Edge Detection Using Deep Learning

Pooja A S[1] and Smitha Vas P[2]

[1]*Department of Computer Science & Engineering, LBS Institute of Technology for Women, Poojappura, Thiruvananthapuram*
[2]*Department of Computer Science & Engineering, LBS Institute of Technology for Women, Poojappura, Thiruvananthapuram*

---***---

Abstract - *The paper describes the implementation of new deep learning based edge detection in image processing applications. A set of processes which aim at identifying points in an image at which the image brightness changes formally or sharply is called edge detection. The main theme of this work is to understand how efficiency is the new proposed method. The algorithm which increases the efficiency and performance is fast, accurate and simple. The traditional method shows canny edge detector has better performance and it finds more edges. The aim of this article is to find acute and precise edges that are not possible with the pre-existing techniques and to find the misleading edges, here use a deep learning method by combining it with other types of detection techniques. Here canny algorithm is used for making a comparison with the proposed technique with the MATLAB 2017b version. And also improve the canny algorithm by using the filter for reducing the noise in the image. Here uses a deep learning method to find more edge pixels by using the pre-trained neural network.*

Key Words*: *Edge Detection, Image Processing, Canny Edge Detection, Deep Learning, CNN

1. INTRODUCTION

The early stage of vision processing identify the features of images are relevant to estimating the structure and properties of objects in a scene. Edges are one such feature. Edge detection is an image processing technique used to locate and identify the sharp discontinuities in an image. Edge detection is frequently the first step in recovering information from images. One of the most classical studying projects of computer vision and image processing field is edge detection [1]. There are many edge detection operators available, which identifies vertical, horizontal, corner and step edges. The quality of edges detected by these operators is highly dependent on noise, lighting conditions, objects of the same intensities and the density of edges in the scene. This problem can be solved by changing the threshold values and by adjusting the various parameters. A wide range of operator is available for extracting edges from noisy images. But the edges extracting from noisy images are less accurate due to the presence of noise that extracts false edges. The Edge detection steps as follows, 1) Filtering: Images are often corrupted by random variations in intensity values, called noise. 2) Enhancement: it is essential to determine changes in intensity in the neighborhood of a point. 3) Detection: The image point which have a nonzero value for the gradient, and not all this points are edges. The result is a poor localization of edges [1]. Discontinuities in the image intensity can be either step discontinuities or line discontinuities. The early stages of edge detection algorithm give the result as the canny algorithm performs the most and it detects more edges when compared with other techniques. But finding the misleading edges is too difficult. Implementing the new deep learning based method should improve the efficiency and helps in finding the false edges also and to improve the noise reduction by using a filter. Here propose a deep learning method for finding the misleading edges and comparing the result with traditional algorithm. It performs the canny edge detection by improving the efficiency. Compared with common edge detection algorithm, the canny algorithm has the best performance [3]. This chapter introduces the canny edge algorithm for comparison with deep learning based detection of edges.

2. EDGE DETECTION

Image segmentation is one of the fundamental tools for edge detection. Edge detection methods that transform the original images into edge images benefit from the changes of grey color in the image [2] [1]. In image processing, the edge detection treats the localization of important variations of a gray level image and the detection of the physical and geometrical properties of objects of the scene. The three different types of discontinuities in the grey level like point, line, and edges. Spatial masks can be used to detect all the three types of discontinuities in an image [2]. The coordinates of an edge point may be the integer row and column indices of the pixel where the edge was detected, or the coordinates of the edge location at sub-pixel resolution. An edge fragment may be conceptualized as a small line segment about the size of a pixel, or as a point with an orientation attribute. The edge set

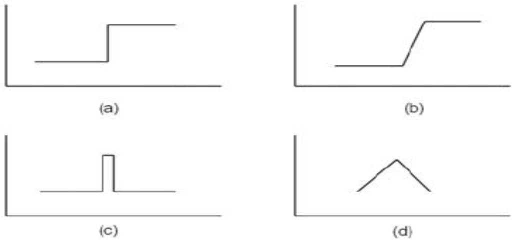

Figure 1: Types of Edges (a) Step Edge (b) Ramp Edge (c) Line Edge (d) Roof Edge

Produced by an edge detector can be partitioned into two subsets: correct edges, which correspond to edges in the scene, and false edges, which do not correspond to edges in the scene. The third set of edges can be defined as those edges in the scene that should have been detected. This is the set of missing edges. The false edges are called false positives, and the missing edges are called false negatives. The difference between edge linking and edge following is that edge linking takes as input an unordered set of edges produced by an edge detector and forms an ordered list of edges. Edge following takes as input an image and produces an ordered list of edges. Edge detection uses local information to decide if a pixel is an edge or not.

