图标

描述已自动生成

**计算机视觉**

**上机实验三报告**

|  |  |  |  |
| --- | --- | --- | --- |
| **专业** | **：** | **计算机电脑** |  |
| **班级** | **：** | **CS2002** |  |
| **学号** | **：** | **I201920029** |  |
| **姓名** | **：** | **冯就康** |  |
| **电话** | **：** | **15623031879** |  |
| **邮箱** | **：** | **sokhorng526@gmail.com** |  |

Contents

[I. 简介 3](#_Toc154479538)

[1. 卷积神经网络的简介 3](#_Toc154479539)

[2. ResNet模块的简介 4](#_Toc154479540)

[II. 网络的架构 5](#_Toc154479541)

[1. 模型的结构 5](#_Toc154479542)

[2. 残差块的结构 5](#_Toc154479543)

[3. 模型的权重和偏差 5](#_Toc154479544)

[III. 模型训练过程 6](#_Toc154479545)

[1. 数据准备： 6](#_Toc154479546)

[2. 损失函数和优化器： 6](#_Toc154479547)

[3. 训练循环： 6](#_Toc154479548)

[IV. 实验结果以及分析 7](#_Toc154479549)

[1. 实验结果 7](#_Toc154479550)

[2. 实验分析 7](#_Toc154479551)

[V. 实验总结 8](#_Toc154479552)

[1. 实验总结： 8](#_Toc154479553)

# 简介

## 实验的简介

设计一个卷积神经网络，输入为两张MNIST手写体数字图片，如果两张图片为同一个数字（注意，非同一张图片），输出为1，否则为0。

从**MNIST**数据集的训练集中选取10%作为本实验的训练图片，从**MNIST**数据集的测试集中选取10%作为本实验的测试图片。请将该部分图片经过适当处理形成一定数量的用于本次实验的训练集和测试集。

## 卷积神经网络的简介

**卷积神经网络**（**Convolutional** **Neural** **Networks**，简称**CNN**）是一种深度学习算法，主要用于图像分类和计算机视觉任务。它由卷积层、池化层和全连接层组成。

* **卷积层**---是卷积神经网络的核心构建块，负责执行大部分计算。
* **池化层**---执行降维操作，旨在减少输入中参数的数量。
* **全连接层**---根据通过先前层及其不同的过滤器提取的特征，执行分类任务。

A diagram of a diagram of a process

Description automatically generated with medium confidence**卷积神经网络**利用线性代数原理（特别是矩阵乘法）来识别图像内的图案。**卷积神经网络**的结构是一个分层结构，随着层级的递进，**卷积神经网络**的复杂性也逐步增加，能够识别图像的更多部分。**卷积神经网络**的出现，提供了更加可扩展的方法来执行图像分类和对象识别任务。

# 网络的架构

## 模型的结构

整个网络架构是由两个卷积层、两个全连接层和一个输出层组成。

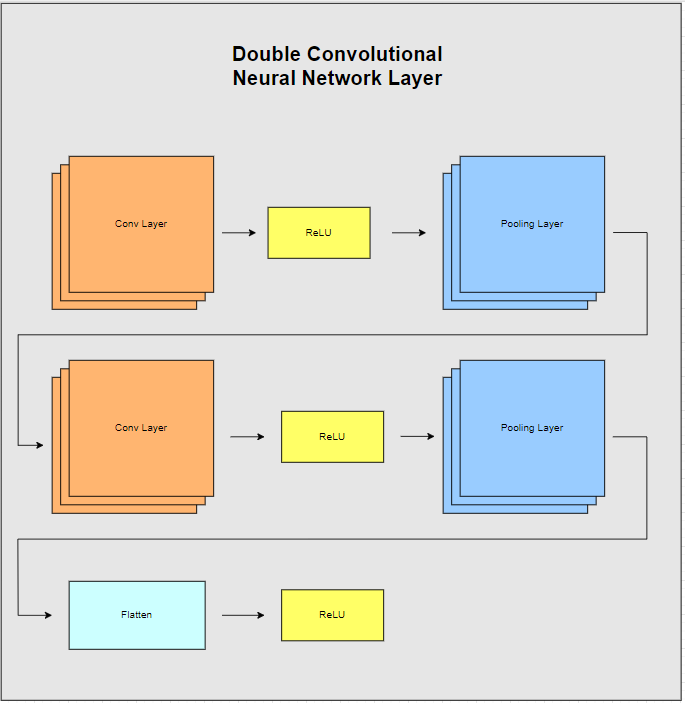
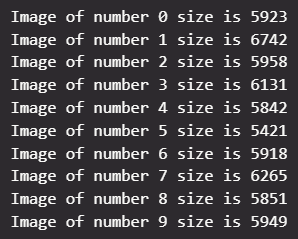
* 网络的输入层由 **nn.Conv2d** 函数定义，该函数接受两张 **MNIST** 手写数字图片作为输入。输入图片是大小为28x28像素的灰度图像。
* 第一个卷积层 (**self.conv1**) 有 32 个滤波器，内核大小为 5x5。
* 第二个卷积层 (**self.conv2**) 有 64 个滤波器，内核大小为 5x5。
* 两个卷积层都使用 ReLU 激活函数，后面是内核大小为 2x2 的最大池层 (**F.max\_pool2d**)。
* 第二个卷积层的输出被展平并通过两个全连接层（**self.fc1** 和 **self.fc2**）。
* 第一个全连接层有 256 个神经元，并使用 **ReLU** 激活函数。
* 第二个全连接层有 2 个神经元，没有激活函数。
* 网络的输出层由前向函数定义，该函数接收两个全连接层的输。
* 最后通过全连接层 (**self.fc2**) 以产生单个输出值。如果两个输入图片的编号相同，则输出值为1，否则为0。

