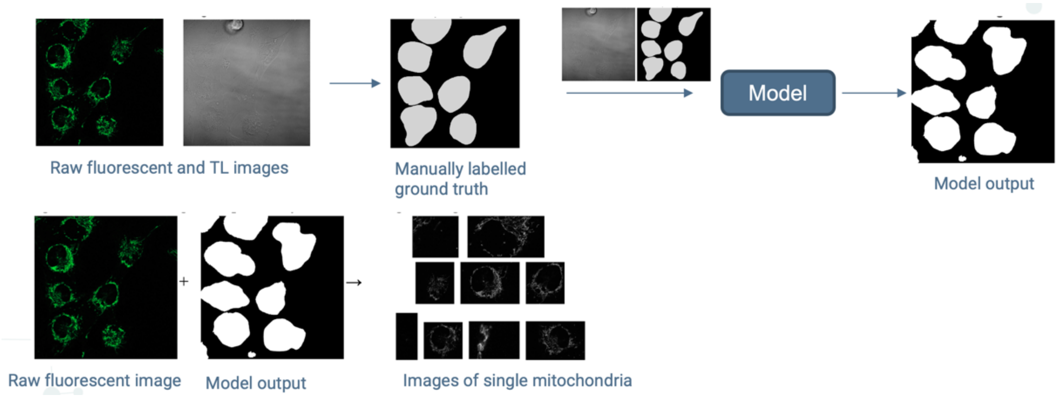
Optimization of Mitochondrial Segmentation using Semi-Supervised Learning

Hsu-Ting Kuo

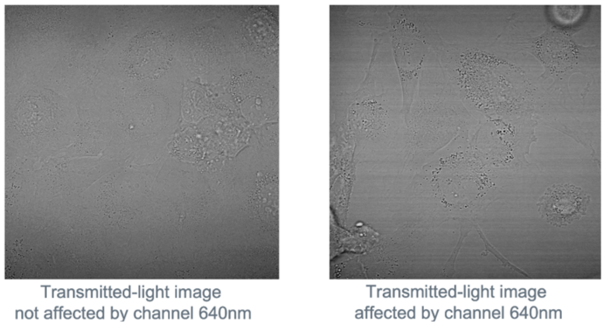
Department of Biomedical Engineering, National Taiwan University
Correspondence: B08508028@ntu.edu.tw

Outline

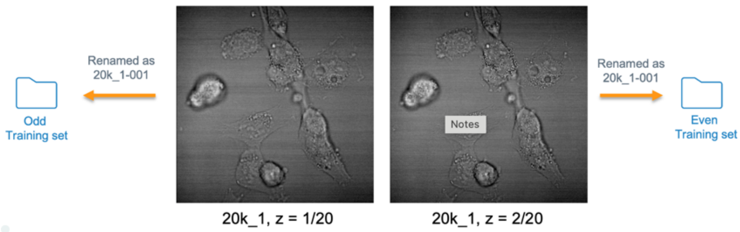
1. Introduction
    a. Background
    Mitochondrial segmentation has always been considered a challenging task due to the unique morphology of mitochondria and the lack of labeled images. Previously, we have proposed a solution to achieve automated mitochondrial segmentation. Through labeling the locations of mitochondria on transmitted-light images, we can train a U-Net or a UNet++ model to predict the binary mask of mitochondrial location from transmitted-light images. After the model output the predicted binary mask, we can use it to crop single mitochondria from the corresponding fluorescent image.



    b. Problems encountered
        i. Lack of accuracy in ground-truth labeling
        For ground-truth labeling, we labeled mitochondria in transmitted-light manually, which is hard to be accurate as the contour of mitochondria is clear in neither the fluorescent images nor in the transmitted-light images.
            1. Solution:
            I acquired images of cells stained with CellMask, a cell membrane dye so that I can be more certain about the contour of mitochondria while labeling the ground truth.
        ii. Data deficiency:
        CellMask staining is tricky and it is not easy to collect images of cells stained with CellMask, which makes the existing problem of data deficiency even worse.
            1. Solution:
            I figured semi-supervised learning might improve model performance significantly as we have few accurately labeled data and relatively more data without accurate labeling. Secondly, while acquiring images of cells stained with CellMask, I used z-stack imaging, which allows us to collect the largest amount of data using limited time and resources.
        iii. Background noise in newly acquired images
        However, adding the laser channel (640nm) for CellMask fluorescence dye during confocal imaging causes unexpected background noises (the horizontal lines shown in the right figure below).



