**深度学习实验报告**

实验2：深度学习框架熟悉

姓名：姚舜宇

学号：1190202107

班级：1903602

**实验环境：**

Windows11

Python3.8

PyTorch1.10.2

**实验内容：**

使用PyTorch实现AlexNet，并在Caltech101数据集上验证。

1. 数据读取

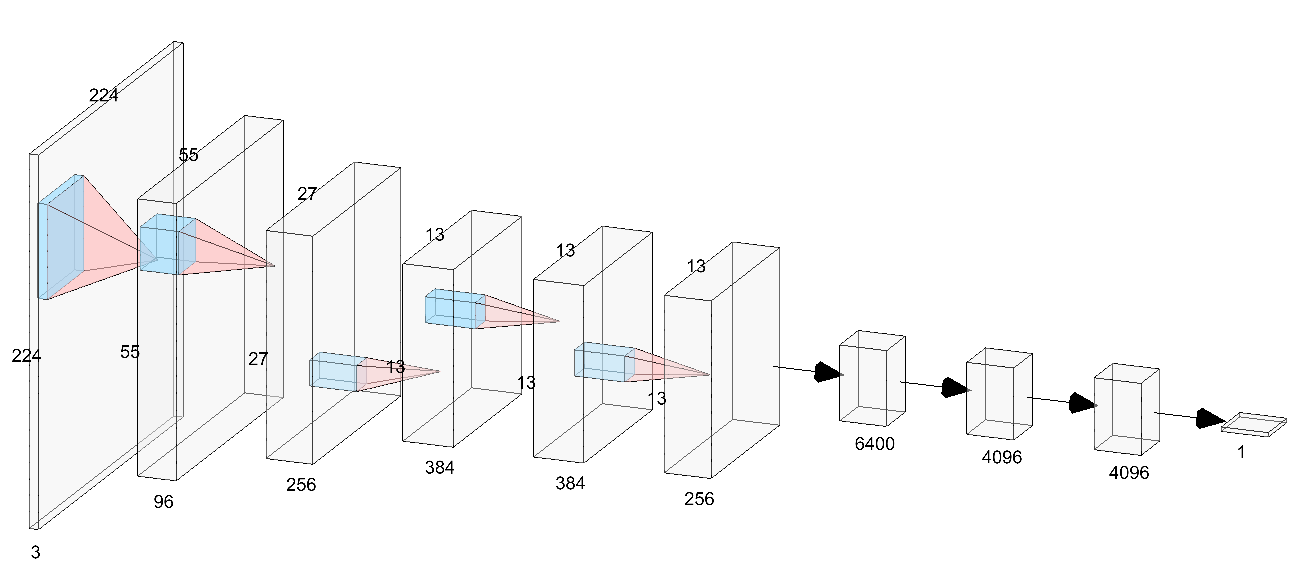
自定义dataset类依次读取数据。

规定数据集前80%用于训练，10%用于验证，10%用于测试，三个集合没有交集。

将图像resize成224\*224，50%的概率水平翻转用于数据增强，并且归一化，作为网络的输入。

由于数据集中每个类别的图像数量不一致，所以根据图像数量依概率随机选择类别，然后在训练/验证/测试集的范围内等概率随机选择图像进行读取。

1. 搭建网络结构



|  |
| --- |
| self.extract\_features = nn.Sequential(  nn.Conv2d(3, 96, 11, 4, 1),  nn.ReLU(),  nn.MaxPool2d(kernel\_size=3, stride=2),  nn.LocalResponseNorm(size=5),  nn.Conv2d(96, 256, 5, 1, 2),  nn.ReLU(),  nn.MaxPool2d(kernel\_size=3, stride=2),  nn.LocalResponseNorm(size=5),  nn.Conv2d(256, 384, 3, 1, 1),  nn.ReLU(),  nn.Conv2d(384, 384, 3, 1, 1),  nn.ReLU(),  nn.Conv2d(384, 256, 3, 1, 1),  nn.ReLU(),  nn.MaxPool2d(3, 2),  )  self.head = nn.Sequential(  nn.Linear(6400, 4096),  nn.ReLU(),  nn.Dropout(p=0.8),  nn.Linear(4096, 4096),  nn.ReLU(),  nn.Dropout(p=0.8),  nn.Linear(4096, output\_num)  ) |

