分类号	TP391	密级	公开
UDC	004.8	学位论文编号	D-10617-308-(2019)-01054

重庆邮电大学硕士学位论文

中文题目	基于深度学习的脱机手写汉字识别技术研究
英文题目	**The Research of Offline Handwritten Chinese Character Recognition Based on Deep Learning**
学　　号	**S160101055**
姓　　名	黄洋
学位类别	工学硕士
学科专业	信息与通信工程
指导教师	谭钦红　副教授
完成日期	**2019 年 6 月 2 日**

摘要

汉字是世界上使用最多的文字，汉字识别在残疾人无障碍阅读、文献自动录入、邮件分拣、银行票据处理、证件识别等领域有着重要的应用价值。汉字数量巨大，手写风格各异，并且汉字中存在大量的形近字，导致脱机手写汉字识别一直存在准确率偏低的问题。近年来，深度学习发展迅速，在模式识别、自然语言处理、语音识别等领域都取得了不错的成绩。因此，本文采用深度学习的方法对脱机手写汉字识别进行研究。

针对汉字识别大分类问题，采用深度学习中卷积神经网络的方法对 GB2312-80 标准中规定的最常用的一级 3755 个汉字进行识别。典型的卷积神经网络是一个端到端的结构，直接接受原图输入，但却无法学习到相关领域知识。常规卷积操作中，图像对应区域的所有通道均被同时考虑，无形中增加了网络的冗余度。本文对此进行改进，使用图像的八方向梯度特征作为卷积神经网络输入，使用深度可分离的卷积方式进行卷积。最后设计多组卷积神经网络进行实验，在 CASIA-HWDB 数据集上的实验结果表明，八方向梯度特征输入与深度可分离卷积能够显著提升汉字识别效果，最终取得了 95.86%的准确率。

针对脱机手写汉字识别中形近字难以识别问题，从两个方面进行改进。方法一，使用卷积神经网络加中心损失的方法对相似手写汉字进行识别。引入度量学习中中心损失函数到卷积神经网络，使用交叉熵损失及中心损失作为卷积神经网络的联合损失，使模型学习到更加具有鉴别能力的特征，减小同类样本之间的距离，增加不同类样本之间的距离。方法二，使用卷积神经网络加支持向量机的方式对相似手写汉字进行识别。将卷积神经网络当作一个特征提取器，使用卷积神经网络全连接层输出的特征向量训练支持向量机分类器，识别相似手写汉字。实验结果表明，使用联合损失函数的方法，及卷积神经网络加支持向量机的方式相对于单独使用卷积神经网络的方式，平均识别准确率能够提升 2.68%，1.59%。

关键词：脱机手写汉字识别，深度学习，卷积神经网络，中心损失，支持向量机

Abstract

Chinese characters are the most widely used words in the world. And Chinese character recognition has important application scenarios in the fields of disabled people's accessible reading, automatic document entry, mail sorting, bank bill processing, and identity recognition. There are a number of Chinese characters and the handwriting styles are different. Apart from that, there are a large number of similar characters in Chinese characters. So, the accuracy remains low in offline handwritten Chinese character recognition. In recent years, deep learning has developed rapidly, and has achieved good results in the fields of pattern recognition, natural language processing, and speech recognition. Therefore, this thesis studies offline handwritten Chinese character recognition based on the deep learning method.

For the large classification problem of Chinese character recognition, the convolutional neural network in deep learning is used to recognize the most commonly used first-class 3755 Chinese characters collected in GB2312-80 standard. A typical convolutional neural network is an end-to-end structure, which directly uses the original image as its input, but it can’t learn the relevant domain knowledge. In the conventional convolution operation, all channels in the corresponding area of the image are considered simultaneously, which inevitably increases the redundancy of the network. In this thesis, the eight-direction gradient feature of the image is used as the input of the convolutional neural network, and the depthwise separable convolution method is used to convolute the image. Finally, multiple groups of convolutional neural network are designed for experiments. The experimental results on the CASIA-HWDB dataset show that the eight-direction gradient feature input and the depthwise separable convolution can significantly improve the recognition effect of Chinese characters, and finally achieve an accuracy of 95.86%.

For the problem that it is difficult to recognize similar characters in offline handwritten Chinese character recognition, this thesis improves it from two aspects. The first method combines convolutional neural network with center loss to recognize similar handwritten Chinese characters. The central loss function in metric learning is introduced to the convolutional neural network, and the cross-entropy loss and the center loss are used as the joint loss of the convolutional neural network. So the model

can learn more discriminative features which can reduce the distance between same samples and increase the distance between different types of samples. The second method combines convolutional neural network with support vector machine to recognize similar handwritten Chinese characters. The convolutional neural network is regarded as a feature extractor, and the feature vectors of the convolutional neural network are used to train the support vector machine classifier to recognize similar handwritten Chinese characters. The experimental results show that the average recognition accuracy can be improved by 2.68% and 1.59% by using joint loss function and convolutional neural network plus support vector machine compared with using convolutional neural network alone.

Keywords: offline handwritten Chinese character recognition, deep learning, convolutional neural network, center loss, support vector machine

目录

图录

表录

注释表

CNN	Convolutional Neural Network，卷积神经网络
MQDF	Modified Quadratic Discriminant Function，改进的二次判别函数
SVM	Support Vector Machine，支持向量机
HMM	Hidden Markov Model，隐马尔科夫模型
DLQDF	Discriminative Learning Quadratic Discriminant Function，鉴别学习二次判决函数
LVQ	Learning Vector Quantity，学习矢量量化
ICDAR	International Conference on Document Analysis and Recognition，文档分析与识别国际会议
HLDA	Heteroscedastic Linear Discriminant Analysis，异方差线性判别分析
DFE	Discriminative Feature Extraction，鉴别特征提取
LDA	Linear Discriminant Analysis，线性判别分析
BP	Back Propagation，反向传播
DBN	Deep Belief Network，深度信念网络
RBM	Restricted Boltzmann Machine，受限玻尔兹曼机
ILSVRC	ImageNet Large Scale Visual Recognition Competition，ImageNet 大规模视觉识别竞赛
ReLU	Rectified Linear Unit，修正线性单元
GPU	Graphics Processing Unit，图形处理器
RNN	Recurrent Neural Network，循环神经网络
RNN	Recursive Neural Network，递归神经网络
LSTM	Long Short-Term Memory，长短期记忆网络
PCA	Principal Component Analysis，主成分分析
OVO	One Versus One，一对一法
OVR	One Versus Rest，一对多法

第 1 章 绪论

1.1 研究背景与意义

随着信息全球化的发展，文字的传播越来越不局限于纸质文档，很多时候需要对纸质文档进行数字化处理。如果借助人力手动输入，不仅效率低下，也很容易出现错误。汉字是全世界范围内使用最多的文字，利用图像识别方法，自动进行汉字识别，可以极大地解决汉字录入慢这一瓶颈。最近几年，人工智能技术发展迅速，汉字识别的应用场景也越来越广泛。研究表明，人类 70%以上的认知来自于视觉，对于视觉障碍人士而言，如果能够借助辅助设备，将外界的文本信息，例如书籍、商品包装上的文字，转换为语音信息，能极大地方便他们的日常生活。无人驾驶中，道路交通牌上的文字提供了丰富的道路交通信息，对道路交通牌上的文字进行识别，可以帮助汽车规范安全驾驶。除此之外，汉字识别在邮件分拣、银行票据处理、证件识别、主观题自动阅卷等领域都有广阔的应用场景[1]。

汉字识别可以分为印刷体汉字识别与手写体汉字识别两大类，对于手写体汉字识别而言，根据书写时媒介的不同，又可细分为脱机手写汉字识别与联机手写汉字识别[2]。汉字识别的分类如图 1.1 所示。

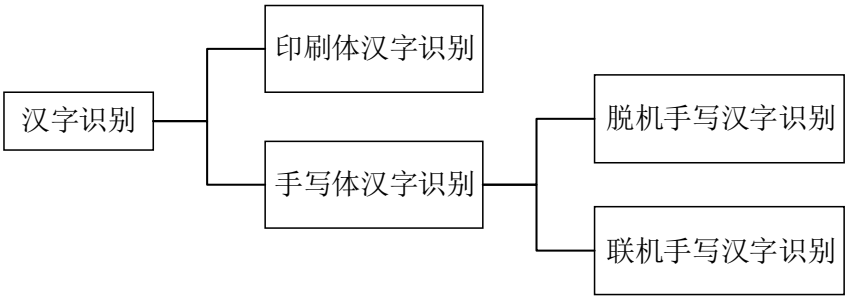


图 1.1 汉字识别分类

很明显，印刷体汉字识别即对打印出来的纸质材料进行识别，汉字字迹工整、清晰；联机手写汉字识别主要指的是对书写在电子屏幕上的汉字进行识别，例如手机、平板等电子元器件的手写输入，它在识别时记录了汉字书写的时序信息及笔画相对位置信息；脱机手写汉字识别所处理的对象为扫描仪或摄像头采集到的

离线的手写汉字二维图片，丢失了时序信息。一般而言，脱机手写汉字识别比联机手写汉字识别更加困难。本文所研究的内容就是脱机手写汉字识别。

深度学习是在传统神经网络的基础上发展起来的一种机器学习方法，它的设计是对人脑的思维学习过程的模拟，采用多层次的结构，具有很强的抽象学习能力。近十年来，由于计算机性能的巨大提升，以及数据样本的极大丰富，深度学习发展迅速，在人脸识别[3, 4]、语音识别[5, 6]、自然语言处理[7, 8]、无人驾驶[9, 10]等领域取得了不少的成就。字符识别是模式识别中一个重要的研究领域，结合深度学习，对脱机手写汉字进行识别具有重要的意义。

1.2 国内外研究现状

1.2.1 汉字识别研究现状

汉字识别的研究起源于上世纪 60 年代，1966 年，IBM 公司的 R. Casey 和 G. Nagy 发表了关于印刷汉字识别的论文，开启了汉字识别研究历程。他们通过模板匹配的方法，对 1000 个印刷体汉字进行了有效的识别[11]。1977 年，日本东芝公司开发出首个可以识别 2000 个不同汉字的印刷体汉字识别系统[12]。80 年代，GB2312-80 字符集[13]的推出极大地加速了汉字识别的研究。1981 年，IBM 公司的 E. F. Yhap 等设计出一套较为成熟的联机手写汉字识别系统，该系统基于汉字笔画，字根编码的思想对汉字进行识别，对 920 个汉字的实验结果准确率为 91.1%，对 2260 个汉字实验结果准确率为 79.9%，但要求书写者使用工整楷书书写。1988 年，国内刘迎键等人提出利用笔段为基元的联机手写汉字识别技术，对于手写正楷汉字，识别字典可达 6763-12000，熟练用户的准确率可达 95%以上。随着印刷汉字与联机手写汉字识别技术的不断提升，脱机手写汉字识别技术也得到了很大的发展，1996 年，中科院自动化研究所开发出一套脱机手写汉字识别系统，其适用于特定的人群，识别率可以达到 93.6%[14]。

经过数十年的发展，目前，在联机手写体汉字识别与印刷体汉字识别方面，国内外研究已比较成熟，出现了许多商用产品。例如，现如今，智能手机、平板电脑都已实现了手写输入的功能，国内也有众多功能较好的印刷体汉字识别软件，

例如扫描全能王、汉王 OCR、科大讯飞 OCR 等。然而，脱机手写汉字识别仍然是一个还未完全解决的课题，是当前模式识别的热点与难点。

传统的脱机手写汉字识别系统主要包括数据预处理、特征提取和分类识别三部分[15]。数据预处理主要包括样本归一化[16]、整形变换、伪样本生成等。特征提取部分可以分为结构特征和统计特征两种，结构特征主要对汉字结构、笔画或部件进行分析来提取[17]。但对于手写字符而言，目前最好的特征基本上都是统计特征，例如方向特征，Gabor 特征及梯度特征[18]。分类器最常用的模型包括改进的二次判决函数(Modified Quadratic Discriminant Function，MQDF)[19]、支持向量机(Support Vector Machine，SVM)[20]、隐马尔科夫模型(Hidden Markov Model，HMM)[21]、鉴别学习二次判决函数(Discriminative Learning Quadratic Discriminant Function，DLQDF)[22]和学习矢量量化(Learning Vector Quantity，LVQ)[23]等。