3. EDGE DETECTION METHODOLOGIES

The below figure shows the different types of edge detection techniques. Each method detects the edges in an image. The Robert cross method performs simple and quick to compute. The input which is given in this method is a grayscale image as same the output produced [2]. The second method is Prewitt algorithm, which is used to estimate the magnitude and orientation of an edge. A quite time-consuming calculation is needed to estimate the direction from the magnitudes in the x andy directions [2].

Prewitt edge detection is simpler than Sobel operator, but it produces more noisy images. The Sobel operator is a directional differential operator on a basis of odd size template. In this algorithm, the edge of the image is located at the place in which the brightness changes significantly, therefore in the neighborhood of the pixel gray value of pixels exceeds a set threshold depending on the specific steps for the edge [2]. The Laplacian of Gaussian (LoG) smoothes and computes the Laplacian, which yields a double edge image. Locating edges then consists of finding the zero crossings between the double edges [2].

4. CANNY ALGORITHM

The edge detection technique which gives the clear response, good localization and gives good detection. In practical applications, the results are not satisfactory and a problem with edge blur, the weak edge loss, the whole edge discontinuity etc. The three criteria for edge detection are proposed, which is expressed in the form of mathematics. The characteristics of a canny algorithm: The performance of edge detection is good, positioning performance is good, Only single edge response.

4.1. Steps

Here perform an image preprocessing and removal of noise. First, we need to select the output image, which may be the color image or grayscale image. If it is a color image convert it into the grayscale image. After taking the image, it needs to perform smoothing by reducing the noise. The noise which contained in the image input I(i,j) with a median filter. Here Euclidean distance is used to enhance the edge of the image. For that, a filter is made, which consider both vertical and horizontal axis direction. The filter first determines the size of the filter template according to the square distance of the template from the center pixel from small to large.

A 3 *3 template is used which is then divided into horizontal and vertical axis direction. The filter is calculated by;

$$F(x,y)=\sqrt{{f_{x,y}}^2(x,y)}+\sqrt{{f_{45^\circ 135^\circ}}^2(x,y)}$$

After the noise removal is performed, next step is to find the gradient in an image. The reason why detect the image gradient because they occur a drastic change at grayscale of an edge of an image. The algorithm usually select a 2*2 neighboring area to get the amplitude and direction in both vertical and horizontal.

$$M(x,y)=\sqrt{k_x^2}(x,y)+\sqrt{k_y^2}(x,y)$$

Where M(x,y) stands for the image gradient magnitude and D(x,y)stands for the image gradient direction. Then it is necessary to use an algorithm to find the high and low threshold values. Here we set high threshold T_h and set $T_l = T_h$ /2 as a low threshold.

D(x,y)=arctan[k_y(x,y)/k_x(x,y)]

The algorithm is used to divide the gradient amplitude and it is easy to ignore the local weak gradient amplitude, which produces a large deviation, resulting in poor edge image continuity.

$$T_h = \frac{1}{2*n} \sum_{i=0}^{n} ti$$

In this section, we compare the performance of traditional and proposed method with the standard Lena image with the pixel size of 512*512. It performs the proposed algorithm to find the maximum edges. After performs the thresholding, the thinning of the output image is also done. Thinning is performed in order to get the sharp edge lines.

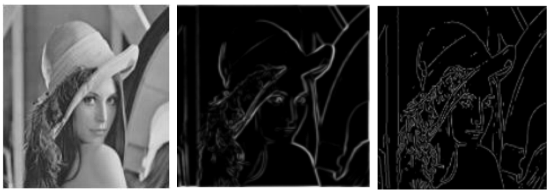

Figure 2: The image shows original, Threshold and Thinning Image

The above figure 2 gives the threshold image. Forgetting the threshold value, we set if the gradient is above, then it should be an edge pixel and if the gradient id below, then it should be a non-edge pixel. Thus from the evaluation, we find that the improved algorithm gives more edges than the traditional methods.

