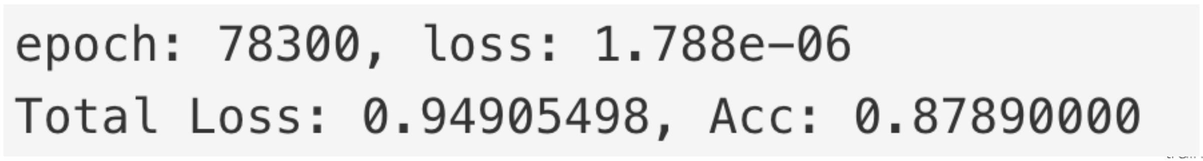
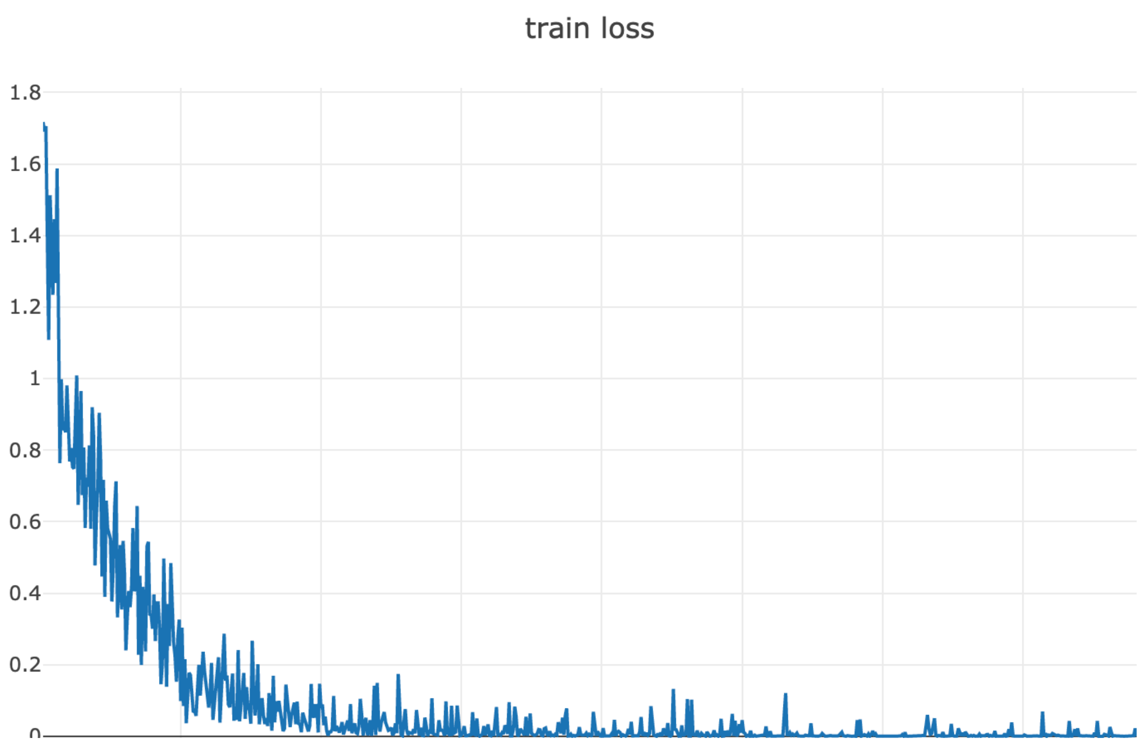
# AI_HOMEWORK_1_REPORT

