1. Result

1.1 Visualize the learning progress

Training parameter
Optimizer:SGD
Momentum:0.9
LR:0.01

I used the visdom to visualize the learning progress, the result is shown below:

Trian loss

After retraining the whole model by 10,000 steps, the train loss is 0.01

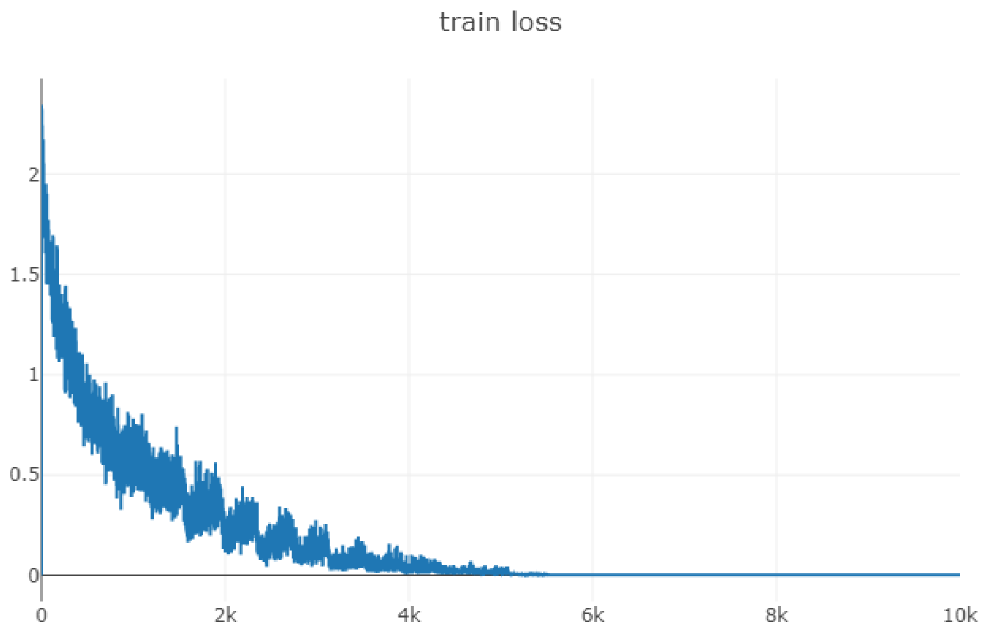


Fig.1 train loss

Train acc

After retraining the whole model by 10,000 steps, the train acc is 100%

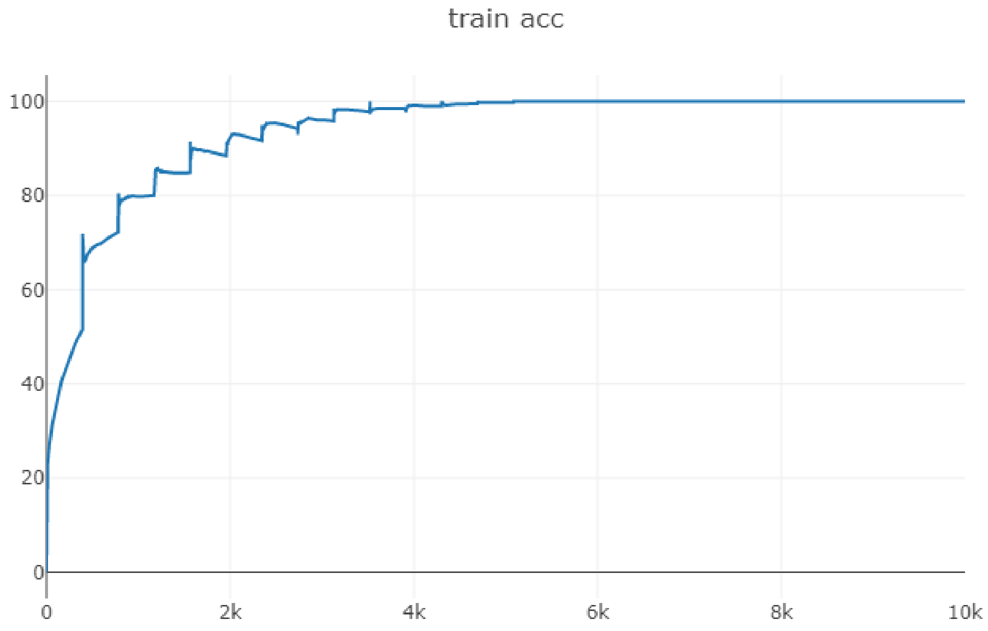


Fig.2 train acc

Test acc

After retraining the whole model by 10,000 steps, the test acc is 86.9%

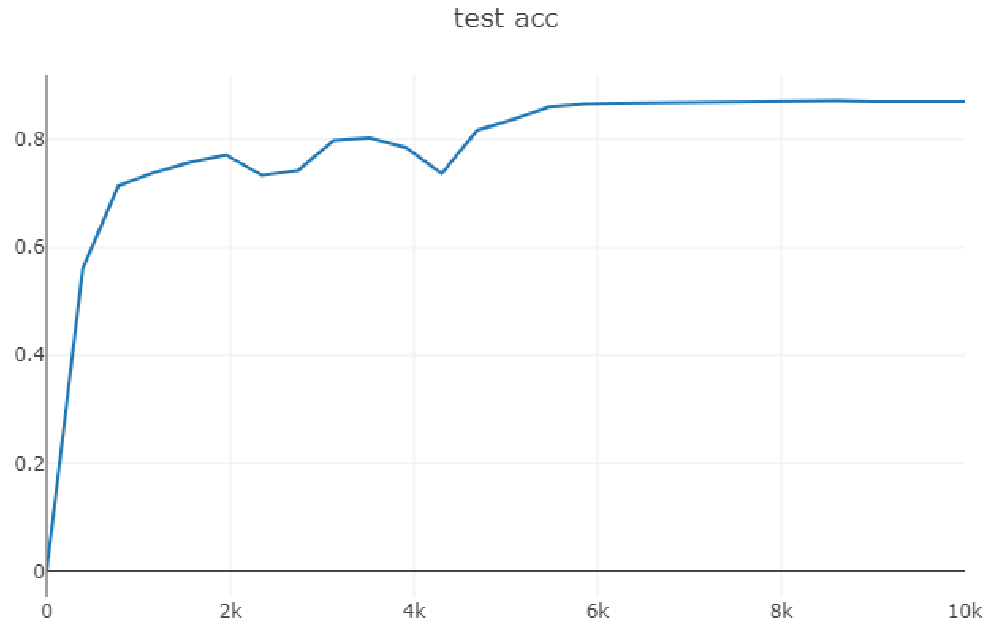


