***2023***



**计算机视觉实验 ·实验报告·**

j0242087[1]

|  |  |
| --- | --- |
| 专 业： | 计算机科学与技术 |
| 班 级： | CS2106 |
| 学 号： | U202115514 |
| 姓 名： | 杨明欣 |
| 电 话： | 13390396012 |
| 邮 件： | ymx@hust.edu.cn |
| 指导老师： | 刘康 |
| 完成日期： | 2023-12-22 |



目录

[1 实验内容 2](#_Toc154419961)

[1.1 实验要求 2](#_Toc154419962)

[1.2 实验内容 2](#_Toc154419963)

[1.2.1 卷积神经网络 2](#_Toc154419964)

[1.2.2 残差神经网络 2](#_Toc154419965)

[1.3 实验方法 3](#_Toc154419966)

[1.3.1 网络架构设计 3](#_Toc154419967)

[1.3.2 网络性能比较 4](#_Toc154419968)

[1.4 实验设计 4](#_Toc154419969)

[1.4.1 数据集 4](#_Toc154419970)

[1.4.2 实验环境 5](#_Toc154419971)

[1.5 实验分析与结果 6](#_Toc154419972)

[1.5.1 分类性能 6](#_Toc154419973)

[1.5.2 网络比较 8](#_Toc154419974)

[1.6 实验总结 9](#_Toc154419975)

[2 实验心得 10](#_Toc154419976)

# 实验内容

## 实验要求

设计一个卷积神经网络，并在其中使用ResNet模块，在MNIST数据集上实现10分类手写体数字识别。

具体要求：不能直接导入现有的ResNet网络。可以尝试不同的网络架构、激活函数、训练超参数等，至少尝试两种，观察并比较网络性能。

## 实验内容

### 卷积神经网络

卷积神经网络（Convolutional Neural Networks，简称CNN）是一种具有局部连接、权值共享等特点的深层前馈神经网络（Feedforward Neural Networks），是深度学习（deep learning）的代表算法之一，擅长处理图像特别是图像识别等相关机器学习问题，比如图像分类、目标检测、图像分割等各种视觉任务中都有显著的提升效果，是目前应用最广泛的模型之一。

卷积神经网络（CNN）具有表征学习能力，能够按其层次结构对输入信息进行平移不变分类。它适用于监督学习和非监督学习，并通过卷积核参数共享和层间连接的稀疏性，以较小的计算量对像素和音频等格点化特征进行学习。CNN能够稳定地实现分类任务，无需额外的特征工程，因此在计算机视觉、自然语言处理等领域得到广泛应用。

### 残差神经网络

深度残差网络（Deep Residual Networks，ResNet）自从2015年首次提出以来，就在深度学习领域产生了深远影响。通过一种创新的“残差学习”机制，ResNet成功地训练了比以往模型更深的神经网络，从而显著提高了多个任务的性能。深度残差网络通过引入残差学习和特殊的网络结构，解决了传统深度神经网络中的梯度消失问题，并实现了高效、可扩展的深层模型。

传统的深度神经网络试图学习目标函数 ( H(x) )，但是在ResNet中，每个网络层实际上学习的是一个残差函数 ( F(x) = H(x) - x )。然后，这个残差结果与输入 ( x ) 相加，形成 ( H(x) = F(x) + x )。这一机制使得网络更容易学习身份映射，进而缓解了梯度消失问题。

## 实验方法

为了实现对MNIST数据集的分类任务，需要设计卷积神经网络的模型架构，通过前向和后向的传播，对卷积神经网络内部参数进行训练，得到最终的分类模型。本实验在此分成两个部分，第一部分设计两种不同的网络架构完成MNIST数据集分类任务，第二部分则对比上面两种架构的网络性能。

### 网络架构设计

本实验中主要通过使用卷积神经网络（包含残差块）进行MNIST数据集分类任务，最终设计实现了两种模型架构，第一种模型架构更多使用卷积层进行特征的提取，最终根据提取的特征进行分类。具体结构示意图如图 1‑1所示。