经过几十年来研究学者的不懈努力，脱机手写汉字识别取得了很大进展。在 2010 年，中科院自动化研究所模式识别国家实验室公开了脱机手写汉字数据集 CASIA-HWDB，并且在 2011 年和 2013 年连续举办了两届 ICDAR(International Conference on Document Analysis and Recognition，文档分析与识别国际会议)国际手写汉字识别比赛[24, 25]，促进了手写汉字识别的进一步发展。清华大学的王阳伟等人，使用级联的 MQDF 分类器，结合异方差线性判别分析(Heteroscedastic Linear Discriminant Analysis，HLDA)方法，在 ICDAR 竞赛数据集上取得了 91.54%的正确率。文献[26]中使用鉴别特征提取方法(Discriminative Feature Extraction，DFE)和 DLQDF 分类器，在 ICDAR 竞赛数据集上取得了 92.72%的正确率。

值得一提的是，连续两届的 ICDAR 手写汉字识别比赛的获胜者都是采用基于深度学习或神经网络的方法。在 2011 年的 ICDAR 脱机手写汉字识别比赛中，瑞士的 U. Meier 和 D. Ciresan 首次采用深度学习的方法对脱机手写汉字进行识别，获得了脱机手写汉字单字识别比赛的第一名，识别率高达 92.18%。2013 年的 ICDAR 手写汉字识别比赛中，来自富士通公司的团队采用改进的 CNN(Convolutional Neural Network，卷积神经网络)，获得了脱机手写汉字识别比赛的第一名，识别率高达 94.77%。可以看出，深度学习在脱机手写汉字识别上具有重要的应用价值，能大幅提高识别准确率。

在脱机手写汉字识别中，相似汉字严重阻碍着汉字识别准确率的进一步提高。近几年来，人们也开始重视相似手写汉字识别。文献[27]使用线性判别分析(Linear Discriminant Analysis，LDA)和相似模式判别分析，结合级联的 MQDF 分类器对相似汉字进行识别。随着深度学习在图像分类领域的巨大成功，更多的学者在使用深度学习识别手写汉字识别的同时，也开始使用深度学习的方法来对相似手写汉字进行识别。文献[28]通过云端服务平台获取海量手写数据，使用 CNN 进行相似手写汉字识别；文献[29]使用弹性形变对数据集进行扩充，结合 CNN 识别了 15 组相似汉字，每组包括 10 个易混淆的汉字；文献[30]使用级联的 CNN 对 172 组，368 个相似汉字进行了识别。

1.2.2 深度学习技术研究现状

1989 年，美国纽约大学 Y. Lecun 教授首次提出了卷积神经网络模型结构，该模型包括 3 个隐藏层，采用局部连接、权值共享策略，使用反向传播(Back Propagation，BP)算法进行训练，并成功应用于手写邮政编码识别中[31]。1998 年，Y. Lecun 在先前研究的基础上，提出了经典的 LeNet-5 模型，并成功应用于美国众多银行的支票识别系统中[32]。该模型的诞生标志着卷积神经网络的成熟。LeNet-5 模型共计 7 层，前六层由卷积层和池化层交替构成，最后由 Softmax 层输出结果，这在当时是一个非常新颖的想法，这种结构对后来的网络设计产生了深远的影响。

但是深度学习由于难以训练等原因，一段时间内发展缓慢。2006 年，加拿大多伦多大学 G. E. Hinton 教授提出了深度信念网络(Deep Belief Network，DBN)的概念[33, 34]。解决了深层神经网络训练难，容易陷入局部最值的问题。2012 年，G. E. Hinton 及其学生 A. Krizhevsky 设计的 AlexNet 卷积神经网络模型[35]，在 ImageNet 大规模视觉识别挑战赛(ImageNet Large Scale Visual Recognition Competition，ILSVRC)上，取得了比赛的冠军，深度学习引起人们侧目。在将 120 万张高分辨率图像分为 1000 个类别的任务中，AlexNet 网络模型取得了 15.3%的 Top5 错误率。在该模型中，首次使用了修正线性单元(Rectified Linear Unit，ReLU)激活函数，Dropout 技术，并且使用了 GPU(Graphics Processing Unit，图形处理器)加速运算，极大地促进了深度学习的发展。2014 年，谷歌公司 C. Szegedy 等人提出的 GoogLeNet 网络模型[36]，牛津大学 K. Simonyan 和 DeepMind 公司 A. Zisserman 联

合推出的 VGGNet 深度学习模型[37]，分别取得了 2014 年 ImageNet 比赛的第一、二名。

目前，深度学习的研究呈现井喷的状态，日新月异，各种网络变体被提出，基于区域的卷积神经网络[38]、深度强化学习[39]、指针网络[40]、深度残差网络[41]、胶囊网络[42]等，在人脸识别、语音识别、目标检测、无人驾驶等领域都取得了很好的成绩。

国内，深度学习也发展迅速，2013 年百度率先成立了深度学习研究院[43, 44]，此后 360、腾讯、阿里等公司相继成立了人工智能实验室。2017 年 11 月，国家公布了首批国家新一代人工智能开放创新平台，依托百度、阿里云、腾讯、科大讯飞公司建立自动驾驶、城市大脑、医疗影像、智能语音新一代人工智能开放创新平台。

1.2.3 脱机手写汉字识别的研究难点

脱机手写汉字识别的研究难点主要表现在以下 3 个方面：

1. 汉字识别是一个大分类任务，汉字的个数超过 50000 个，GB2312-80 标准中规定的最常用一级汉字有 3755 个，次常用二级汉字 3008 个，与英文字母识别（包括大小写）52 分类及数字识别 10 分类相比，脱机手机汉字识别无疑更加具有挑战性；

2. 脱机手写汉字风格各异，不同人的笔迹差别较大，即使是同一人，不同状态下书写的字迹也不尽相同，再加上横、竖、撇、捺、弯钩等笔画书写时极易变形，大大增加了识别的难度；

3. 汉字中存在大量的形近字，例如“我、找、伐”等，“日、曰”等，这些字有些仅相差一个笔画，有些就是长宽比例的不同，对于刚接触汉字的人来说，正确分辨这些形近字也是一件头疼的事情，对于机器而言，就更容易混淆。

1.3 论文主要研究内容

本文对基于深度学习的脱机手写汉字识别技术进行了研究。后文内容中，如无特别说明，所提手写汉字识别均指脱机手写汉字识别。主要研究内容包括以下几个部分：

1. 对手写汉字识别研究现状、相关技术进行了调研与分析，提出使用深度学习中卷积神经网络的方法对 GB2312-80 标准中规定的最常用的 3755 个汉字进行识别。

2. 卷积神经网络的输入一般采用原图输入的方式，在一定程度上，避免了特征提取过程，但是却丢失了一些神经网络无法学习的先验领域知识。本文提取汉字图像的八方向梯度特征，使用八方向梯度特征图像作为卷积神经网络的输入，提升了汉字识别的准确率。

3. 卷积神经网络中最重要的概念就是“卷积”，卷积神经网络的性能很大程度上受到卷积方式的影响，本文使用深度可分离卷积方式进行卷积，相比于常规卷积，即减少了模型参数又提升了模型识别效果。最后设计多组卷积神经网络对手写汉字进行识别，统计 3755 个汉字错误识别数据，详细分析汉字错误识别原因。

4. 针对手写汉字识别中，形近字难以识别问题，对卷积神经网络损失函数进行改进，使用 Softmax 损失函数和 Center Loss 函数作为卷积神经网络的联合损失函数，增大不同类别汉字的类间距离，提升相似手写汉字识别准确率。

5. 同样针对形近字难以识别问题，提出使用卷积神经网络加支持向量机的方法对相似手写汉字进行识别。将卷积神经网络当作一个特征提取器，使用卷积神经网络全连接层输出的特征向量训练 SVM 分类器，识别相似手写汉字。

1.4 论文的组织结构

本文基于深度学习对脱机手写汉字进行识别，全文内容划分为五章。

第 1 章，介绍了手写汉字识别的研究背景与意义，并且详细介绍了汉字识别以及深度学习的国内外研究现状，分析了汉字识别的研究难点，并阐述了本文的主要研究内容及章节安排。

第 2 章，首先分析对比了传统脱机手写汉字识别方法与基于卷积神经网络的脱机手写汉字识别方法；然后详细介绍了全连接神经网络及卷积神经网络的技术原理；最后介绍了卷积神经网络设计中常用的分类器、损失函数及避免过拟合和加速网络训练的方法。

第 3 章，设计卷积神经网络，对 GB2312-80 规定的一级 3755 个汉字进行识别。首先介绍了手写汉字识别基本流程及图像预处理方法，接着介绍了汉字图像八方

向梯度特征提取方法，然后构建卷积神经网络对手写汉字进行识别，分析了卷积神经网络输入、卷积方式及神经网络复杂度对汉字识别准确率的影响，最后对错误结果进行分析，发现错误原因主要为相似汉字的影响。

第 4 章，相似手写汉字专项识别，首先根据一级 3755 类汉字错误识别统计结果，挑选出具有代表性的 15 组相似汉字，每组包括 10 种汉字。然后使用两种不同方法对相似手写汉字进行识别。方法一，使用 Softmax Loss 损失函数和 Center Loss 损失函数作为 CNN 的损失函数。方法二，使用 SVM 分类器替代卷积神经网络中常用的 Softmax 分类器，对相似手写汉字进行分类。最终验证了 CNN+Center Loss 函数及 CNN+SVM 在手写汉字识别中的有效性。

第 5 章，总结与展望，对课题所做工作进行总结，并对手写汉字识别的下一步研究方向进行展望。

第 2 章 手写汉字识别技术分析

2.1 引言

深度学习发展至今，演变出许多经典的深度学习模型，主要包括深度信念网络、卷积神经网络、循环神经网络(Recurrent Neural Network，RNN)，递归神经网络(Recursive Neural Network，RNN)、生成对抗网络等，如图 2.1 所示。每种网络都有其侧重的应用场景。

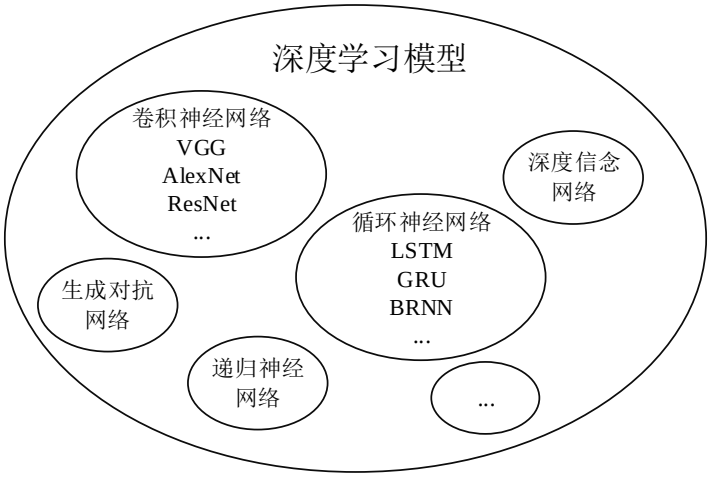


图 2.1 深度学习常用模型

深度信念网络是一种生成模型，经典的 DBN 网络结构由若干个受限玻尔兹曼机及一层 BP 网络组成。DBN 可以用于特征提取，对一维数据的建模比较有效，例如语音识别，在其他领域中应用较少，更多是了解深度学习“哲学”和“思维模式”的一个手段。

循环神经网络常用于处理序列信息，例如利用自然语言处理中的语言模型建模，视频图像分析等，典型的循环神经网络为长短期记忆网络(Long Short-Term Memory，LSTM)。递归神经网络是循环神经网络的推广，主要用于处理具有树或者图结构的信息。生成对抗网络是一种无监督学习方法，通过生成模型和判别模型的相互博弈产生好的输出，主要用于图像生成和数据增强。

卷积神经网络是在全连接神经网络的基础上发展起来的，主要由卷积层、池化层和全连接层组成，通过不断地堆叠卷积层和池化层来构建深层的网络模型。

卷积神经网络和图像紧密相连，非常适合处理结构化数据，由于卷积神经网络在图像分类上已经取得的良好效果，本文主要使用卷积神经网络来对手写汉字进行分类识别。

2.2 手写汉字识别方法对比

手写汉字识别方法，主要分为两大类，传统的“特征提取+分类器”的方法，以及最近几年兴起的基于卷积神经网络的方法。

2.2.1 传统脱机手写汉字识别技术分析

传统的手写汉字识别方法在多年的研究过程中，已经形成了一套较为成熟的流程，主要包括图像预处理，特征提取，特征压缩，分类器设计几个步骤。以梯度特征为例，传统手写汉字识别流程如图 2.2 所示[45]。