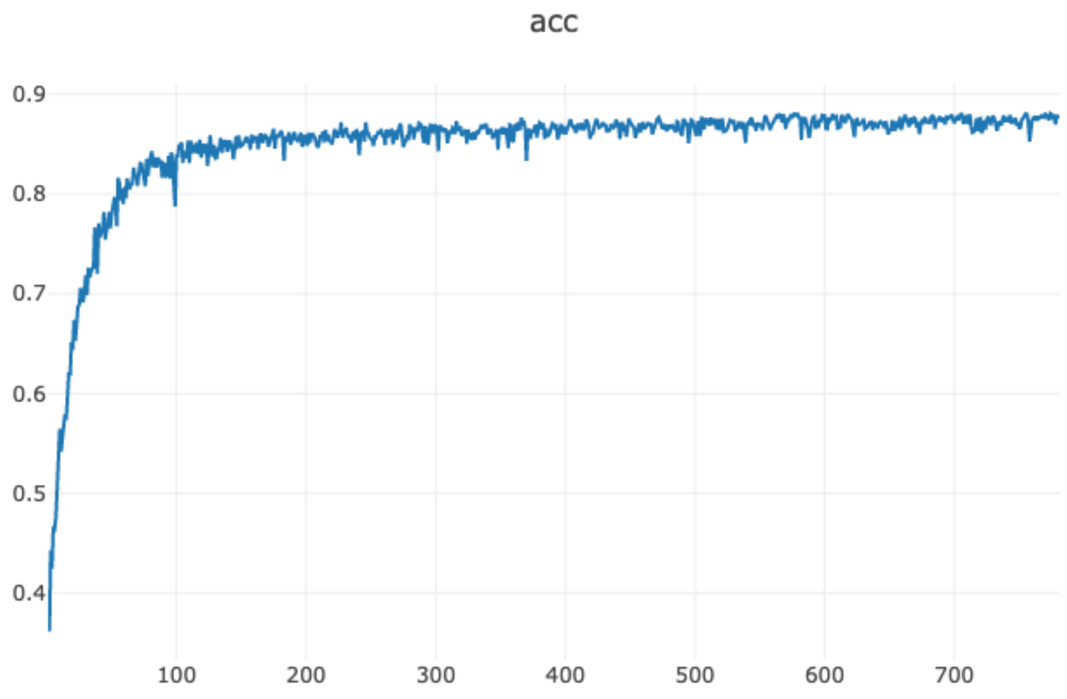
## 1.My own ResNet work

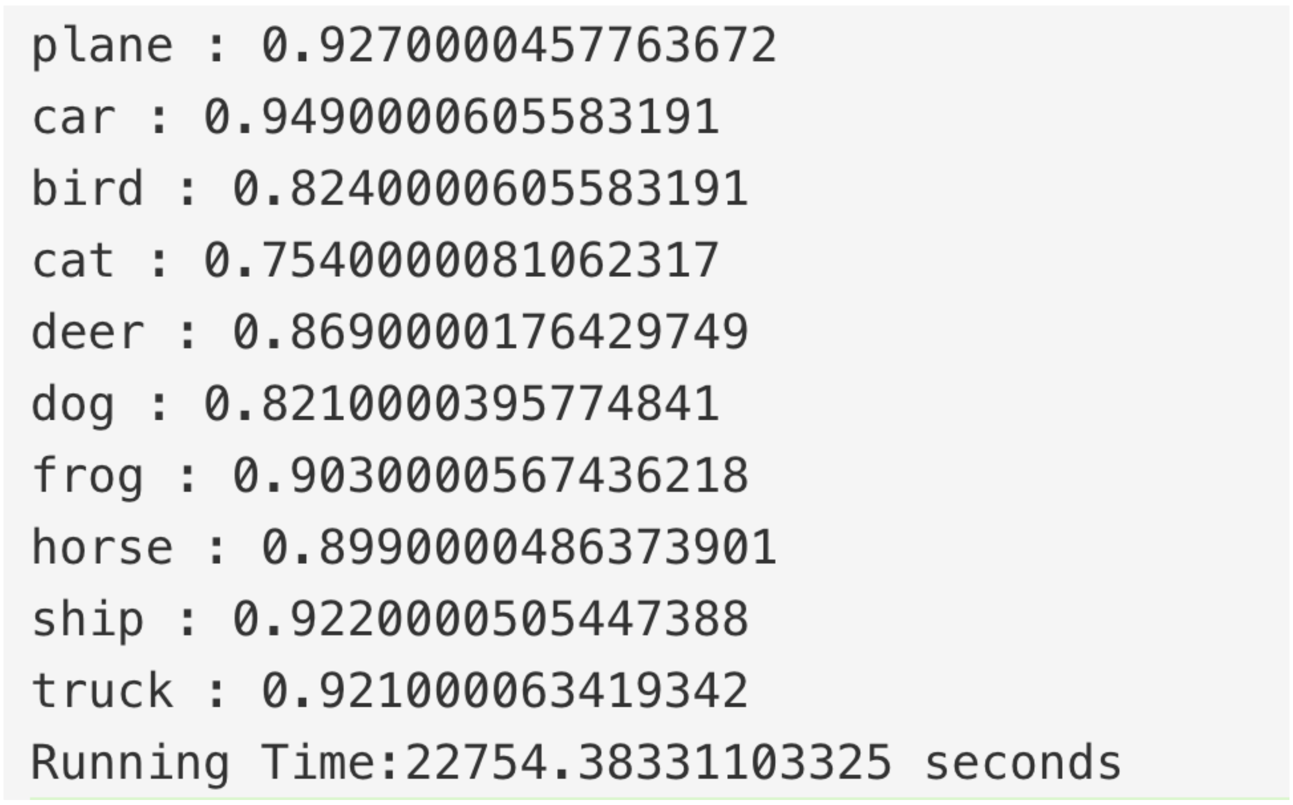
- My own ResNet model is the same as the model which is shown in the Table 1 in the assignment 1.The code of my own model is in the REsnet.py.
- For training my model,I used 78300 iterations.The specific results are shown in the figures below.



```
epoch: 78300, loss: 1.788e-06
Total Loss: 0.94905498, Acc: 0.87890000
```







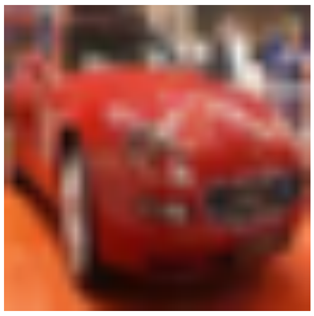
```
plane : 0.9270000457763672
car : 0.9490000605583191
bird : 0.8240000605583191
cat : 0.7540000081062317
deer : 0.8690000176429749
dog : 0.8210000395774841
frog : 0.9030000567436218
horse : 0.8990000486373901
ship : 0.9220000505447388
truck : 0.921000063419342
Running Time:22754.38331103325 seconds
```

