International Journal on Applications in Electrical and Electronics Engineering
Volume 4: Issue 2: April 2018 , pp 1- 7 . www.aetsjournal.com ISSN (Online) : 2395 - 3527

NEURON SEGMENTATION USING CONVOLUTION NEURAL NETWORK (CNN)

S. ARTHI , B. DHARANI , K. DURGADEVI , K. GAYATHRI , K. NANDHITHA

***Abstract*—** We proposed the paper to the problem of neuron segmentation in image volume acquired by Electron Microscopy. Neuron segmentation in a 3 dimensional volume is conducted by associating the corresponding neuron region in each image. Experiments of real world datasets demonstrate that our paper out performs neuron segmentation based on graph based semi supervised learning, the supervised CNN and variants of semi supervised CNN. We presented frame work by using a convolution neural network (CNN) with a semi supervised regularization term. It can reduce the human efforts in without affecting the performance.

***Keywords*—** Segmentation, Electron microscopy, CNN, Graph based semi supervised learning.

I. INTRODUCTION

Graph based semi supervised learning provides a helpful information in modeling of data in the problem of classification. During this scenario, relation between labeled and unlabeled information impact the efficiency. It uses the graph to represents the information. It can be designed using domain knowledge. There are mainly two specific methods which are used to connect each data point to its neighbor. The neighbor should be present within a particular design. The graph is created based on input value similarity.

Semi supervised learning is a type of supervised learning. It makes use of unlabeled information for training a small amount of labeled information with large amount of unlabeled information. It exists between supervised learning and unsupervised learning. Supervised l Learning includes labeled training information. Unsupervised learning includes no labeled training information. Semi supervised learning also refers to inductive learning.

S. Arthi , Department of ECE, V.S.B Engineering College, Karur.
B. Dharani , Department of ECE, V.S.B Engineering College, Karur.
K. Durgadevi , Department of ECE, V.S.B Engineering College, Karur.
K. Gayathri , Department of ECE, V.S.B Engineering College, Karur.
K. Nandhitha , Department of ECE, V.S.B Engineering College, Karur.

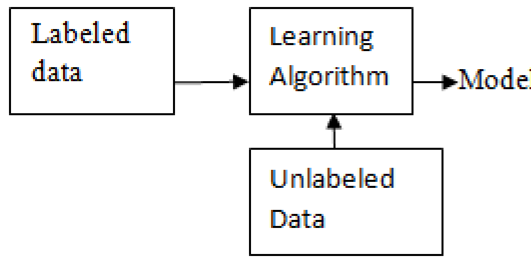


Figure 1 : Semi supervised learning

Supervised learning is type of machine learning. Machine learning is a type of learning in which both inputs and outputs are already given input and output data are labeled for future processing.

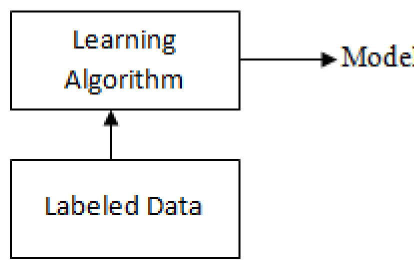


Figure 2: Supervised learning

Unsupervised learning is a type of machine learning. It consists of input label without labeled output. The most common type of this method is cluster analysis. Cluster analyzing is the task of grouping the set of object in which object belongs to same group is called cluster. This algorithm performs the difficult task easily than the supervised algorithm. It is used in bio-informatics for sequence analysis and genetic clustering and data mining and pattern mining in medical imaging for image segmentation and in computer vision for object recognition. This type of learning is more unpredictable than the supervised learning. Deep learning algorithm is used to unsupervised learning tasks. This is an important benefit because unlabeled data are more abundant than labeled data. For example the deep structure that can be trained in an unsupervised manner.