Fig.3 test acc

1.2 Filter visualization

After retraining the whole model by10,000 steps , I saved the model called net_10000.pth. We can visualize the 64 filters in the layer'Conv1' by reload the parameters of this model. As shown in Fig.4:

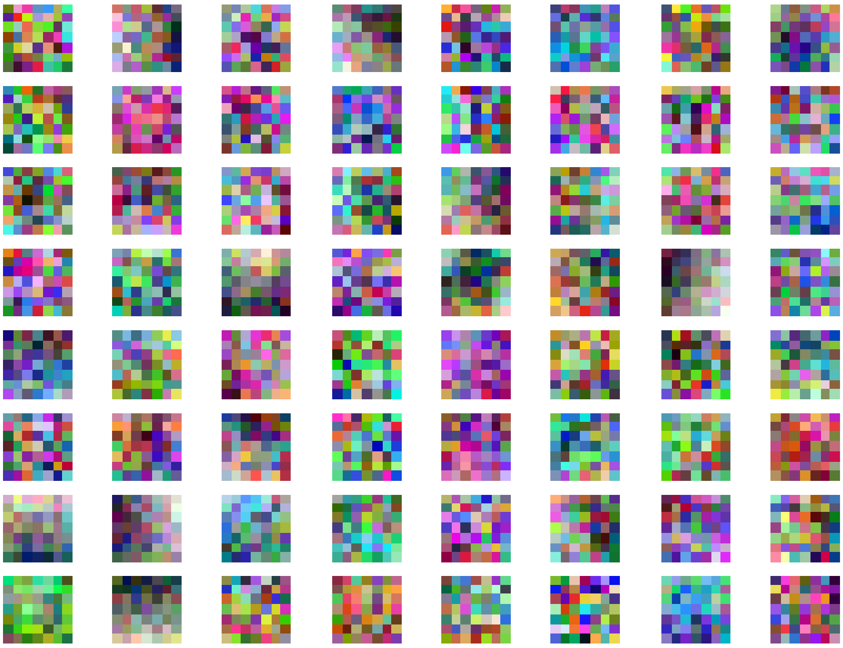


Fig.4 Filter visualization

1.3 Feature mapping visualization

First, we defined model “LastLayer” and “FirstLayer” in the resnet.py. Second ,we got the parameters that have been trained by reload the model called net_10000.pth. The last, we can visualize the feature map.