2. Part 1: Image preprocessing using U-Net
    a. Noise2Noise: Learning Image Restoration without Clean Data
    The study, Noise2Noise: Learning Image Restoration without Clean Data, presents a method to remove background noises using a single U-Net that can learn photographic noise removal, denoising synthetic Monte Carlo images, and reconstruction of undersampled MRI scan based on noisy data only. This study had been used to removed background noise in EM images by comparing the frame 1 image to the frame 2 image in a video. The model learned to recognize the changing parts in two images and remove them. I thought this could be used to remove the background noise in our images as I collected the z-stack of images. The image in a plane of focus is not very different from the one in the next plane of focus (the interval between each image is not more than 10uM), which is similar to the situation in the study of video noise removal.
    b. Dataset
    To prepare the dataset, I separated images at odd z-level from those at even z-level (as shown in. the figure below). I trained the U-Net model using two different datasets: one includes out-of-focus images, and the other one has only focused images.

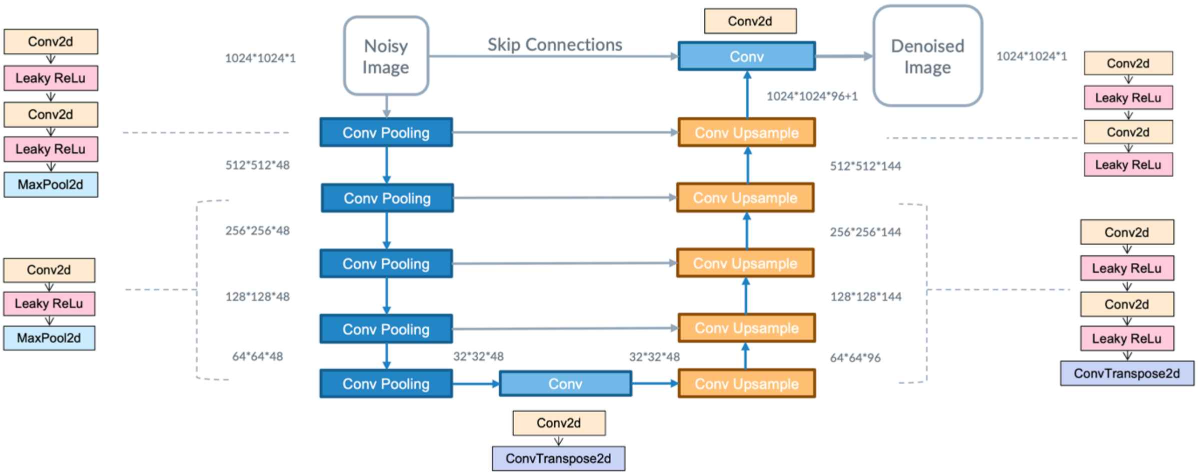


        i. Dataset with blurred images
        source: Lab
        Z-interval between each z-level image: 4uM - 9uM
        Training: 216*2 images
        Odd: 216
        Even: 216
        ii. Dataset with only selected images
        source: Lab
        Z-interval between each z-level image: 4uM - 9uM
        Training: 130*2 images
        Odd: 130
        Even: 130

iii. Model

1. Network architecture

   The model is a U-Net with four down-sampling layers and four up-sampling layers.



2. Implementation details

   Epoch: 500

   Batch size: 2

   Optimizer: Adam

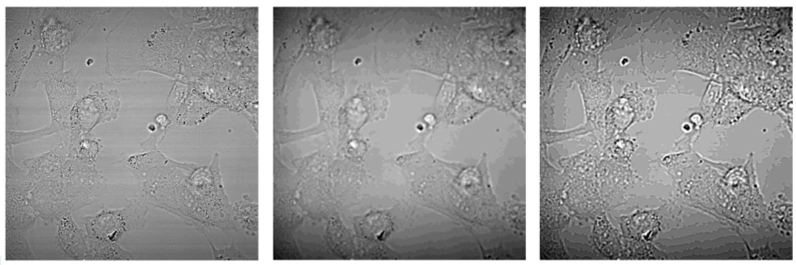
   Learning rate: 0.001

   Loss function: L2 (MSE) Loss

c. Results

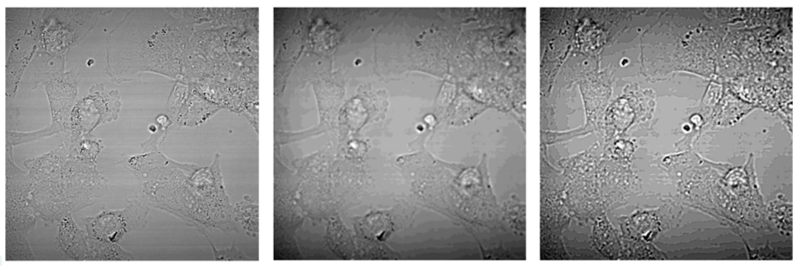
The images output from the U-Net model were post-processed with Unsharp Mask (radius = 0.5 pixels, mask weight = 0.70)