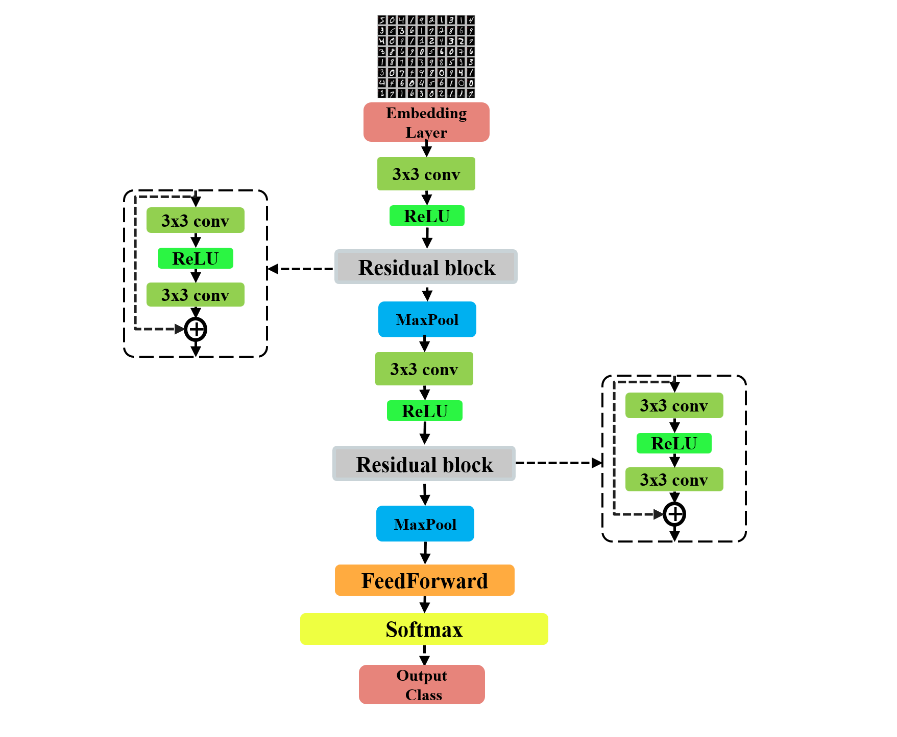


图 1‑1 模型架构（一）示意图

在此部分选择超参数epoch为3，batchsize为128，使用交叉熵损失函数，学习率设置为0.001，权重衰退值设置为0.0001，使用Adam优化器迭代模型。

