

# SRTP training summary report I

---

- Training and testing data indicators and visualization
- Comparison of training and test sets
- To be improved

## 1.Training and testing data indicators and visualization

---

### Training and testing data indicators

Epoch	Average loss of train data	Error rate of train data	Error rate of test data
1	0.8848	4.012%	
5	0.6941	3.075%	
10	0.6196	2.881%	
15	0.5565	2.788%	
20	0.5286	2.690%	2.662%

#### Basic Notes:

- Loss function: Calculate the average loss of training set and test set under different epochs using **cross entropy loss function**. Since U-net uses the method of dense prediction. Therefore, for each pixel of the image, it applies softmax for prediction (pixel-wise softmax). The predicted output is a 3d vector (width, height, class\_nums) representing the classification scores of pixels of different heights and widths in the image. The error function here is the cross entropy between the predicted score of each pixel under different categories and the classification of each pixel of the original label image.
- Validation error of the training set: the error index of the training set under different epochs is calculated using the error calculation function of the source code. For each pixel, we take the label category with the highest score as our classification prediction, and then compare it with the label image to calculate the error.
- Test set verification error: we call the error calculation function of the source code for the test set error index calculation, the principle is the same as the training set error index calculation.

### Visualization

1. Average loss of training data under different epochs



2. Index of error rate of training sets under different epochs



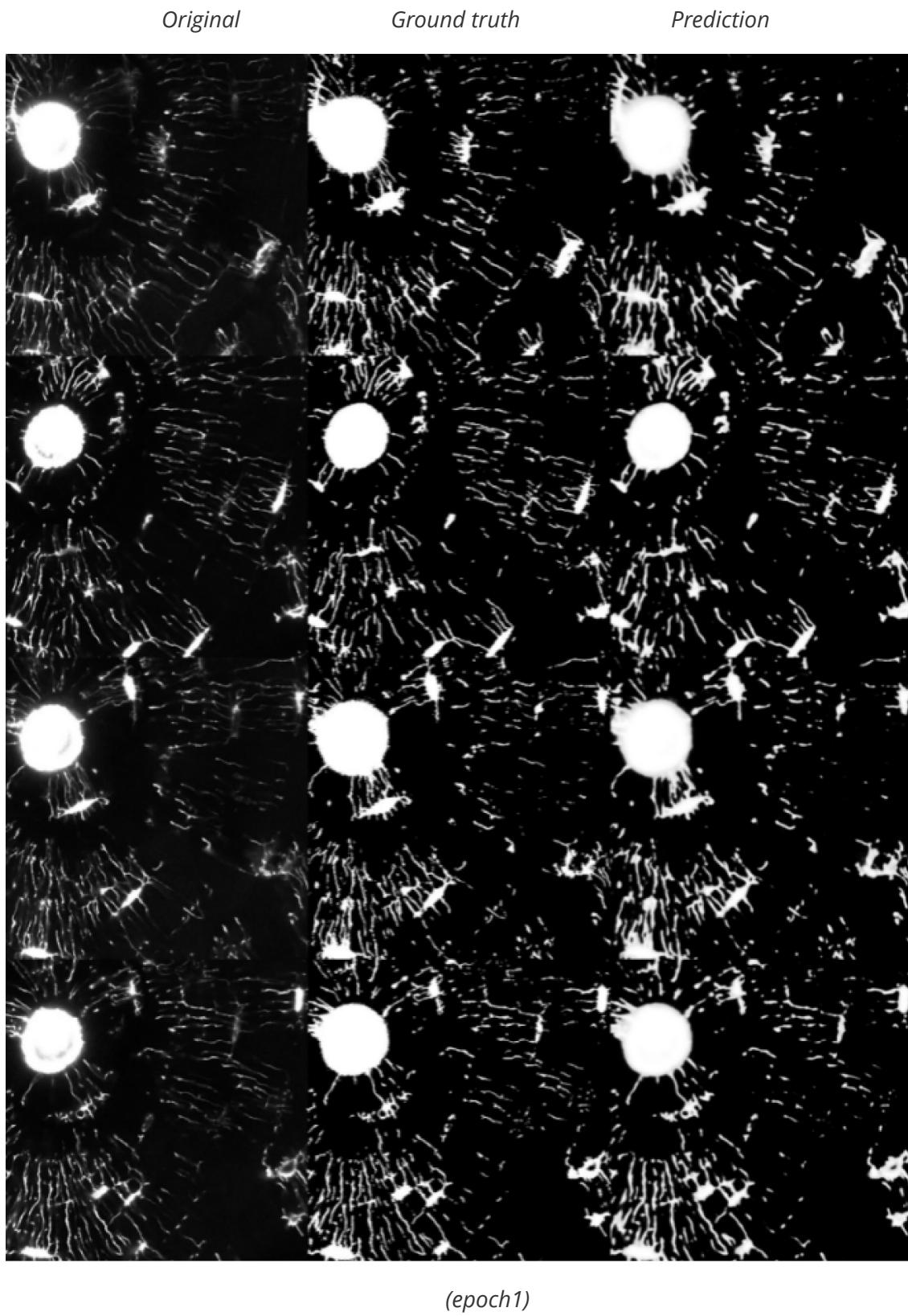
**Notes:**

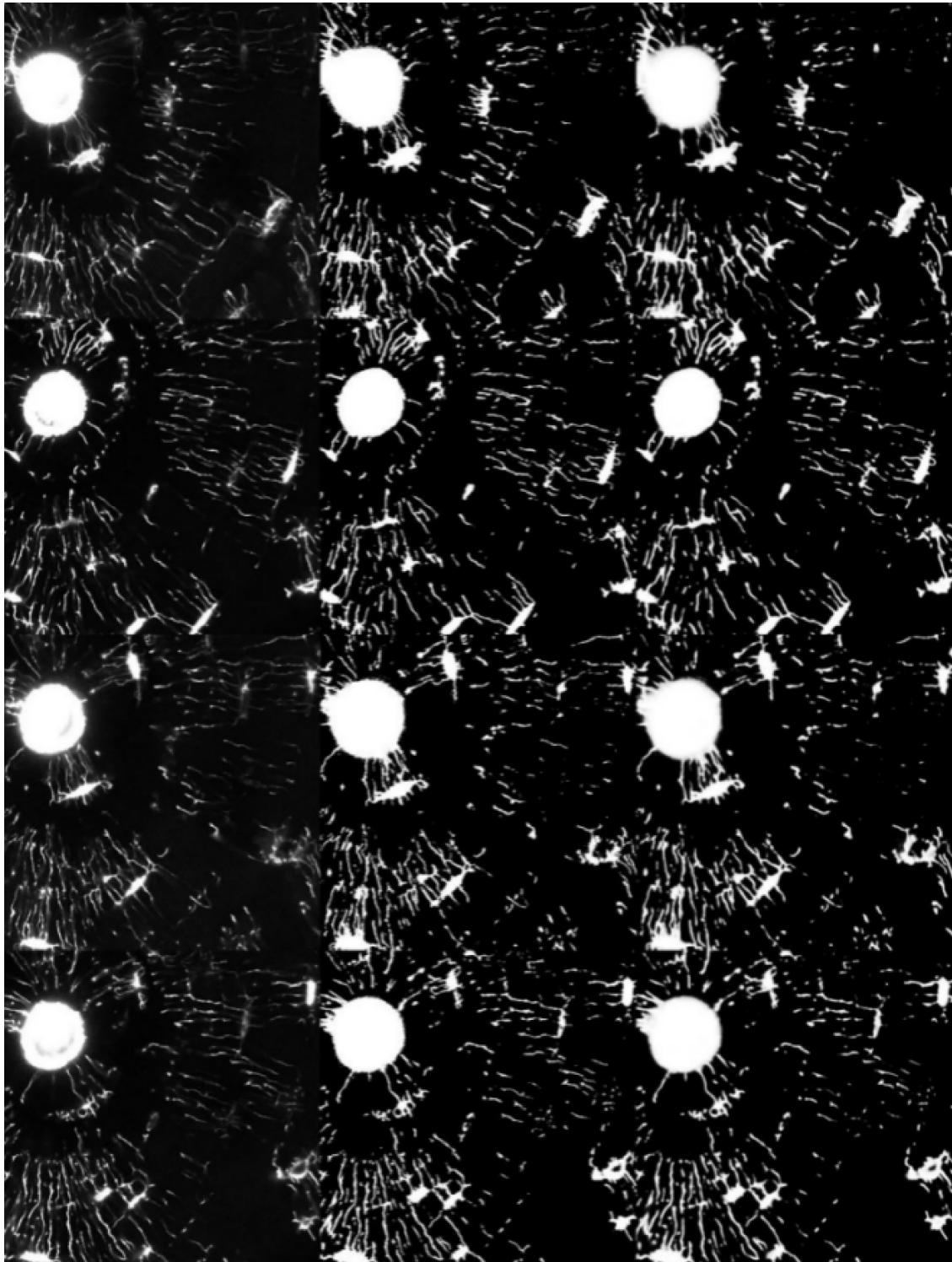
- Since the loss value of the first epoch of the training was too large compared with the error value, the two calculated values of the first epoch were deleted to draw a diagram.
- There was an interruption during the training. After the recovery of the training, the two adjacent epochs would have very similar loss values and error values. We thought that there was no reference significance, so we deleted the two calculated values under one of the epochs and drew the picture.