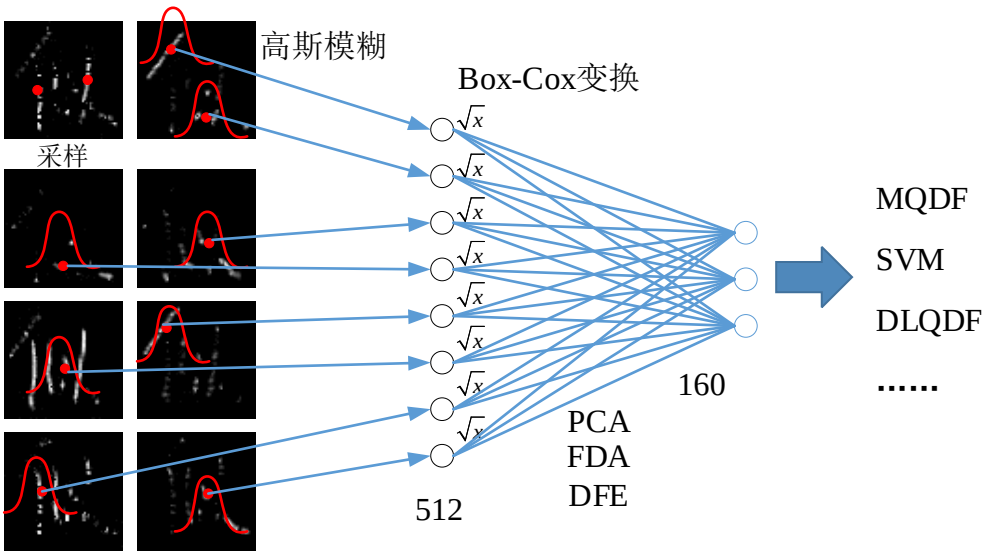


图 2.2 传统手写汉字识别方法流程图

首先对汉字图像进行预处理，主要包括灰度归一化和尺寸归一化。然后进行特征提取，如图 2.2 所示，生成汉字图像的梯度特征图后，选择特征图中具有代表性的特征点，在每个特征点处，使用高斯模糊来减小笔画位置变动的影响，生成特征向量，并且使用 Box-Cox 转换[46]来增加特征向量的高斯性。接着进行特征压缩，降低特征维度，加快运算速度。常见的特征压缩方法有主成分分析(Principal

Component Analysis，PCA)，线性判别分析，鉴别特征提取等。最后选择合适的分类器进行分类，常用的分类器有 MQDF、QLQDF、SVM 等。

2.2.2 基于卷积神经网络的脱机手写汉字识别

基于卷积神经网络的手写汉字识别示意图如图 2.3 所示。网络主要由输入层、卷积层、池化层、全连接层和输出层组成。使用卷积神经网络识别手写汉字时，需要使用大量带标签的汉字样本对网络进行训练，使网络学习到合适的模型参数，再使用训练好的模型对样本进行识别。

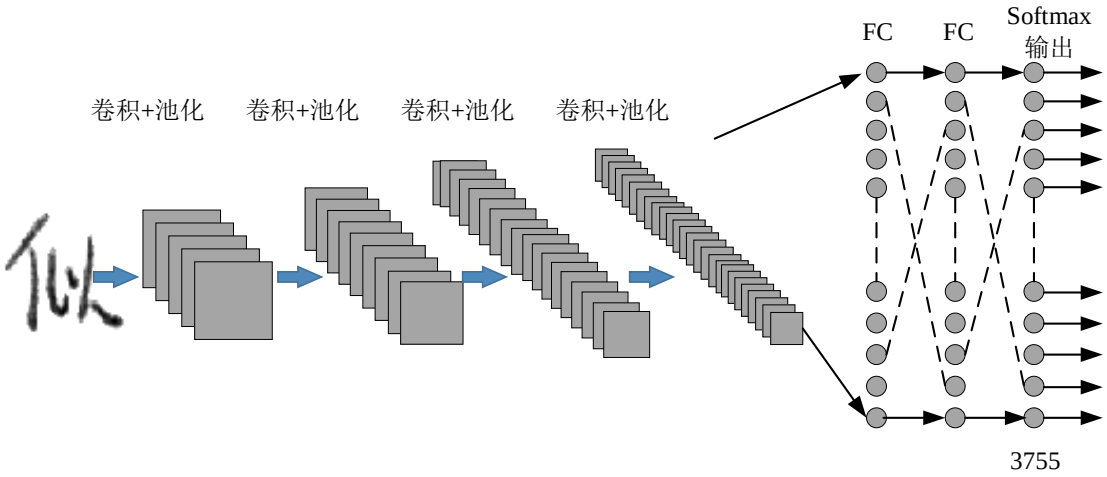


图 2.3 基于卷积神经网络的手写汉字识别示意图

基于卷积神经网络识别手写汉字时，一般直接使用汉字图像作为输入，自动从图像中学习特征，进行分类，避免了手动特征提取过程。整个过程包括模型训练和预测阶段。模型训练包括前向传播和反向传播两个部分。模型的前向传播就是根据输入数据计算得到输出数据的过程。汉字图像送入卷积神经网络后，首先在卷积层使用多个卷积核对图像进行卷积，提取汉字图像特征，生成多通道的特征图；然后在池化层进行池化操作，降低特征图的维度；多次卷积和池化操作后，将抽取到的特征图送入全连接层，经过多次非线性映射后生成一个特征向量，最终送往输出层的 Softmax 分类器进行分类。

模型预测时，只涉及到前向计算过程，汉字图像输入卷积神经网络之后，一级一级地向前传递，到达输出层，产生预测结果。

利用卷积神经网络识别手写汉字，需要设计合适的卷积神经网络结构。涉及到众多超参数的设置，例如全连接层神经元个数，卷积层每层卷积核个数，激活函数选择等，往往需要多次实验进行确定。

传统手写汉字识别方法和基于卷积神经网络的手写汉字识别方法虽然看起来并不相同，但是有很多地方也是类似的。例如，传统方法中的高斯模糊可以看作卷积神经网络中卷积操作，虽然高斯模糊参数是预定义的，卷积核参数是从数据中学习到的；Box-Cox 转换与卷积神经网络中非线性激活函数类似；分类器对应于卷积神经网络中 Softmax 层。可以认为，使用卷积神经网络的方法进行手写汉字识别，与传统的方法有着异曲同工之意。

传统“特征提取+分类器”的手写汉字识别方法虽然成熟，但是存在正确率偏低的缺点。特别是最近几年，传统的手写汉字识别方法鲜有大的突破，而基于卷积神经网络的汉字识别方法经常虽然耗时，但是却能够取得较高的识别准确率。因此，本文基于卷积神经网络对手写汉字进行识别。

2.3 全连接神经网络

2.3.1 全连接神经网络结构

全连接神经网络是最简单的神经网络，卷积神经网络是在全连接神经网络的基础上发展起来的，卷积神经网络中很多基本概念都和全连接神经网络中一样。

神经元是神经网络最基础的构成单元，一个神经元接受多个输入，并产生一个输出。单个神经元的结构如图 2.4 所示，包括输入、输入权值、激活函数、输出几个部分。

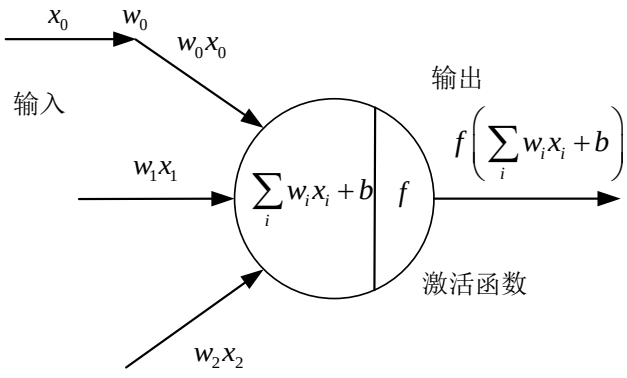


图 2.4 基本神经元结构

全连接神经网络其实就是按照一定规则连接起来的多个神经元，其网络结构如图 2.5 所示。神经网络按照层来布局，从左到右依次为输入层、隐藏层、输出层。“隐藏”这一术语稍显神秘，其实它仅仅意味着“既非输入也非输出”。全连接神经网络中同一层神经元之间没有连接，前一层的每个神经元和后一层的所有神经元相连，前一层的神经元输出就是后一层神经元的输入，每个连接上都对应一个权值。

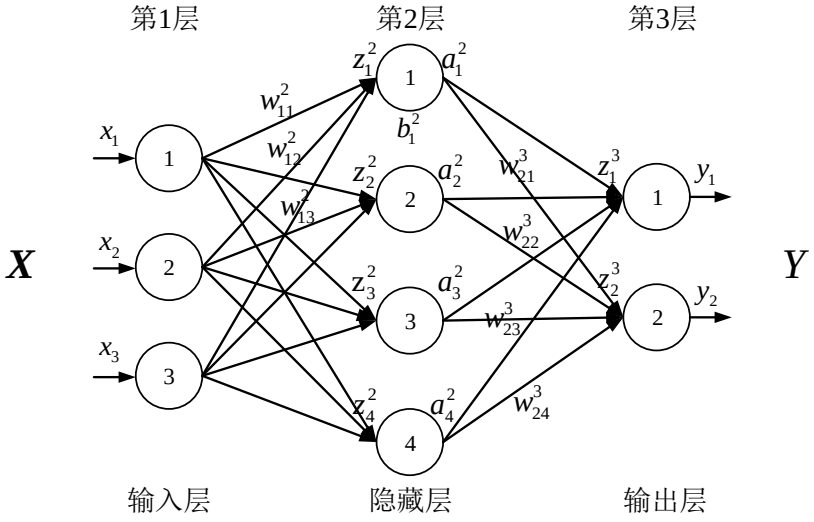


图 2.5 全连接神经网络结构

利用全连接神经网络可以完成分类或回归预测任务，为此，需要大量带有标签的样本来训练网络。当模型学习到足够多的样本之后，就能总结出其中的一些规律，学习到合适的模型参数，从而预测那些它没有遇见过的样本所对应的答案。

2.3.2 模型的前向传播和反向传播

全连接神经网络的训练包括前向传播和反向传播两个部分。模型的前向传播就是根据输入数据计算得到输出数据的过程。如图 2.5 所示，w_{jk}^{l} 为第 l 层上第 j 个节点与第 $l-1$ 层上第 k 个节点的权值，b_{j}^{l} 为第 l 层上第 j 个节点的偏置值，z_{j}^{l} 为第 l 层第 j 个节点的加权输入，a_{j}^{l} 为第 l 层第 j 个节点的激活输出值。假设 f 为第 $l-1$ 层

激活函数，网络的损失函数为C，δ_j^l为第l层第j个节点上的误差，$\delta_j^l=\frac{\partial C}{\partial z_j^l}$。则，模型第$l$层上第$j$个神经元的输出值为：

$$a_j^l=f\left(\sum_k w_{jk}^l a_k^{l-1}+b_j^l\right) \tag{2.1}$$

模型的反向传播过程就是更新参数最优化损失函数C的过程，C为权重w和偏置b的函数，根据梯度下降算法策略来更新权重w和b：

$$w_{jk}^l=w_{jk}^l-\eta\frac{\partial C}{\partial w_{jk}^l} \tag{2.2}$$

$$b_j^l=b_j^l-\eta\frac{\partial C}{\partial b_j^l} \tag{2.3}$$

式中η为学习率，控制w，b调整的步伐，一般为很小的正数。反向传播实质是求$\frac{\partial C}{\partial w_{jk}^l}$，$\frac{\partial C}{\partial b_j^l}$。第$l$层神经元误差为：

$$\delta_j^l=\frac{\partial C}{\partial a_j^l}f'\left(z_j^l\right) \tag{2.4}$$

使用下一层的误差对上一层的误差进行表示：

$$\delta_j^l=\sum_i w_{ij}^{l+1}\delta_i^{l+1}f'\left(z_j^l\right) \tag{2.5}$$

损失函数关于权重的偏导为：

$$\frac{\partial C}{\partial w_{jk}^l}=\delta_j^l a_k^{l-1} \tag{2.6}$$

$$\frac{\partial C}{\partial b_j^l}=\delta_j^l \tag{2.7}$$

则损失函数对权重和偏置的偏导可以根据公式(2.4)-(2.7)求出。

2.4 卷积神经网络

2.4.1 卷积神经网络结构

卷积神经网络是在全连接神经网络的基础上发展起来的，典型的卷积神经网络结构如图 2.6 所示。一个卷积神经网络由输入层、卷积层、池化层、全连接层和

输出层组成，它在全连接神经网络的基础上增加了局部连接、权值共享、下采样等策略。

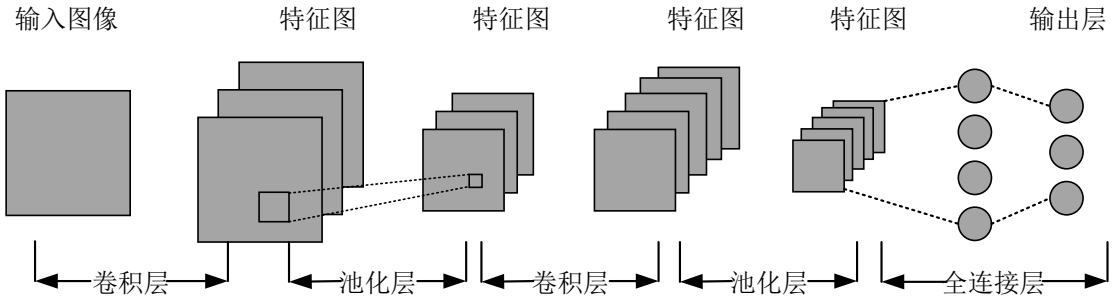


图 2.6 卷积神经网络结构

卷积神经网络的训练流程与全连接神经网络基本一致，根据梯度下降策略和反向传播算法，更新模型参数，来最小化损失函数。

在卷积层中，使用卷积核对输入图像进行卷积，生成特征图。特征图中每个像素与输入图像的一小块区域相对应，这个区域也叫做局部感受野。卷积时，平等对待图像的不同区域，所以每个特征图共享一组卷积核权重参数。这样的局部连接和权值共享策略，减少了网络中大量参数。实际中，通常使用多个卷积核来获得更多有用的特征。