而第二种模型架构则参考LeNet-5，同时使用卷积层和全连接层实现对于特征的提取，最终根据提取的特征进行分类。具体结构示意图如图 1‑2所示。

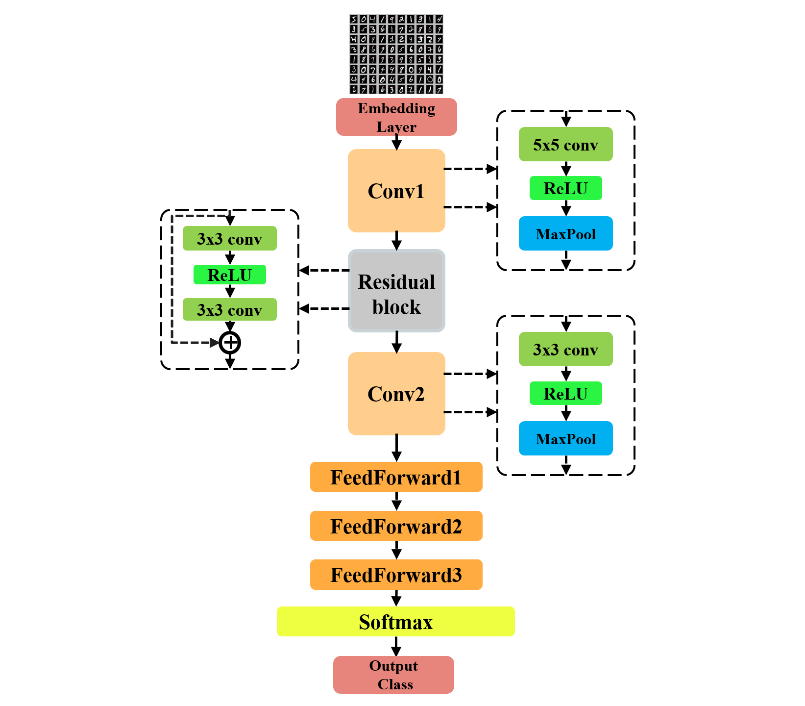


图 1‑2 模型架构（二）示意图

在此部分选择超参数epoch为4，batchsize为128，使用交叉熵损失函数，学习率设置为0.001，权重衰退值设置为0.0001，使用Adam优化器迭代模型。

### 网络性能比较

在前一部分的网络架构中，我们设计实现了两种模型架构，第一种模型架构更多使用卷积层进行特征的提取，第二种模型架构则参考LeNet-5，同时使用卷积层和全连接层实现对于特征的提取，最终根据提取的特征进行分类。在此部分，我们将对比两个模型的模型参数量，收敛速度，准确率等。

## 实验设计

### 数据集

MNIST是一个手写体数字的图片数据集，由美国国家标准与技术研究所（National Institute of Standards and Technology，NIST）发起整理。这个数据集包含了70000个样本，其中60000个是训练样本，10000个是测试样本。每个样本都是一个28x28像素的手写数字图像。

MNIST数据集的内容来自250个不同的人手写数字，其中50%是高中生，50%来自人口普查局的工作人员。这个数据集的收集目的是希望通过算法，实现对手写数字的识别。1998年，Yan LeCun等人发表了论文《Gradient-Based Learning Applied to Document Recognition》，首次提出了LeNet-5网络，利用上述数据集实现了手写字体的识别。

MNIST 数据集包含了四个部分：

* Training set images: train-images-idx3-ubyte.gz (9.9 MB, 解压后 47 MB, 包含 60,000 个样本)
* Training set labels: train-labels-idx1-ubyte.gz (29 KB, 解压后 60 KB, 包含 60,000 个标签)
* Test set images: t10k-images-idx3-ubyte.gz (1.6 MB, 解压后 7.8 MB, 包含 10,000 个样本)
* Test set labels: t10k-labels-idx1-ubyte.gz (5KB, 解压后 10 KB, 包含 10,000 个标签)

接下来首先将数据集加载，然后进行打乱（shuffle），以备模型训练和测试。

### 实验环境

系统版本：Windows 11 专业版

处理器：13th Gen Intel(R) Core(TM) i9-13900HX 2.20 GHz

机带RAM：16.0 GB (15.7 GB 可用)

系统类型：64 位操作系统, 基于 x64 的处理器

显卡版本：NVIDIA GeForce RTX 4060 Laptop GPU

显存：8188MiB

CUDA版本：12.0

Python版本：3.9.18

torch版本：2.1.1+cu118

## 实验分析与结果

### 分类性能

对于1.3.1中的第一个网络架构进行训练，使用交叉熵损失函数（Cross\_entropy），使用Adam优化器，epoch为3轮，batch­size为128，学习率设置为0.001，训练过程中训练集的损失和准确率如图 1‑3所示。