## 2.Comparison of training and test data

---

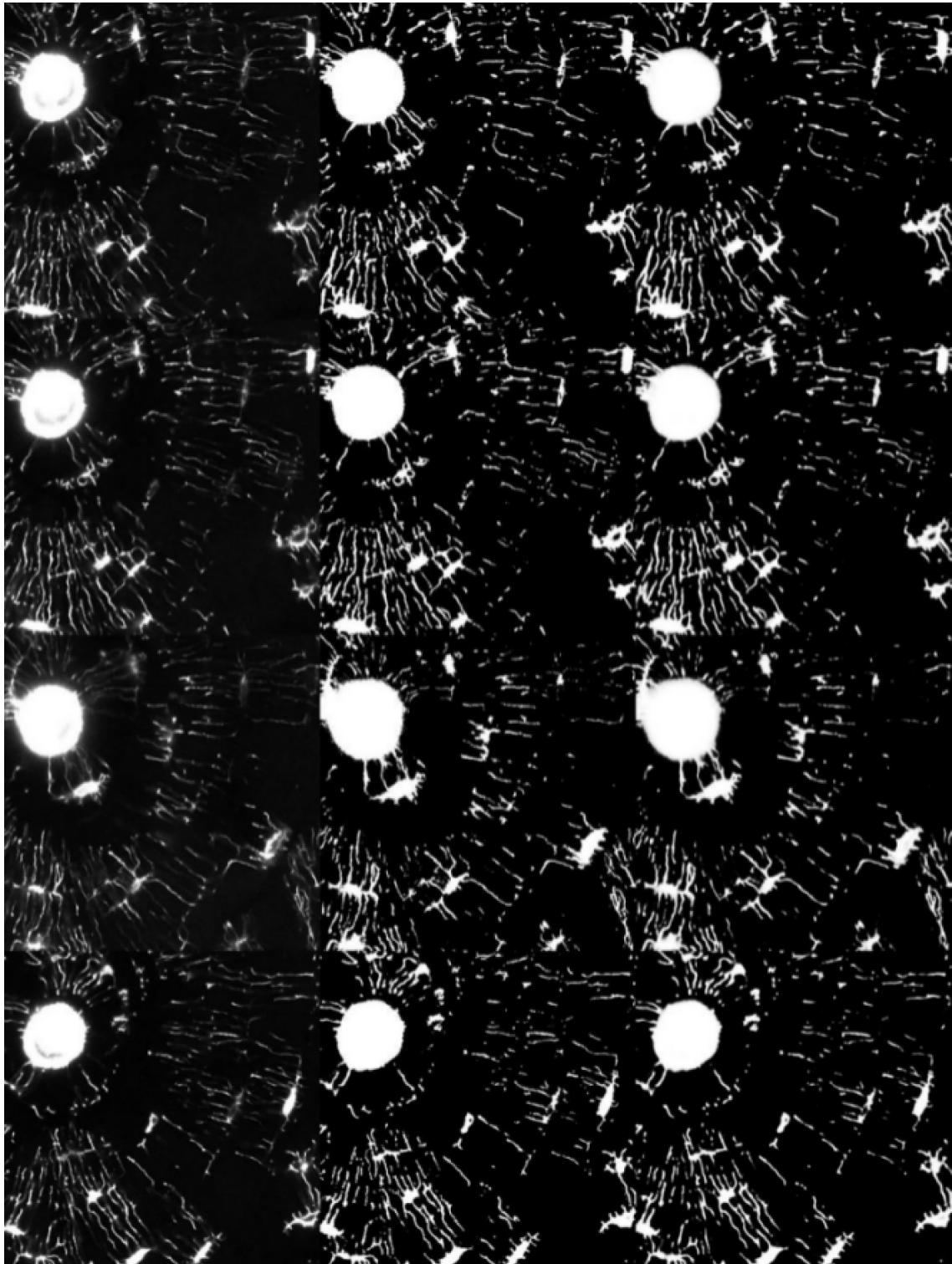
1. Performs on training data and original prichture(Verification Batchsize = 4)





