基于改进 CNN 的实时手写数字识别

任峻东 周 丽

四川大学锦城学院 计算机与软件学院 四川 成都 611731

【摘 要】手写数字识别，即识别程序对读入的手写体数字进行识别，并反馈出识别结果的一种技术。伴随随着深度学习的火爆，识别技术也得到了良好的发展，其中包括数字识别，汉字识别，英文识别等等，其中数字识别拥有广泛的使用场景，例如可用于教育行业，实现批量打分，或者用于财务金融领域，实现报表的自动导入等等。本文详细介绍了 LeNet-5 网络的基本概念和关键结构，并引入了空洞卷积的概念来改进网络。采用了成熟的手写数字集 MNIST，进行两个的模型训练，最后配合可视化窗口来实现数字书写和识别的同时化。

【关键词】深度学习；MNIST；LeNet-5 卷积神经网络；空洞卷积

1 引言

LeNet-5 神经网络的出现可以说是卷积神经网络的开端。直到现在仍然使用的是输入层、卷积层、激励层、池化层和全连接层的基本结构。因为结构简单，易于上手，所以被大众广泛运用。

数字识别分为规范的印刷体数字识别和不规范手写体数字识别。前者利用数字不变的几何形状，可以十分容易的提取出特征值实现识别，但手写数字的形状就因人而异了，就拿“5”来说，有人喜欢一笔写完，有人喜欢写两笔，这样用提取出的特征来识别有可能是 5，也有可能是 3。所以需要一个拥有较高识别率的方法。可以使用卷积神经网络来提高手写数字分类的正确率，故本文通过使用成熟的 LeNet-5 神经网络完成手写的数字识别。

2 数据集简介

本文使用了目前最常用的手写数字数据集 MNIST，它是由美国国家标准与技术研究所（NIST）收集的，拥有 60000 个训练集和 10000 个测试集，标签则是 0-9 这十个类别，数据来自 250 个不同人的手写数字，考虑到学生的字是变化率最高的，故 50% 的数据来自高中学生。数字样本图像大小统一为 28*28 大小，并且处于正中心的位置 [1]。这样更有利于做数字识别的训练和测试。如下图。

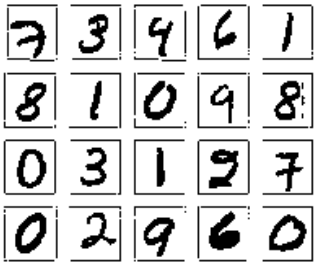

图 1 MINIST 数据集部分样本

3 卷积神经网络简介

CNN(Convolutional Neural Network) 网络，也叫卷积神经网络。通常是由卷积层、激活层和池化层三部分构成 [2]。CNN 网络的输入通常是一张图片，通过卷积层从图片中提取图像的特征矩阵，然后模型将这个特征矩阵再传入到全连接层中作为输入，利用全连接层的运算得到图片到标签（label）的映射值。

3.1 卷积层原理

CNN 中的一个重要的概念就是卷积 (Convolution)。而卷积中最重要的概念就是卷积核。在图像领域里卷积核叫做离散二维滤波器，其实就相当于一个二阶矩阵，通常大小为 3*3、5*5 等等。图像与卷积核做卷积操作就是将卷积核从图片的最左上角开始按步长 (Stride) 滑动，同时做内积运算，将得到的值放入新的矩阵中，也就是特征矩阵 [3]，如图 2 所示。这样的操作在图像处理领域内应用的十分广泛，对同一张图片使用不同的卷积核也会得到不同的特征矩阵，这都取决于实验者的需要。

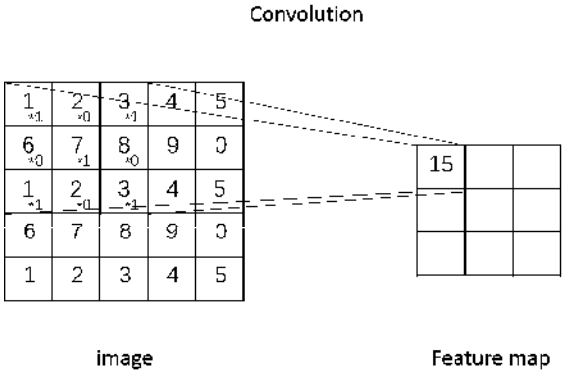

图 2 卷积运算示例

3.2LeNet-5 网络

LeNet-5 是 LeCun 于上世纪 90 年代提出的用于解决美国银行支票上的数字识别问题的网络，也是最早的卷积神经网络之一 [4]。LeNet-5 网络可以分为 5 层，分别为卷积池化层 1，卷积池化层 2，全连接层 1，全连接层 2，全连接层 3[5]。如图 3 所示。