Conv1_feature_map

Original image   Conv1_feature_map

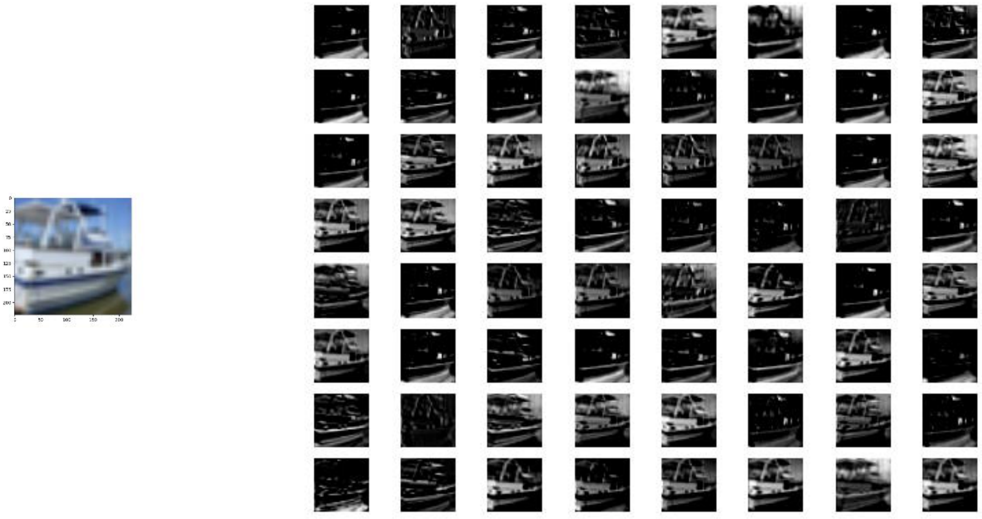


Fig.5 Conv1_feature_map

Conv5_feature_map

Original image   Conv5_feature_map

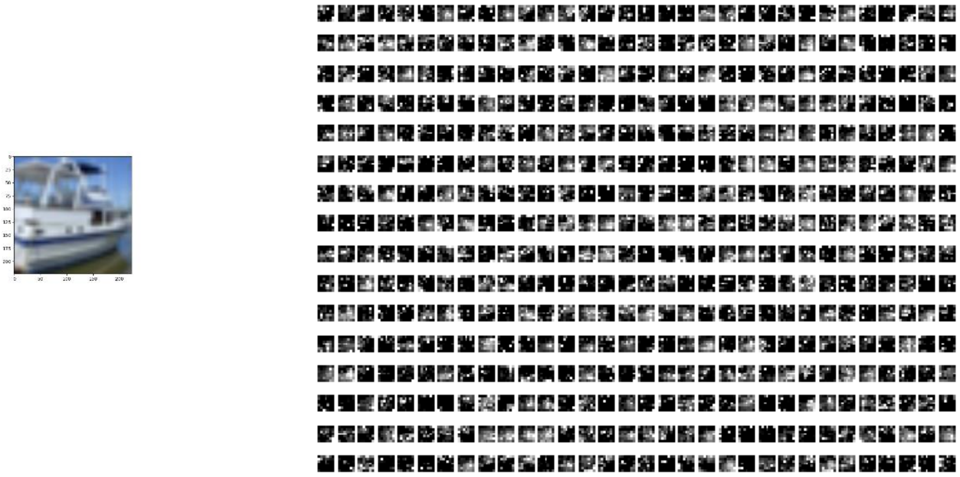


Fig.6 Conv5_feature_map

Conclusion

We can know the connection between the original image and the feature map. Each channel learned some different features from original image. With the deepening of network , the features learned are more and more local.