将网络分为两个部分，提取特征extract\_features部分和分类头head部分。特征提取部分按照AlexNet的结构使用5个卷积层，ReLU作为激活函数，MaxPool2d用于下采样。分类头使用3层的MLP，ReLU激活，加入dropout层用于防止过拟合。

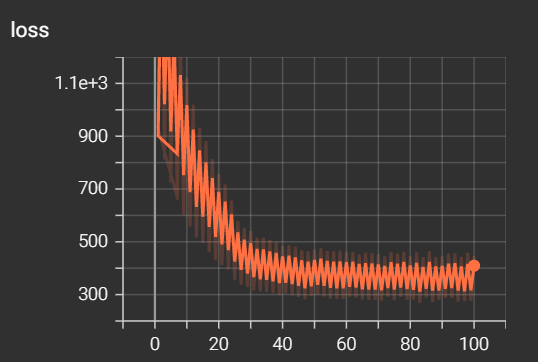
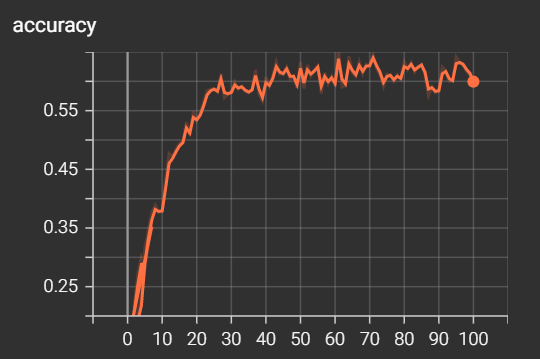
1. 定义优化器

使用Adam优化器，初始学习率设置为1e-4，weight decay设置为2e-5，学习率调整方式使用MultiStepLR，milestone设置为12，24，36，gamma为0.1，共训练epoch数为50次。

|  |
| --- |
| optimizer = optim.Adam(filter(lambda p: p.requires\_grad, model.parameters()), lr=opt.lr\_initial,  betas=(0.9, 0.999), eps=1e-8, weight\_decay=opt.weight\_decay)  scheduler = torch.optim.lr\_scheduler.MultiStepLR(optimizer, milestones=[12, 24, 36], gamma=0.1) |

1. 定义损失函数并训练

使用交叉熵损失函数进行训练。每个epoch进行两次测试并且记录。使用tensorboard绘制训练loss和测试准确率图如下：

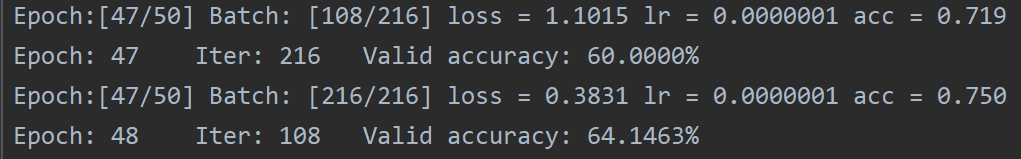


最终在验证集上的准确率约为63%。

**实验结果与分析：**

在验证集上的准确率约为63%。使用在验证集上效果最好的参数在测试集进行测试。由于读取数据部分有随机性，所以每次测试的结果有一定差别，平均约在63%上下，基本与在验证集上的结果持平。

根据训练过程中的log，发现模型在最后有一定程度上过拟合的现象，在训练集上的准确率能够达到70%，所以最后进行了提前结束，使用第25个epoch的参数进行测试并保存。



加载训练参数代码：

|  |
| --- |
| net = AlexNet(output\_num=101).cuda()  net.load\_state\_dict(torch.load('./checkpoint/alexnet\_caltech-101\_e25.pth')['state\_dict']) |