold is used to identify potential edges.

Hysteresis-Based Edge Tracking: The trailing edge is found by suppressing all edges unrelated to a very specific edge.

Canny Edge Detector has the following loopholes: a) It's taking too long. b) Lack of edge localization found as a result of Gaussian smoothing. d) Gaussian smoothing blurs the image. d) To extract whole edges from an image, the edges of the image should be connected. Therefore, a new effective strategy is proposed to address the above limitations in edge detection.

III.PROPOSED METHOD

Below is a description of the B edge detection algorithm.

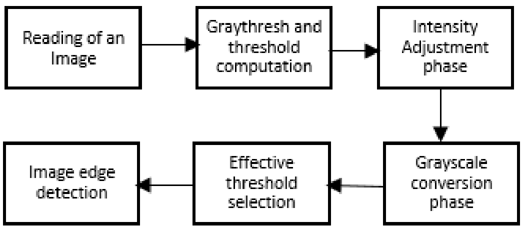


Figure1: B-edge detection

Step1:

Reading the image: Read the input image from the database.

Step2:

Threshold and graythresh calculation phase: The grayscale value of each pixel is determined by the formula

$$\psi=\frac{(\sum_{\alpha=1}^{n}\cdot\sum_{\beta=1}^{m} ara)}{m.n} \qquad (3)$$

Where ara is the input image array and m, n is the pixel dimensions. Image averaging is provided by Graythresh.

Step3:

Intensity adjustment phase: The relevant threshold for the image is calculated as:

$$\emptyset=\frac{\psi.\ 20}{8.33} \qquad (4)$$

Multiplying the calculated value by 20/8.33 to improve the image contrast. This 20/8.33 is taken into account because further simulations show the value to be greater than 20/8.33. This will give you a higher value. As a result, the additional blurring step is heavily implemented, resulting in very smooth images and loss of edge information. Values less than 20/8.33 have lower values, but this results in a weaker implementation of the blurring that produces double edges and automatic noise generation, resulting in unacceptable results.

The image may not be suitable for the capture of the result due to lack of image contrast, in which case the calculated value will be lower. So we try to increase the contrast of the image and our method suggests it. As a result, the first condition "edge thickness" is satisfied by implementing an additional step of blurring in the algorithm.

Step4:

Grayscale conversion phase: A grayscale image is technically varied over an intensity range of 0 to 255 during the grayscale conversion phase. A key problem in computer vision has been how to choose an efficient edge detection threshold. It has been observed that

ignoring the entire intensity range does not improve edge detection sufficiently. Even the Canny edge operator double-threshold method, which uses gradient magnitude-based data to detect edges, needs to improve edge connectivity and image data. Therefore, the proposed solution to improve edge connectivity uses a multi-threshold approach.
In this case, a triple intensity threshold has specified that attempts to fully cover the grayscale intensity range, thus not producing edge results with compromised intensity. Therefore, using these triple intensities provides full coverage of the intensity range, improved edge connectivity, and successful edge detection.

Step5:

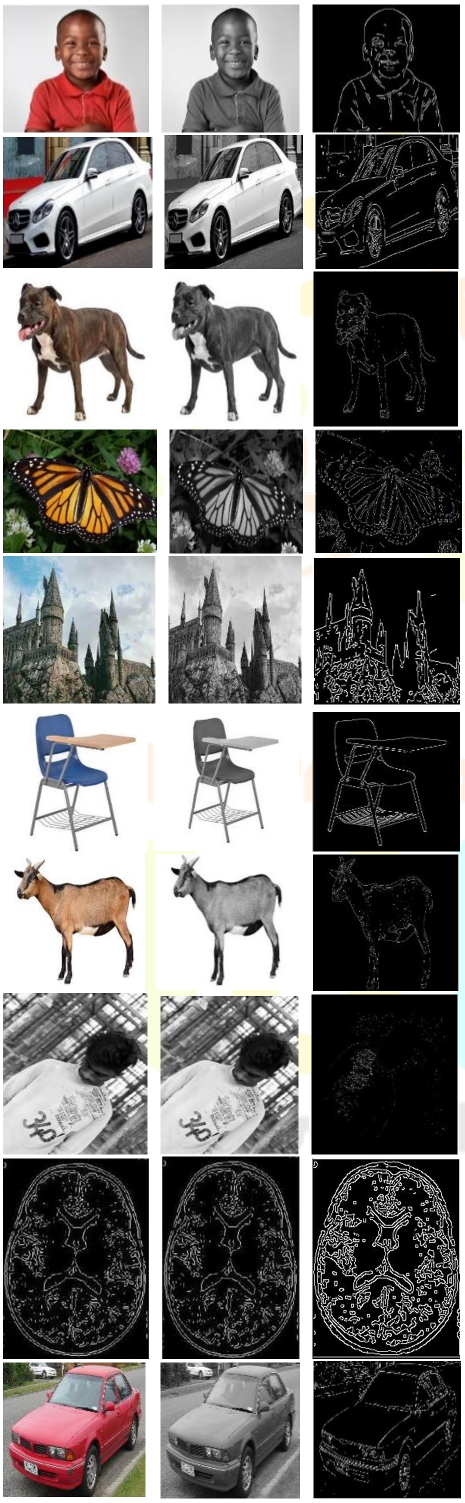
Best choice of solution: It showed correct range checking. One of the three final intensities for edge detection is a specific threshold that falls within the optimal band of final shortlisted intensities. Contrary to the above case, if the individual thresholds do not fall within the optimal range, the general threshold will remain in the good threshold position. From the above situation, it is similar to scanning pixels in the horizontal direction and determining three thresholds to detect edges. The resulting edge image is also detected.

