A SEMINAR REPORT
on

Large-Scale Mitochondria Segmentation

Submitted in requirement for the course

TRAINING AND SEMINAR (CSN - 499)

OF BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING

by

Aarush Gupta
(Enrollment Number : 16114002)

Project Taken up at

Visual Computing Group, Harvard University
Boston, MA, USA

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY ROORKEE

ROORKEE, 247667 (INDIA)

SUMMER, 2019

Contents

1 Acknowledgement

I had the privilege and pleasure of joining **Harvard University** at Boston, MA, USA as a research intern for a period of 11 weeks, from 13th May, 2019 to 26th July, 2019. It was the first time I got the opportunity to work at a research lab, and that too on a big project with huge potential and real-life applications.

I worked with the **Visual Computing Group** at the **School of Engineering and Applied Sciences** at Harvard, and am grateful to **Prof. Hanspeter Pfister** for giving me an opportunity to work with him during the summers. I would also express my gratitute to my mentor, **Dr. Donglai Wei**, who guided me in the best possible way. He is a postdoctoral research fellow with the group and oversees the Connectomics Project. There was always something to learn from his experience. Besides, I would like to thank the whole group and other interns who provided me valuable feedback on the project as well as future career prospects.

My internship was a part of the **SN Bose Scholars Program**, a joint initiative by Science & Engineering Board (SERB), Department of Science and Technology (DST), Govt. of India, the Indo-U.S. Science and Technology Forum (IUSSTF) and WINStep Forward, aimed at developing a dynamic and transformative student exchange program between premier institutions in Indian and the United States. I am grateful to the program organizers for financially supporting my internship and providing a platform to interact with other scholars working in various disciplines.

2 Introduction

2.1 University

Harvard University is the oldest higher education institution in the United States. Situated in Cambridge, MA, it is an Ivy League college and has the highest annual endowment in the world for an educational institution (around $40 billion). It is famous for the business, law, medical and engineering schools as well for its notable alumni, including Barack Obama, Franklin Roosevelt, Mark Zuckerberg, Bill Gates, and Natalie Portman. Being an engineering student, I worked at the Harvard John A. Paulson School of Engineering and Applied Sciences.

2.2 Research Group

I worked with the **Visual Computing Group** at Harvard SEAS. The group is led by **Prof. Hanspeter Pfister**, and explores and provides visual analysis tools and methods to help scientists and researchers better process and understand large, multi-dimensional data sets in various domains, such as neuroscience, genomics, systems biology, astronomy, and medicine. I worked under the supervision of **Dr. Donglai Wei**, a postdoctoral research fellow with the group.

2.3 Project

During my internship, I worked on the Connectomics project. This project is in collaboration with the Litchmann Lab at Harvard, and is funded by IARPA, NSF, NIH and some private foundations. Specifically, my task was to develop a neural network that automatically segments and labels different instances of mitochondria in neurons of the human brain. I was further assigned the task to develop a post-processing pipeline that could correct some of the prominent prediction errors made by the neural network without any supervision.

3 Motivation for the Project

The Connectomics project broadly involving the segmentation and analysis of brain scans obtained by electron microscopy (EM). The overall objective is to develop 3D maps of connections in the human brain to help in neuroscience. There are many diseases and disorders that are attributed to wrong wiring somewhere in the brain. With interactive models of neuron wirings, neuroscientists can better understand these diseases and possibly arrive at a cure [1]. The project further aims to improve the understanding of how the human brain actually works, how memories are stored and so on. Neuroscience also draws correlation from different attributes of mitochondria, synapses, and other organelles inside brain cells, such as their number density, structure, etc.

Multiple research groups at Harvard have been working in collaboration on this project for about ten years now. The reason why its taking such a long time is the staggering amount of data associated with a single human brain. It takes around 100 terabytes (roughly 100000 gigabytes) just to record all the connections in a salt grain-sized block of the brain!

One of the tasks associated with this project is to detect mitochondria in neurons automatically. It turns out that mitochondria are very different in shape from what most people imagine them to be like. Instead of the globular structure that is shown in most high school textbooks, mitochondria look like what is shown in figure 1.