i. Visualization: model trained with blurred images



Original Model output Postprocessed output

ii. Visualization: model trained with selected images



Original Model output Postprocessed output

iii. Performance of models trained with original images and denoised images

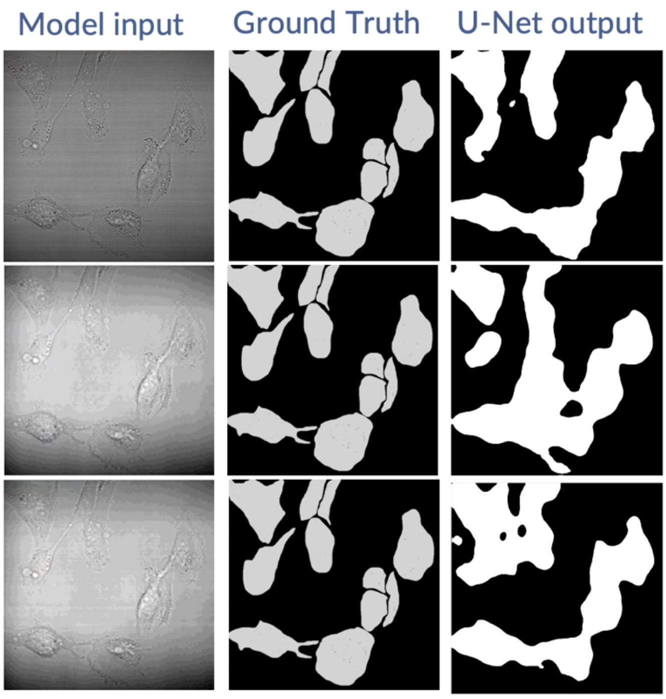
1. **Original images**
   Train mean IoU: **0.753399**
   Test mean IoU: **0.679553**
2. Denoised images - model trained with blurred images
   Train mean IoU: 0.61672
   Test mean IoU: 0.518328
3. Denoised images - model trained with selected images
   Train mean IoU: 0.5440
   Test mean IoU: 0.47996

It appeared that the results from models trained with denoised training datasets were worse than those from the model trained with original images. Therefore, in the second part of this project, semi-supervised learning, I did not use the denoised images as the training dataset.

Original images

Denoised images - model trained with blurred images

Denoised images - model trained with selected images

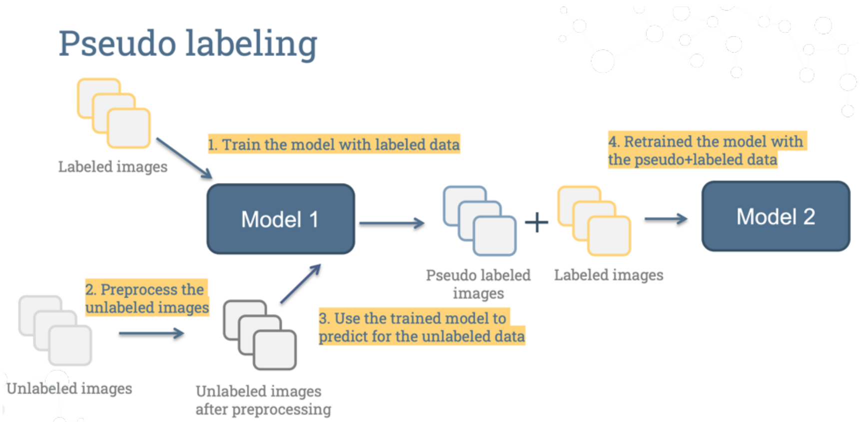


3. Part 2: Semi-supervised learning
   a. Datasets
      i. Labeled images:
         Cells in the images were not treated with chemicals
         - Source: Lab
         - In total: 52 images
         - Training: 30 images
         - Validation: 9 images
         - Test: 13 images