卷积其实就是对图像进行滤波，设输入图像的尺寸为(W_1,H_1)，卷积核的尺寸为(M,N)，卷积的步长为S，则卷积后，生成特征图的尺寸为(W_2,H_2)，其可由式(2.8)求出。

$$\begin{cases} W_2=(W_1-M+2P)/S+1 \\ H_2=(H_1-N+2P)/S+1 \end{cases} \tag{2.8}$$

其中P为 0 填充数量，指的是在原始图像周围补几圈 0，一般用于保持卷积后图像尺寸大小不变。卷积的示意图如图 2.7 所示。

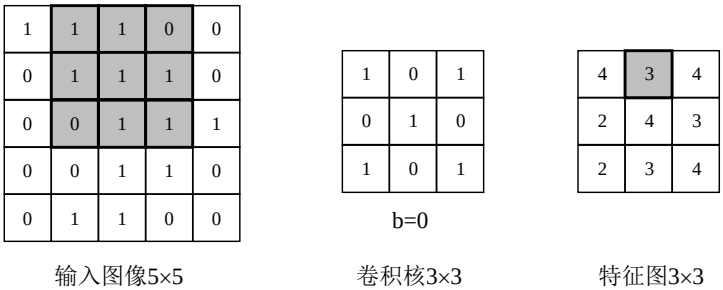


图 2.7 卷积示意图

用 x_{ij} 表示输入图像的第 i 行第 j 列元素，w_{mn} 表示卷积核第 m 行第 n 列权重，b 表示卷积核的偏置项，a_{ij} 表示卷积后生成的特征图第 i 行第 j 列的元素，f 表示激活函数。则卷积后输出值 a_{ij} 为：

$$a_{ij} = f\left(\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} w_{mn}x_{i+m,j+n} + b\right) \tag{2.9}$$

池化就是降采样，使用池化来减小图片的维度，进一步减少参数数量，同时还可以提升模型的鲁棒性。池化方法有很多，常用的为最大值池化，平均值池化。最大值池化实际就是在 $n\times n$ 的样本中取最大值，作为采样后的样本值。池化步长为 2 时，2×2 最大值池化示意图如图 2.8 所示。

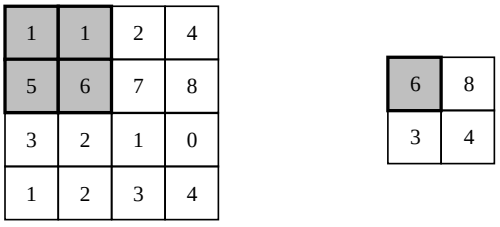


图 2.8 最大值池化示意图

卷积神经网络相比全连接神经网络，增加了卷积层和池化层，反向传播时，都是先根据当前层的误差 $\boldsymbol{\delta}^{l}$，表示出上一层的误差 $\boldsymbol{\delta}^{l-1}$，最后求出损失函数对权重的偏导 $\frac{\partial C}{\partial \boldsymbol{W}^{l}}$ 与 $\frac{\partial C}{\partial b^{l}}$。在卷积层中，

$$\boldsymbol{\delta}^{l-1} = \boldsymbol{\delta}^{l} * rot180\left(\boldsymbol{W}^{l}\right) \odot f'\left(\boldsymbol{z}^{l-1}\right) \tag{2.10}$$

其中 $*$ 表示卷积操作，符号 $\odot$ 表示矩阵对应位置元素相乘，相当于把第 l 层的特征图周围补一圈 0，再与 180 度翻转后的卷积核进行乘积。卷积层损失函数对权重和偏置的导数即可由式(2.11)与式(2.12)求出。

$$\frac{\partial C}{\partial \boldsymbol{W}^{l}} = \boldsymbol{a}^{l-1} * \boldsymbol{\delta}^{l} \tag{2.11}$$

$$\frac{\partial C}{\partial b^{l}} = \sum_{m}\sum_{n} \delta^{l}_{mn} \tag{2.12}$$

池化层没有需要学习的参数，在卷积神经网络的反向传播中，池化层需要做的仅仅是将误差项传递到上一层，而没有梯度的计算。对于最大值池化，下一层

的误差项的值会原封不动的传递到上一层对应区块中的最大值所对应的神经元，而其他神经元的误差项的值都是 0。

2.4.2 Softmax 分类器与交叉熵损失函数

卷积神经网络用于分类任务时，输出层一般为 Softmax 层。Softmax 函数公式为：

$$s_i = \frac{e^{z_i}}{\sum_k e^{z_k}} \tag{2.13}$$

其中 s_i 代表第 i 个神经元的输出，z_i 代表第 i 个神经元的加权输入。s_i 取值范围为 [0,1]，$\sum s_i = 1$，所以 Softmax 层的输出可以看成是一个概率分布，每个神经元的输出值 s_i 可以理解为被分类为该类别的概率。具体在汉字识别中，输出即为识别为某个汉字的概率，不仅如此，按照输出概率从大到小排列，可以获得汉字的 Top-n 分类概率，在文本识别中具有重要意义，结合上下文语境与 Top-n 分类概率可以帮助纠正许多分类错误。

使用 Softmax 分类器时，损失函数常使用交叉熵损失函数，公式为：

$$C = -\sum_i y_i \ln a_i \tag{2.14}$$

其中 y_i 为样本的标签值，a_i 为样本的真实输出。交叉熵刻画的是通过概率分布 a_i 来表达概率分布 y_i 的困难程度，反映的是两个概率分布的距离，两个概率分布越接近时，交叉熵值越小。

因为 Softmax 分类器与交叉熵损失函数经常一起使用，有时直接把卷积神经网络最后一个全连接层的输出经过 Softmax 函数后再计算得到的交叉熵损失称为 Softmax 损失。本文第 4 章中的 Softmax 损失就是这个含义。

2.5 Dropout 与 Batch Normalization

2.5.1 Dropout

卷积神经网络模型容易出现过拟合问题，即网络在训练集上性能很好，但在测试集上性能大幅下降。过拟合产生的主要原因为网络模型参数太多，而训练样

本太少。常见的减轻过拟合的方法有增加样本数量、正则化、Dropout（弃权）等。本文在识别手写汉字时，使用了 Dropout 策略来减轻过拟合。

在训练神经网络时，通常输入神经元与输出神经元之间都是全连接的。使用 Dropout 技术后，训练时，会随机地临时删除网络中一定比例的隐藏层神经元，同时保持输入和输出层的神经元不变。然后把训练数据通过修改后的网络前向传播，并把得到的损失结果通过修改后的网络反向传播；一小批训练样本执行完这个过程后，在没有被删除的神经元上按照梯度下降法更新对应的参数。重复这一过程，随机地删除相应比例的神经元进行训练，直至最终模型训练结束。在实际预测时，使用整个网络进行预测，不进行弃权。

Dropout 技术减轻过拟合的原理类似于训练多个卷积神经网络求平均。当弃权掉不同的神经元集合时，就如同在训练不同的神经网络，不同的网络会以不同的方式过拟合，平均法能够在一定程度上消除过拟合。同时 Dropout 技术避免了训练多个神经网络带来的昂贵时间代价。

2.5.2 Batch Normalization

卷积神经网络模型复杂时，训练会非常困难，甚至会出现不收敛的情况，使用 Batch Normalization[47]可以加速训练过程，加快卷积神经网络收敛。

Batch Normalization，即批规范化，它对卷积神经网络中每一层神经元的加权输入进行调整，使每一层神经元的加权输入满足均值为 0，方差为 1 的正态分布。批规范化的这种操作使得激活函数的输入值能够落在比较敏感的区域，增大导数值，增强信息的反向传播，减轻梯度消失问题。在训练卷积神经网络时，一次训练过程包括一个 Batch 中 n 个训练实例，批规范化操作就是对每个神经元的加权输入进行如下变换：

$$x^{(k)} = \frac{x^{(k)} - \mathrm{E}\left[x^{(k)}\right]}{\sqrt{\mathrm{Var}\left[x^{(k)}\right]}} \tag{2.15}$$

$$y^{(k)} = \gamma^{(k)} x^{(k)} + \beta^{(k)} \tag{2.16}$$

其中 $x^{(k)}$ 为第 k 个神经元加权输入，$\mathrm{E}\left(x^{(k)}\right)$ 为 k 个神经元加权输入的均值，$\mathrm{Var}\left[x^{(k)}\right]$ 为 k 个神经元的方差。γ、β 为调节参数，$y^{(k)}$ 为批规范化后最终结果。

公式(2.15)含义明显，是将神经元的输入调整为均值为 0，方差为 1 的正态分布。公式(2.16)的目的是通过训练过程中学习到的两个调节因子γ、β对式(2.16)的结果进行微调。因为式(2.15)的操作有可能降低神经网络的非线性表达能力，以此方式来进行补偿。

批规范化一般应用在卷积神经网络中的全连接层与卷积层。因为每一层每个神经元的输入都是一个数值，而批规范化需要知道输入的均值与方差，是一个统计量，因此需要指定一个神经元的集合 S，在这个指定的神经元集合中，利用集合中每个神经元的加权输入来计算出所需的均值和方差。应用在全连接层时，对于神经元 k 而言，一个 Batch 内的 n 个样本都会在神经元 k 处产生一个输入，即 Batch 中的 n 个训练实例分别通过同一个神经元 k 产生了 n 个输入，批规范化的集合 S 就是这 n 个样本在同一个神经元处的加权输入。应用在卷积层时，每个训练样本被一个卷积核进行卷积后，都会生成一个二维激活平面，Batch 中的 n 个训练实例分别通过同一个卷积核后产生 n 个激活平面，批规范化的集合 S 就由这 n 个激活平面内包含的所有激活值构成。

2.6 本章小结

本章首先介绍了深度学习中的常用方法，指出不同深度学习方法常见的应用场合，选择卷积神经网络来对手写汉字进行识别，然后分析对比了传统手写汉字识别方法与基于卷积神经网络的汉字识别方法，接着详细讲解了全连接神经网络和卷积神经网络的基本知识及前向传播和后向传播过程，最后讲解了卷积神经网络设计中常用的Softmax分类器与交叉熵损失函数，Dropout减轻过拟合方法，Batch Normalization 加快网络训练方法。

第 3 章 基于梯度特征和深度可分离卷积的手写汉字识别

汉字数量众多，具体的汉字数量没有确切数字。《中华字海》收录的汉字数量为 85568 个[48]，北京国安咨询设备公司汉字字库收入的有出处汉字就有 91251 个。对所有类别汉字进行识别即不现实，又没有必要。据统计，3000 个汉字就已经满足日常 99%的使用。因此，本章对 GB2312-80 标准中规定的最常用的一级 3755 个汉字进行识别。本章首先讲解了手写汉字识别基本流程及相关预处理方法，之后从卷积神经网络输入、卷积方式两个方面对卷积神经网络进行改进，使用汉字图像八方向梯度特征作为卷积神经网络输入，使用深度可分离卷积方式进行卷积，最后设计多组卷积神经网络对手写汉字进行识别，并进行对比。

3.1 手写汉字识别基本流程

3.1.1 基本流程

手写汉字识别方法流程图如图 3.1 所示。主要包括图像预处理，八方向梯度特征提取，卷积神经网络设计三个部分。卷积神经网络本身自带特征提取和分类功能，可以直接接受原始汉字图像输入，分类产生结果。但是考虑到梯度特征在传统手写汉字识别中的良好性能，本文提取汉字八方向梯度特征作为卷积神经网络输入。在卷积神经网络中，最重要的操作即为卷积，但常规卷积同时对图像的所有通道进行卷积，冗余度高，影响分类效果，本文对此进行改进，使用深度可分离卷积对图像进行卷积。