1.4 reconstructed patterns

reconstruction conv1

As shown in Fig.7, I randomly selected four images to show the results. I used all 64 channels of conv1 to reconstruct it. The image on the left is the reconstructed image, and the original image is on the right. The reconstructed image is similar to original image.

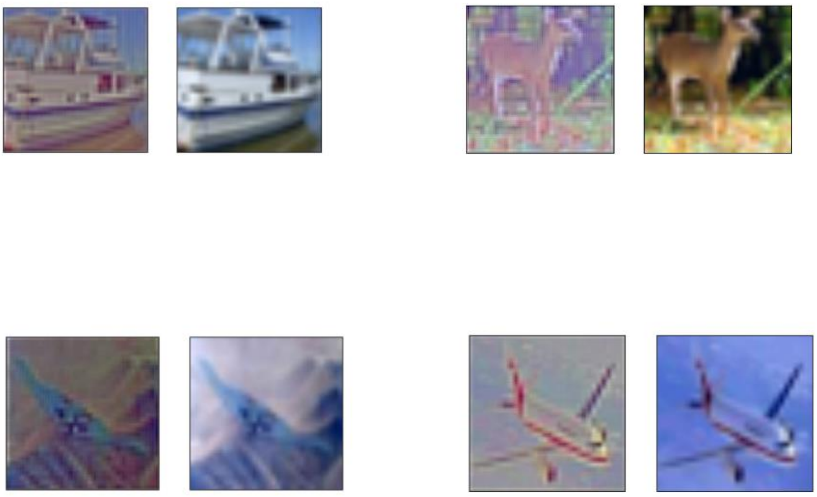


Fig.7 reconstruction conv1

As shown in Fig.8, I randomly selected two images to show the results. I used per channel of conv1 to reconstruct it.

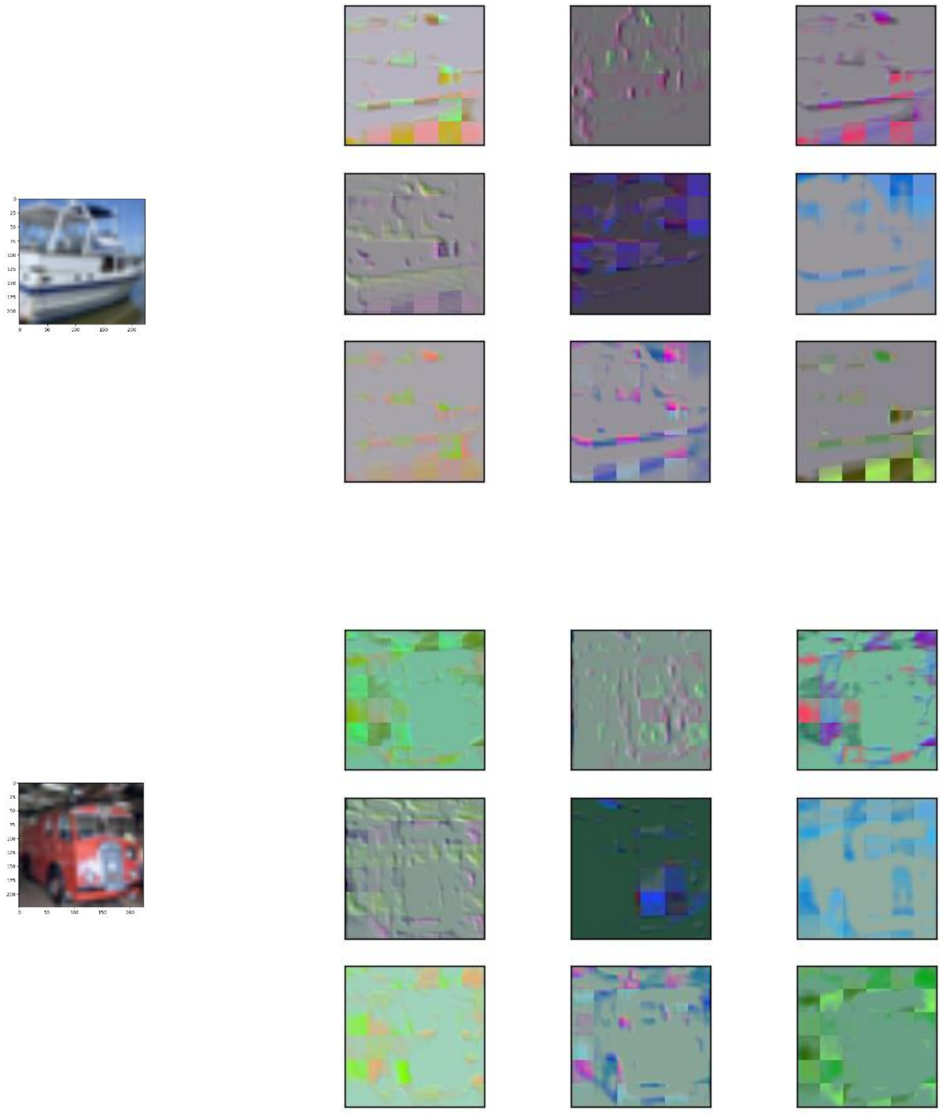
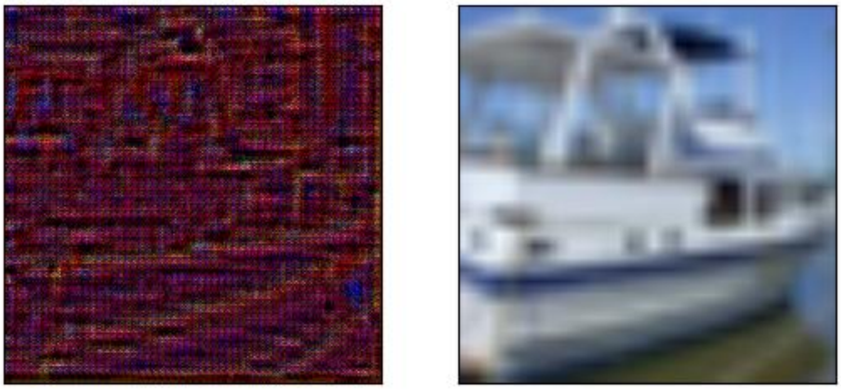