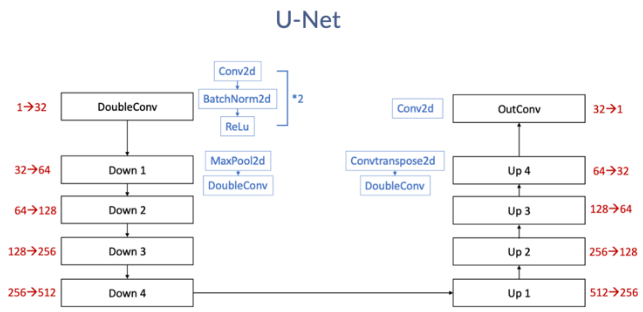
      ii. Unlabeled images:
         Cells in the images were treated with FCCP, Oligomycin, Antimycin A, and Rotenone
         The images were pre-processed using unsharp mask (radius = 0.5 pixels, mask weight = 0.70)

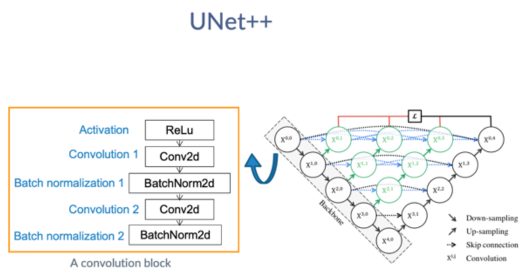
- Source: Lab
- In total: 40 images

b. Pseudo labeling
The workflow for pseudo labeling is as shown below.
Model 1 was trained with labeled images, which was later used for pseudo labeling the unlabeled images. Finally, pseudo labeled images and labeled images were both used as a training dataset to train model 2.



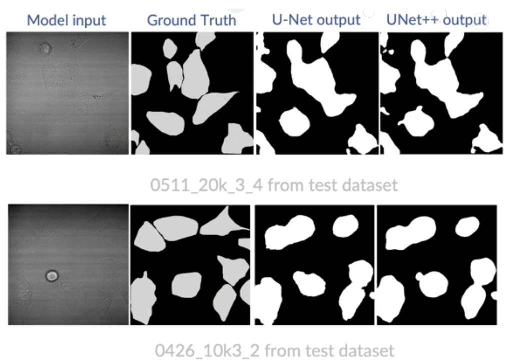
c. Model
U-Net and UNet++ were both used as models 1 and 2. Their performance will be compared later.





d. Model 1
i. Implementation details
- Epoch: 100
- Batch size for training: 4
- Batch size for testing: 32
- Optimizer: Adam
- Learning rate: 0.002
- Loss function: Binary Cross-Entropy

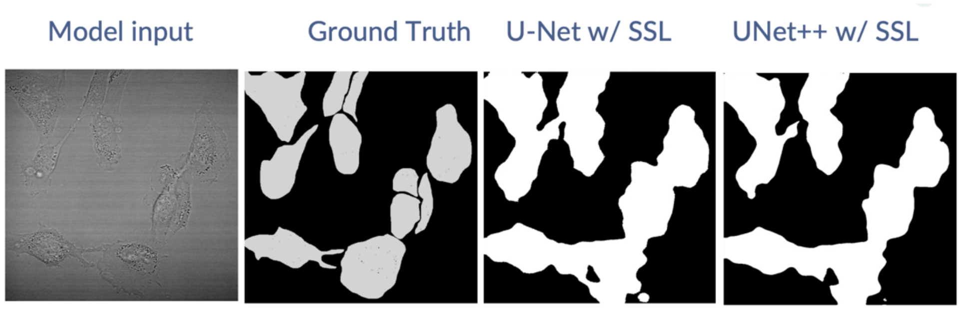
ii. Performance
1. U-Net
Train mean IoU: 0.66729
Test mean IoU: 0.66722
2. UNet++
Train mean IoU: 0.63561
Test mean IoU: 0.62304



e. Model 2
   i. Datasets
      Both labeled and pseudo labeled images were used as training data
      - Source: Lab
      - In total: 92 images
      - Training: 52 images
      - Validation: 13 images
      - Test: 27 images
   ii. Implementation details
      - Epoch: 100
      - Batch size for training: 4
      - Batch size for testing: 32
      - Optimizer: Adam
      - Learning rate: 0.002
      - Loss function: Binary Cross-Entropy