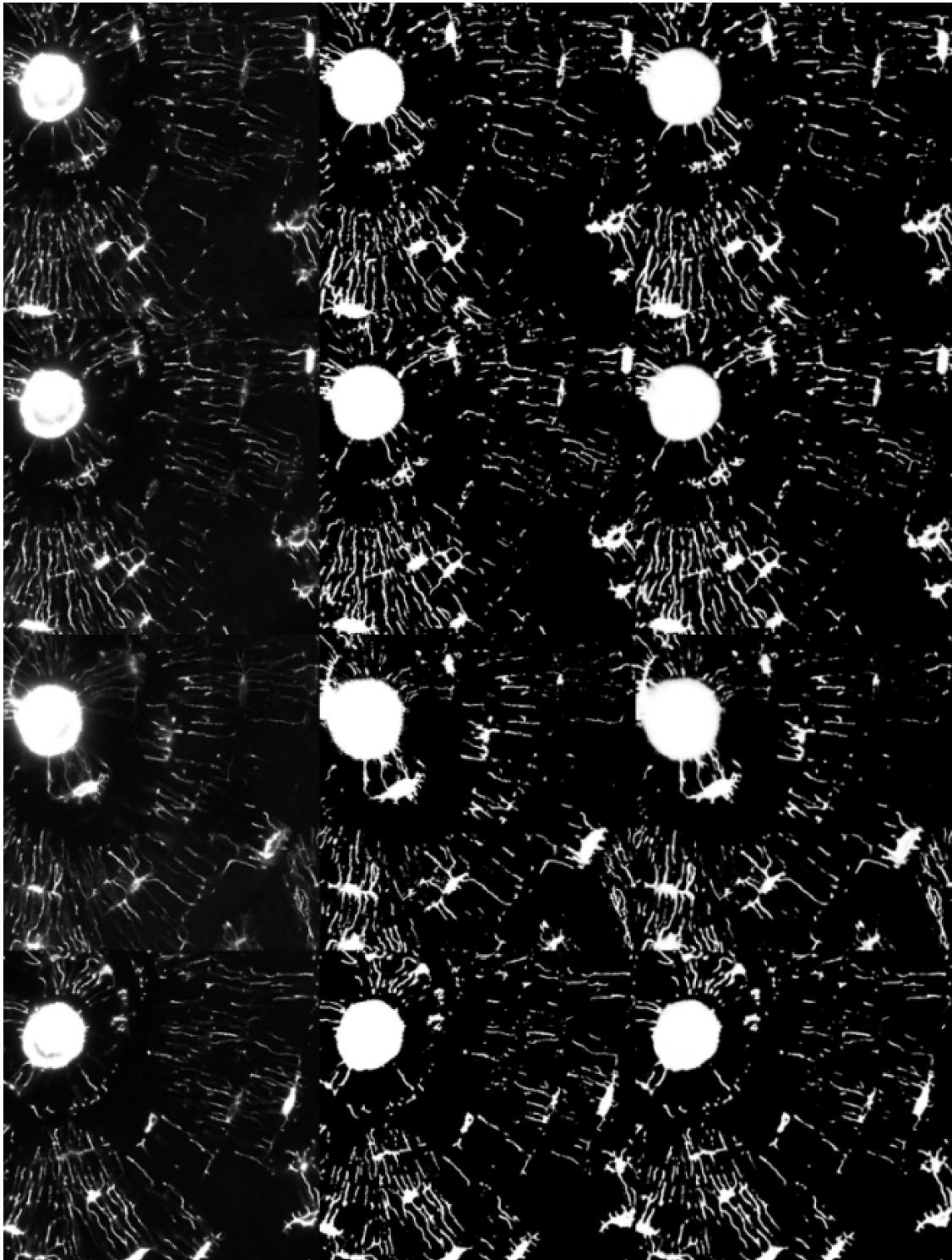
(epoch5)