5. CONVOLUTIONAL NEURAL NETWORK

The new area of machine learning that has been introduced with the moving object close to the original one, ie Artificial Intelligence. The another benefit of deep learning models is the ability to perform automatic feature extraction from data. Here we use a pixel-based edge detection, In image processing, we can write an image as a set of pixels and an edge detection filter which detects edges with direction φ as a matrix with elements $W_{n,m}$. Then we can determine whether the pixel $f_{p,q}$, is an edge pixel or not, with the help of neighborhood pixels. The neighborhood has the same size as the edge detector template,(2N +1)×(2M +1). Then calculate the convolution. In order to reduce the computation time, the convolution layers are interspersed with a sub-sampling layer.The input receive images and each unit in a layer receives input from a set of units located in a small neighborhood. With local receptive fields, neurons can extract elementary visual features such as oriented edges, end points, corners. This features are then combined by the subsequent layers in order to detect higher-order features.

The training of Convolutional Neural Networks is very similar to the training of other types of NNs. Here uses the pre-trained model as VGG16 with great depth, great density, and multiple stages. The fine-tuning deep neural networks pre-trained on the general image classification task is useful to the low-level edge detection task that connect our side output layer to the last convolutional layer in each stage as conv1 2, conv2 2, conv3 3, conv4 3, conv5 3. The receptive field of each of the layers is identical to the side-output layer, then the last stage of VGGNet, including the 5th pooling layer and all the fully connected layers.

6. IMPLEMENTATION

This section describes how to prepare the training and validation data set for training a CNN model. Take the BSDS500 dataset (Berkeley Segmentation Dataset) as an example. Each image has hand labeled ground truth edges. For each image, several people were asked to draw a contour map separating different objects based on their own understanding. BSDS500 images are divided into 3 subsets, with 200 for training, 100 for validation and 200 for testing. Before training, prepare image patches and corresponding ground truth that can be acceptable by our neural networks. First, we could apply some pre-processing techniques to remove noise from the original images. For each image in BSDS500, there are multiple corresponding edge maps. In edge detection, however, we are limited by the number of training images available in the existing benchmarks.

During training, 17 patterns including 8 patterns for "edge" and 9 patterns for "non-edge" are randomly selected. The training process passes to the stages according to training epoch's number to reach the weight values. The epoch's number value ranges from 100 epoch to 100000 epoch as a maximum weight values. The epoch's number value ranges from 100 epoch to 100000 epoch as a maximum number performed. The below figure 10 shows the edge features from the 1st convolution layer. During training first connect the side output layer to the last convolutional layer in each layers conv1_2, conv2_ 2, conv3_3, conv4_3, conv5_3. Here the pooling layers and fully connected layers are not used.

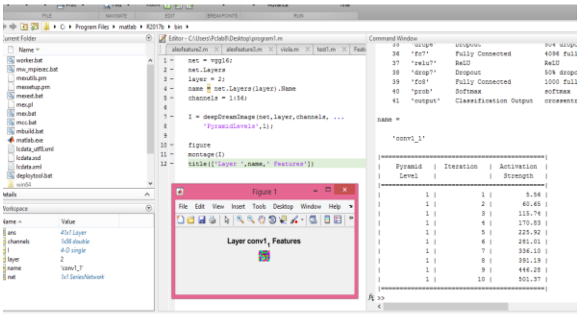

Figure 3: Edge Features in 1st convolution layer

The PSNR (peak signal-to-noise ratio) is used to evaluate the network output during raising the epoch's number. The below figure shows the changes of the edge detected output image of the proposed technique. And shows the changes of the noise ratio in the output edge detected image when applied to the proposed system during increasing the training epochs number from 400 to 100000 epoch, a significant change is occurred when raised the epoch number to its maximum value.

Figure 4: Comparison of proposed method with canny

7. CONCLUSION

In this paper, the traditional edge detection techniques are discussed. The performance of various edge detection techniques is carried out with an image by using MATLAB software. From that, we find canny edge detection gives a good result because this algorithm is sensitive to noise but computationally very expensive when compared with traditional ones. An improved canny algorithm is performed as a proposed method. The proposed method uses median filtering that replaces the traditional method for denoising. The thresholding gives the gradient value with an edge and non-edge pixels also. The accuracy of the edge detection technique can be performed by deep learning technique for getting more true and false edges. This can be done with the help of pre-trained model. From that we get this method to detect more edges and improve the efficiency also.

REFERENCES

[1] Lei Zhai, Shouping Dong, Honglian Marie, "Recent Methods and Applications on Image Edge Detection", *2008 International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing.*

[2] Muthukrishnan.R and M.Radha, "Edge Detection Techniques for Image Segmentation", *International Journal of Computer Science & Information Technology (IJCSIT)* Vol 3, No 6, Dec 2011

[3] Yingke Feng, Jinmin Zhang, and Siming Wang, "A new edge detection algorithm based on Canny idea" *2nd International Conference on Materials Science, Resource and Environmental Engineering (MSREE 2017)*

[4] Otsu N.A. Threshold Selection Method from gray level Histograms, IEEE Trans on Systems. Man and Cybernetics, SMC - 9,1979,9(1):62-66.