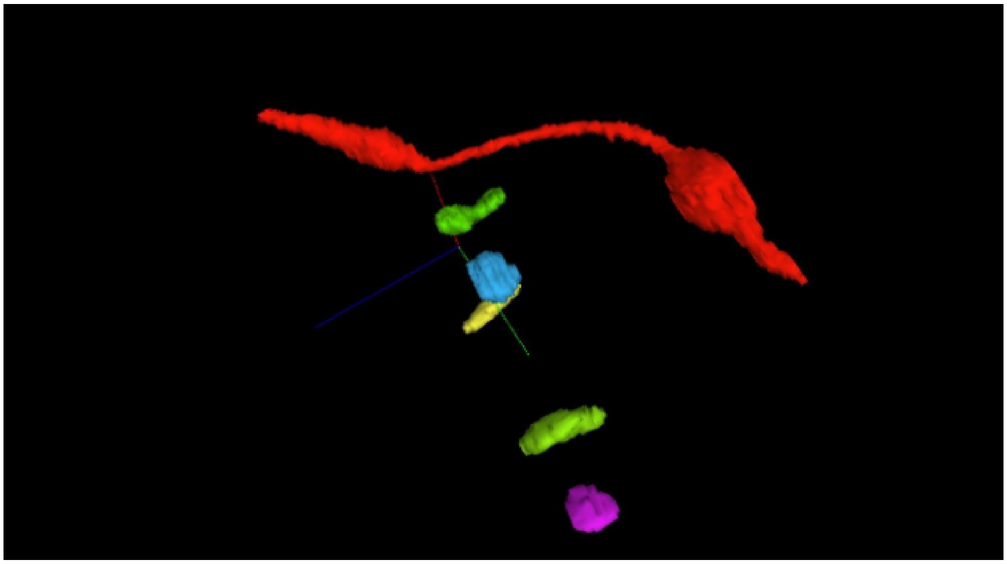

Figure 1: A sample of mitochondria as viewed in Neuroglancer

The conventional methods used for medical image segmentation fail for such non-convex morphologies. Another problem is that a large number of mitochondria are in contact with each other, making the problem more difficult. My project was aimed to arrive at a solution to these problems by developing a neural network architecture which take into account these priors and give better results than the standard methods. The input to the model and the required output are shown in figure 2.

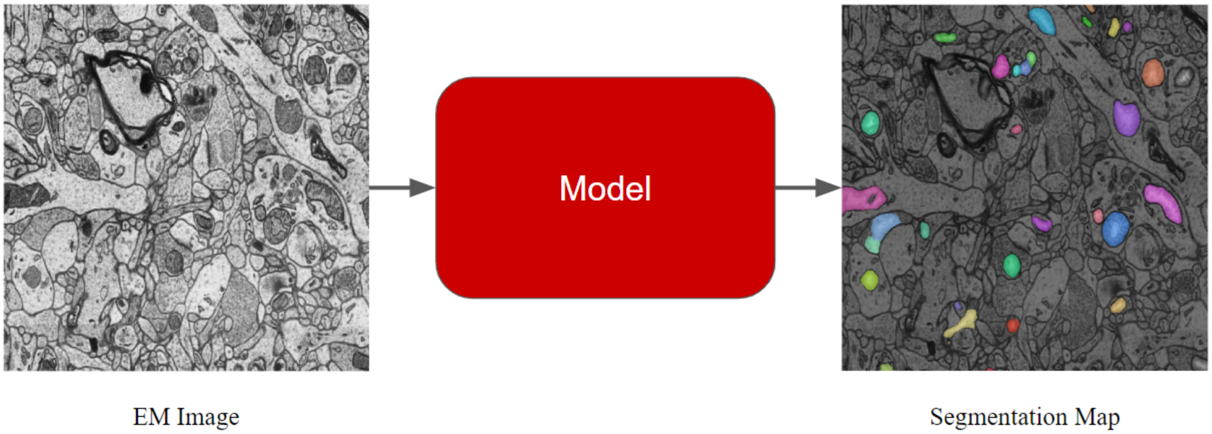

Figure 2: Model Overview

4 Objectives of Work Done

The aim of the whole project was to develop an end-to-end trainable neural network and a suitable post-processing pipeline for the instance segmentation of mitochondria in brain images obtained by a multi-bean electron microscope.The following sections illustrate the objectives of each part.