---



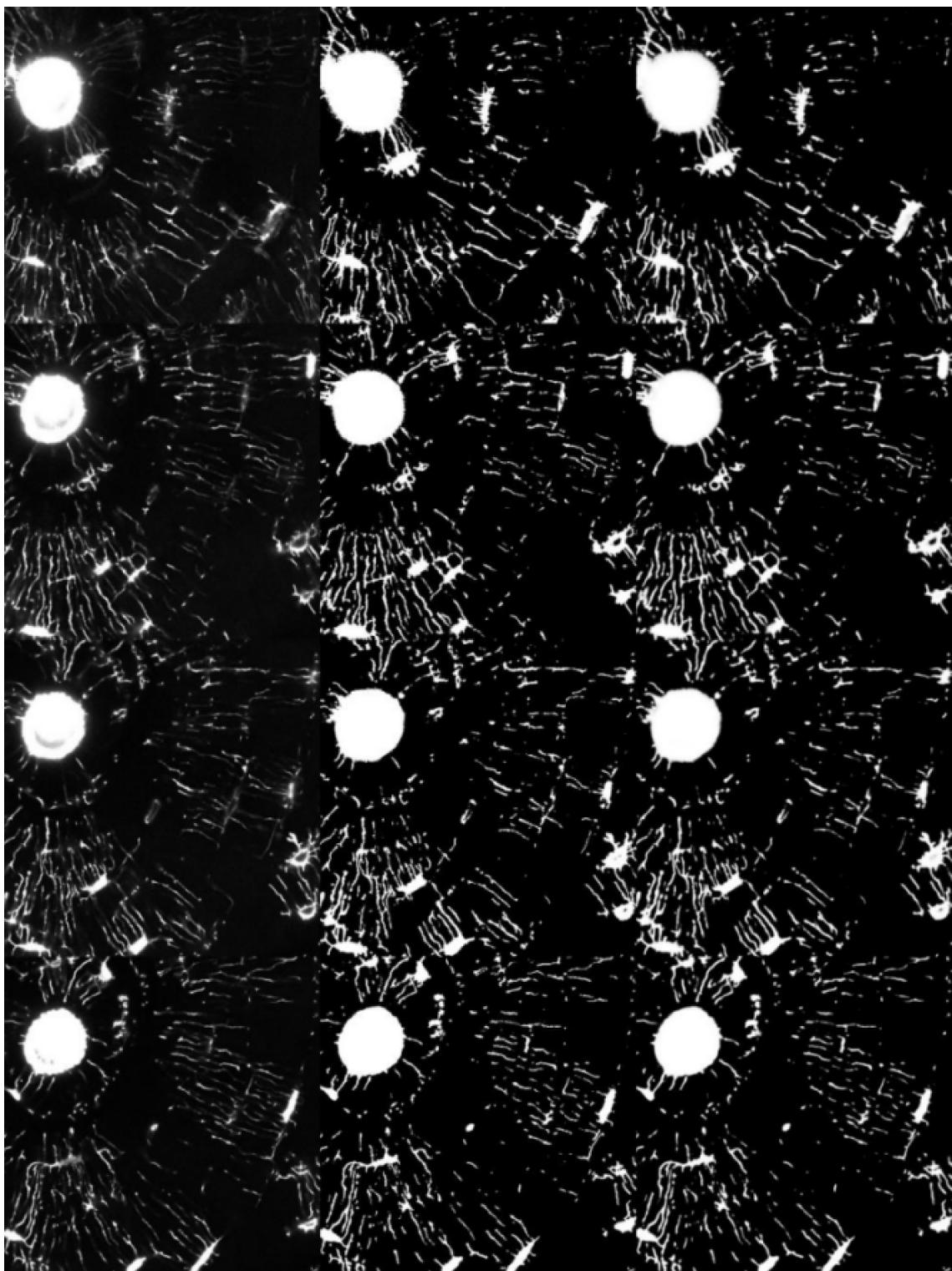
(epoch 10)

