Volume 8, No. 5, May-June 2017
International Journal of Advanced Research in Computer Science
SURVEY REPORT
Available Online at www.ijarcs.info
ISSN No. 0976-5697

A Survey on Convolutional Neural Network based Edge Detection Techniques

Anuja J A
PG Student
Department of Computer Engineering
Govt. Model Engineering College
Thrikkakkara, Kerala, India

Jumana Nahas
Assistant Professor
Department of Computer Engineering
Govt. Model Engineering College
Thrikkakkara, Kerala, India

Abstract: Edge detection is the process of finding meaningful transitions in an image. Edge detection is a well developed area in its own in the field of Image Processing and Computer Vision. Various techniques are available for solving the non-trivial problem of edge detection. In this paper, the various convolutional neural networks based techniques for edge detection are explored.

Keywords: Image Processing, Edge detection, Deep neural networks, Deep learning, Convolutional neural networks

I. INTRODUCTION

Edge detection is the process of identifying presence and location of edges constituted by sharp changes in intensity or brightness in images [20]. It has many applications in the fields of object recognition, computer vision etc. The importance of edge detection was first emphasized by Hubel and Wiesel with their famous cat experiment in 1959, for which they were awarded the Nobel Prize in Physiology or Medicine. Since then many literatures have been published in the field of edge detection, upto the extent that now this field has been deemed stagnant.

The area of computer vision saw a breakthrough in 2012 where a deep convolutional neural net, won the 2012 ILSVRC. The ImageNet Classification with Deep Convolutional Neural Networks [1], achieved an error rate of 16%, and paved way for tackling many computer vision problems such as classification, localization, detection using deep convolutional neural networks. In this survey, the different edge detection techniques based on convolutional neural networks are discussed.

II. OVERVIEW

A. *Edge Detection Techniques*

Edge detection is the process of identifying presence and location of edges constituted by sharp changes in intensity or brightness in images. There are many factors which makes edge detection a difficult process. Variations in illumination and intensity, background clutter, poor image quality, difference in depth, surface orientation are a few of those factors which affects edge detection. An ideal edge prediction would be done at object level, and preferably with thinner edges.

Earlier edge detectors were gradient operators which predicts edges by taking the image gradient. Classical edge detection techniques such as Roberts, Prewitt and Sobel are first order gradient operators. Second derivative methods include Canny edge detector, Laplacian of Gaussian (LOG) filter etc.

Edge detection techniques maybe further divided into information theory based methods such as Statistical edges[13], and learning based methods such as Structured edges[14].

Edge detection is a preprocessing step to many higher level tasks such as segmentation, pattern recognition and classification. [12] discusses about soft computing edge detection approaches for segmentation. They are divided into fuzzy logic based approaches, approaches based on genetic algorithms and neural network based approaches [15].

Table I. Summary of edge detection techniques

Edge Detection Techniques	***Gradient operators***	First order derivative methods
		Second order derivative methods
	Soft computing based approaches	Fuzzy logic based
		Genetic algorithm based
		Neural Network based
	Information theory based approaches	
	Learning based methods	

B. *Convolutional Neural Networks*

An artificial neural network or ANN is a computationL system modelled after biological neural network. It consists of nodes called neurons which performs some specified function on its inputs and produces a thresholded output. Based on its architecture can be classified into single layer neural networks and multi layer neural networks. A multi layer neural network will have layers in between the input layer and output which are termed hidden layers. MADALINE, multi layer perceptron, backpropogation network are examples of multi layer networks. Compared to single layer networks, multi layer networks can solve more complicated problems. More the number of hidden layers, better will be the results, but at the cost of training effort it takes.

Deep neural networks are networks with four or more hidden layers. Deep learning is a machine learning technique trending in the fields of pattern recognition, object recognition

,classification, computer vision etc. Deep learning is about learning multiple levels of representation and abstraction that help to make sense of data such as images, sound, and text. Alexnet[1], VGGNet[18], GoogLeNet[19] are some of the pioneering deep network architectures.

Convolutional neural networks are deep networks mostly used for image processing and computer vision tasks. They are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. They usually accept images as input. Each neuron in the input and hidden layer will be connected to a set of pixels in the input image. The number of pixels or the size of region in the input image to which a neuron is connected is called its receptive field. All the pixels in the receptive field of a neuron shares the weights and biases of the corresponding link. At each layer, convolution is done on pixels in the receptive field and a randomly initialized filter. Output obtained after convolution maybe upsampled or downsampled using pooling operations such as max pooling, min pooling, average pooling etc. Finally, a non-linearity such as SoftMax or ReLu (Rectified non-linear unit) is applied. The manner at which a network is trained can be controlled by specifying certain parameters such as learning rate, momentum and regularization. Momentum is used to decrease the learning rate after sometime into the training process. Regularization techniques are employed to prevent overfitting, in which a network performs well for the data using which it is trained but fails for unseen test images. A summary of the advances in convolutional neural networks is available in [16].