Fig.8 reconstruction conv1 per channel

reconstruction conv5



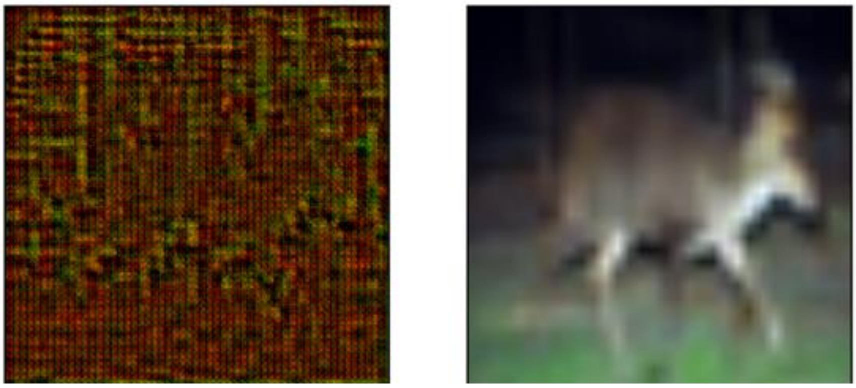
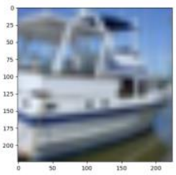
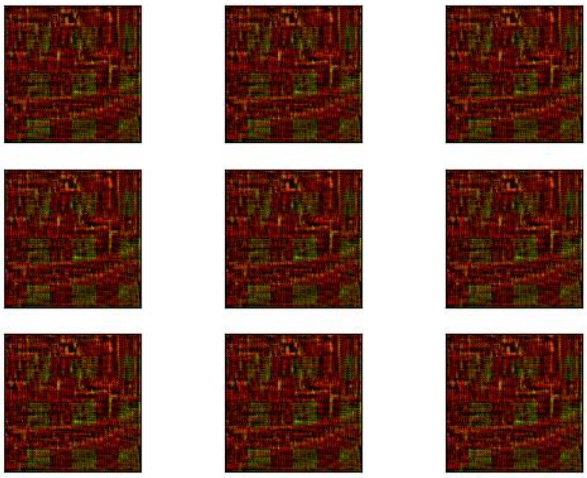