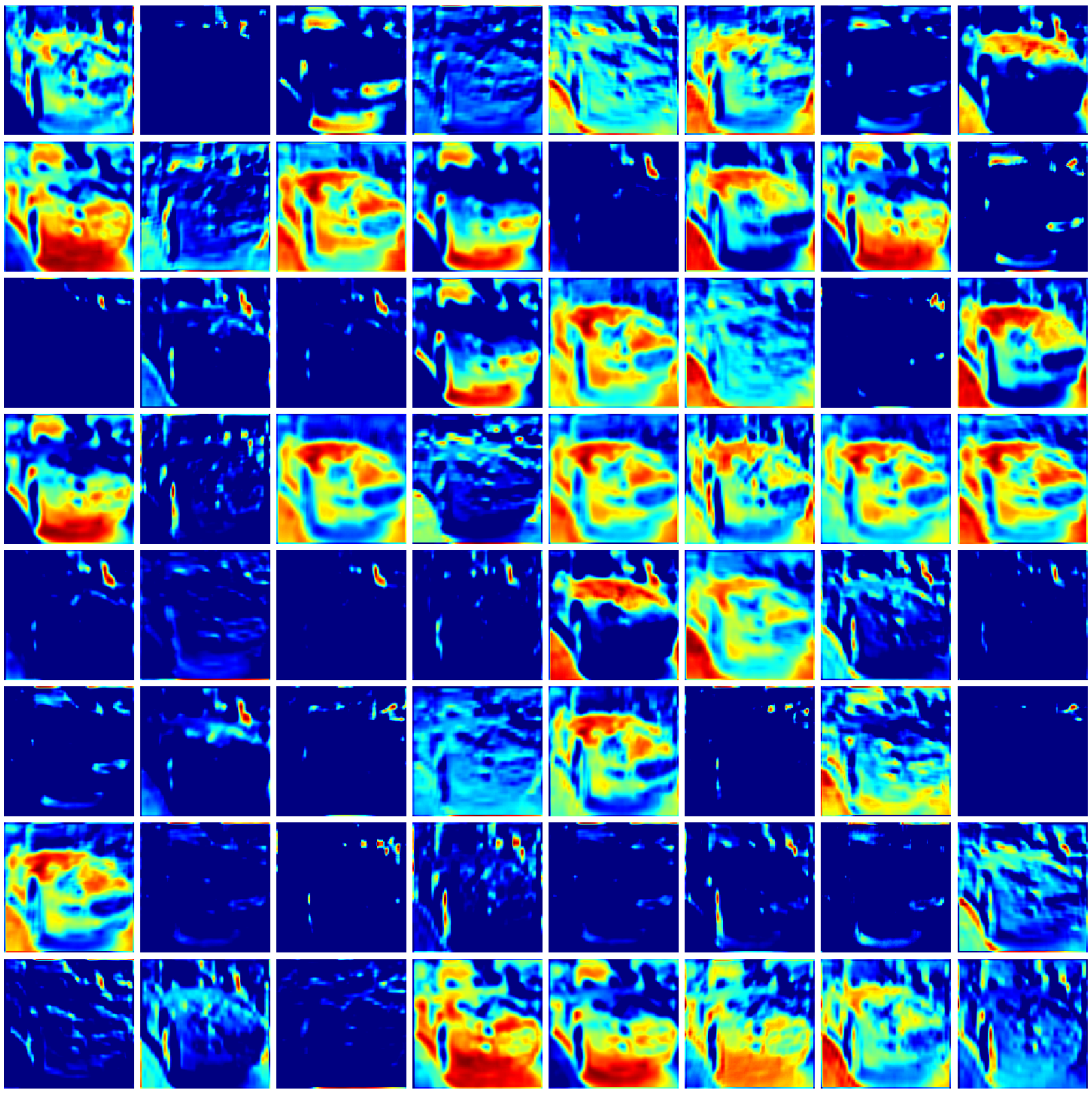
I also did specific tests for different accuracy rates for ten different categories.In terms of the specifics of the iterative process,you can see it in the cn_1022_f.out.