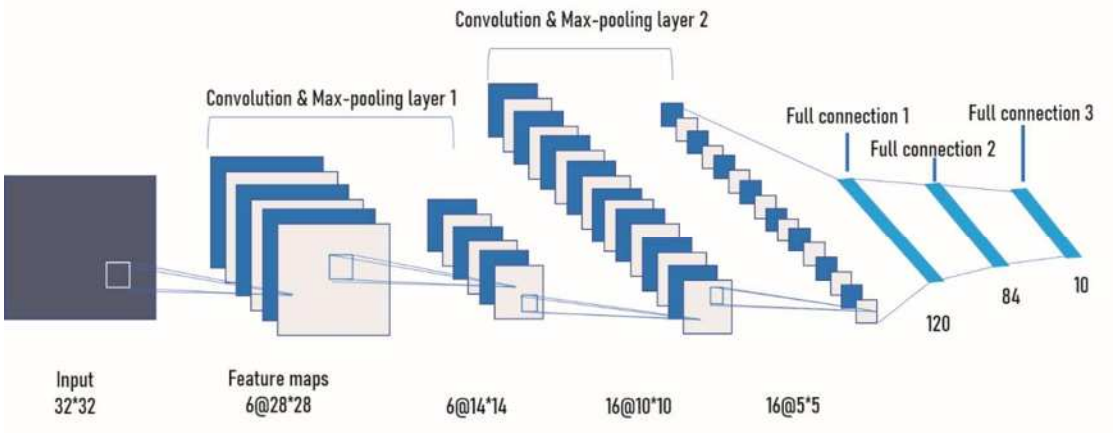

图 3 LeNet-5 网络结构

图 3 中一共有 5 层，其中不包括输入层，以下详细对各层做介绍。

第 1 层是卷积池化层 1，本层包括一次卷积和一次池化。卷积层使用 6 个 5*5 的卷积核，以步长（stride）为 1，在边缘填充了 2 层的 32*32 图片上做滑动内积，得到了 6 个 28*28 的特征图。池化层使用了 2*2 的核，以步长为 2 做特征降维（最大池化）操作，结果为 6 个 14*14 的特征图。

第 2 层是卷积池化层 2，本层将做第二次卷积和池化操作。本层使用了 16 个 5*5 的卷积核，以步长为 1，在 14*14 的特征图上卷积，形成了 16 个 10*10 的特征图。池化层依旧使用以步长是 2 的 2*2 的核来池化，得到 16 个 5*5 的特征图。

第 3 层是全连接层 1，把卷积池化层 2 的结果扁平化后作为输入，该层有 120 个神经元，和全连接层 2 的 84 个神经元全连接。

第 4 层是全连接层 2，本层有 84 个神经元每个神经元都和上一层的 120 个神经元相连接，所以加上一个偏置项，共存在 84*(120+1)=10164 个权值参数。

第 5 层是全连接层 3，将 84 个神经元映射到 10 个神经元上得出 0-9 这 10 个数的相应概率。最大值就是识别的结果。

4 实时手写数字的实现

4.1 使用 LetNet-5 训练

损失函数是用来衡量预测值和真实值之间的差异大小。LetNet-5 网络使用的则是交叉熵损失函数（CrossEntropyLoss）。本文使用 LetNet-5 对 MINIST 训练集的 60000 个数据按 batch_size 为 64 进行训练和按同样的 batch_size 对 10000 个验证数据进行验证。得到的损失值曲线和准确率曲线如下图。可以看到损失值和准确值都在第 10 个 epoch 左右开始收敛，最后的准确值稳定在了 98.70% 左右。由此可见，LeNet-5 网络对该问题的解决效率可以说是很不错了。但精益求精，本文将对 LeNet-5 网络进行部分修改以得到更高的准确率。