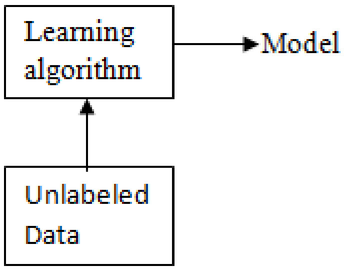


Figure 3 : Unsupervised learning

II. RELATED WORK

Segmentation of neurons has proven to be the most challenging part of the EM image analysis automation and over the years multiple groups have proposed different approaches to solve the problem. Manual segmentation has clear and well understood limitations for the analysis of EM stacks. One notable example can be found in [12], where the successful segmentation of the nervous system of a nematode worm, containing only 302 neurons, necessitated a sustained effort over a 10 year period. The need for expert knowledge and the growing size of EM data set render manual segmentation impractical and highlight the need for automation. Semi automated methods based on active contours and level sets [4], [13], [7] as well as graph cuts [8] have achieve some measure of success on EM images.

For the neuron segmentation problem, motivation for the inclusion of semantic labeling is a in difficult cases, where the edge evidence given by membrane detection is in conclusive, semantic labels provide additional clues that support merge decision of neighboring nodes.

For information with isotropic resolution, this advantage is inherently present and most segmentation algorithm operates directly in 3D. The step wise segmentation approach, first proposed by [25], is currently used by many other 3D segmentation pipelines. While slice-by-slice methods have been shown to provide both reasonable segmentation results and computational savings, they fail to leverage the consistency of the structures in all 3 dimensions.

Nearest to our work state-of-the-art method of Kreshuk. A [20]. The initial supervoxel over-segmentation introduces structure and drastically reduces the problem size, thus allowing more advanced algorithms to be used. The method of [19] is still, however, limited to the segmentations contained in the merge-tree. Computationally, we rely on convolution neural network algorithm. This extends the neuron segmentation by a simultaneous semantic node labeling.

Our work needs synapse detections, which provide important ideas about the biological priors.

III. METHODS

The key part of our pipeline is the convolution neural network (CNN) segmentation algorithm is a class of deep, feed-forward artificial neural networks that has successfully applied to analyzing visual imaginary. The convolution neural network use relatively pre-processing compared to other image classifications algorithms. This means that the network learns the filters that in traditional algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major disadvantage. The feed forward architecture of convolution neural network was extended in the neural abstraction pyramid[18] by lateral feedback connections. Additionally, we have to think in a different way of the input in terms of 3D volume has a depth associated with a normal image. This is also called the number of channels. For examples an RGB of has 3 channels so the full shape is really. On the other hand, a grayscale image of the same size only has one value for pixel locations. Since these are technically, they are sometimes called image volumes. The layers of CNN have neurons arranged in 3 dimensions which include width, height and depth. The neurons inside a layer are attached to a small place of the layer before it, called as receptive field. It includes 3 layers they are,

1. Convolution layer 2. Pooling layer 3. Fully connected layer.

1) Convolution layer is the basic building block of convolution neural network. During the forward

pass, each filter is convolved across the width and height of the input volume. As a result, the network learns filters that activate when it detects some specific type of features at some position in the input. The output volume of the convolution layer is calculated by an equation W×H×F. The rectified linear unit or RELU is most frequently used for CNNs. The function is defined by the following function is defined by the following equation.

$$F(x) = \max(0,x)$$

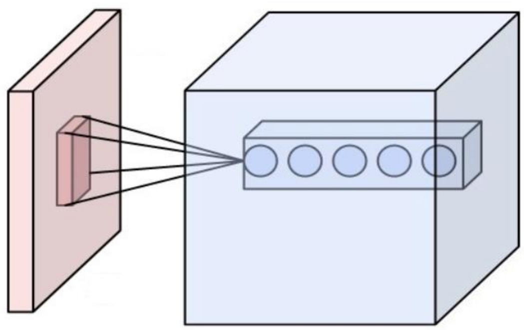


Figure 4 : Neurons of a convolution layer (blue), connected to their receptive field(red)

2) Pooling layer is a form of non linear down sampling. It separates the input image into a set of non overlapping rectangles and, for each sub region, output the maximum. The pooling layer serves to progressively minimize the size of the representation, to reduce the number of parameters and also to control over fitting. There are several non linear function to introduce pooling among which max pooling is the most important.