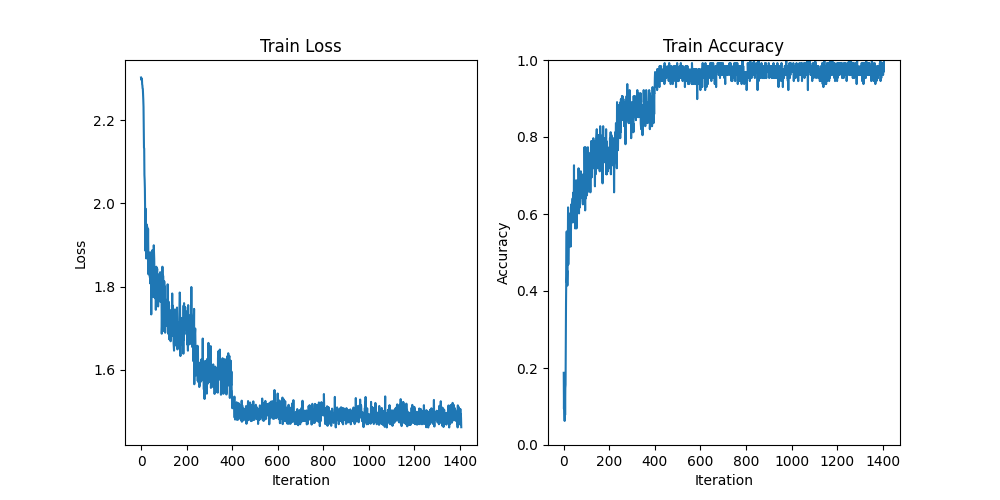


图 1‑3 模型（一）训练过程中训练集的损失和准确率

在训练集上进行测试可得整体的准确率为98.07%，其中数字0的准确率为98.33%，数字1的准确率为98.62%，数字2的准确率为97.35%，数字3的准确率为97.80%， 数字4的准确率为99.11%，数字5的准确率为98.03%，数字6的准确率为98.70%，数字7的准确率为99.28%，数字8的准确率为97.03%，数字9的准确率为96.35%。

训练过程中测试集的损失和准确率如图 1‑4所示。

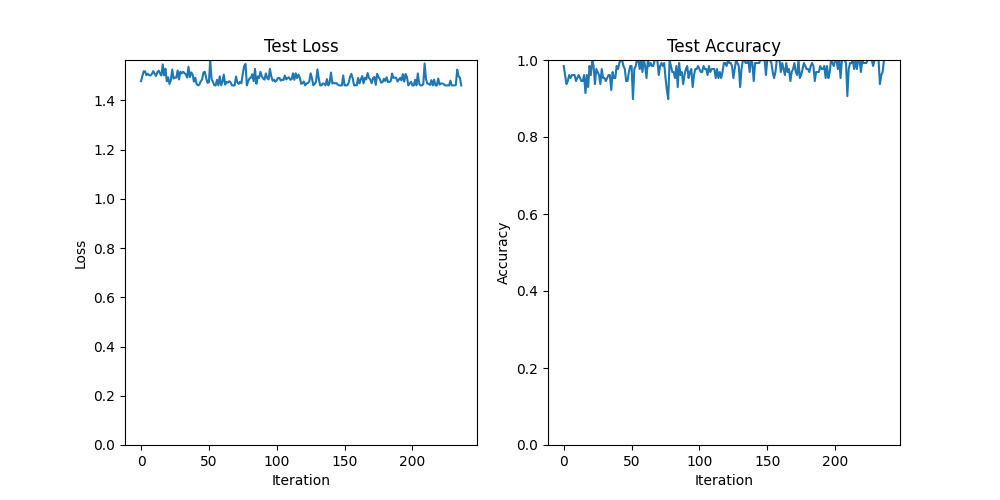


图 1‑4 模型（一）测试过程中测试集的损失和准确率

最终在测试集上进行测试可得整体的准确率为98.14%，其中数字0的准确率为98.57%，数字1的准确率为98.85%，数字2的准确率为97.48%，数字3的准确率为98.22%， 数字4的准确率为99.08%，数字5的准确率为98.43%，数字6的准确率为97.60%，数字7的准确率为99.22%，数字8的准确率为96.92%，数字9的准确率为96.93%。