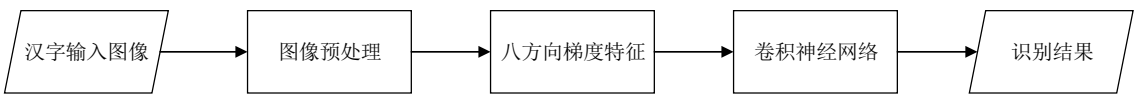


图 3.1 手写汉字识别基本流程

3.1.2 图像预处理

实际应用中，由于图像采集设备和环境的不同，采集到的原始汉字图像易于

受到多种干扰和限制而不能直接应用于识别任务，图像预处理可以有效地减小这些因素的影响。在本文手写汉字识别任务中，使用到的图像预处理操作包括对比度增强和尺寸归一化。

1. 对比度增强：图像在采集过程中，容易受到光照、采集设备的影响，导致采集到的图像出现笔画较轻、亮度不均匀等现象。增强对比度，可以有效地解决这些问题。通过灰度变换，将图像灰度值拉伸到[0,255]：

$$D(x,y)=\frac{\left(I(x,y)-I_{\min}\right)\times 255}{I_{\max}-I_{\min}} \tag{3.1}$$

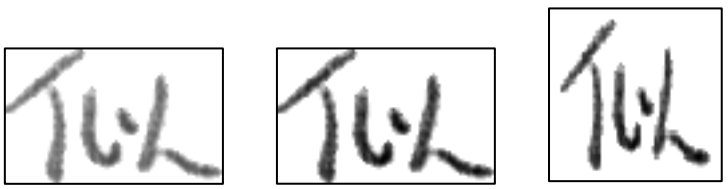
其中 I_{max}、I_{min} 为原图像的最大、最小灰度像素值，$I(x,y)$ 为原图像像素值，$D(x,y)$ 为目标图像像素值。

2. 尺寸归一化：为满足卷积神经网络的输入要求，一般需要调整网络输入图片的尺寸为固定尺寸。在经典的 MNIST 手写数字识别中，输入图像通常被归一化为 28×28 大小，已能够取得良好的识别效果。输入图片尺寸过大会增加网络的训练负担，过小则影响网络的识别性能，考虑到汉字图像比数字更加复杂，本文将汉字图像归一化为 56×56 大小，并且在图像四周各添加四个空白像素，使汉字内容居于图像正中，最终的输入图片尺寸为 64×64。图像尺寸归一化方法很多，本文采用简单的线性归一化方法，使用最近邻插值策略，坐标映射公式为：

$$\begin{cases} x' = \dfrac{W_2}{W_1}x \\ y' = \dfrac{H_2}{H_1}y \end{cases} \tag{3.2}$$

其中，W_1、H_1 为原图像的宽度、高度，W_2、H_2 为目标图像的宽度、高度，(x,y) 为原图像坐标，(x',y') 为目标图像坐标。

随机挑选 CASIA-HWDB1.1 数据集中一张汉字图片“似”，对比度增强及尺度归一化效果如图 3.2 所示，图片边缘黑框起辅助显示作用。



(a) 原图　(b) 对比度增强　(c) 尺寸归一化

图 3.2 图像预处理前后效果对比图

3.2 八方向梯度特征

卷积神经网络是一个端到端的结构，集特征提取与分类功能于一体。其输入通常为原始图像，简单便捷，但是却无法学习到一些有效的先验领域信息，例如在传统手写识别中被证明是行之有效的梯度特征。特征提取是传统手写汉字识别中重要的一步，其中梯度特征得到了最广泛的应用。汉字由基本的横、竖、撇、捺等笔画组成，提取汉字多个方向的梯度特征，可以更加有效地表达汉字图像。本文提取汉字八个方向的梯度特征图像替代原始汉字图像作为卷积神经网络的输入。

使用 sobel 算子计算图像水平方向和垂直方向的梯度，sobel 算子如图 3.3 所示。设图像的坐标为 $f(x,y)$，图像上点 (x,y) 处的梯度为 $G(x,y)=(G_x,G_y)$，则，

$$\begin{cases} G_x(x,y)=f(x+1,y-1)+2f(x+1,y)+f(x+1,y+1) \\ \qquad\quad -f(x-1,y-1)-2f(x-1,y)-f(x-1,y+1) \\ G_y(x,y)=f(x-1,y+1)+2f(x,y+1)+f(x+1,y+1) \\ \qquad\quad -f(x-1,y-1)-2f(x,y-1)-f(x+1,y-1) \end{cases} \tag{3.3}$$

-1	0	1
-2	0	2
-1	0	1

1	2	1
0	0	0
-1	-2	-1

图 3.3 水平和垂直方向 sobel 算子模板

为了提取汉字多个方向的特征，将梯度 $G(x,y)$ 向 0、$\pi/4$、$\pi/2$、$3\pi/4$、π、$5\pi/4$、$3\pi/2$、$7\pi/4$、8 个方向进行分解，生成 8 个与输入汉字图像尺寸相同的特征图。对于每一点的梯度，按照平行四边形法则，将梯度分解到相邻的两个标准方向，对应方向梯度大小即可相应求出，其余 6 个方向对应位置梯度为 0，分解示意图如图 3.4 所示。

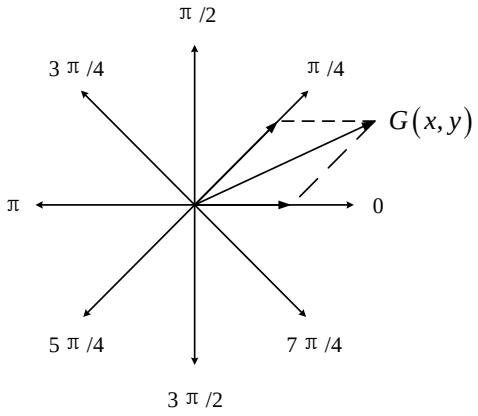
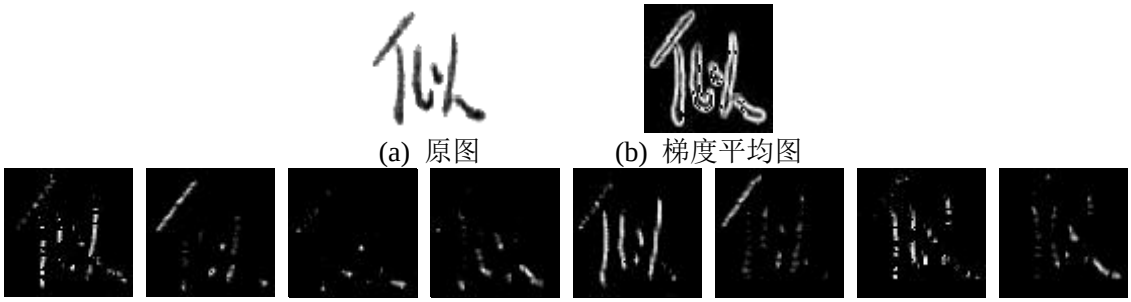


图 3.4 梯度分解示意图

以 CASIA-HWDB1.1 数据集中汉字图片“似”为例，图像预处理后，八方向梯度示意图如图 3.5 所示。



(a) 原图 (b) 梯度平均图

(c) 0 (d) $\pi/4$ (e) $\pi/2$ (f) $3\pi/4$ (g) π (h) $5\pi/4$ (i) $3\pi/2$ (j) $7\pi/4$

图 3.5 八方向梯度平面示意图

图 3.5(a)为原始汉字图像“似”，3.5(b)为八个方向的梯度叠加在一起后的平均梯度图像，3.5(c)到 3.5(j)共 8 幅图像为对应方向的梯度图像。使用 8 幅特征图替代原始图像作为卷积神经网络输入。

3.3 手写汉字识别卷积神经网络结构设计

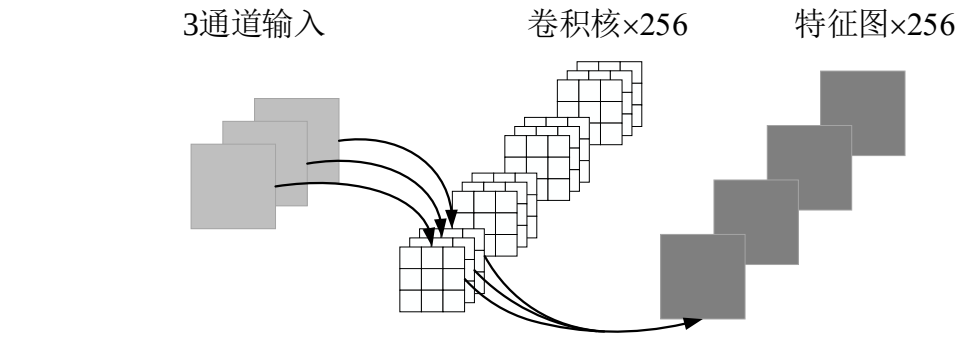
卷积神经网络结构是影响手写汉字识别效果最重要的因素，从 1998 年 LeNet5 卷积神经网络诞生至今，演变出多种多样的网络结构，如 AlexNet、VGGNet、GoogLeNet、ResNet 等，这些卷积神经网络都在其各自应用场景取得了优异的识别性能，但是无法直接用于手写汉字识别。卷积神经网络类似于一个黑匣子，针对特定的应用，需要构建特定的具体的网络结构，无法从一开始就设计出最合适的

网络结构，往往需要多次尝试，反复微调。本文从卷积神经网络输入，卷积方式，卷积神经网络复杂度三个方面，构建不同卷积神经网络来对手写汉字进行识别。

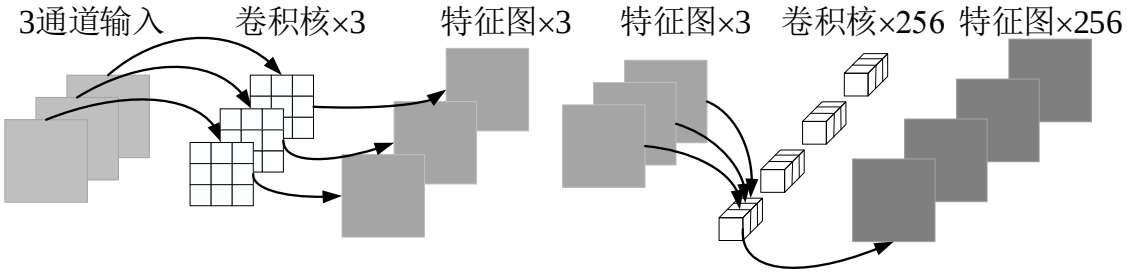
3.3.1 深度可分离卷积

卷积神经网络中，卷积起着至关重要的作用。卷积运算具有强大的特征提取能力，相比全连接层消耗更少的参数。常规卷积操作中，图像对应区域的所有通道均被同时考虑，但是图像的每个通道是不相关的，无形中增加了网络的冗余度，还会影响网络的分类效果。本文使用深度可分离卷积，将卷积分成两步进行，先在每个图像通道上进行卷积，再使用 1×1 跨通道卷积，将各个通道结合起来，生成最终的特征图。通过使用深度可分离卷积，减小图像通道之间的相关性，提高网络的分类效果。

假设输入层为一个 64×64×3 的图像，其中 64×64 为图像尺寸，3 为图像通道数，卷积后，输出层为 64×64×256 大小图片。常规卷积时，直接使用 256 个 3×3×3 的卷积核进行卷积，使用深度可分离卷积，分两步完成，先使用 3 个 3×3 卷积核进行分层卷积，再使用 256 个 1×1×3 卷积核进行卷积，在深度方向上进行加权组合。常规卷积与深度可分离卷积对比示意图如图 3.6 所示。



(a) 常规卷积



(b) 深度可分离卷积

图 3.6 常规卷积与深度可分离卷积示意图

进行分层卷积后，还需再进行 1×1 的跨通道卷积，是因为分层卷积完全是在二维平面内进行的，没有有效的利用不同通道图片在相同空间位置上的信息。1×1 维卷积类似于一个全连接结构，实现跨通道信息融合，并且 1×1 跨通道卷积还起到升维的作用。

3.3.2 网络结构设计

一般而言，增加卷积神经网络模型的复杂度，可以提升模型的识别效果。但是模型复杂度不能无限提升，模型越复杂，模型训练更加困难，训练时间代价更大，并且可能会出现不收敛和过拟合问题。在此基础上，本文设计了 20 个不同结构的卷积神经网络，来分析输入数据、卷积方式、模型复杂度对手写汉字识别的影响，避免了单个网络带来的偶然性。使用常规卷积时的网络结构如表 3.1 所示。