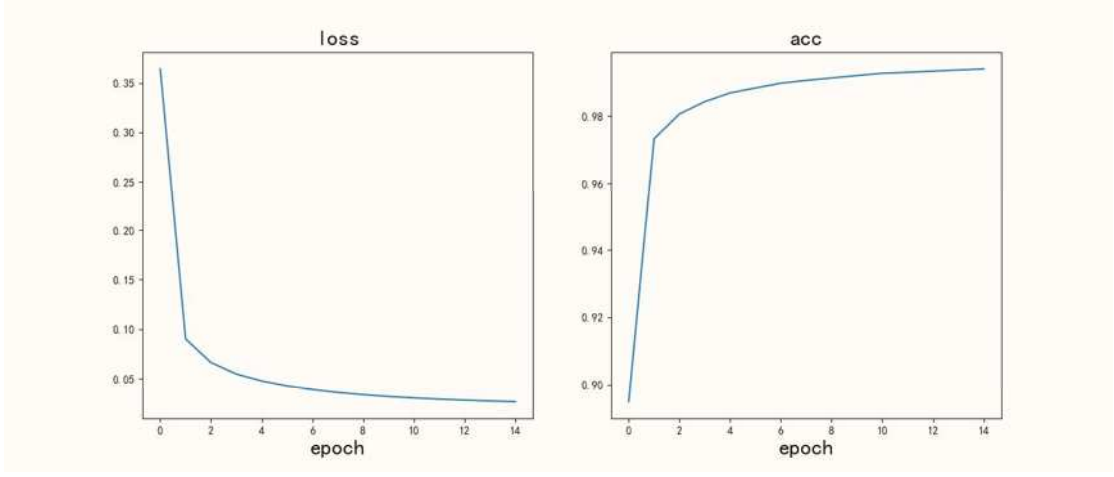

图 4 损失值和准确值曲线

4.2 优化 LeNet-5 网络

从上面的 MINIST 数据集可以看出绝大部分的数字都是在图片的中心位置，这样得到的特征矩阵也会大量包含背景值，如果选用较大的卷积核会产生巨大的计算负荷。故本文采用了图像修复领域内的一个重要概念——空洞卷积。

空洞卷积（Dilated Convolution），也叫做扩张卷积。其主要的作用就是在不改变特征图大小和计算量的情况下还能增大感受野 [6]。简单地说，就是在标准卷积核的元素之间加上若干个空洞 d(dilated) 来撑大卷积核，比如本文将 3*3 的卷积核加上 4 个空洞后变换成了 9*9 的卷积核。计算公式如下图。

$$n = k + (k-1) * (d-1)$$

图 5 空洞卷积计算公式

本文将空洞卷积应用在原本第一个卷积层之前，应用 9*9 的空洞卷积核在 28*28 的输入图像上做填充为 4 的卷积，得到 28*28 的特征图。

得到的前 15 个 epoch 的损失值和准确率如下图右侧所示。可以看见测试集上的准确率相对于 LeNet-5 网络（下图左侧所示）的准确率有所提高。

```
train_epoch:10  loss:0.0305  acc:99.27%
========testloss:0.0430  acc:98.64%========

train_epoch:11  loss:0.0293  acc:99.30%
========testloss:0.0423  acc:98.71%========

train_epoch:12  loss:0.0282  acc:99.33%
========testloss:0.0418  acc:98.72%========

train_epoch:13  loss:0.0274  acc:99.37%
========testloss:0.0413  acc:98.70%========

train_epoch:14  loss:0.0267  acc:99.40%
========testloss:0.0411  acc:98.70%========
```

```
train_epoch:10  loss:0.0246  acc:99.42%
========testloss:0.0383  acc:98.74%========

train_epoch:11  loss:0.0233  acc:99.47%
========testloss:0.0374  acc:98.80%========

train_epoch:12  loss:0.0222  acc:99.51%
========testloss:0.0373  acc:98.81%========

train_epoch:13  loss:0.0212  acc:99.54%
========testloss:0.0365  acc:98.86%========

train_epoch:14  loss:0.0202  acc:99.56%
========testloss:0.0361  acc:98.94%========

train_epoch:15  loss:0.0193  acc:99.59%
========testloss:0.0359  acc:98.95%========
```

图 6 优化后的结果对比

4.3 手写板实现

本实验将显示窗口设置为全黑，不仅仅是便于显示，更是为了验证模型的性能，因为训练集和验证集都是白色的底黑色的字，故特意设置成全黑。书写的逻辑可类比真实写字的流程，得到三个主要鼠标事件：左键按下、左键松开、左键按下的同时移动鼠标。到此就完成了绘制功能[7]。

截图识别功能需要将窗口转换类型后，做图像增强和维度变换并传入模型中进行训练。结果会得到 10 个值，但这并不是理想的结果。所以还需要利用 Softmax 函数将值映射成概率[8]，并求出最大值。