对于1.3.1中的第二个网络架构进行训练，使用交叉熵损失函数（Cross\_entropy），使用Adam优化器，epoch为4轮，batch­size为128，学习率设置为0.001，训练过程中训练集的损失和准确率如图 1‑5所示。

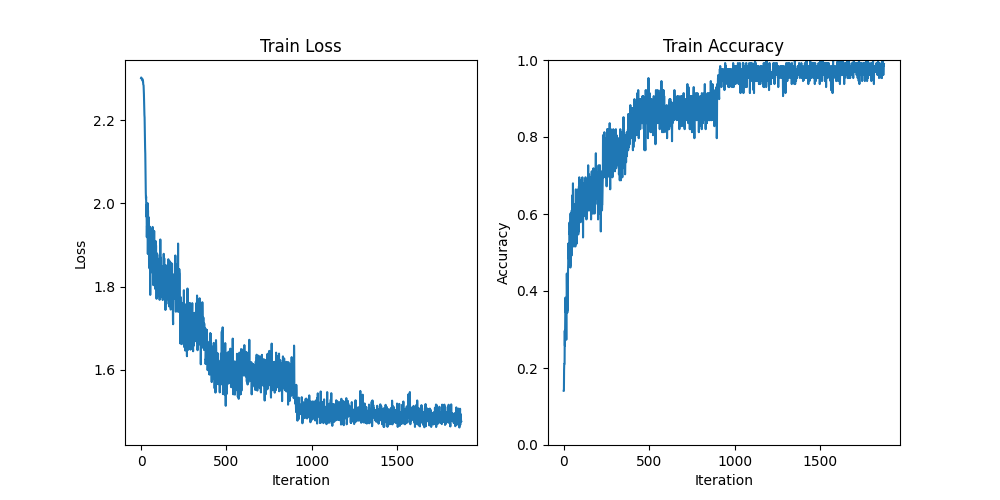


图 1‑5 模型（二）训练过程中训练集的损失和准确率

在训练集上进行测试可得整体的准确率为97.74%，其中数字0的准确率为98.95%，数字1的准确率为97.30%，数字2的准确率为98.57%，数字3的准确率为95.99%， 数字4的准确率为99.32%，数字5的准确率为98.19%，数字6的准确率为98.99%，数字7的准确率为98.29%，数字8的准确率为96.24%，数字9的准确率为95.66%。