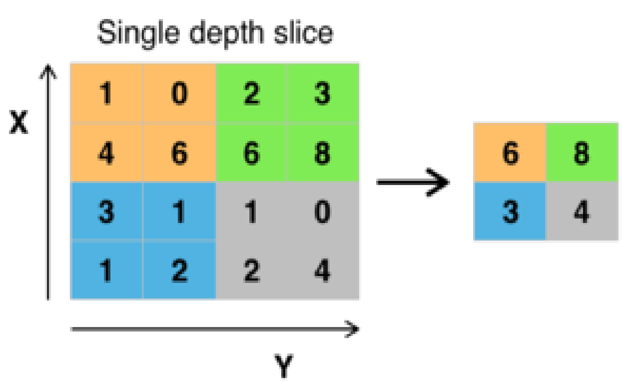


Figure 5 : Max pooling with 2×2 filter and stride=2

The pooling layer operates independently on every depth slice of the input and resizes it spatially. The most important thing is pooling layer with size 2×2 applied with a stride of down samples at every depth slice in the input by 2 along both width and height. In addition to the max pooling the pooling units can use other functions, such as average pooling or L2-norm pooling. Average pooling was used historically but has recently fallen out of favor compared to max pooling, which is better in practice. Pooling is an important component of convolution neural networks for object detection applications.

3) *Fully connected layer* in this layer neurons in a fully connected layer have connections to all activations in the previous layer, as seen in regular neural networks. Their activations can hence be computed with a matrix multiplication followed by a bias offset. The amount of parallelism to keep pace with convolution layers would quickly saturate the FPGA off chip memory, therefore it is proposed that his stage of the input layer either batched or pruned. As fully connected layer is placed in the layer stages of a CNN network, many possible features have already been eliminated. Therefore, pruning can significantly reduce the amount of work required.

While traditional multilayer perception models were successfully used for image recognition, due to the full connectivity between nodes the suffer from the curse of dimensionality, and thus do not scale well to higher resolution images. Convolution neural network are biologically inspired variants of multilayer perception designed to emulate the behavior of a visual cortex and the 3D volumes of neurons arranged in the 3 dimensions that is width, height and depth. The neurons inside a layer are connected to only a small region of the layer before it, called a receptive field. Distinct types of layers, both locally and completely connected, are stacked to form CNN architecture.

When training the neural network, there is additional layer called the loss layer. This layer provides feedback to the neural network on whether it identified inputs correctly. This helps to guide the neural network to reinforce the right concepts as it trains. This is always the last layer during the training.

The above is not directly applicable to our case. The inputs we will be dealing with are actually 28×28×1. But we can still perform a convolution so that the resulting volume is 28×28×6 by using some zero padding. We perform a convolution with 6 features maps to get a resulting activation volume. Then we pool to get 14×14×6 activation volume. We have halved the width and height so we use a pooling size of 2×2. The last convolution layer is 120 filters of size 5×5, which also flatter it, which usually in the case. Finally, we add a hidden layer with 84 neurons and the output layer with 10 neurons. The output layer neurons produce 10 members' numbers.

1. Edge Detection

Edge detection includes the variety of mathematical methods that identifying points in digital image at which the image brightness changes sharply has discontinuities. A problem finding discontinuities in one dimensional signals called step detection. The problem of finding signal discontinuities over time is called change detection. Edge detection is a fundamental tool in image processing. Edges obtained from non-trivial are often hampered by fragmentation meaning that the edge curves are not connected, missing edge segments. The edges extracted from a two-dimensional image of a three-dimensional scene can be divided into view point dependent or view point independent.