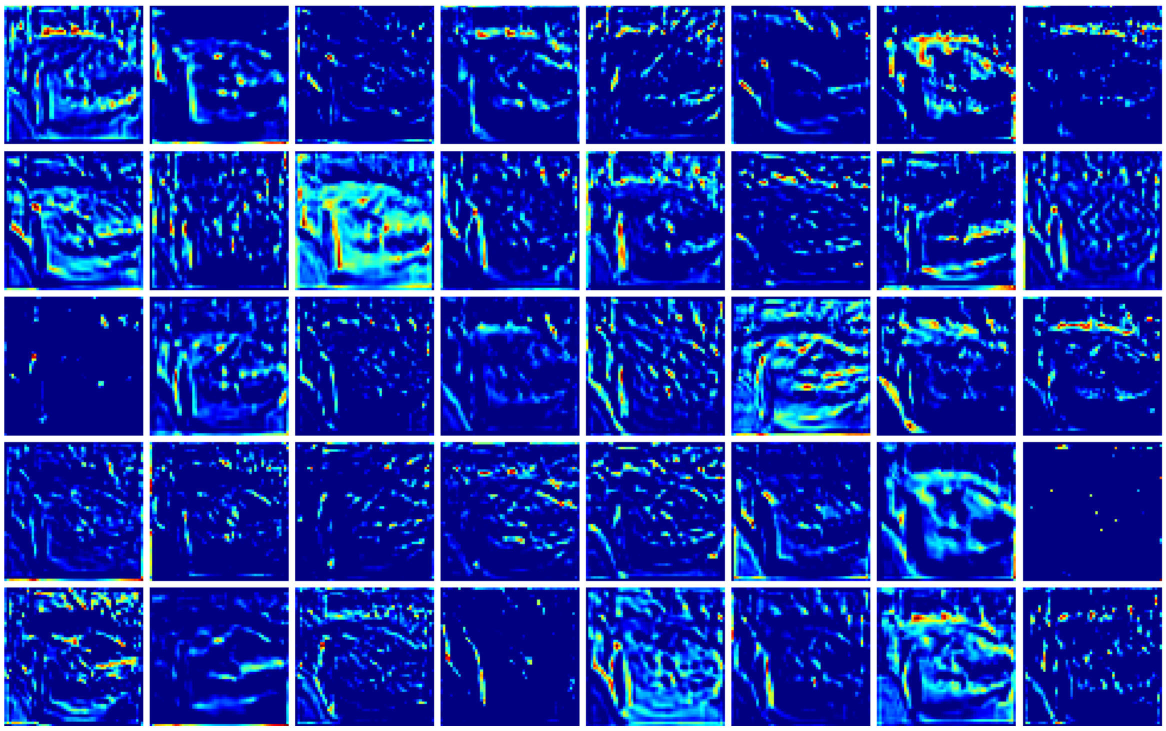