4.1 Neural Network to Predict Distance Transform

To maintain the long-range connectivity in non-convex shapes, we hypothesized that the neural network should be designed in such a manner that it is ’aware’ of any connectivity changes in the predictions while training, so that it could restore the connectivity by adjusting weights accordingly. Therefore, the objectives of this part can be summarized as follows:

- Design a connectivity-aware neural network to predict the energy maps.
- Train the network on a single annotated cell body
- Validate the trained model on other cell bodies

4.2 Post-Processing Pipeline to Convert Energy Maps to Segmentation Maps

This part of the project was concerned with getting the final segmentation maps from the distance transform predicted by the neural network mentioned above. The pipeline also automatically corrected some of the prominent errors in the predictions. To summarize, the objectives of this part are:

- Use the predicted energy maps to get an initial segmentation map
- Analyze some prominent error cases in the initial segmentation map
- Develop a suitable unsupervised error correction method to deal with prominent errors
- Deploy the combined pipeline to network predictions on a 49x4096x4096x1000 sized volume

4.3 Class Wrapper for Neuroglancer

Neuroglancer is a WebGL-based viewer for visualizing volumetric data. I found the design of the library a little complicated and cluttered for my purpose. To ease out setup and addition of new layers to the Neuroglancer server, I wrote a wrapper class that abstracts away some of the functionalities and boilerplate code, and enables quick and quick setup of the server.

5 Work Plan

The following work plan was drafted out by the group to meet the objectives within a period of 11 weeks:

- **Week 1:** Setting up my personal machine, installing the required libraries, getting a research computing account to access Harvard's computing cluster, setting up the required libraries and a development environment for prototyping, training, and deploying trained neural networks to the full dataset.
- **Week 2:** Getting acquainted with PyTorch Connectomics, a library primarily developed by Zudi Lin, one of the PhDs working on the project.
- **Week 3-4:** Design and build a prototype the connectivity-aware energy prediction model.
- **Week 5:** Train the model, and test for different hyperparameters.
- **Week 6:** Write the validation script, run validation on the annotated cell body, and post-process the output using vanilla watershed algorithm.
- **Week 7:** Analyze the results of the network manually to find out instances where the model fails.
- **Week 8-9:** Build an appropriate post-processing pipeline for automatic error detection and correction.
- **Week 10:** Try out 3D CNNs instead of 2D ones to deal with poor layer-wise connectivity.
- **Week 11:** Wrapping up the work, cleaning up and documenting the codebase, final presentation.

In addition to the above, I attended various events and talks held at Harvard University and MIT to get to know about related works, and see if I could incorporate anything into my work. I also presented my work to Prof. Jeff W. Lichtman's group and got some feedback from him as well.

6 Work Description

6.1 Watershed Energy Prediction

Previous approaches used for instance segmentation don't work properly for mitochondria for two main reasons, these approach cannot capture the non-convex morphology of the mitochondria, and that the mitochondria are generally in close vicinity of each other. To add to the problem, the amount of time and money required to annotate a sufficient amount of data is too high. Simple region proposal based methods [2] [3] [4] [5] don't work really well because of the wiry or articulate shape.

One of the previous approaches uses boundary prediction [6] [7] for the task. The model predicts the boundary, and a flood filling rule is applied on the resultant map. But this approach fails for mitochondria as it results in a lot of false merges, as the boundary can break at places where the mitochondria are in close vicinity.

Another approach uses a neural network to predict the watershed energy [8] in a discrete setting. They use an intermediate task to help train the model for the main task. But the discretization of the energy into levels leads to loss of long-range connectivity of large mitochondria leading to false splits. There are also some recurrence-based methods [9] [10] but those are very computationally expensive.

To alleviate the above-mentioned problems without using a large amount of training data and computational resources, we use a connectivity-aware model that is aware of the connectivity changes in the predictions with the help of appropriate loss functions. Whenever the connectivity breaks, the penalty on the prediction increases, and backpropagation adjusts the model weights accordingly. The ground truth is also pre-processed to facilitate this type of training. The model has a UNet [6] architecture with Squeeze-and-Exciation [11] layers in both the encoder and decoder. The model also has auxiliary losses which act as deep supervision and promote better feature learning at each level.