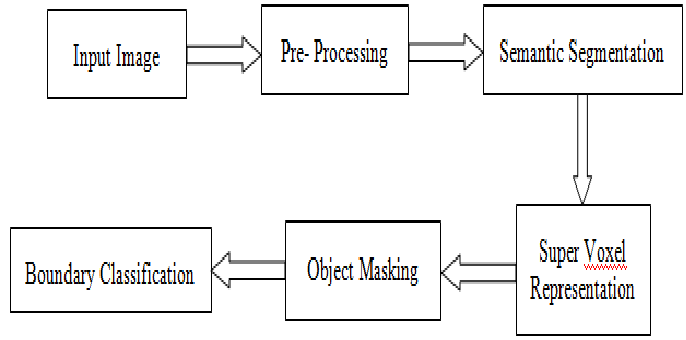
In cutting-plane approach, where violated constraints are added iteratively. The cut formulation in the edge domain relies on binary random variables, which can be joined with edges in the adjacent graph.

$$f(x) = \frac{I_r - I_l}{2}\left(erf\left(\frac{x}{\sqrt{2\sigma}}\right) + 1\right) + I_l$$

2. Semantic class affinities

In the setup of semantic segmentation, establishing the semantic class of every pixel is the end goal of the algorithm. The prediction of the classifier does not need to represent the semantic class affinity directly. Example: Neural segmentation.

IV. Block Diagram



V. Module Explanation

1) Input Image

The input image is obtained from Electron Microscopy). We present a parametric image model consisting of two levels set functions to represent the neuron images. Then we formulate an optimization problem merging image restoration and segmentation. Even after restoration, image segmentation must still be performed on the restored images for the purpose of image analysis in our method performs the restoration and segmentation simultaneously. So no other major segmentation operation is needed. Due to parametric representation both dendrites and somas can be segmented simultaneously.

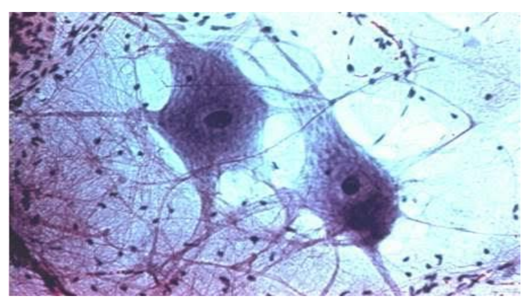


Figure 6 : Input Image

2) Pre Processing

We can simply merge the mitochondria segments with the neuron segments, with which they have the longest common boundary. However, occasionally the mitochondria are (wrongly) connected over the neuron boundary and we need to apply the merging stepwise. This strategy is as follows:

International Journal on Applications in Electrical and Electronics Engineering
Volume 4: Issue 2: April 2018 , pp 1- 7 . www.aetsjournal.com ISSN (Online) : 2395 - 3527

1. We find the *connector* supervoxels, which keep the two mitochondria joined. Then we remove the connectors and merge the remaining pieces by simple procedure above.
2. The connector themselves merged in the last step. The connectors are found by running a shortest path algorithm on the adjacency graph of the g-supervoxels for all pairs of nodes for each connected component of mitochondria class.
3. Each node in this adjacency graph counts how often it works used for crossing. The local maxima of this estimate are taken as connectors. If more than one connector is found in a mitochondria segment, the procedure is iterated until all connectors are merged. Using this approach, all mitochondria of our test block have been correctly merged with their containing neuron.

Figure 7 : pre-processing

3) Semantic Segmentation

For the neuron segmentation problem, the motivation for the inclusion of semantic labeling is that in difficult cases, where the edge evidence given by the membrane detection is inconclusive, semantic labels provide additional clues that support merge decisions of neighboring nodes.