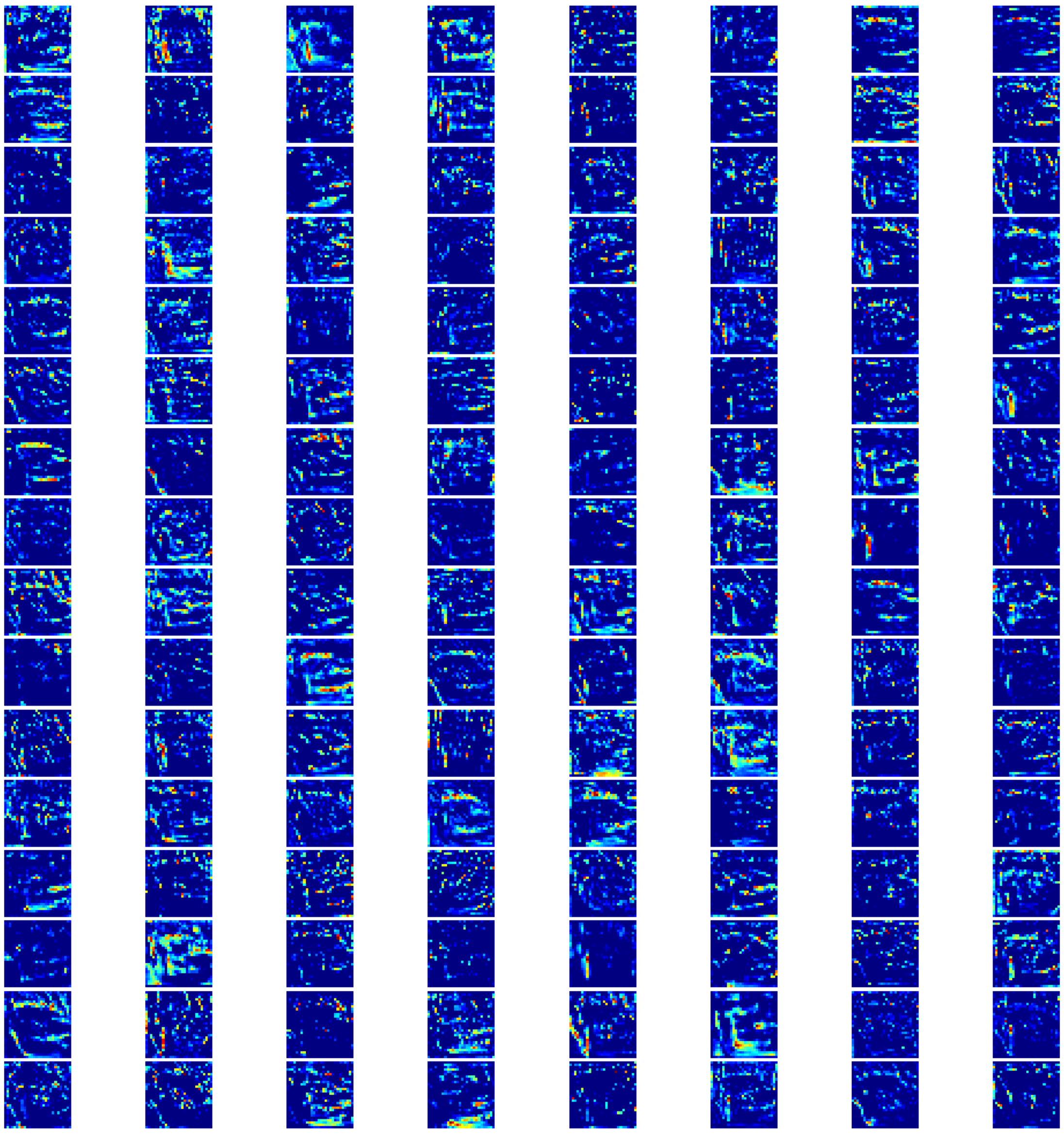
- Filter visualization