表 3.1 使用常规卷积时的卷积神经网络结构

<table>
<tr><td colspan="5">CNN 网络模型</td></tr>
<tr><td>Net1
（5 层）</td><td>Net2
（6 层）</td><td>Net3
（7 层）</td><td>Net4
（8 层）</td><td>Net5
（11 层）</td></tr>
<tr><td colspan="5">输入(64×64×1/64×64×8)</td></tr>
<tr><td>50C3</td><td>50C3</td><td>50C3</td><td>50C3</td><td>50C3
50C3</td></tr>
<tr><td colspan="5">MP2</td></tr>
<tr><td>100C3</td><td>100C3</td><td>100C3</td><td>100C3</td><td>100C3
100C3</td></tr>
<tr><td colspan="5">MP2</td></tr>
<tr><td>200C3</td><td>200C3</td><td>200C3</td><td>200C3
200C3</td><td>200C3
200C3</td></tr>
<tr><td colspan="5">MP2</td></tr>
<tr><td></td><td>400C3</td><td>400C3
400C3</td><td>400C3
400C3</td><td>400C3
400C3</td></tr>
<tr><td></td><td colspan="4">MP2</td></tr>
<tr><td colspan="4">1000FC</td><td>1000FC
1000FC</td></tr>
<tr><td colspan="5">3755FC</td></tr>
<tr><td colspan="5">Softmax</td></tr>
</table>

表 3.1 中展示的是使用常规卷积的网络结构，每一列表示一个网络，总共包括 5 个网络模型，记为 Net1~Net5。从 Net1 到 Net5，网络复杂度递增，分别包括 5、

6、7、8、11 层，其中层数只计算卷积层和全连接层数量。输入 64×64×1 表示原图输入，64×64×8 表示使用八方向梯度特征图像作为输入；xC3 表示该层为常规卷积层，有 x 个卷积核，卷积核尺寸为 3×3；MP2 表示使用 2×2 的最大值池化方式；xFC 表示该层为具有 x 个神经元的全连接层；Softmax 表示神经网络采用 Softmax 分类函数输出结果。以八方向梯度输入时，Net2 网络为例，其网络结构可表示为：64×64×8Input-50C3-MP2-100C3-MP2-200C3-MP2-400C3-MP2-1000FC-3755FC-Softmax-Output。

表 3.1 中，对应于每一个 Net，将常规卷积全部改为深度可分离卷积，得到 5 个新的模型，记为 Net1-DS~Net5-DS。使用 Conv_ds3-x 的形式表示使用深度可分离卷积。对于使用常规卷积的神经网络 Net1~Net5，使用深度可分离卷积的网络 Net1-DS~Net5-DS，输入都有两种不同方式，所以总共有 20 个具体的卷积神经网络。

卷积时，为了保持输入输出图片尺寸相同，卷积步长设置为 1，周围进行 0 填充。池化时步长设置为 2。以 Net2-DS 网络为例，使用八方向梯度特征图像作为输入，各层输入输出图片尺寸情况如图 3.7 所示。

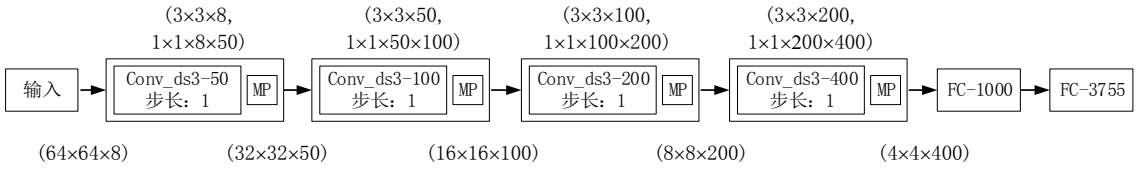


图 3.7 Net2-DS 网络各层输入输出情况

图 3.7 最下面一行数据表示各层输入数据，最上面一行数据表示卷积时所使用的卷积核情况，例如 3×3×8 表示分层卷积时使用的 8 个 3×3 卷积核，1×1×8×50 表示跨通道卷积时使用的 50 个 1×1×8 卷积核。

3.4 模型训练

3.4.1 实验环境

深度学习是数据驱动的，深度学习的训练，往往需要很长时间。最近几年，得益于图形处理器对深度学习的加速，以及众多深度学习框架的支持，搭建和训

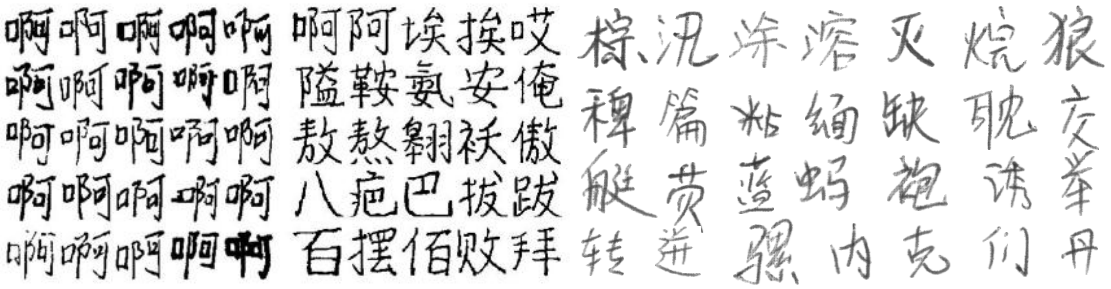
练卷积神经网络更加高效。本文使用 Keras 深度学习框架搭建卷积神经网络模型，使用 GPU 进行加速训练。具体实验环境如表 3.2 所示。

表 3.2 实验相关环境

操作系统	Ubuntu 18.04
CPU	Intel Core i7-8700K CPU @3.70GHzX12
GPU	Nvidia GTX1081Ti 11G
RAM	DDR4 3200 16G
深度学习框架	Keras2.1.0，后端 Tensorflow1.4.0
编程语言	Python3.6.3

3.4.2 CASIA-HWDB 数据集

深度学习是在数据驱动下发展的，没有数据，深度学习就没有落脚点。常用的手写汉字数据集有北京邮电大学模式识别实验室收集的 HCL2000 数据集[49]，以及中科院自动化研究所模式识别国家重点实验室收集的 CASIA-HWDB 数据集[50]。相对于 HCL2000 数据集，CASIA-HWDB 数据集中手写汉字更加随意、多变，识别难度更大，因此本文选择 CASIA-HWDB 数据集作为实验数据集。HCL2000 与 CASIA-HWDB 数据集样本图像如图 3.8 所示。



(a) HCL2000 数据集　　(b) CASIA-HWDB 数据集

图 3.8 HCL2000 与 CASIA-HWDB 数据集样本图像

CASIA-HWDB 数据集包括 CASIA-HWDB1.0-1.2 三个子集以及 ICDAR-2013 竞赛数据集，CASIA-HWDB1.0-1.2 数据集一般用于训练集、验证集，ICDAR-2013 数据集一般用于评判数据集。CASIA-HWDB 数据集统计情况如表 3.3 所示。

表 3.3 CASIA-HWDB 数据集统计数据

数据集	贡献者人数/个	样本数量/个	汉字类别/个
CASIA-HWDB1.0	420	1609136	3866
CASIA-HWDB1.1	300	1121749	3755
CASIA-HWDB1.2	300	990989	3319
ICDAR-2013	60	224419	3755

CASIA-HWDB1.1 数据集囊括了 GB2312-80 规定的所有 3755 个一级汉字，其由 300 人书写而成，每人书写对应的 3755 个汉字，总共包括 1121749 个样本，每个类别汉字约 300 个样本。CASIA-HWDB1.0 数据集包括 3866 个汉字，其中 3740 个属于一级汉字，其可以作为 CASIA-HWDB1.1 数据集的补充训练集。CASIA-HWDB1.2 包括 3319 个汉字，这 3319 个汉字和 CASIA-HWDB1.0 中的汉字类别是不相交的，CASIA-HWDB1.0 和 CASIA-HWDB1.2 结合后囊括了 GB2312-80 中规定的所有一级和二级汉字，可作为更大类别汉字分类的数据集。ICDAR-2013 数据集与 CASIA-HWDB1.1 数据集相对应，其包括 224419 个样本，每个类别汉字包括约 60 个样本。在 2011 年和 2013 年中科院自动化研究所模式识别国家重点实验室连续举办的两届 ICDAR 比赛中，均使用 ICDAR-2013 竞赛数据集作为评判数据集。

本文主要内容为对一级 3755 个汉字进行识别，选择 CASIA-HWDB1.1 数据集作为模型训练集，ICDAR-2013 竞赛数据集作为测试集。

3.4.3 模型训练

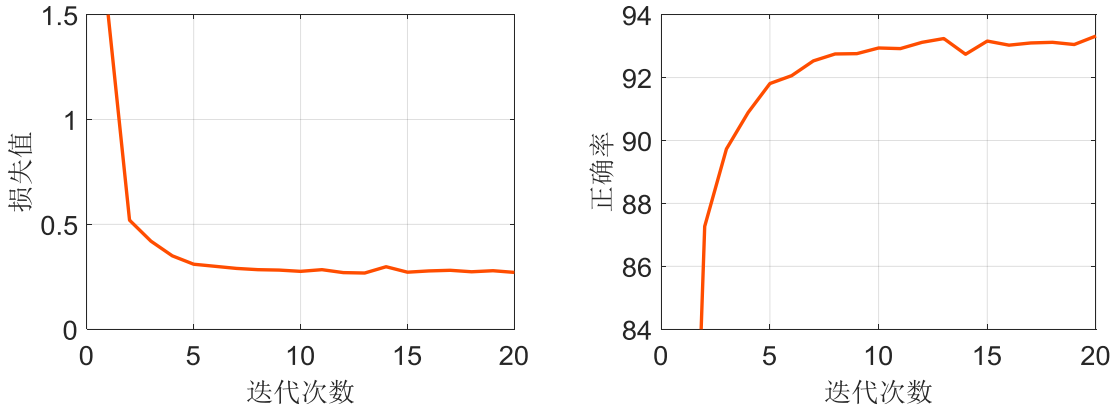
将 CASIA-HWDB1.1 数据集划分为训练集和验证集两个部分，训练集上每个汉字包括约 240 个样本，样本总数为 897758 个，验证集上每个汉字包括约 60 个样本，样本总数为 223991。

卷积神经网络的训练，涉及众多超参数的设置。模型超参数如表 3.4 所示。

表 3.4 卷积神经网络超参数设置

分类器	Softmax
损失函数	交叉熵损失函数
优化器	Adadelta
权重初始化	W：均值为 0，标准差：0.01 的高斯分布；b：0
激活函数	ReLU
Dropout	0.5
Batch Size	128

训练时，使用 Softmax 分类器作为卷积神经网络最终输出，使用交叉熵损失函数作为代价函数，采用 Adadelta 优化器更新权重，来最小化代价函数。模型卷积层和全连接层初始权重服从均值为 0，标准差为 0.01 的高斯分布，激活函数均为 ReLU 函数，在全连接层后设置了 Dropout 来减小模型过拟合，Dropout 设置为 0.5。在卷积层和全连接层使用 Batch Normalization 加快网络收敛，Batch Size 设置为 128，每次迭代结束时，在验证集上进行验证。图 3.9 为 Net2 网络模型在原图输入情况下，训练过程中验证集上损失值及正确率变化情况。Net2 网络每个周期训练时间约 12 分钟，总共迭代次数为 20 次，共计所需训练时间约 4 个小时。

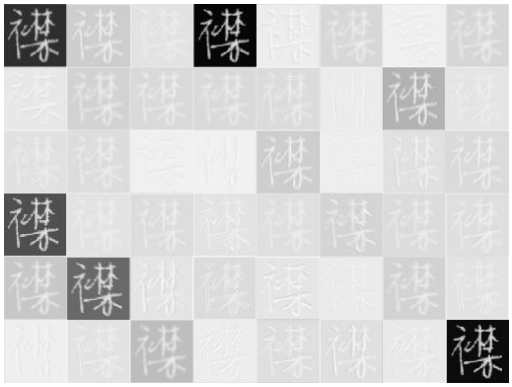


(a) 损失值　　(b) 正确率

图 3.9 Net2 卷积神经网络损失值及正确率变化曲线图

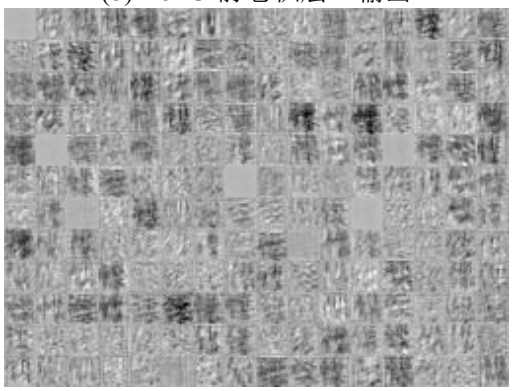
为了更好地理解卷积神经网络训练过程中到底在学习什么，对卷积神经网络卷积层每层输出情况进行可视化。因为卷积之后经过 ReLU 激活函数后，小于 0 的数据都变为 0，显示出来都是黑框，不利于观察，故主要选择 ReLU 前的数值进行可视化，并给出前两层卷积操作后经过 ReLU 激活函数之后的图作为对比。以手写汉字“襟”为例，使用原始图像作为输入时，Net2 网络每个卷积层输出的可视化情况如图 3.10 所示。

