Automated annotation of cellular cryo-electron tomograms using U-Net

Yanyan Zhao
Department of Bioengineering
Stanford University
yanyanz@stanford.edu

Weijiang Zhou
Department of Bioengineering
Stanford University
weijianz@stanford.edu

Abstract

Cryo-electron tomography is a powerful technique by which 3-dimensional cellular features can be observed in nanometer resolution. Accurate and fast annotation of features within tomograms are essential and demanding in current research field. Here we propose to apply deep learning to tomography annotation. Our preliminary results show that deep learning offers an relative accurate annotation for several cellular features including microtubule, double membrane and single membrane.

1 Introduction.

Electron cryo-tomography(cryo-ET) is a popular tomography technique used by biologists in recent years to determine nanometer resolution 3-D sub-cellular structures. This technique is normally featured with imaging a rotating sample for 120 degrees, generating a series of 2D tilt images that can be combined to produce a 3D volume. The main method researchers use in reconstructing the 3D volume of the sample starts with manually annotating various features such as cytoskeletal filaments, cell wall elements and internal compartments in each 2D image slice. Due to the diversity and crowdedness of features in 2D each slices as well as the large number of slices in each 3D volume, low efficiency becomes the biggest challenge in manual annotation. Although remaining as the most accurate method, the concern of this low efficiency becomes more serious especially when data collection is speeding up these days. Here we propose to apply deep learning to cryo-ET tomogram annotation to provide users a good trained neural network for general cryo-ET cellular feature prediction and therefore help them accelerate cell biology study. Our goal is to generate an annotated tomogram by taking each 2D image slice of the 3D tomogram and passing it through our pre-trained neutral network. In this project, we are aiming at annotating six features, including background, microtubules, ribosome, double layer membrane, single layer membrane and carbon edge. The input image to our algorithm is 512 by 512 pixels while the output from the neural network is in shape of 6 by 512 by 512 predicting six class probabilities for each pixel.

2 Related Work.

Very few studies have been reported so far to address the cryo-ET tomogram annotation issue except Muyuan Chen[1] *et al* tried to use convolutional neural network method to annotate cellular features semi-automatically. Although his method largely accelerates the tomogram annotation compared to manual annotation, it's still less efficient considering that his method only allows for single feature prediction. With multiple features in the image, users have to train separate models by repeatedly going through same set of tomograms. In addition, given the very small training dataset and shallow neural network of only four layers, this method often predicts false positives and false negatives thus resulting in more efforts in postprocessing.

3 Dataset and features.

The raw data comes from three sources. The first source is a PC-12 cell tomogram in size of (96, 864, 868) used in Muyuan Chen's paper. This data is acquired at low magnification on cryo-electron microscopy. The second part of the data consists of thirty-eight neuron cell tomograms acquired at medium magnification of size (n, 960, 960) where n is between 113 and 630 and with median 281. And the third part includes four high-magnification ribosome tomograms from EMPIAR 10064 in size of (256, 1024,1024). Of these three sets of data, only the first dataset has been manually annotated.

Lots of efforts are devoted in data preprocessing. First, top and bottom slices from neuron and ribosome tomograms are excluded since these images mostly have nothing but noise. Then 38 neuron cell tomograms are combined into 8 large tomograms in order to obtain more accurate labelling. All neuron cell and ribosome tomograms are semi-automatically labelled using e2tomoseg_convnet.py developed by Muyuan Chen. The annotated features includes microtubule, ribosome, double layer membrane, single layer membrane and carbon edge. False positives from semi-automatic annotation are largely excluded by thresholding and manual cleaning. Mask for noise and five features are further encoded with 0 <= number < 6(class number). All cleaned 3D tomograms and their corresponding 3D masks are extracted into 2D images with each image being cropped into four images of size 512 by 512. The final dataset contains 16,856 512x512 images. Dataset is randomly shuffled and divided into train set, dev set and test set(80/10/10).