---



(epoch 15)

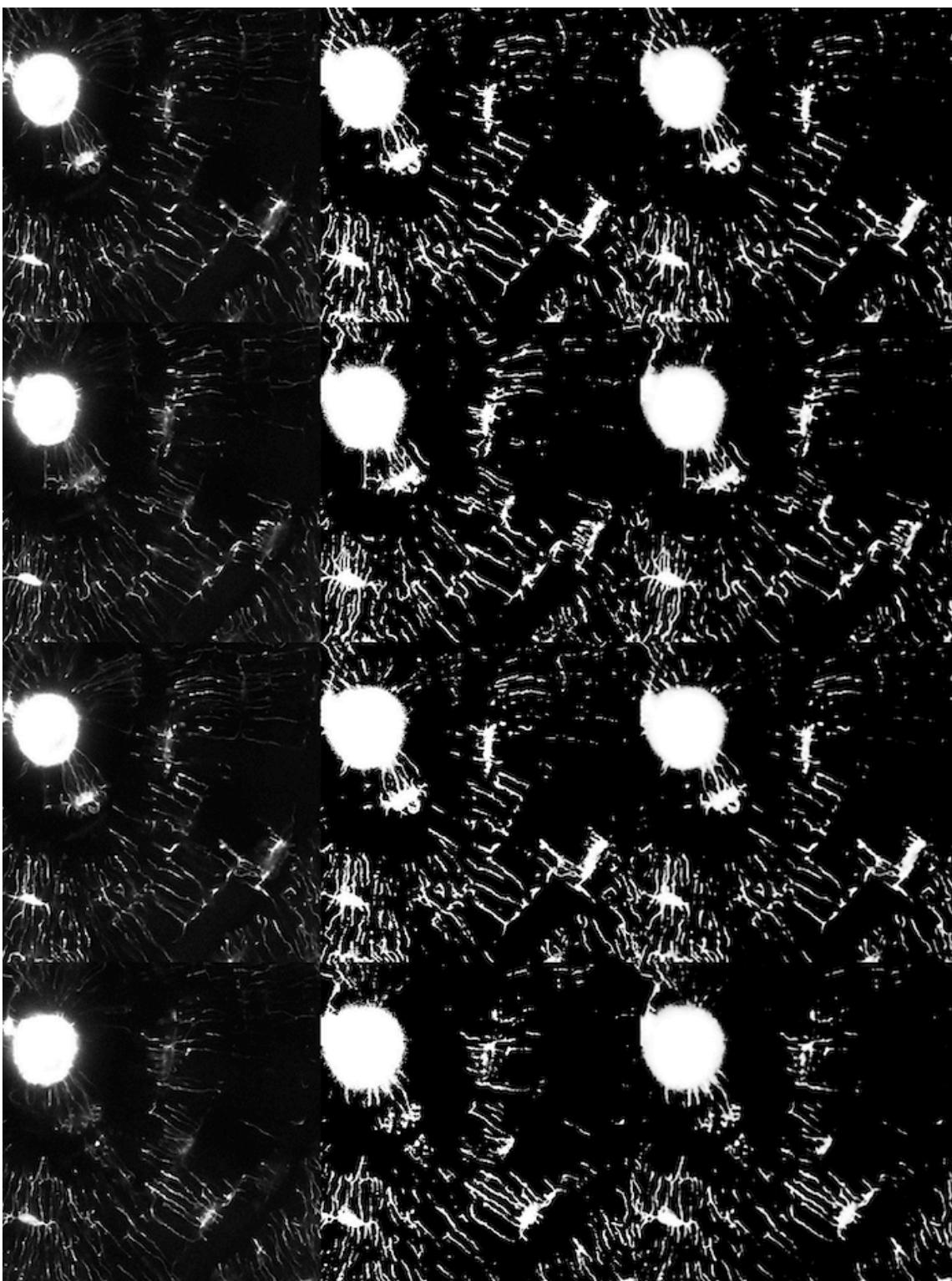
