

**计算机视觉实践实验报告**

|  |  |  |  |
| --- | --- | --- | --- |
| **姓 名:** | 燕语晨 | **学 号:** | 123106222779 |
| **学 院:** | 计算机科学与工程学院 | | |

**2024年 4 月 26 日**

# 实验二 图像拼接

## 【1】实验目的

实现LeNet-5在MNIST数据集上的训练和测试，并进行分析

## 【2】算法原理

LeNet-5是由Yann LeCun等人在1998年提出的卷积神经网络（CNN）结构，是早期用于手写数字识别的经典模型之一，对于推动深度学习的发展具有重要意义。

卷积层（Convolutional Layers）：

LeNet-5包含两个卷积层。在每个卷积层中，都会有多个卷积核对输入图像进行卷积操作，提取不同的特征。卷积操作通过滑动卷积核在输入图像上进行运算，生成特征图（Feature Map）。

池化层（Pooling Layers）：

在每个卷积层之后，LeNet-5使用池化层进行下采样操作，常用的是最大池化（Max Pooling）。池化层可以减少特征图的大小，并且保留主要特征，从而降低后续层级的计算复杂度。

全连接层（Fully Connected Layers）：

LeNet-5包含三个全连接层，用于将卷积层和池化层提取的特征映射到输出类别上。全连接层通过权重矩阵将上一层的特征映射到下一层，并通过激活函数（通常是Sigmoid或ReLU）引入非线性。

Softmax层：

在LeNet-5的最后一层是一个Softmax层，用于将全连接层的输出转换为概率分布。Softmax函数将每个输出单元的原始分数转换为对应类别的概率值，使得模型可以进行分类预测。

反向传播算法（Backpropagation）：

LeNet-5使用反向传播算法进行训练，在训练过程中通过计算损失函数的梯度来更新网络参数。反向传播算法通过将梯度从输出层向输入层传播，调整网络中的权重和偏置，使得模型的预测结果与实际标签更加接近。

## 【3】实验步骤

数据准备： 使用PyTorch的torchvision库加载MNIST数据集，并进行数据变换（转换为Tensor并进行归一化）。

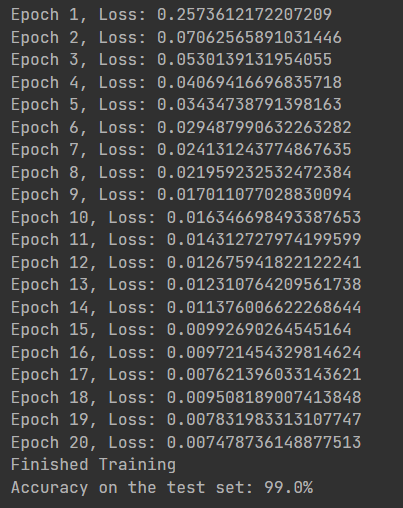
LeNet-5模型搭建： 构建了LeNet-5模型，包括两个卷积层、两个最大池化层和三个全连接层，使用ReLU作为激活函数。

损失函数和优化器： 使用交叉熵损失函数和Adam优化器。

训练过程： 进行了20个Epoch的训练，每个Epoch遍历训练集并更新模型参数。

测试过程： 使用训练好的模型对测试集进行预测，并计算准确率。

## 【4】结果分析



训练过程分析：

损失值下降： 随着训练Epoch的增加，损失值逐渐下降，这表明模型在训练集上学习到了有效的特征表示和分类规律。

收敛速度： 初始几个Epoch中损失值下降较快，后续逐渐趋于平稳，这说明模型在初期快速收敛并且没有出现明显的过拟合现象。

训练时间： LeNet-5模型相对较小且简单，训练过程较快，可以在较短的时间内完成训练。

测试结果分析：

准确率： 在测试集上达到了99.0%的准确率，这表明模型在对未见过的数据进行分类时具有很高的正确性。

泛化能力： 高准确率表明模型具有较好的泛化能力，能够有效地推广到新的样本上，并且在处理手写数字识别任务上表现优异。

误差分析： 可以进一步分析模型在测试集上错误分类的样本，探究错误的原因，比如模糊的数字、不规范的书写等，以便进一步改进模型或优化数据预处理方法。

## 【5】实验心得

本实验成功使用LeNet-5模型对MNIST数据集进行了训练和测试，训练过程中损失逐渐降低，测试结果显示模型在手写数字识别上取得了很高的准确率。LeNet-5作为经典的卷积神经网络模型，在简单数据集上展现了较好的性能，对于入门深度学习和图像分类任务具有重要意义。