4 Methods.

The U-net model[2] consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. The contracting path includes four block units, with two convolutional layers and one dropout layer followed by a max pooling layer in each block unit. Similar to contracting path, the expansive path also has four block units. However, two convolutional layers are followed by an up-sampling layer and a convolution layer. In each unit, the convolutional output from contracting path is concatenated to enable

precise localization. The output layer of the U-net model has six channels with each representing a class. Softmax function isn't applied at this point since it's covered by the cross entropy loss function in Pytorch.

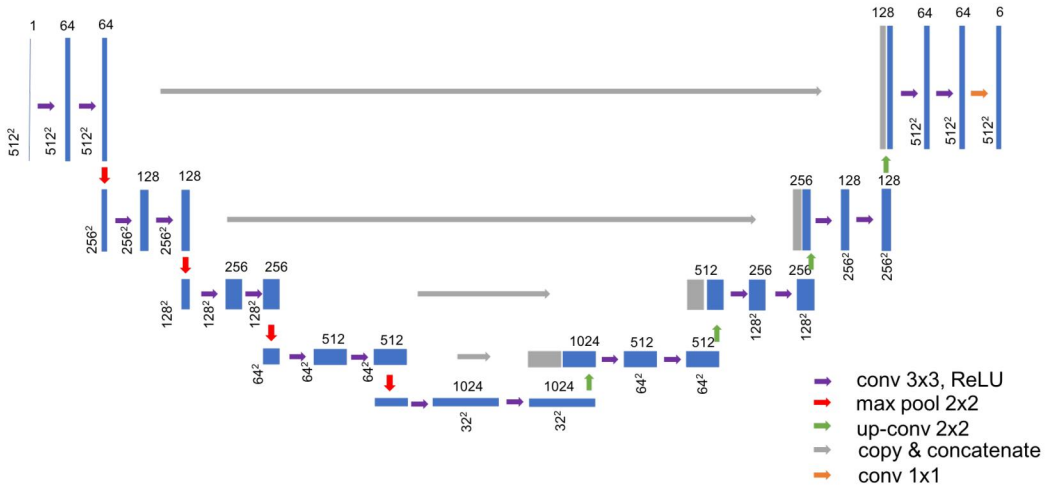

Figure 1. U-net model.

For this multi-feature classification task, we employ a weighted cross entropy loss function,

$$Loss(x,\ class) = \text{weight[class]} * (-x[class] + log(\sum_j exp(x[j])))$$

because we have a very imbalanced dataset with noise class accounting for more than 90% pixels in the dataset. Our model uses Adam optimizer algorithm that takes the running average of previous gradients for parameter updating during training. This model is implemented from the previous work at https://github.com/zhixuhao/unet, but with pytorch.

5 Experiments, results and discussion.

	NOISE	MICROTUBULE	RIBOSOME	DOUBLE MEMBRANE	SINGLE MEMBRANE	CARBON EDGE
TRAINING F1 SCORE	0.94	0.57 (0.22)	0.47 (0.22)	0.48 (0.21)	0.71 (0.50)	0.60 (0.40)
TEST F1 SCORE	0.98	0.77 (0.36)	0.01	0.66 (0.40)	0.79 (0.52)	0.40 (0.10)

Table 1. Statistics of training and test for the final model(class weight is [1,2000,3000,2000,2000,3000], learning rate is 0.001, batch size is 4, epoch number is 50, Adam optimizer). Six features are targeted in training and test, including noise, microtubule, ribosome, double membrane, single membrane and carbon edge. Number in brackets represents the mean F1 score from the last epoch.

During hyperparameter search experiments, we primarily explored batch size, learning rate, class weights and number of epochs. Throughout the training, we switched our evaluation metric from general accuracy to F1 score specific for

each class. In general, the metric F1 score for cell feature classes is relatively low compared to noise class. Given that noise accounts for more than 90% of the pixels in the dataset, this class imbalance poses a potentially challenging problem for better neural network training. We observed training loss dropping throughout the training epochs, however, for each feature class, especially for ribosome and carbon edge, the high variance of F1 score across batches in each epoch training suggests the bias in data loading during training and testing.