---

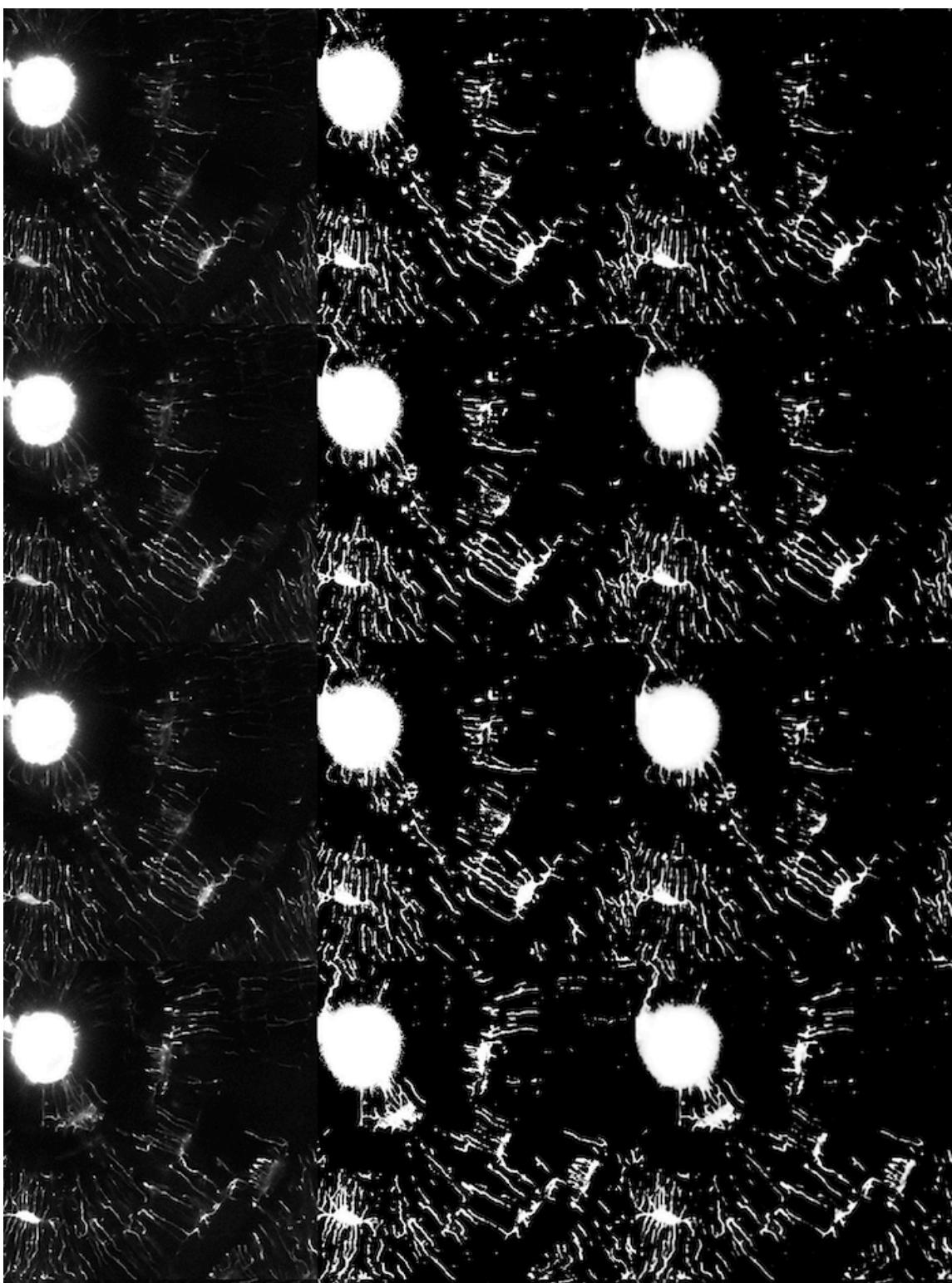


(epoch20)

