

1)

转化 dataset 为 28\*28\*1

```
transform=transforms.ToTensor()
```

第一个卷积层:

```
nn.Conv2d(1, 25, kernel_size=12, stride=(2, 2), padding='valid'),
```

第二个卷积层:

```
nn.Conv2d(25, 64, kernel_size=5, stride=(1, 1), padding=2),
```

池化层:

```
nn.MaxPool2d(kernel_size=2, stride=2
```

全连接层:

```
nn.Linear(4*4*64,1024)
```

```
nn.ReLU()
```

```
self.fc = nn.Linear(64*4*1024, num_classes)
```

2)

损失函数和学习率:

```
criterion = nn.CrossEntropyLoss()
```

```
optimizer = torch.optim.Adam(model.parameters(), lr=0.0001)
```

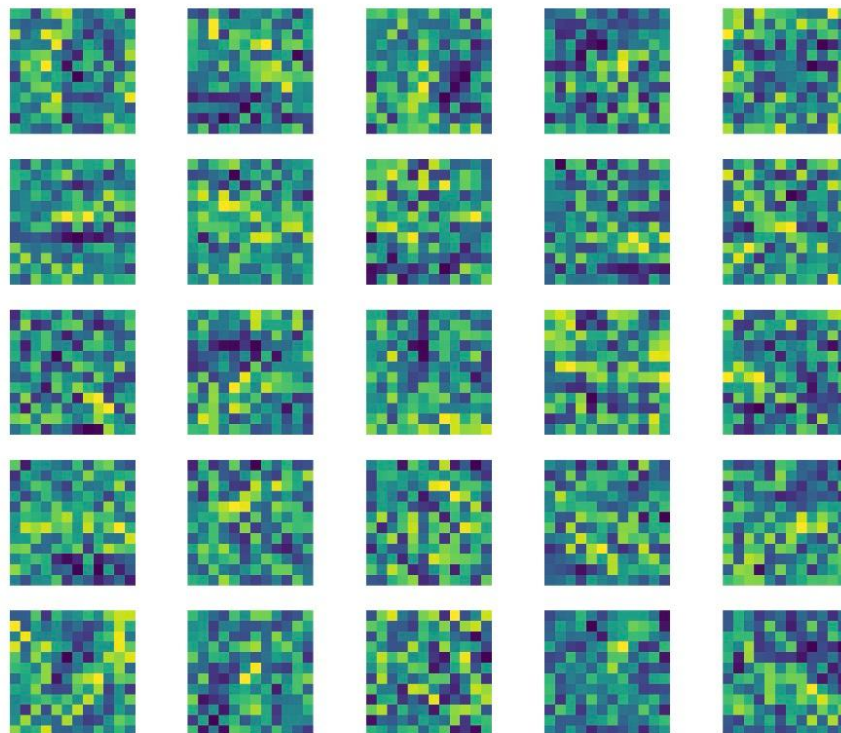
准确率:

```
format(100 * correct / total)
```

3) 训练 epoch=3

```
cnn (1) ×
Accuracy of the network on the 100 iterations train : 98.28667225481978 %
Accuracy of the network on the 100 iterations train : 98.28810720268007 %
Accuracy of the network on the 100 iterations train : 98.28953974895397 %
Accuracy of the network on the 100 iterations train : 98.28929765886288 %
Accuracy of the network on the 100 iterations train : 98.2890559732665 %
Accuracy of the network on the 100 iterations train : 98.29048414023373 %
Accuracy of the network on the 100 iterations train : 98.29190992493744 %
Epoch [3/3], Step [1200/1200], Loss: 0.0310
Accuracy of the network on the 100 iterations train : 98.29333333333334 %
Accuracy of the network on the 10000 test images: 98.48 %
```

#### 4) 可视化



5) 提交代码和回复:  
文件中 `cnn.py` 为代码文件。