5 实验结果

结果如下图所示。能够对标准书写的数字进行预测并显示结果。到此本实验的主要功能已经实现完成。

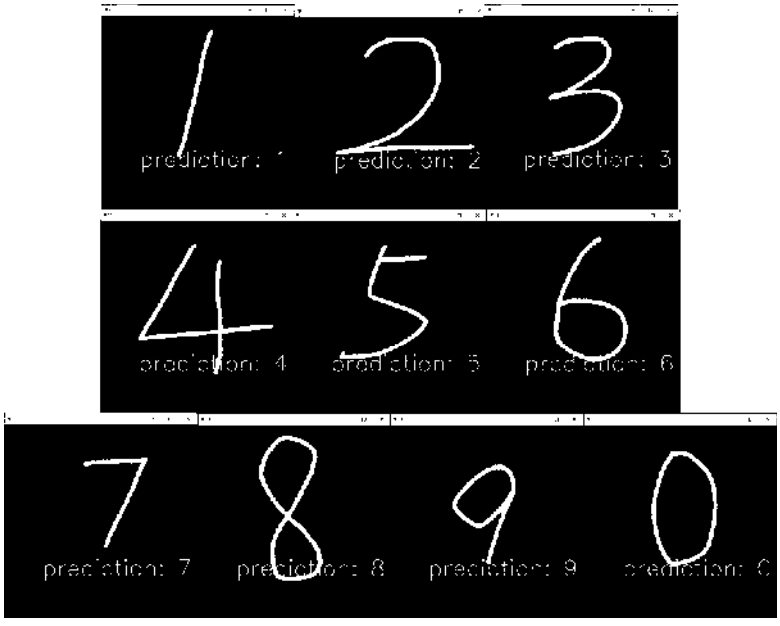

图 7 测试结果图

下图则是对同一数字分别使用改进后和改进前的模型所得到的结果。很显然，对一个书写轨迹十分相近的数字来说，改进后的网络相比于 LeNet-5 网络来说能更准确的识别出结果来。

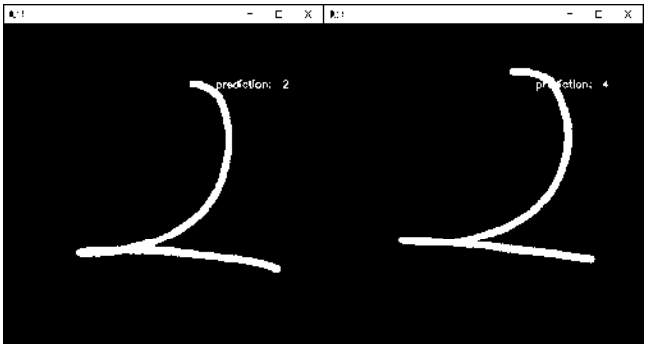

图 8 改进前后对比（左边是改进后、右边是改进前）

总结

LeNet-5 网络因简单且重要的特性深受新手的推崇。随着硬件和一些算法的提出，在这之后也陆续出现了一些更好的网络结构，比如加深了网络的 VGG16，增强了卷积模块功能的 NIN 网络，还有从分类任务迁移到检测任务的 R-CNN 网络等等。但这些模型沿用的仍然是 LeNet-5 网络的基本结构，只是更深更复杂了，所以对 LeNet-5 网络的学习会给自己对神经网络的学习打下坚实的基础。

本文通过实验完成了手写数字识别并可视化了实验结果，而不是依靠将手写的数字拍成照片再传入网络中这样繁琐的过程。简化了传统预测流程，对以后的实验思路提供了有效启发。

通过实验发现，当数字书写的不规范时，得出来的结果也和期望值不一样，最主要的原因就是数据集不全。因为每个人有每个人的书写习惯，而书写者的特征可能不包括在 MNIST 数据集当中，所以想要提高准确率可以把自己书写的数字添加到训练和测试集当中，但值得注意的是加入的数据必须是大量的而且将格式统一成 28*28 大小，否则最后结果并不会很理想甚至出现错误。

本文不足的地方就是没有将 LeNet-5 网络和 VGG16 网络作对比实验而是采用了自己加深的网络结构做改进模型，使得实验结果一般。

【参考文献】

[1] 翟高粤 . 基于卷积神经网络的手写数字识别应用 [J]. 甘肃科技纵横 , 2021, 50(01):1-3.

[2] 魏峰 , 山磊 . 基于 CNN 优化的手写数字识别技术研究 [J]. 连云港职业技术学院学报 , 2020, 33(02):5-7.

[3] 代贺 , 陈洪密 , 李志申 . 基于卷积神经网络的数字识别 [J]. 贵州师范大学学报 (自然科学版), 2017, 35(05):96-101.

[4] 李大华 , 王宇 , 高强 , 于晓 . 基于改进 LeNet-5 网络的文件编号识别 [J]. 现代计算机 , 2021(02):62-66.

[5] 何凯 , 黄婉蓉 , 刘坤 , 高圣楠 . 基于改进 LeNet-5 模型的手写体中文识别 [J]. 天津大学学报 (自然科学与工程技术版), 2020, 53(08):847-853.

[6] 何帅 . 卷积神经网络在手写数字识别中的应用 [J]. 电脑知识与技术 , 2020, 16(21):13-15.

[7] 刘宝宝 , 杨雪 , 吴治虎 , 侯飞 , 穆姣 . 改进的全卷积神经网络在手写数字识别上的应用 [J]. 电脑知识与技术 , 2020, 16(35):1-3.