(a) 输入图像

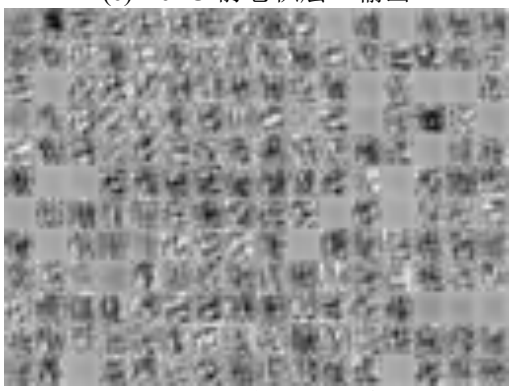


(b) ReLU 前卷积层 1 输出

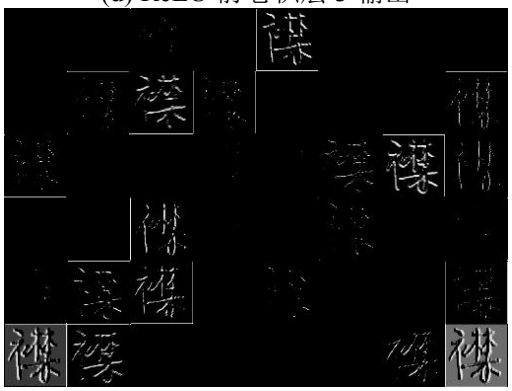
(c) ReLU 前卷积层 2 输出



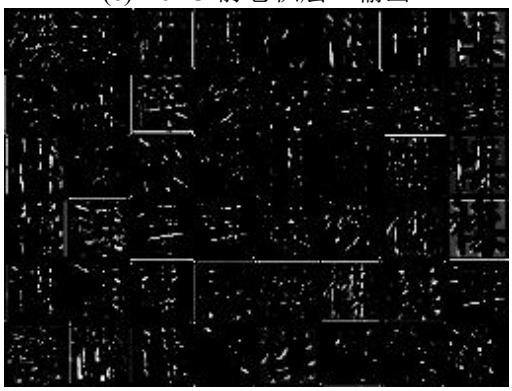
(d) ReLU 前卷积层 3 输出



(e) ReLU 前卷积层 4 输出



(f) ReLU 后卷积层 1 输出



(g) ReLU 后卷积层 2 输出

图 3.10 卷积层输出可视化

由于空间限制，卷积层 1 和卷积层 2 只可视化了 48 个卷积输出，卷积层 3 和卷积层 4 可视化了 192 个卷积输出。仔细观察，可以发现卷积层 1 和卷积层 2 的输出中还可以看到输入汉字“襟”的轮廓，卷积层 3 和卷积层 4 的可视化输出则是些朦胧抽象的图，无法直观理解。对比图(b)与(f)，图(c)与图(g)，经过激活函数 ReLU 后，大部分输出都变为 0，这些特征可以被认为不是很重要的特征，对于最终的分类不起太大帮助。

通过观察，可以猜测，低层卷积层具有类似边缘检测的功能，在这个阶段里，卷积基本保留了图像所有信息。随着层数的增加，卷积层输出的内容也越来越抽

象，保留的图像细节信息也越来越少，在学习图像更深层次的特征。这些信息是对低层视觉特征的综合，更加有利于最终的分类任务。

这类似于人类感知世界的方式，我们脑海中储存着各种物体的概念图，例如大象、飞机、企鹅等，当我们看见这些物体时，一眼就可以分辨出它属于哪个种类，但是当我们试图在脑海中画出这件物体时，却发现非常困难。因为我们没有根本就没有记住事物的具体外观，也没有必要全部记住。这是因为大脑已经学会了完全抽象物体的视觉输入，过滤掉不重要的视觉细节，把它转换为高层次的视觉概念。

3.5 实验结果与分析

一级 3755 类手写汉字实验结果如表 3.5、表 3.6 所示。

表 3.5 常规卷积时不同网络汉字识别准确率结果

网络模型	原图输入	八方向梯度特征图像输入
Net1	90.12%	91.28%
Net2	92.05%	93.17%
Net3	93.38%	94.14%
Net4	93.77%	94.66%
Net5	94.16%	94.95%

表 3.6 深度可分离卷积时不同网络汉字识别准确率结果

网络模型	原图输入	八方向梯度特征图像输入
Net1-DS	91.01%	92.55%
Net2-DS	92.35%	93.74%
Net3-DS	93.42%	94.61%
Net4-DS	94.27%	95.50%
Net5-DS	94.53%	95.86%

3.5.1 模型复杂度对汉字识别性能的影响

模型复杂度对模型识别性能具有重要的影响，一般而言，深层神经网络分类效果要优于浅层神经网络。从最早的 7 层 LeNet-5 网络结构，到后来的 16 层 VGGNet，22 层 GoogLeNet 网络，模型规模整体趋向于复杂化。

如表 3.5 所示，无论输入是原图还是八方向梯度特征图像，随着模型深度、宽度的增加，模型的分类效果均相应提升。当输入为八方向梯度特征图像时，Net5

取得了最好的识别效果 94.95%，比较 5 个卷积神经网络模型识别结果，Net2 相对于 Net1 准确率提升 1.89%，Net3 相对于 Net2 提升 0.97%，Net4 相对于 Net3 提升 0.52%，Net5 相对于 Net4 提升 0.29%。发现，模型复杂度的增加带来的准确率增益却逐渐减小，趋于饱和，与此同时，复杂的卷积神经网络模型带来的训练代价却是巨大的。表 3.6 中深度可分离卷积时的结果具有类似的规律。实验发现，常规卷积与深度可分离卷积单个周期训练时间几乎相同，八方向梯度特征输入训练时间略高于原图输入。表 3.7 为采用八方向梯度特征输入时，Net1-Net5 卷积神经网络训练时迭代周期，及训练时间统计数据。

表 3.7 网络训练迭代周期及训练时间统计

	Net1	Net2	Net3	Net4	Net5
单个周期训练时间/分钟	17	18	19	20	28
迭代周期/次	20	20	20	20	40
总时间/小时	5.66	6.0	6.33	6.66	18.66

从表 3.7 可以看出，Net1-Net4 四个模型单个周期训练时间差异不大，并且迭代周期为 20 次时就已收敛。而分类效果最好的 Net5 模型无论单个周期训练时间还是迭代周期均远大于前四个模型，其训练时间是 Net1 的 3.3 倍，主要原因为全连接层相对于卷积层更难训练。如果在 Net5 的基础上继续增加模型的复杂度，训练将更加困难。因此，不能简单的堆砌卷积神经网络，需要平衡模型训练时间代价与汉字识别准确率之间的关系。

3.5.2 八方向梯度特征对汉字识别性能的影响

梯度特征在传统手写汉字识别中应用非常广泛。一般来说，提取的特征越优良，输入分类器后的识别效果越好。当卷积神经网络接受原图作为输入时，可以自己从图片中提取特征，但是却无法学习和利用事先提取好的特征。本节分析对比了八方向梯度特征对汉字识别性能的影响。

为了方便讨论，将表 3.5 和表 3.6 识别结果展示在图 3.11 中。图 3.11 为不同输入时手写汉字识别准确率对比结果，左图为采用常规卷积的神经网络，右图是采用深度可分离卷积的网络。可以看出，无论采用何种卷积方式，使用八方向梯度

作为卷积神经网络输入，相对于原图输入都能够取得更好的识别效果。常规卷积时平均准确率提升 0.94%，深度可分离卷积时平均准确率提升 1.33%。

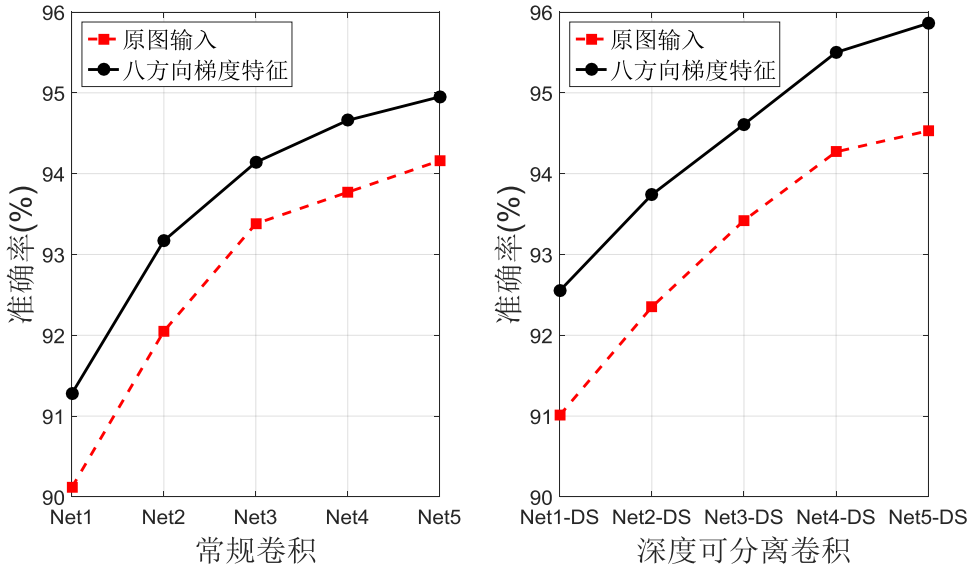


图 3.11 不同输入时手写汉字识别准确率对比图

由此可以看出，使用卷积神经网络对手写汉字进行识别时，虽然可以像其他模式识别任务一样直接使用原图作为输入，方便快捷，但是却存在相应弊端。使用汉字图像整体作为输入，比较笼统，无法学习到有效的先验的领域知识。提取领域内良好的八方向梯度特征信息，结合卷积神经网络能够显著提升汉字识别准确率。

3.5.3 深度可分离卷积对汉字识别性能的影响

不同卷积方式时手写汉字识别准确率对比结果如图 3.12 所示，其中左图为原图输入，右图为使用八方向梯度特征输入。两种输入情况下，对于不同复杂度的卷积神经网络，使用深度可分离卷积时的网络识别准确率，都高于使用常规卷积时的准确率。原图输入时，深度可分离卷积相比常规卷积准确率平均提升 0.42%，八方向梯度输入时准确率平均提升 0.81%。

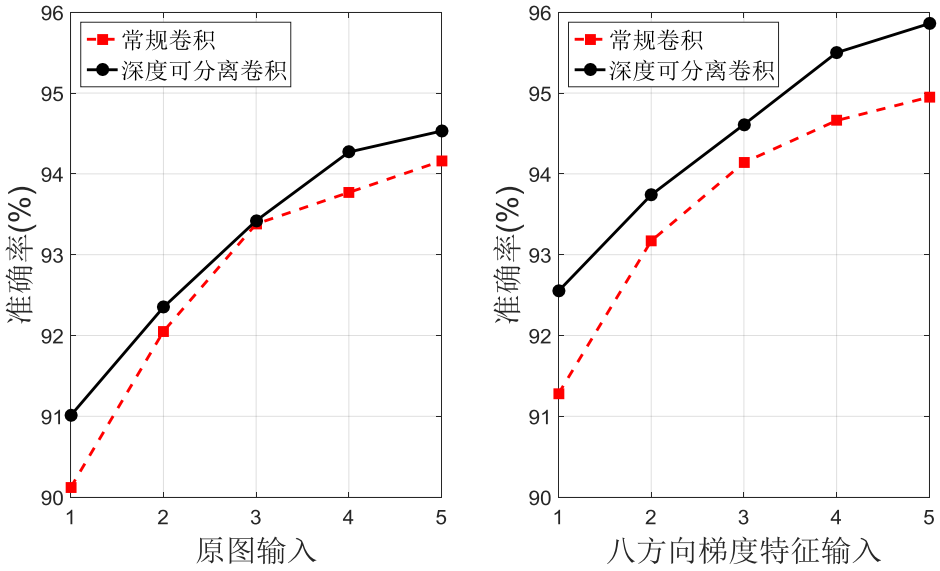


图 3.12 不同卷积方式时手写汉字识别准确率对比图

由此可以看出，卷积时，减小图像通道之间的相关性，有利于取得更好的识别效果。伴随着减小网络冗余度的同时，深度可分离卷积还可以减少卷积层的参数数量，减小模型存储量。

3.5.4 与其他相关算法对比分析

当使用八方向梯度特征图像作为输入，采用深度可分离卷积时，Net5-DS 网络取得了本文最好的手写汉字识别准确率 95.86%。将其与其他主流方法在 ICDAR-2013 竞赛数据集上的识别效果进行对比，结果如表 3.8 所示。