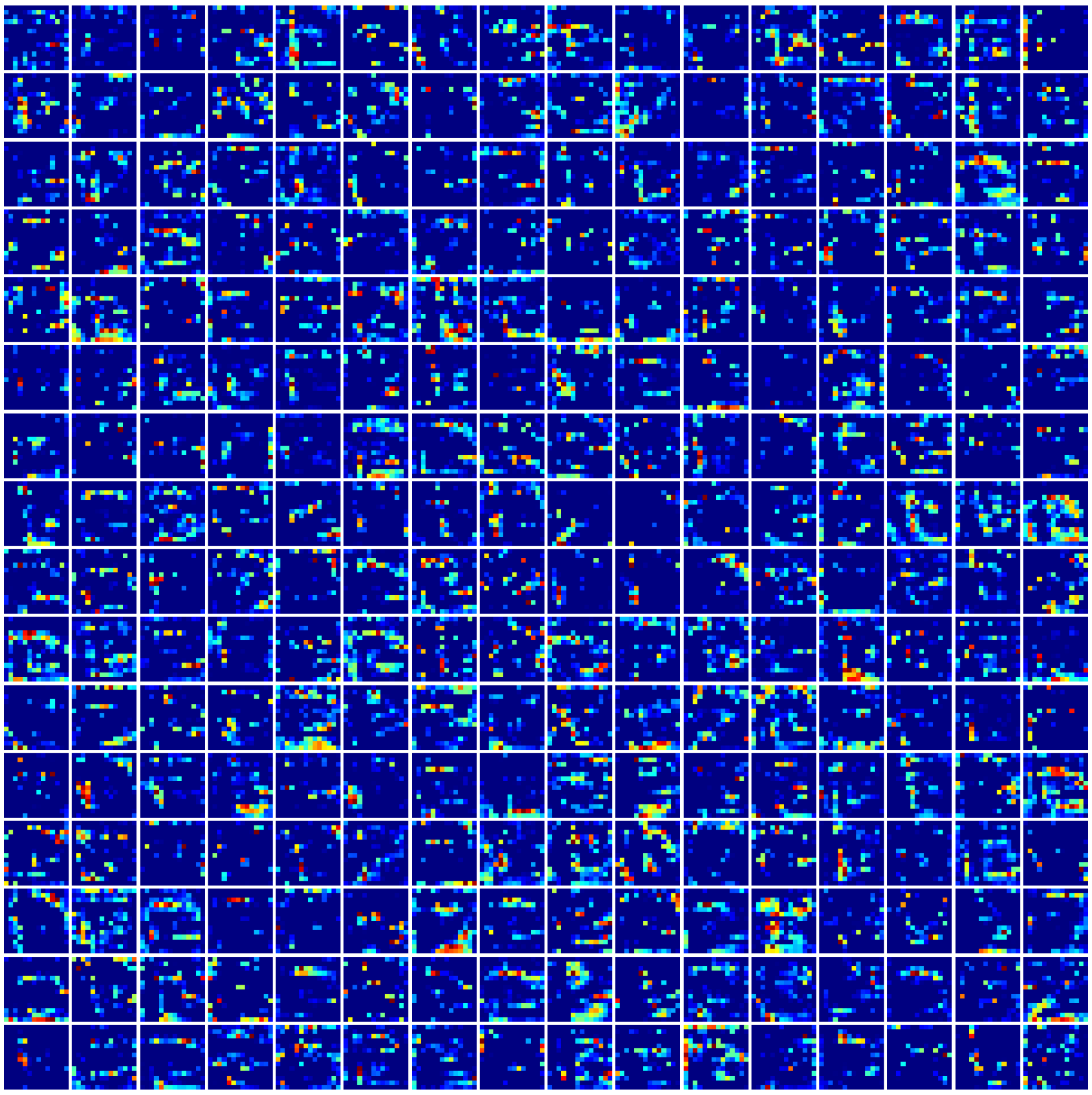
For seeing more clear images, you can look at the filter_visualization.png and filter_visualization_1.png.As for the concrete implementation of the code， you can find it in the visualization_filter.py.