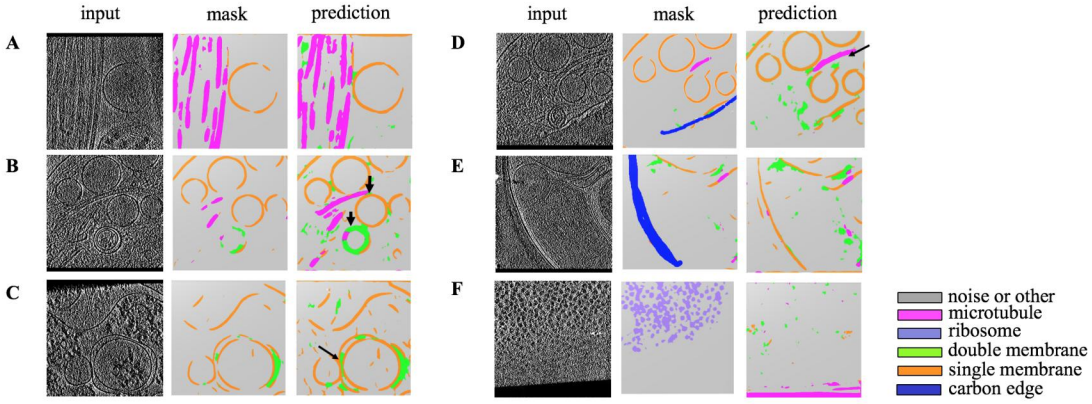

Figure 2. Neural network prediction. Our U-net model offers an relative accurate annotation for microtubule, double membrane and single membrane, but not for carbon edge and ribosome. Arrows point to areas where the model outperforms the labeling.

Despite the imperfect performance by F1 score evaluation metric, we find that our model shows relatively good accuracy in predicting microtubule, double membrane and single membrane. Especially, it accurately predicts microtubules and double membranes in areas where these features are mislabeled as false negatives. However, the model performs poorly in terms of prediction on ribosome and carbon edge. Considering the relatively rare occurrence of these two features in the dataset, up-sampling or application of higher weights will be tried to solve the problem.

6 Conclusion and Future Work.

The preliminary results of our U-net model shows that deep learning offers an relative accurate annotation for several cellular features including microtubule, double membrane and single membrane. Compared to shallow neural network reported previously which targets only one feature, the U-net model performs better in terms of low false positives. However, because of the class imbalance especially for ribosome and carbon edge, our model does not show perfect prediction on these features so far. Up-sampling of minority classes and introducing more data of these features is the next step we're aiming to try.

As good data labeling plays a critical role in model performance, in the next step, we plan to manually label the dataset to exclude false positives and false negatives. Considering significant class imbalance, we also need to get rid of images with little cellular features to down-sample the noise class, and meanwhile add more data with ribosome and carbon edge to balance classes. Even though we shuffled data before data splitting, we still observe the

existence of data bias based on the high variance of F1 score for each class and testing loss. We speculate that this is caused by large difference in the diversity and crowdedness of features among images. Therefore, we have to be more cautious in balancing all classes in training set, development set and test set.

In the future, we plan to try 3D U-Net on tomogram annotation considering some features e.g. microtubules has little information on 2D slices when they go in Z direction. Due to the limitation of computing source at present, we were not able to try large batch size in training the model, but in the future, we hope to try it with more GPUs.

7 Contributions.

Since the data preprocessing step demands lots of efforts, both team members Weijiang Zhou and Yanyan Zhao participated in labeling and cleaning the data. Pipeline code was written by Yanyan Zhao.

8 pipeline code.

https://github.com/YANYANZH/cs230_project

Acknowledgment.

We would like to thank Aarti Bagul for her helpful advice during office hours. We would also like to thank Dr. Gong-Her Wu for providing original cryo-electron tomography datasets and useful discussion.

Reference.

1. Chen, M. et al. Convolutional neural networks for automated annotation of cellular cryo- electron tomograms. Nat. Methods 14, 983–985 (2017).

2. Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. in Medical Image Computing and Computer-Assisted Intervention -- MICCAI 2015 (eds. Navab, N., Hornegger, J., Wells, W. M. & Frangi, A. F.) 234–241 (Springer International Publishing, 2015).