6.2 Post-Processing Pipeline

The post-processing pipeline is concerned with getting the final segmentation map from the energy map predictions from the neural network. To do this, the energy map is thresholded at a certain gray scale value to get a binary map, which acts as a mask for the watershed algorithm. To deal with false merges, we use the skeletons predicted by the model. These skeletons act as markers for each instance in the energy map. Finally, the watershed algorithm is used with the mask and the markers to get the final segmentation map. The use of markers helps in curbing watershed's over segmentation problem.

7 Implementation Details

7.1 Experimental Setup

All the models were prototyped locally on a Linux machine. The model and post-processing pipeline were coded using Python and some open-source libraries such as PyTorch, NumPy, Scikit-Learn, etc. The model was trained with the help of PyTorch Connectomics, a library made by the Visual Computing Group. The library supports Synchronized BatchNorm for training on multiple nodes and GPUs. For training and deployment, Harvard's RC cluster was used, which had a variety of machines, filesystems and GPUs. I trained most of the models on the cluster's Nvidia DGX servers. These servers are based on Nvidia's Volta architecture and are state of the art in Deep Learning hardware. Each server consists of 8 GPUs with 16GB memory each. The model training was parallelized on 4 or 8 GPUs depending on the need.

7.2 Datasets

The whole project is mostly based on the dataset collected by the Litchmann Lab. Small brain samples are cut into slices which a few manometres thin. These slices are then imaged by a multi-beam electron microscope developed by Zeiss. The resultant images are stacked on top of each other and processed. The processed image stack is then passed on to an annotation team which labels a small part of the image stack for training and validation purposes. The data is then used by the Visual Computing Group to develop tools and models for automating detection and visualization. Apart from this, the group uses the SNEMI dataset [12], and the CREMI dataset [13] for pre-training and prototyping networks.

7.3 Tools

- **Neuroglancer:** It is a WebGL-based viewer for visualizing volumetric data. It helps in viewing brain image stacks, and segmentation maps by superimposing the maps on the images layer-by-layer.

- **Tensorboard:** It is a web-based open source tool developed by the Tensorflow team to keep track of the training statistics and for on the fly validation of the model.

- **Slurm:** It is a workload manager and is used by Harvard's RC cluster for scheduling and managing jobs on different nodes.
- **Nano:** It is a command line text editor for making small changes to the codebase.