Figure Overview of CNN Model

# 模型训练过程

## 数据准备：

我们需要使用的数据集还是 **MNIST** 数据集。 **MNIST** 数据集是一个大型手写数字数据库，通常用于训练各种图像处理系统。它还广泛用于机器学习领域的训练和测试。该数据集包含 60,000 张训练图像和 10,000 张测试图像。图像为灰度图像，分辨率为 28x28 像素。我们可以从**torch**.**vision**下载**MNIST**数据集，并用**torchvison**.**transform**.**ToTensor**()把数据变成**Tensor**。

由于我们只是使用MNIST训练集集的10%作为本次实验的训练集，使用MNIST测试集的10%作为本次实验的训练集。我们要保证实验的训练集包括每个数字（0-9），所以我引入了**torch.manual\_seed(0**),而且把所有的数字在训练集的数量都打印出来。

## 损失函数和优化器：

* **nn.CrossEntropyLoss** 函数用于定义损失函数，用于计算预测标签和实际标签之间的差异。
* **optim.Adam**函数用于定义优化器，用于在训练期间更新模型的权重。

## 训练循环：

训练过程是使用嵌套的 **for** 循环来实现的。外循环迭代10个**epoch**，而内循环则分批迭代训练数据。在内循环的每次迭代中，从批次中提取输入和标签，并将输入沿维度 0 拆分为两个张量。标签也沿维度 0 拆分为两个张量，并对两组标签进行比较生成一个二进制标签，指示两个输入图像是否表示相同的数字。

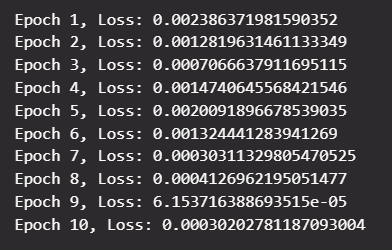
优化器以 0.001 的学习率初始化，并使用反向传播算法更新网络的权重。该任务使用的损失函数是交叉熵损失，它是分类任务的标准损失函数。

在训练期间，调整神经元之间连接的权重以最小化预测误差。这是使用反向传播算法完成的，该算法计算损失函数相对于网络权重的梯度。然后权重在负梯度的方向上更新，这减少了损失函数。对每批数据重复此过程，并且每个时期的权重更新多次。

数据集多次通过网络的迭代过程称为一个纪元。在每个时期，整个数据集分批通过网络，并使用反向传播算法更新网络的权重。纪元数是一个超参数，需要进行调整才能实现任务的最佳性能。

# 实验结果以及分析

## 实验结果

经过**10个Epochs**训练的训练集的损失值：

A black background with white text

Description automatically generated测试集的准确率：

## 实验分析

从实验结果的训练损失和测试准确性表明，卷积神经网络 (**CNN**) 模型对于比较两张 **MNIST** 手写数字图片的给定任务表现良好。训练损失随着每个时期的推移而减少，这表明模型正在学习做出更好的预测。 98%的测试准确率表明该模型能够以高置信度对输入图像进行准确分类。

混淆矩阵显示，该模型在总共 500 张测试图像中做出了 464 个**True** **Positive**预测和 26 个**Ture** **Negative**预测。该模型做出了 3 个**False** **Positive**预测和 7 **False** **Negative**预测。大量的**True** **Positive**和**True** **Negative**预测表明该模型对于给定任务表现良好。

总体而言，所提供的 **CNN** 模型对于比较两张 **MNIST** 手写数字图片的给定任务表现良好。

# 实验总结

## 实验总结：

本次实验涉及到设计一个卷积神经网络 (**CNN**) 来比较两张 **MNIST** 手写数字图片。 **CNN** 架构由两个卷积层、两个全连接层和一个输出层组成。该模型使用学习率为 0.001 的 **Adam** 优化器和交叉熵损失函数进行训练。训练损失随着每个**epoch**的增加而减少，测试准确率为98%。

实验结果表明，**CNN** 模型对于比较两张 **MNIST** 手写数字图片的给定任务表现良好。高测试精度和**False** **Positive**和**False** **Negative**预测数量表明该模型能够以高置信度对输入图像进行准确分类。

本次实验对于计算机视觉和图像识别领域具有重要意义。 表明了**CNN** 模型可用于对医学成像、安全系统和自动驾驶车辆等各种应用中的图像进行比较和分类。

未来的工作可能涉及探索不同的 **CNN** 架构和超参数以提高模型的性能。此外，可以在更大的数据集上测试该模型，以评估其可扩展性和鲁棒性。