Table II. Components of Convolutional Neural Networks

Components of convolutional neural network	***Pooling operations***	Max pooling
		Min pooling
		Average pooling
		Stochastic pooling
	Activation functions	Identity function
		Sigmoid function
		Step function
		ReLu
		Leaky ReLu
	Loss functions	Softmax
		ReLu
	Regularization techniques	L1- Normalization
		L2-Normalization
		Dropout

III. Convolutional Neural Network Based Edge Detection Techniques

A. Holistically Nested Edge Detection

Holistically nested edge detection [2] is a technique which uses convolutional neural network for edge detection. It automatically learns features in a hierarchical fashion, as training progresses. From each layer of the network, an initial edge map is produced, which is fed to the next layer which in turn produces more refined edge predictions. The average of all the set of such predictions is the final edge map. The base architecture is VGG-16.

B. DeepEdge: A Multi-Scale Bifurcated Deep Network for Top-Down Contour Detection

Instead of using localized or low-level edge information for object detection, this technique [3] uses object-level information for predicting contours. It extracts candidate contour points from the input image by applying canny edge detection technique. Around each candidate point, four patches at four different scales are extracted and fed to first five convolutional layers of [1], to obtain object-level information. These convolutional layers are connected to two separately trained network branches. One is trained for classification and the other for regression. While testing, the scalar outputs from the branches are averaged to produce the final score.

C. DeepContour: A Deep Convolutional Feature Learned by Positive-sharing Loss for Contour Detection

Contour detection is the underlying process in object recognition and classification and other computer vision tasks. Available positive classes are divided into subclasses and model parameters are set for each subclass [6]. A positive-sharing loss function is proposed for better learning. Using this function the overall loss for a class is shared among all its subclasses.

D. Automated Edge Detection using Convolutional Neural Network

This technique [10] makes use of momentum feature extraction, for edge detection. Unlike previous techniques, it can process images of any size. The proposed architecture automatically learns filters, which is a spatially local pattern. Further operations are that of a basic convolutional neural network.

E. High-for-Low and Low-for-High: Efficient Boundary Detection from Deep Object Features and its Applications to High-Level Vision

Like [3], it [4] uses object-level features to predict edges. It reuses the features from VGGnet [18] for boundary detection, semantic segmentation and classification. It uses structured edges method to extract a set of candidate contour points, which is fed to VGGNet after upsampling. For each edge map obtained from the convolutional layers of the network, feature interpolation is applied to obtain the feature descriptor of the candidate points. These feature vectors are then fed to two fully connected layers which is trained to predict if an edge is present in a pixel or not. The predictions from these fully connected layers is accumulated to give the final edge map

F. Pixel-wise Deep Learning for Contour Detection

This technique [5] tackles the problem of per-pixel classification for contour detection. Like [3], it resuses features from [1] for efficient per-pixel classification. First the input image is fed to DenseNet [17] which produces multiscale versions of the image and stitches it to a larger scale. After processing the whole plane by CNNs, DenseNet would unstitch the descriptor planes and then obtain multiresolution CNN descriptors. It also uses an SVM classifier for further contour detection

G. Edge Detection Using Convolutional Neural Network

The proposed technique [7] is pretty straightforward. A three layer convolutional neural network is trained with images after noise removal. During testing, post processing steps like non maximal suppression and morphological operations maybe applied on the output to get thinner edge maps.

H. Using Deep Convolutional Networks for Occlusion Edge Detection in RGB-D Frames

An occluded edge is an edge masked by other edges, and is very difficult to distinguish even by naked eyes. Identifying occluded edges is important and a non trivial task in any classification, object recognition and computer vision applications. The proposed technique [9] uses a three pairs of convolution-pooling layers followed by softmax non-linearity. For training the network, the images are divided into patches of size 32x32 where each is patch is labelled as occluded or non occluded patch. Further postprocessing such as extrapolation using Gaussian distribution with Full Width at Half Maximum (FWHM) is done during testing. Gaussian kernels from overlapping patches are fused in a mixture model to generate smooth occlusion edges in the testing frame.

