

pj2实验报告

1. 完整实现图像分类(模型训练和测试)

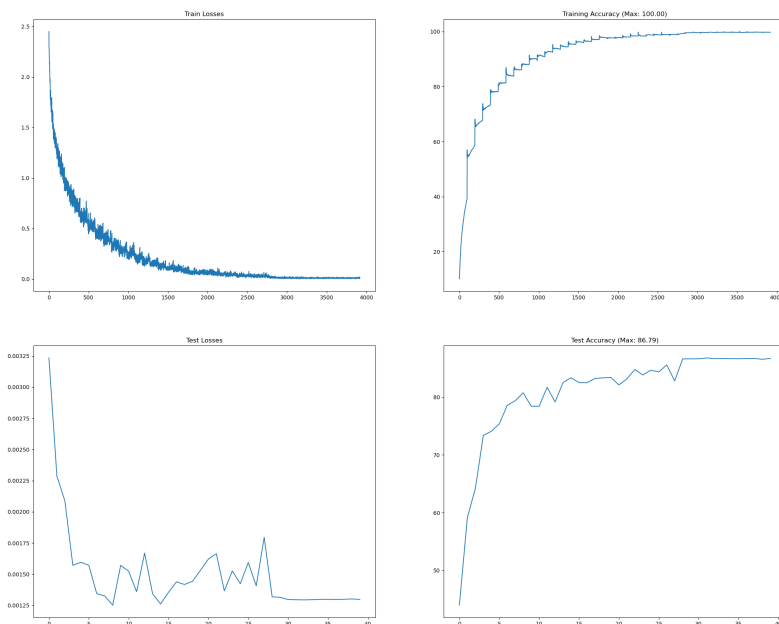
model_training关键代码：

```
1 # 定义损失函数
2 criterion = nn.CrossEntropyLoss()
3 # 清空梯度
4 optimizer.zero_grad()
5 # 前向传播：计算预测值
6 y_pred = model(data)
7 # 计算损失
8 loss = criterion(y_pred, target)
9 # 反向传播
10 loss.backward()
11 # 更新模型参数
12 optimizer.step()
```

model_testing关键代码：

```
1 #补全内容:获取模型输出，loss计算
2
3 # 获取模型输出
4 output = model(data)
5 # 计算损失
6 loss = F.cross_entropy(output, target)
7 test_loss += loss.item() # 累加损失值
```

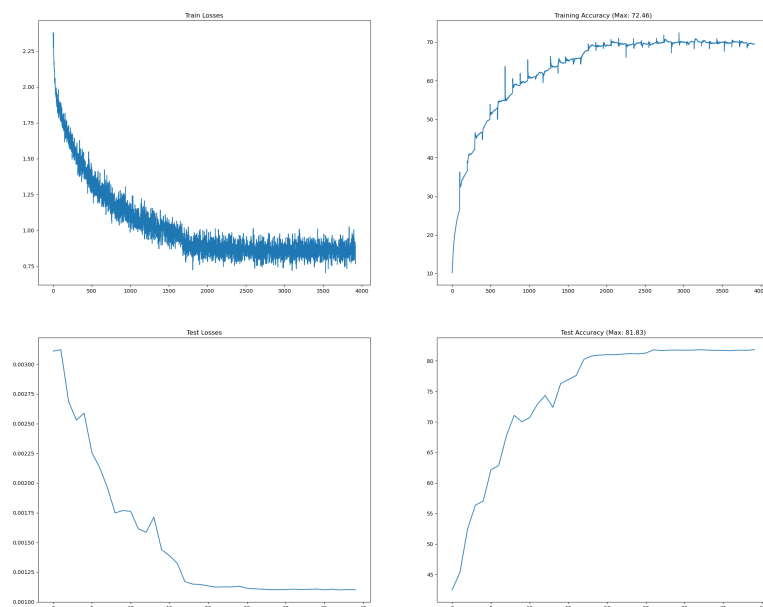
结果：



2. 数据增强技术的设计以提高准确性

关键代码：

```
1 transforms.RandomResizedCrop(32), # 随机裁剪并调整大小
2 transforms.RandomHorizontalFlip(), # 随机水平翻转
3 transforms.ColorJitter(brightness=0.2, contrast=0.2, saturation=0.2, hue=0.2),
  # 随机颜色抖动
4 transforms.RandomRotation(10), # 随机旋转
```