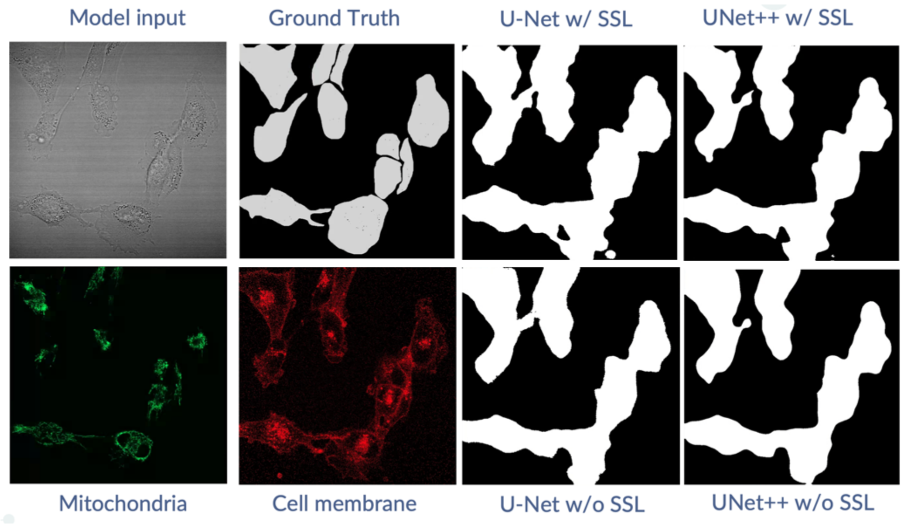
4. Results

a. Model 2 performance
   From the figures below, we can see that despite having lower mean IoUs, UNet++ successfully captured and predicted more details than U-Net did.
   i. U-Net
      Train mean IoU: **0.56776**
      Test mean IoU: **0.65433**
   ii. UNet++
      Train mean IoU: 0.52400
      Test mean IoU: 0.62310



b. Comparison of models
   The results show that U-Net without semi-supervised has the best performance. However, from the output images, we can see that UNet++ with semi-supervised learning can segment mitochondria more accurately.

	Batch size	Epoch	Train Mean IoU	Test Mean IoU
U-Net w/o SSL	8	100	0.644485	0.684533
UNet++ w/o SSL	8	100	0.482820	0.671528
U-Net w/ SSL	8	100	0.567776	0.654332
UNet++ w/ SSL	8	100	0.524004	0.623103



5. Discussions
    a. Part 1: Noise Removal using U-Net
    The horizontal lines in images were successfully removed, however, some details were lost during the training process. The differences between each frame of images used in video noise removal might be smaller than the ones between each z-level of images, which explains why the noise removal method did not work well on our images.
    b. Part 2: Semi-Supervised Learning
        i. The performance of model 1 was not as good as expected
        Previously, we thought that treating cells with chemicals did not affect the quality of transmitted-light images, however, the results from pseudo labeling showed that images of cells treated with certain chemicals couldn't be recognized by model 1, which was trained with images of normal cells only, and therefore affect the model performance and the results of pseudo labeling. Secondly, there are differences between the qualities of old and new training images, for example, the background noise as mentioned earlier.
    c. Future work
    To improve the model performance, I think the size of the training dataset should be increased. Moreover, model 1 should be trained with images of cells treated with chemicals, so that it could recognize this type of image during pseudo labeling. Finally, the workflow of semi-supervised learning should be optimized and better organized.
6. References

    a. Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
    b. M. (2021). *GitHub - milesial/Pytorch-UNet: PyTorch implementation of the U-Net for image semantic segmentation with high quality images*. GitHub.https://github.com/milesial/Pytorch-UNet
    c. Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., & Liang, J. (2018). UNet++: A nested U-Net Architecture for Medical Image segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, 3–11. https://doi.org/10.1007/978-3-030-00889-5_1

d. ZhenyuTan. “GitHub - ZhenyuTan/Noise2Noise-Cryo-EM-Image-Denoising: Pytorch Implementation of Noise2noise for Cryo-EM Image Denoising.” *GitHub*, github.com, 5 Dec. 2018, https://github.com/ZhenyuTan/Noise2Noise-Cryo-EM-image-denoising.
e. Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., & Aila, T. (2018). Noise2Noise: Learning image restoration without clean data. arXiv preprint arXiv:1803.04189.
f. NVlabs. (n.d.). *NVlabs/noise2noise: Noise2Noise: Learning Image Restoration without clean data - official tensorflow implementation of the ICML 2018 paper*. https://github.com/NVlabs/noise2noise