I. Convolutional Neural Network based Edge Detector Learned via Contrast Sensitivity Function

Contrast sensitivity is a parameter used to assess overall vision. Contrast sensitivity function (CSF) is used in visual psychology to evaluate whether an edge is visible to human visual system. The proposed network [8] is trained using synthesized images which were labelled using CSF. This convent-CSF model is more robust to contrast variation.

J. Semantic Image Segmentation with Task-Specific Edge Detection using CNNs and a Discriminatively Trained Domain Transform

The proposed technique [11] focuses on performing semantic image segmentation with the help of available edge information. They propose a network called EdgeNet which predict edges with the help of features extracted from intermediate layers of DeepLab[21]. The final edge prediction is obtained by applying domain transform [22] on the output from Edgenet. Domain transform is an edge-preserving filtering method in which the amount of smoothing is controlled by a referenced edge map.

IV. REFERENCES

[1] Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. “Imagenet classification with deep convolutional neural networks.” In Advances in neural information processing systems, pp. 1097–1105, 2012.

[2] S.Xie and Z.Tu. “Holistically-nested edge detection.” In ICCV, 2015.

[3] G. Bertasius, J. Shi, and L. Torresani. “Deepedge: A multi-scale bifurcated deep network for top-down contour detection.” In CVPR, 2015.

[4] G. Bertasius, J. Shi, and L. Torresani. “High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision.” In ICCV, 2015.

[5] J.-J. Hwang and T.-L. Liu. “Pixel-wise deep learning for contour detection.” In ICLR, 2015.

[6] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang. “Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection.” In CVPR, 2015

[7] Wang R. (2016) “Edge Detection Using Convolutional Neural Network.” In: Cheng L., Liu Q., Ronzhin A. (eds) Advances in Neural Networks – ISNN 2016. ISNN 2016. Lecture Notes in Computer Science, vol 9719. Springer, Cham

[8] Dou H., Liu W., Zhang J., Wu X. (2015) “Convolutional Networks Based Edge Detector Learned via Contrast Sensitivity Function.” In: Arik S., Huang T., Lai W., Liu Q. (eds) Neural Information Processing. Lecture Notes in Computer Science, vol 9489. Springer, Cham

[9] S. Sarkar, V. Venugopalan, K. Reddy, M. Giering, J. Ryde, and N. Jaitly, “Occlusion edge detection in rgb-d frames using deep convolutional networks,” Proceedings of IEEE High Performance Extreme Computing Conference, Waltham, MA, 2015.

[10] Mohamed A. El-Sayed, Yarob Istitieh, Mohamed A. Khafagy: “Automated Edge Detection Using Convolutional Neural Network. International Journal Of Advanced Computer Science And Applications” (IJACSA. 4(10): 11-17 (2013)

[11] Liang-Chieh Chen, Jonathan T. Barron, George Papandreou, Kevin Murphy, and Alan L. Yuille. 2015. “Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform.” CoRR, abs/1511.03328.

[12] Senthilkumaran, N., and R. Rajesh. "Edge detection techniques for image segmentation–a survey of soft computing approaches." *International journal of recent trends in engineering* 1.2 (2009): 250-254.

[13] S. Konishi, A. L. Yuille, J. M. Coughlan, and S. C. Zhu, “Statistical edge detection: Learning and evaluating edge cues,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 25, no. 1, pp. 57–74, 2003

[14] Dollár, Piotr, and C. Lawrence Zitnick. "Fast edge detection using structured forests." *IEEE transactions on pattern analysis and machine intelligence* 37.8 (2015): 1558-1570.

[15] Pinho A.J., Almeida L.B. (1995) “Edge detection filters based on artificial neural networks.” In: Braccini C., DeFloriani L., Vernazza G. (eds) Image Analysis and Processing. ICIAP 1995. Lecture Notes in Computer Science, vol 974. Springer, Berlin, Heidelberg

[16] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, and G. Wang, “Recent advances in convolutional neural networks,” arXiv preprint arXiv:1512.07108, 2015

[17] Iandola, Forrest, et al. "Densenet: Implementing efficient convnet descriptor pyramids." *arXiv preprint arXiv:1404.1869* (2014).

[18] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

[19] Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

[20] R. C. Gonzalez and R. E. Woods. “Digital Image Processing”. 2nd ed. Prentice Hall, 2002

[21] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In ICLR, 2015

[22] E. S. L. Gastal and M. M. Oliveira. Domain transform for edge-aware image and video processing. In SIGGRAPH, 2011.