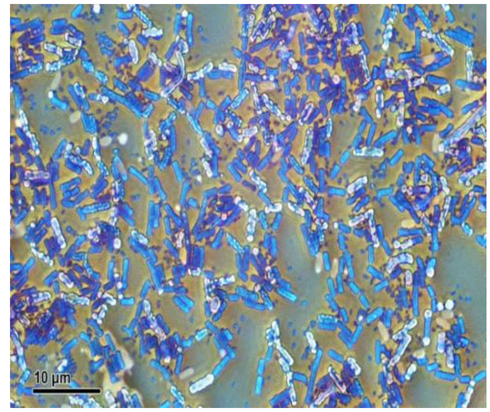


Figure 8 : Segmented Image

4) SuperVoxel Representation

The over segmentation of the image volume into super voxels is based on a pixel wise membrane probability map. This probability map is de-noised by non local mean algorithm. It is in image processing for the image de-noising. This result in much greater post filtering clarity and less noise of detail in the image compare with the local mean algorithm. Local minima on the smoothed map serve as seeds for a watershed algorithm. The key behind the watershed algorithm is to change our image into another image whose catchment basins are objects we want to identify. It mainly used in medical image segmentation. There are two approaches in watershed segmentation,

1. *Frontier approach*
2. *Region approach*

The supervoxels, resulting from the watershed algorithm, are termed W – Supervoxels.

The W-supervoxels are coarse enough to compute the first set of pair wise features and to train a Random Forest classifier for the respective affinities.

5) Object Masking

Image masking is the process of separating an image from its background, either to cause the image to stand out on its own or to place the image over another back ground.

Figure 9 : Object Masking

6) Boundary Classification

The pixels are classified into four spatial classes: core boundary, isolated region, corridor and branch. The rest of the object area, a pair of initial image which does not belong to internal contour will be referred to as core regions. Classes are detected in a step wise process. At each state one classes excluded from the internal boundary following the class hierarchy isolated regions pixel can be detected using the opening the reconstruction. That is reconstruction by dilation with the initial image used as a mask image. The core areas image i.e., the result of erosion of the original image is used as marker image.

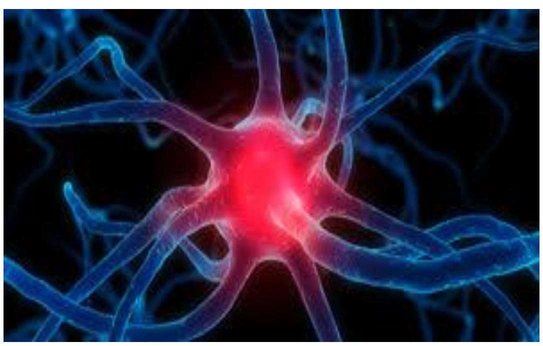


Figure 10 : Output Image

VI. Conclusion

We have introduced a principled approach to incorporate high-level biological priors into the neural network segmentation procedure [11]. While the convolution neural network is already superior to greedy approaches due to its globally optimal nature. We show that it can be further improved by considering sparse global priors of neuron type in addition to local boundary evidence.

References

[1] G. Knott, H. Marchman, D. Wall, and B. Lich, "Serial section scanning electron microscopy of adult brain tissue using focused ion beam milling.," The Journal of Neuroscience, 2008.

[2] S.-Y. Takemura, C. S. Xu, Z. Lu, P. K. Rivlin, T. Parag, D. J. Olbris, S. Plaza, T. Zhao, W. T. Katz, L. Umayam, C. Weaver, H. F. Hess, J. A. Horne, J. Nunez-Iglesias, R. Aniceto, L.-A. Chang, S. Lauchie, A. Nasca, O. Ogundeyi, C. Sigmund, S. Takemura, J. Tran, C. Langille, K. Le Lacheur, S. McLin, A. Shinomiya, D. B. Chklovskii, I. A. Meinertzhagen, and L. K. Scheffer, "Synaptic circuits and their variations within different columns in the visual system of Drosophila.," Proceedings of the National Academy of Sciences of the United States of America, vol. 112, no. 44, 2015.