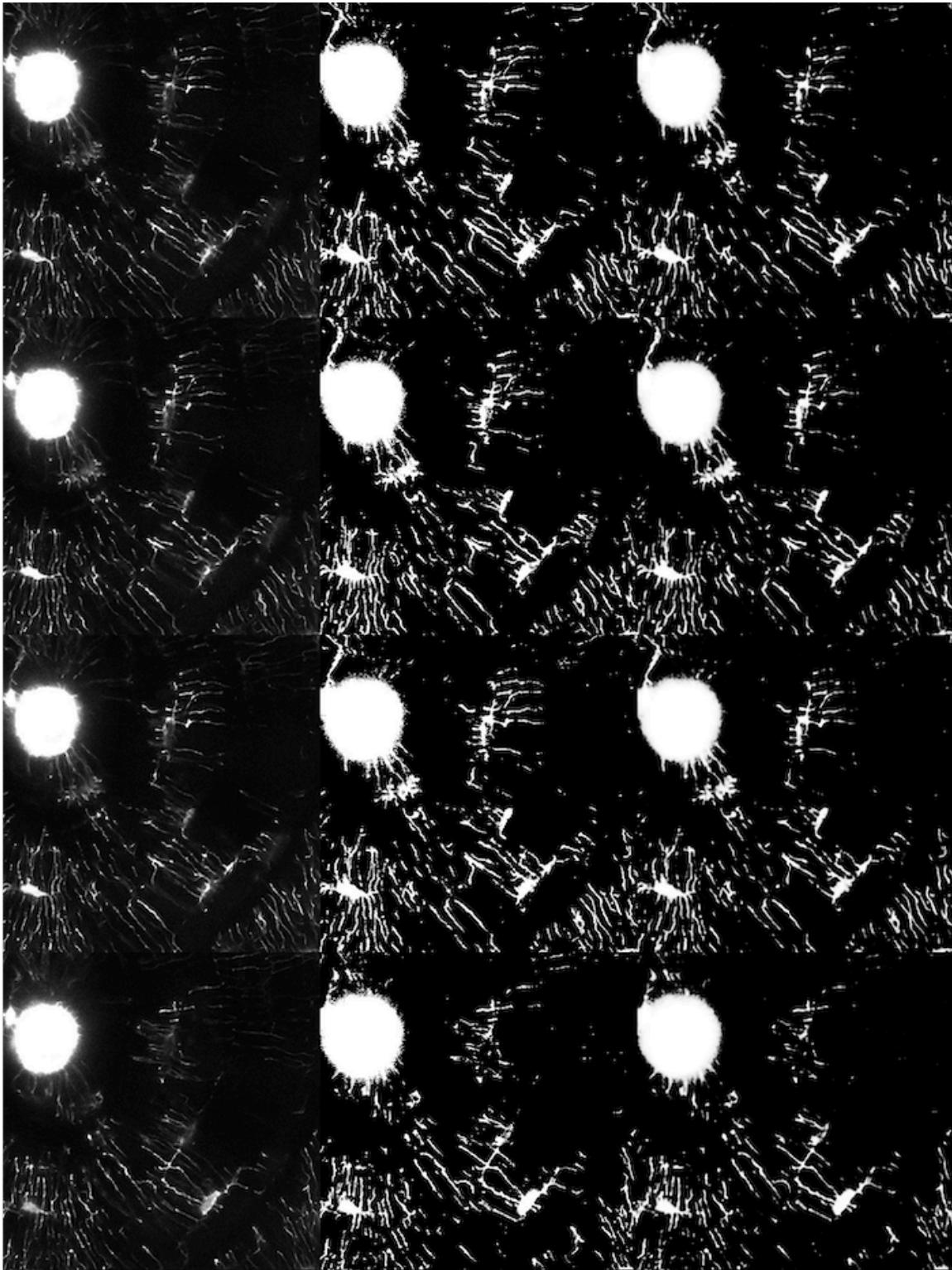
---

2. Performs on test data and original prichture(Verification Batchsize = 4)





---



### 3. To be improved

---

1. From the loss function images of the test set and the training set, it can be seen that the loss of both decreased significantly with the increase of epoch. But neither converges. We realized that we might get better results by continuing our training.
2. As for the error calculation of the test set, we have tried to set an appropriate threshold to export the predicted picture of the test set into a binary image. However, the average error of the training set will decrease with the increase of the threshold value. When the threshold

is 255, the mean error reaches a minimum of 7.979%. However, when the image represented by the binary matrix is displayed, it is obvious that it does not match the label image. On the contrary, the image processed by a smaller threshold can better correspond to the label image. This is a problem I don't understand.

**For instance**

*label picture*      *prediction threshold = 255 (with minimum error rate)*      *prediction threshold = 127*