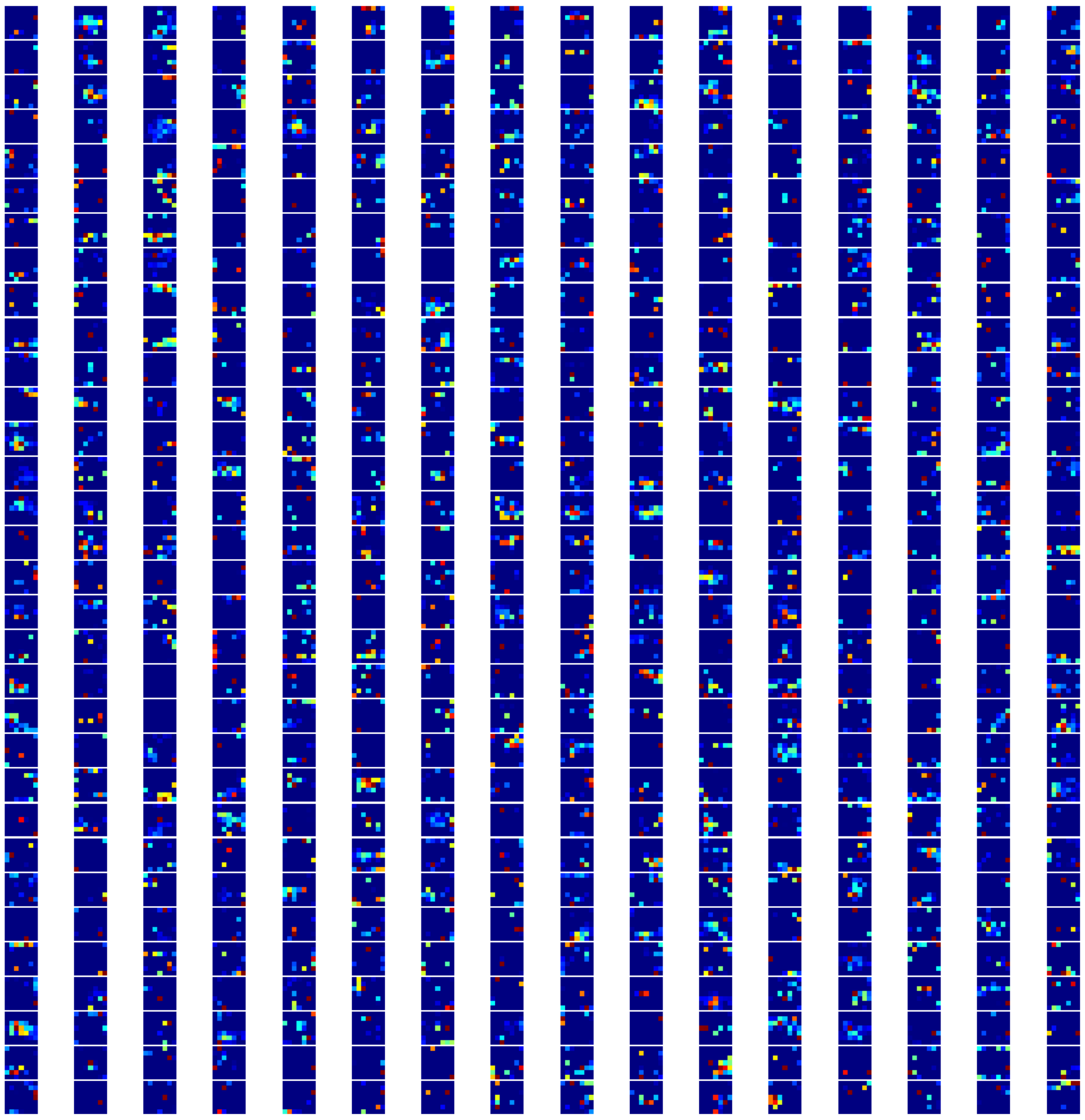
- Feature mapping visualization









The feature map of convolution layer is realized from one to five to visualization, including the feature maps of layers'Conv1'、'Conv2'、'Conv3'、Conv4'、Conv5'.As for the concrete implementation of the code， you can find it in the resnet_visualization.py.By the way,You can find the source file of these pictures in this package.

## Feedback

- I already spent  more than forty hours for this assignment.
- I think the teacher should more concentrate in some specific knowledge.
- I think the description of the next assignment should be clearer.

## Note

If you want to run these code on a server, you should first run the following command：

- `ssh -L 18097:127.0.0.1:8097 username@remote_server_ip`
- `python -m visdom.server`

Then enter the following address in your local browser:127.0.0.1:18097
By running the above command, you can see the process of  visualizing the loss function and accuracy.
What is more, if you want to run visualization_filter.py on a server ,you should first run the following command:

- `ssh -L 16006:127.0.0.1:6006 username@remote_server_ip`

- tensorboard —logdir=visual_weights(Of course, you should make sure you have created this package.)

Then enter the following address in your local browser:http://127.0.0.1:16006

## confusion

I have some confusion when I accomplish this assingment:

- When I trained the model, there was a gradient explosion.Then I didn't adjust any parameters, and the next time it went away.I'm not sure if this is the norm for training models.
- When I did specific tests for different accuracy rates for ten different categories.I find whatever I use 100 ,1000 or more than 10000 iterations, the accuracy of cat is significantly lower than other categories.I even tried to tune the model and parameters, the accuracy of cat is still significantly lower than other categories.And I don't find some reasonable explanations, so I really hope Mr.chen can have some ideas about this.

## Reference

1. visualization_filter.py.
   - Reference code link: https://github.com/TingsongYu/PyTorch_Tutorial
2. resnet_visualization.py
   - Reference code link: https://blog.csdn.net/weixin_40500230/article/details/93845890