Fig.9 reconstruction conv5





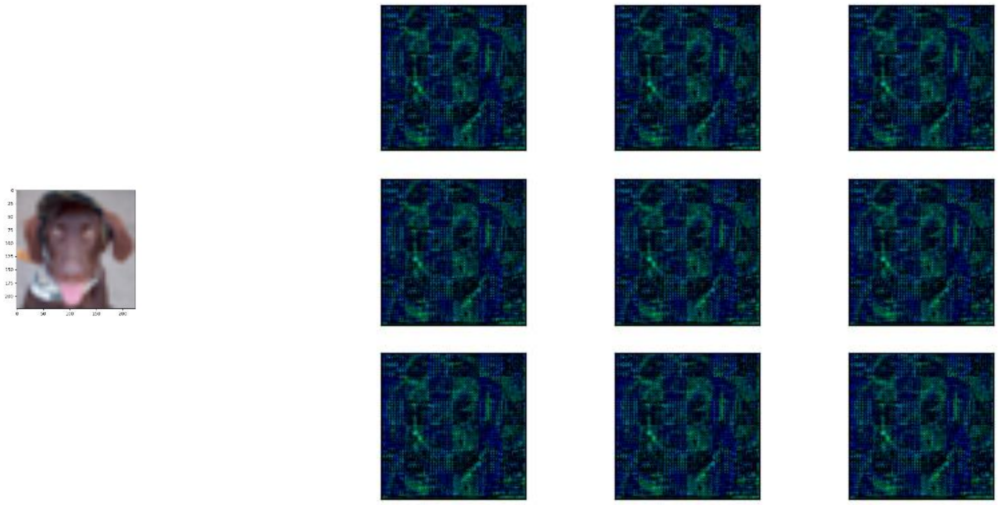
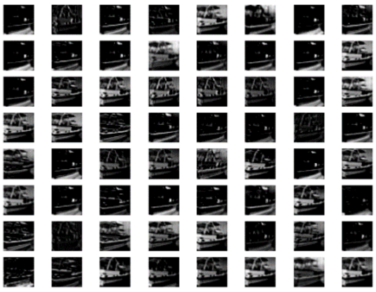


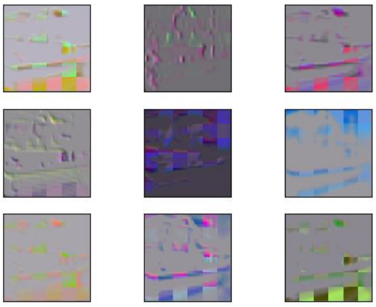
Fig.10 reconstruction conv5 per channel

Conclusion

Conv1_feature_map

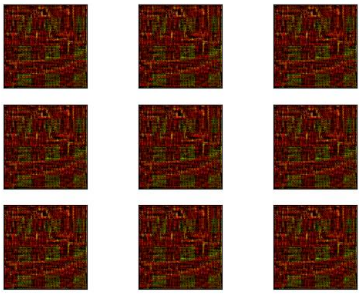


reconstruction conv1



Conv5_feature_map

reconstruction conv5



According to the results, the reconstruction of each channel retains some patterns of the original image.

2. Feedback

- **Time I spend for this assignment**

  About 80 hours or more. I have been doing this assignment in my spare time in the last two weeks.
- **Comments for this course**

  AI is a required course, instead of an elective course. Maybe, teacher should consider some students whose research direction not related to artificial intelligence.

  My research fields is traditional video coding. I don't have the basis of artificial intelligence. Somtimes , I can't understand what the teacher said. So, I hope the teacher will talk more about the basis of artificial intelligence.
- **Comments for this assignment**

  This assignment was a big challenging for me so that it took me a lot of time. I learned a lot in the process of coding. But I suggest the amount of assignment should be smaller because it cost me much time.
- **Suggestion for the following lectures**

  I suggest the following lectures tends to discuss interesting ideas rather than in-depth implementation. Because although my research has nothing to do with artificial intelligence, I can be inspired by these interesting ideas.

Reference

resnet.py

[1] https://pytorch.org/docs/stable/_modules/torchvision/models/resnet.html

visualization.py

[1] https://github.com/grishasergei/conviz
[2] https://github.com/kvfrans/feature-visualization
[3] https://github.com/utkuozbulak/pytorch-cnn-visualizations