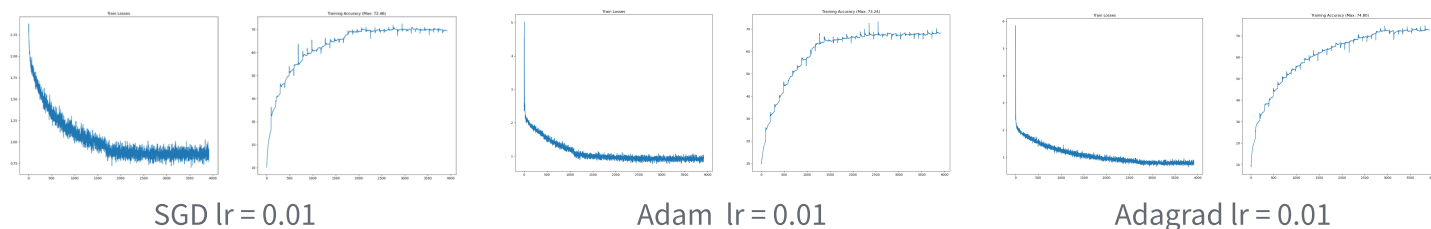
3. 优化器与学习率计划设计的分析与探讨

在phase2的基础上选择不同优化器(lr = 0.01)

	SGD	Adagrad	Adam
--	-----	---------	------

avg_test_loss	0.0010	0.0010	0.0013
max_test_accuracy(%)	81.83	83.45	78.72

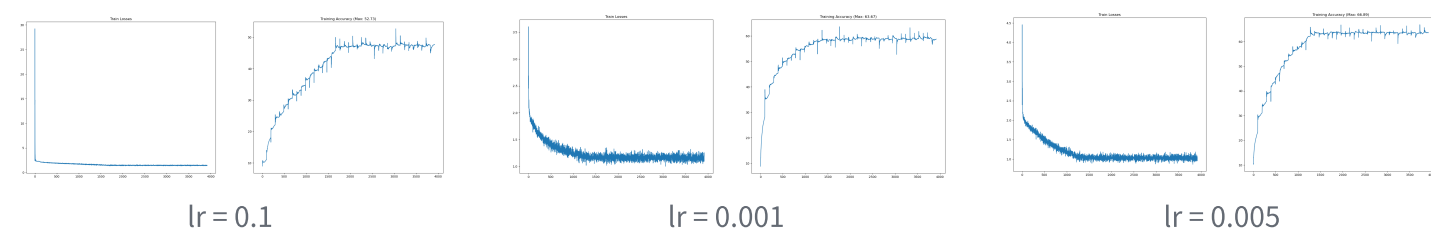
收敛速度：



在phase2的基础上选择不同学习率(optimizer = optim.Adagrad)