[3] M. Helmstaedter, K. L. Briggman, S. C. Turaga, V. Jain, H. S. Seung, and W. Denk, "Connectomic reconstruction of the inner plexiform layer in the mouse retina," Nature, vol. 500, no. 7461, pp. 168–174, 2013.

[4] J. Anderson, B. Jones, C. Watt, M. Shaw, J. Yang, D. Demill, J. Lauritzen, Y. Lin, K. Rapp, D. Mastronarde, P. Koshevoy, B. Grimm, T. Tasdizen, R. Whitaker, and R. Marc, "Exploring the retinal connectome," Molecular Vision, 2011.

[5] J. S. Kim, M. J. Greene, A. Zlateski, K. Lee, M. Richardson, S. C. Turaga, M. Purcaro, M. Balkam, A. Robinson, B. F. Behabadi, et al., "Space-time wiring specificity supports direction selectivity in the retina," Nature, vol. 509, no. 7500, pp. 331–336, 2014.

[6] S. Takemura, A. Bharioke, Z. Lu, A. Nern, S. Vitaladevuni, P. K. Rivlin, W. T. Katz, D. J. Olbris, S. M. Plaza, P. Winston, et al., "A visual motion detection circuit suggested by drosophila connectomics," Nature, vol. 500, no. 7461, pp. 175–181, 2013.

[7] N. Kasthuri, K. J. Hayworth, D. R. Berger, R. L. Schalek, J. A. Conchello, S. Knowles-Barley, D. Lee, A. Vazquez-Reina, V. Kaynig, ´ T. R. Jones, et al., "Saturated reconstruction of a volume of neocortex," Cell, vol. 162, no. 3, pp. 648–661, 2015.

[8] N. Krasowski, T. Beier, G. Knott, U. Koethe, F. Hamprecht, and A. Kreshuk, "Improving 3d em data segmentation by joint optimization over boundary evidence and biological priors,"

[9] T. Kroeger, J. Kappes, T. Beier, U. Koethe, and F. Hamprecht, "Joint supervised-unsupervised segmentation," Proceedings of GCPR, 2014.

[10] J. Nunez-Iglesias, R. Kennedy, T. Parag, J. Shi, and D. Chklovskii, "Machine Learning of Hierarchical Clustering to Segment 2D and 3D Images.," PLoS ONE, 2013.

[11] J. C. Fiala and K. M. Harris, "Extending unbiased stereology of brain ultrastructure to three-dimensional volumes," J. Am. Med. Informat. Assoc., vol. 8, no. 1, pp. 1–16, 2001.

[12] F. Fleuret and D. Geman, "Stationary features and cat detection," J. Machine Learn. Res., vol. 9, pp. 2549–2578, 2008.

[13] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in Proc. Int. Conf. Machine Learn., 1996, pp. 148–156.

[14] J. Herold, W. Schubert, and T. W. Nattkemper, "Automated detection and quantification of fluorescently labeled synapses in murine brain tissue sections for high throughput applications," J. Biotechnology., vol. 149, no. 4, pp. 299–309, Sep. 2010.

[15] Q. J. H. Nguyen, "Shape-driven three-dimensional watersnake segmentation of biological membranes in electron tomography," IEEE Trans. Med. Image., vol. 27, no. 5, pp. 616–628, May 2008.

[16] E. P. Xing and M. I. Jordan. On semidefinite relaxation of normalized k-cut and connections to spectral clustering. Technical Report UCB/CSD-3-1265, Computer Science Div., U. C. Berkeley, Berkeley, CA, June 2003.

[17] S. X. Yu and J. Shi. Understanding pop out through repulsion. In CVPR-2001, volume 2, pages 752–757, 2001.