Step6:

Step 5 should be repeated in the vertical direction.

RESULTS:

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.



(a) (b) (c)

Figure2: illustrates (a) the input image (b) the grayscale image (c) our output image

Figure 2 shows the application of the proposed B-edge algorithm to the image and the output of method (c). Figure 2 shows (a) the current image selected for manipulation, (b) the grayscale input image, and (c) the result of the proposed B-edge detection method. The proposed method effectively dealt with noise, edge connectivity, and thickness.

IV.PERFORMANCE METRICS

1. Mean Square Error(MSE):

MSE determines the difference in pixel values between a reference image and a proposed edge detection image. MSE can be calculated as:

$$\text{MSE}=\frac{1}{mn}\sum_{i=0}^{m}\sum_{j=0}^{n}([g(i,j)-f(i,j)])^2 \qquad (5)$$

Where m and n are used to denote the total number of rows and columns in the image, and g and f are used to denote the input and output images, respectively. If the mean squared error is minimal, the result is good.
The calculated MSE values are shown in the table.

2. Peak Signal to Noise Ratio(PSNR):

PSNR measures the similarity between a reference and a proposed image result.

$$\text{PSNR}=20log_{10}\left[\frac{[L]^2}{\frac{1}{mn}\sum_{i=0}^{m}\sum_{j=0}^{n}([g(i,j)-f(i,j)])^2}\right] \qquad (6)$$

Here, the number of gray levels in the image is indicated by the letter L.
The obtained PSNR values are shown in the table.

TABLE1: Parametric evaluation of selected images

Images	MSE of the outcome	PSNR of the outcome
Image1	3.229E+04	3.0729
Image2	2.128E+04	4.883
Image3	5.14E+04	1.0546
Image4	7.276E+04	9.5455
Image5	2.661E+04	3.9131
Image6	5.209E+04	0.9965
Image7	5.262E+04	0.9528
Image8	4.152E+04	1.9817
Image9	1.135E+04	7.6119
Image10	1.722E+04	5.8042

V.CONCLUSION

The proposed method produces effective edge detection with higher edge connectivity and thickness. This proposed methodology can be used in both normal and medical images. The study also provides performance metrics such as mean squared error (MSE) and peak signal-to-noise ratio (PSNR). From our findings, we can conclude that the presented method can effectively find valid edges.

VI.FUTURE SCOPE

We can improve the proposed B-edge algorithm as:
This algorithm will not effectively work on blur images so we can improve in obtaining the results with blur images. The computation time can also be improved in further implementations.

VII.REFERENCES

[1] MAMTA MITTAL1, AMIT VERMA, IQBALDEEP KAUR, BHAVNEET KAUR, MEENAKSHI SHARMA, LALIT MOHAN GOYAL, SUDIPTA ROY, AND TAI-HOON KIM, "An Efficient Edge Detection Approach to Provide Better Edge Connectivity for Image Analysis", IEEE transaction published on 13 March.

[2] S.R. Dixit; A.Y. Deshmukh "Design Strategies for EDGE Based Image Representation System", Published in 2013 6th International conference on emerging trends in Engineering and Technology.

[3] J.J. Lim, C.L. Zitnick and P. Dollar, "Sketch tokens: A learned mid-level representation for contour and object detection," in Proc. IEEE Conf. Comput Vis. Pattern Recognit, Jun. 2013, pp. 3158–3165.

[4] Weibin Rong, Zhanjing Li, Wei Zhang, and Lining Sun, "An improved CANNY edge detection algorithm," in Proc. IEEE Int. Conf. Mechatron Automat., 2014, vol. 2, no. 2, pp. 577–582.

[5] L. Xuan and Z. Hong, "An improved CANNY edge detection algorithm," in Proc. 8th IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS), Nov. 2017, pp. 275–278.

[6] Kun Zhang, Yuming Zhang, Pu Wang, Ye Tian, and Jun Yang, "An improved Sobel edge algorithm and FPGA implementation," in Proc. 8th Int. Congr. Inf. Commun. Technol., 2018, pp. 243–248.