8 Results and Discussions

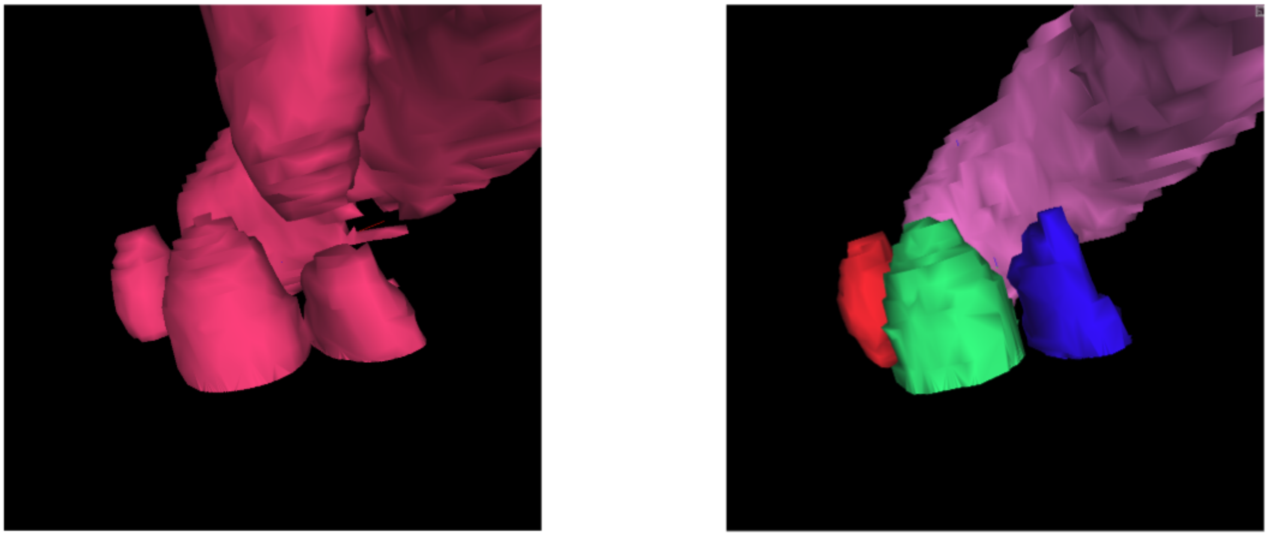

Figure 3: Results before and after Error Correction

The results by the model and the post-processing pipeline combined are shown in table 1. We use a strict IoU threshold of 0.8 for matching the instances.

Table 1: Results on Validation Data (IoU>0.8)

Data / Metric	**True Positives**	**False Positives**	**False Negatives**
Initial Results	49	12	20
After Error Correction	59	12	10

The pixel-level precision and recall scores are 94% and 96% respectively. Most of the false positives are due to poor-layerwise connectivity across the depth of the volume. Also, as evident in figure 3, error correction works pretty well. This can be mitigated by using a 3D CNN model rather than a 2D one. The false negatives are mostly due to faulty annotations in small mitochondria, or due to anomalies at the edges of the volume.

For further work, we aim to improve the layerwise connectivity along the depth of the image stack, as well as test the robustness of our model on similar instance segmentation tasks. Subject to the quality of the results, we aim to submit our work to a top-tier Computer Vision or Medical Imaging conference.

9 Conclusion

The summer internship with the Visual Computing Group was a great learning experience for me. The group provided me with the flexibility to choose my project, work at my own pace and time. I had been working on the theoretical aspects of Deep Learning for the past two years. The internship gave me an opportunity to work on a real-world application of Deep Learning. I also got a chance to meet many professors and students, who have been pursuing pioneering work in Deep Learning and driving the field forward. Overall, it was a truly enriching experience for me and the internship would go a long way in helping me with my career as well.

References

[1] Knowles-Barley, Seymour and Jones, Thouis R and Morgan, Josh and Lee, Dongil and Kasthuri, Narayanan and Lichtman, Jeff W and Pfister, Hanspeter, "Deep learning for the connectome," *GPU Technology Conference*, 2014.

[2] L.-C. Chen, A. Hermans, G. Papandreou, F. Schroff, P. Wang, and H. Adam. Masklab: "Instance segmentation by refining object detection with semantic and direction features" *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2018.

[3] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. "Path aggregation network for instance segmentation." *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018. 1, 2

[4] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. "Fully convolutional instance-aware semantic segmentation." *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2359–2367, 2017. 1, 2

[5] K. He, G. Gkioxari, P. Dollar, and R. Girshick. "Mask R-CNN." *In Proceedings of the IEEE International Conference on Computer Vision*, pages 2961–2969, 2017.

[6] O. Ronneberger, P. Fischer, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation." *In International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 1, 3, 6, 7

[7] H. Chen, X. Qi, L. Yu, and P.-A. Heng. "DCAN: Deep Contour Aware Networks for Accurate Gland Segmentation." *In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2487–2496, 2016. 1, 3, 7

[8] M. Bai and R. Urtasun. "Deep Watershed Transform for Instance Segmentation". *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5221–5229, 2017. 2, 3, 5

[9] M. Ren and R. S. Zemel. “End-to-end instance segmentation with recurrent attention.“ *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6656–6664, 2017. 1, 2

[10] B. Romera-Paredes and P. H. S. Torr. “Recurrent instance segmentation.“ *In European conference on computer vision*, pages 312–329. Springer, 2016. 1, 2

[11] Jie Hu and Li Shen and Gang Sun. “Squeeze-and-Excitation Networks.“ *IEEE Conference on Computer Vision and Pattern Recognition 2018*

[12] Arganda-Carreras I., Turaga S. C., Berger D. R., Cirean D., Giusti A., Gambardella L. M., et al. (2015). “Crowdsourcing the creation of image segmentation algorithms for connectomics.“ *Front. Neuroanat.*

[13] CREMI (2016). “Miccai Challenge on Circuit Reconstruction From Electron Microscopy Images (CREMI)“.