	0.1	0.01	0.005	0.001
max_test_accuracy(%)	56.61	83.45	75.39	71.04

收敛速度：

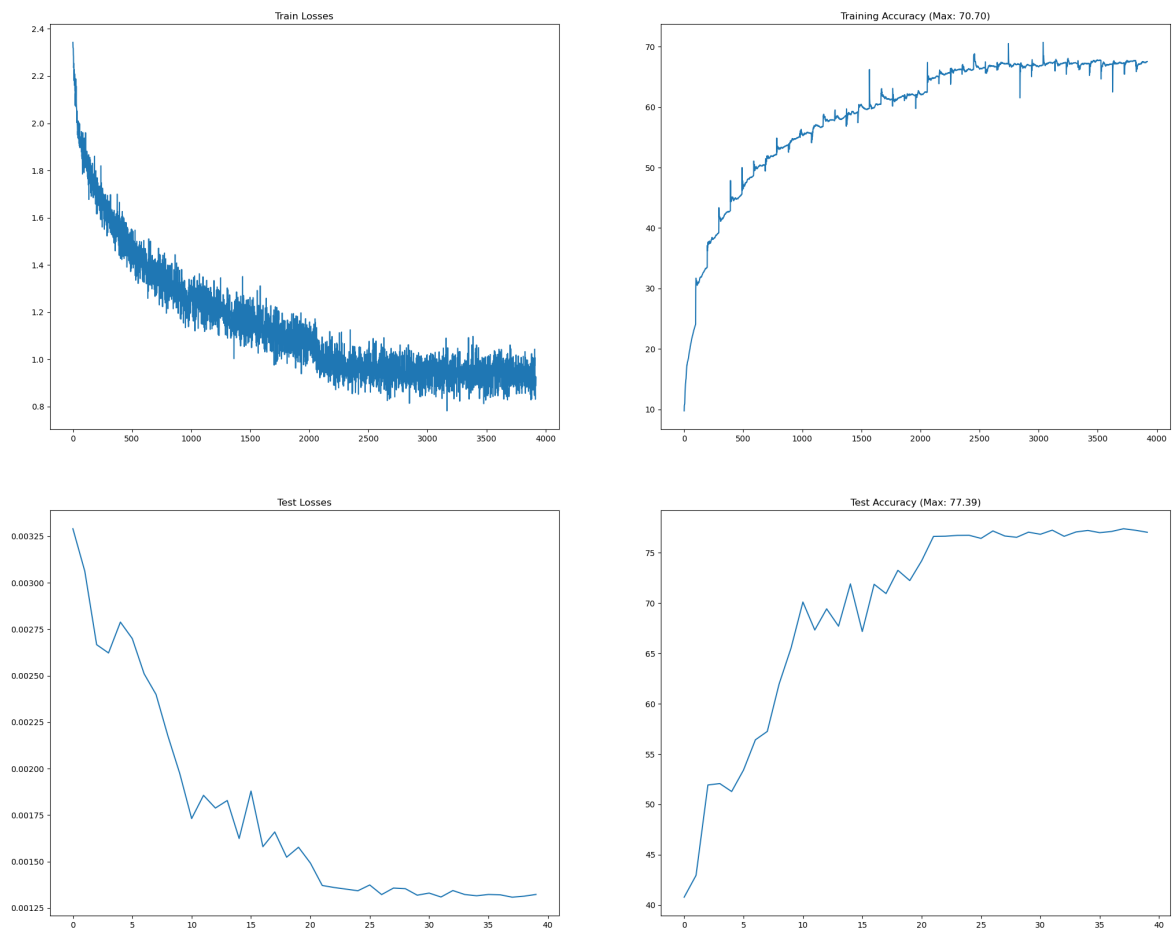


学习率越高，收敛越快；过高的学习率和过低的学习率都会导致准确度降低

不同优化器表现 收敛速度：Adagrad>Adam>SGD;准确度没有太大差别

4. 不同CNN架构的分析与讨论

max_training_accuracy = 70.70;max_test_accuracy = 77.39



VGG lr = 0.01

- 性能

分类精度：通常情况下，ResNet-18 在图像分类任务中表现更好。这是因为 ResNet-18 引入了残差连接，使得网络更深时可以更容易地训练。相比之下，VGG 虽然也可以达到较高的分类精度，但在深度增加时容易出现梯度消失或爆炸的问题，对训练造成了困难。

泛化能力：ResNet-18 通常具有更好的泛化能力，因为残差连接有助于减轻梯度消失的问题，从而使得模型更容易优化并具有更好的泛化性能。相比之下，VGG 由于较深的网络结构可能在一定程度上受到过拟合的影响。

- 复杂度

ResNet-18 拥有更少的参数数量。它通过使用残差块和全局平均池化来减少网络的参数数量，使得模型更加简洁高效。相比之下，VGG 的参数数量较多，采用了更多的卷积层和全连接层。