[5] A. Bansal, X. Chen, B. Russell, A. Gupta, and D. Ramanan"pixelnet: representation *of* the pixels, *by* the pixels, and *for* the pixels"(2017).

[6] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *TPAMI*, 2011.

[7] Yuqin Yao , Hui Ju," A Sub-pixel edge detection method based on canny operator", 2009 *Sixth International Conference on Fuzzy Systems and Knowledge Discovery.*

[8] Chinu and Amit Chhabra, "Overview and Comparative Analysis of Edge Detection Techniques in Digital Image Processing", *International Journal of Information & Computation Technology*. ISSN 0974-2239 Volume 4, Number 10 (2014), pp. 973-980

[9] SF Ghoreishi, et. al, "Adaptive Uncertainty Propagation for Coupled Multidisciplinary Systems", AIAA Journal (2017).

[10] Shuilian Xie., et. al. ,Nonstationary Linear Discriminant Analysis, Proceedings of the 51st Asilomar Conference on Signals, Systems, and Computers, At Pacific Grove, CA, 2018.

[11] Beant Kaur , Anil Garg," Mathematical Morphological Edge Detection For Remote Sensing Images",IEEE,2011,pp: 324-327.

[12] Syed Jahanzeb Hussain Pirzada,Ayesha Siddiqui ," Analysis of EdgeDetection Algorithms for Feature Extraction in Satellite Images", *Proceeding of the IEEE International Conference on Space Science and Communication (IconSpace*), July 2013.

[13] Jaspreet Kaur Anand Sharma" Review Paper on Edge Detection Techniques in Digital Image Processing", *International Journal of Innovations & Advancement in Computer Science IJIACS* ISSN 2347 – 8616 Volume 5, Issue 11 November 2016

[14] Gao W., Yang L., Zhang X. and Liu H. 2010. An Improved Sobel Edge Detection. *IEEE International conference on computer science and information technology (ICCSIT),* vol. 5, 67-71.

[15] Joshi, Shashidhar Ram, and Roshan Koju. "Study and comparison of edge detection algorithms." *2012 Third Asian Himalayas International Conference on Internet. IEEE, 2012*

[16] Norah Saad Al Harbi. Supervisor: Dr. Msater Prince Syed." A novel approach to gradient edge detection", Volume 5, No. 10, October 2014 Journal of Global Research in Computer Science

[17] Canny. J, "A computational approach to edge detection", *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol 8, pp. 679-714, 1986

[18] Shokhan M. H" An efficient approach for improving canny edge detection algorithm", *International Journal of Advances in Engineering & Technology, Mar. 2014*

[19] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR,* abs/1606.00915, 2016. 7, 10, 11

[20] R. Garg, B. G. V. Kumar, G. Carneiro, and I. D. Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. In *ECCV*, pages 740–756, 2016.

[21] P. Krahenb ¨uhl and V. Koltun. Efficient inference in fully ¨ connected crfs with gaussian edge potentials. In *NIPS*, 2011. 2

[22] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015. 2

[23] S. Xie and Z. Tu. Holistically-nested edge detection. In *ICCV*, 2015. 1, 2, 8, 12, 13, 14

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *In NIPS*, 2012.

[25] Qian Xu, Srinivas V J, Chaitali C. A distributed Canny edge detector: algorithm and FPGA implementation [J]. *IEEE Transactions on Image Processing*, 2014, 23(7): 2944-2960

BIOGRAPHIES

"Pooja A S received her B.Tech degree in Computer Science and Engineering from Anna University. She is now pursuing her M.Tech degree in Computer Science and Engineering from LBS Institute of Technology for Women, Thiruvananthapuram affiliated to APJ Abdul Kalam Technological University"

"

Smitha Vas P received her B.Tech degree in Computer Engineering from Cochin University of Science & Technology, Kerala and M.Tech degree in Computer Science & Engineering from the University of Kerala. Currently, she is working as an Assistant Professor in the Department of Computer Science and Engineering, LBS Institute of Technology for Women, Thiruvananthapuram affiliated to APJ Abdul Kalam Technological University. Her areas of interest are Image Processing and Machine Learning."