训练过程中测试集的损失和准确率如图 1‑6所示。

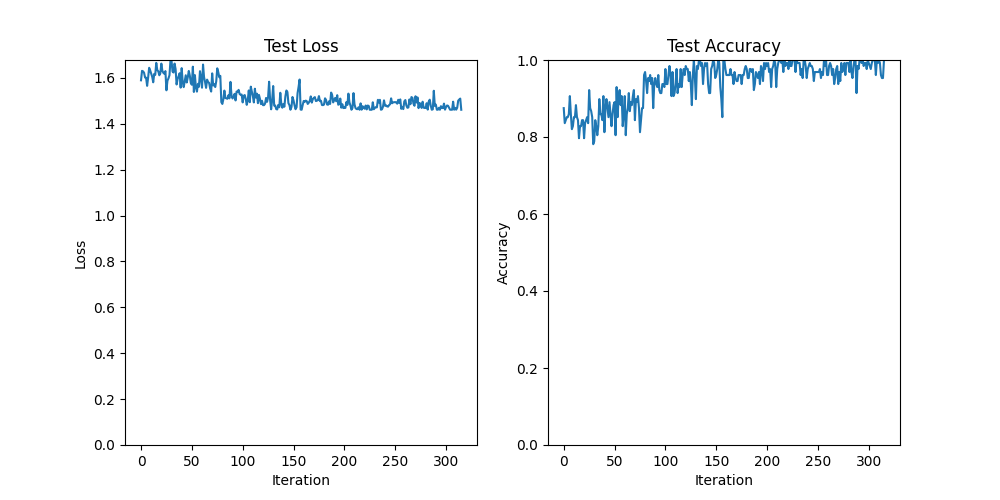


图 1‑6 模型（二）训练过程中测试集的损失和准确率

最终在测试集上进行测试可得整体的准确率为97.83%，其中数字0的准确率为98.98%，数字1的准确率为98.68%，数字2的准确率为98.64%，数字3的准确率为96.73%， 数字4的准确率为98.98%，数字5的准确率为98.32%，数字6的准确率为98.02%，数字7的准确率为97.57%，数字8的准确率为97.33%，数字9的准确率为95.04%。

最终可以得出结论我们的模型最终收敛到了一个比较好的效果，能够很好地完成手写数字识别分类任务。

### 网络比较

首先比较两个模型框架的差异和相关性能等，具体结果如表 1‑1所示。

表 1‑1 模型差异比较

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 模型 | 卷积层数量 | 全连接层数量 | 激活函数 | 参数量 | 收敛速度 | 准确度/% |
| （1） | 6 | 1 | ReLU | 2 | 1[[1]](#footnote-1) | 98.14 |
| （2） | 4 | 3 | ReLU | 1 | 2 | 97.83 |

通过上面的分析总结可得，模型的参数量越多，收敛速度越慢，同时模型的准确度也不一定效果更好，因此需要设计更加优秀适配的网络结构，才能取得更好的分类效果。

进一步比较不同的optimizer发现，SGD优化器会使得模型难以收敛，因此适当的优化器选择才会使模型取得更好的效果。

## 实验总结

本次实验主要完成了如下几点工作：

1. 设计实现了网络架构进行训练，对MINIS数据集实现分类任务。
2. 比较两种不同的模型架构对于分类性能的影响。
3. 得出结论，模型的参数量越多，收敛速度越慢，同时模型的准确度也不一定效果更好，因此需要设计更加优秀适配的网络结构，才能取得更好的分类效果。

# 实验心得

在完成MINSI数据集分类任务的实验过程中，我获得了许多宝贵的经验。

1. 模型设计的重要性：在设计两个卷积神经网络模型时，我深刻体会到了模型设计对性能的关键影响。尽管两个模型都采用了残差层，但在网络深度、宽度、卷积核大小等方面的细微差别导致了模型在参数量、收敛速度和准确率方面的显著差异。这让我认识到了在设计神经网络时，需要谨慎选择各种超参数和结构，以便找到最优的模型配置。
2. 残差层的有效性：通过实验，我观察到残差层在提高模型性能方面的显著效果。残差层有助于解决深度神经网络中的梯度消失和表示瓶颈问题，使得网络能够更有效地学习特征表示。在两个模型中，残差层的引入都显著提高了模型的收敛速度和准确率，证明了其在改善模型性能方面的有效性。

总的来说，这次实验让我深入了解了卷积神经网络的设计、训练和优化过程，并让我体会到了模型设计、参数选择和实验结果分析等方面的重要性，收获很大。

|  |
| --- |
| 一、原创性声明 |
| 本人郑重声明本报告内容，是由作者本人独立完成的。有关观点、方法、数据和文献等的引用已在文中指出。除文中已注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品成果，不存在剽窃、抄袭行为。  特此声明！  **作者签字: 杨明欣** |
| 二、对课程实验的学术评语（教师填写） |
|  |
| 三、对课程实验的评分（教师填写） |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | 评分项目  （分值） | 课程目标1  工具应用  （10分） | 课程目标2  设计实现  （70分） | 课程目标3  验收与报告  （20分） | 最终评定  （100分） | | 得分 |  |  |  |  | |
| **指导教师签字:** |

1. 参数量和收敛速度为相对排名 [↑](#footnote-ref-1)