表 3.8 不同算法在 ICDAR-2013 数据集上识别效果

方法	准确率	训练数据集	模型存储量
MQDF-HIT[25]	92.61%	HWDB1.0+1.1	120MB
MQDF-THU[25]	92.56%	HWDB1.0+1.1+2.0+2.1+2.2	198MB
CNN-Fujitsu[25]	94.77%	HWDB1.1	2460MB
ATR-CNN[51]	95.04%	HWDB1.1	51.64MB
HCCR-Ensemble-GoogLeNet (average of 10 models)[52]	96.74%	HWDB1.0+1.1	270MB
人眼[25]	96.13%	—	—
本文	95.86%	HWDB1.1	53.6MB

表 3.8 中展示了几种典型方法在 ICDAR-2013 数据集上的识别效果，其中人眼一行表示的是人类在该数据集上的识别效果。MQDF-HIT 和 MQDF-THU 采用的是传统汉字识别策略，CNN-Fujitsu、ATR-CNN、HCCR-Ensemble-GoogLeNet 均基

于卷积神经网络。MQDF-HIT 使用双向矩归一化方法归一化灰度字符图像，然后提取 512 维特征向量，再使用 LDA 降维为 160 维，使用感知器学习算法训练的 MQDF 模型进行分类。MQDF-THU 从灰度图像中提取 588 维梯度特征，使用 HLDA 将其降维到 200 维，使用级联的 MQDF 分类器进行分类。CNN-Fujitsu 为 ICDAR-2013 汉字识别竞赛的冠军模型，其使用 4 个 CNN 模型进行投票产生最终的输出结果。ATR-CNN 改变传统卷积中一个特征图内共享卷积核的策略，使用松弛卷积神经网络识别手写汉字。HCCR-Ensemble-GoogLeNet 对 GoogLeNet 进行改进，其使用 10 个改进后的 GoogLeNet 进行投票产生最终的输出结果。

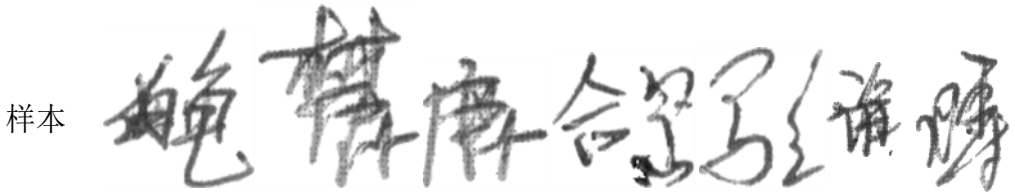
从表 3.8 可以看出，本文的方法识别准确率大幅领先于传统的以 MQDF 为代表的汉字识别方法，相较于其他卷积神经网络方法 CNN-Fujitsu，ATR-CNN，识别准确率也有小幅提升，仅次于 HCCR-Ensemble-GoogLeNet 方法，主要是因为它使用了 10 个卷积神经网络模型求取平均，并且在 HWDB1.1 数据集的基础上使用了扩增数据集 HWDB1.0 进行训练。对 HCCR-Ensemble-GoogLeNet 进行实验，单个模型训练时间约 21 个小时，主要是因为训练数据量过于巨大。除此之外，本文训练得到的模型存储量大小只有 53.6M，除了 ATR-CNN，远小于其他算法，更加适合移植到移动端，方便工程应用。因为时间因素，本文只使用了 HWDB1.1 数据集作为训练集，如果使用更多的数据集 HWDB1.0 进行训练，预计能获得更好的识别效果。

3.6 错误结果分析

以八方向梯度特征输入时，Net5-DS 网络分类结果为例，统计一级 3755 类汉字错误识别结果。Net5-DS 网络在 ICDAR-2013 数据集上取得了 95.86%的正确率，共计有 9291 个错误分类样本，平均每个汉字有 2.4 个样本被错误分类。ICDAR-2013 数据集上，每个汉字约有 60 个样本，统计每个汉字错误样本个数，其区间在[0,26]之间。其中 230 个汉字全部被分类正确，529 个汉字仅有 1 个样本分类错误，最差的为汉字“已”，有 26 个样本分类错误。

首先从数据集本身入手，分析汉字错误识别原因。数据集采集的是几百人的手写样本，采集过程中难免会出现汉字涂改、汉字严重变形、甚至书写错误等情况，导致该样本无法被识别，我们称这些数据为“脏数据”。另外汉字采集后需

要人为给其打上种类标签，这一过程中会出现标签错误情况，导致样本真实值与标签值不一致，我们称之为误标注。ICDAR-2013 数据集中部分“脏数据”及误标注样本如图 3.13 所示。



(a) 一些“脏数据”

(b) 一些误标注的样本

图 3.13 部分“脏数据”及误标注样本

这些少量的“脏数据”及误标注样本，会在一定程度上影响汉字识别结果，导致某些汉字样本识别错误。

除此之外，对识别错误次数较多的汉字进行分析，这些汉字的错误总数占总错误数的绝大部分。统计每个汉字具体错误情况，观测其被误识别为哪些汉字。部分错误识别次数较多的汉字统计情况如表 3.9 所示：

表 3.9 部分错误识别次数较多的汉字具体错误情况

汉字类别	错误情况						错误总数/个
已	己(15)	巳(8)	乙(1)	卫(1)	山(1)	其他(0)	26
淮	谁(8)	准(4)	渡(2)	濉(1)	推(1)	其他(2)	18
乎	手(7)	平(5)	伞(3)	于(1)	呼(1)	其他(0)	17
汪	证(4)	注(3)	汇(2)	迁(2)	江(1)	其他(5)	17
曰	日(8)	目(4)	回(2)	四(1)	田(1)	其他(0)	16
困	团(5)	囤(3)	田(3)	因(3)	园(1)	其他(1)	16

表 3.9 最左侧一栏表示汉字类别，中间 6 列表示汉字被误识别为哪些其他汉字，括号中数字表示误识别次数，最后一列为汉字错误识别总数。以汉字“己”为例，汉字“己”共识别错了 26 次，其中 15 次被误识别为“己”，8 次被误识别为“巳”，

“乙”、“卫”、“山”各一次。从表 3.9 可以看出，被误识别成的汉字都和原来的汉字比较相似，这些字有些差别只是一个笔画的长短，例如“已、己、巳”等，有些差别只是偏旁的不同，例如“淮、谁、准”等。图 3.14 为 ICDAR-2013 数据集上部分形近字示例。

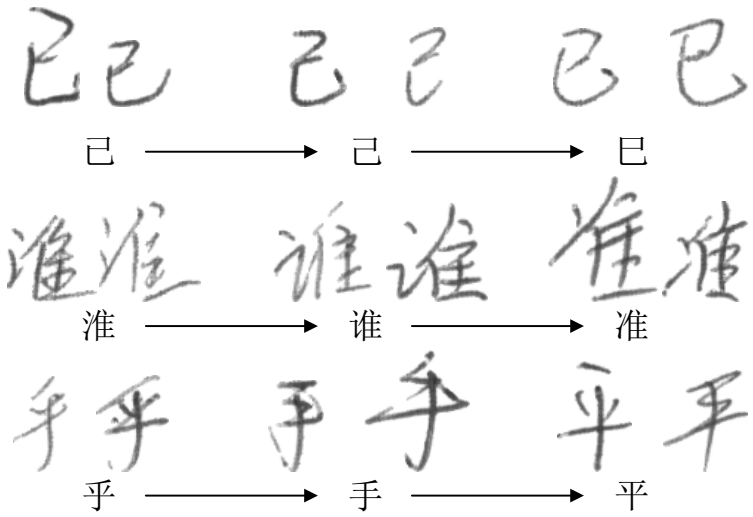


图 3.14 ICDAR-2013 数据集上部分形近字示例

通过以上分析，可以看出，一级 3755 类汉字识别中，除去脏数据”及误标注样本的干扰，错误主要来源于形近字的影响。大多数样本分类错误，是形近字的干扰造成的，这也符合人们的直观感受。

3.7 本章小结

本章研究了一级 3755 类手写汉字识别方法。首先介绍了手写汉字识别基本流程及相关预处理方法，之后从卷积神经网络输入、卷积方式两个方面对卷积神经网络进行改进，设计多组卷积神经网络结构对手写汉字进行识别，分析对比了模型结构复杂度、八方向梯度特征，不同的卷积方式对汉字识别的影响，并与其他相关算法进行对比。实验结果表明，八方向梯度特征和深度可分离卷积能够显著提升手写汉字识别效果，最优的模型准确率达到了 95.86%。最后对错误识别结果进行分析，发现相似字是导致识别错误最主要的原因。

第 4 章 相似手写汉字识别

第 3 章中对一级 3755 个汉字识别结果表明，汉字的识别误差主要来源于相似汉字的干扰。要进一步提高手写汉字识别效果，最有效的途径就是提高相似手写汉字的识别效果。基于此，本章在第 3 章的基础上，研究相似手写汉字识别方法。

卷积神经网络是一个端到端的结构，集特征提取与分类的功能为一体。在使用 Softmax 分类器时，样本的最终分类完全取决于最后一个全连接层的最大值，其他输出值没有起到任何作用，过于极端。Softmax 分类器只追求将不同类样本分开，但不需要分离很多，在数值上的表现就是最大值比次大值大即可，而不需要大很多。当汉字比较相似时，最后一层输出的最大值和次大值比较接近，差值不大，导致汉字分类结果容易波动，出现误分类。再加上模型泛化时不可避免存在准确率下降的现象，导致形近字更加难以区分。

以一组形近字“已己”为例，假设有 70%的“已”、“己”样本正确分类，30%的“已”、“己”被误分类为对方。全连接层输出结果中只保留最大值及次大值，绘制其二维映射图像，如图 4.1 所示。靠近直线 y=x 两边的点为误分类的点，相比于正确分类的点，它们最大值和次大值比较接近，聚集在 Y=X 附近。因为大多数样本毕竟还是更像同类样本，即使被误分类为相似样本，误分类样本对应的最大值也不会比本类样本对应的次大值大很多。

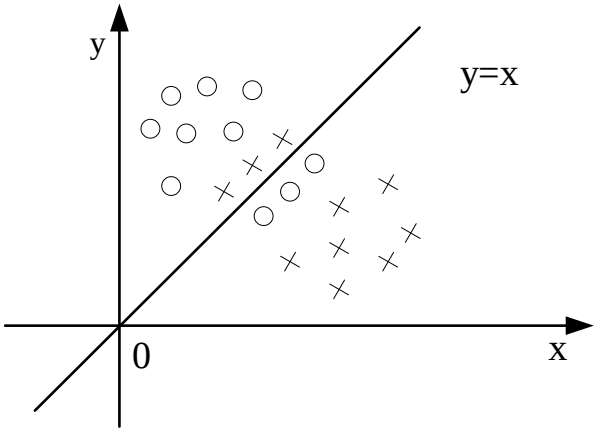


图 4.1 全连接层输出二维平面映射图

针对 CNN 中 Softmax 分类器在相似手写汉字识别上的不佳表现，从两个角度进行改进。方法一，引入 Center Loss 函数，将 Center Loss 和 Softmax 损失结合起来，作为卷积神经网络的联合损失。Softmax 分类器用于将不同类汉字分开；Center Loss 函数为每一类汉字维持一个特征中心，使提取到的该类样本特征都尽量靠近这个中心，减小同类样本之间距离，使不同类样本之间的特征具有明显的区别。方法二，剥离 CNN 的 Softmax 层，将 CNN 当作特征提取器，使用 SVM 分类器将 CNN 全连接层输出的特征数据转换到更高维的空间进行分类。

4.1 相似手写汉字选择

选择相似手写汉字是实验的基础，相似手写汉字的评判没有一个统一的标准，选择具有代表性的相似汉字对实验有重要意义。汉字种类繁多，不同汉字之间相似方式多种多样，拆开来，最基本的大约可以分为六类[53]，如表 4.1 所示。

表 4.1 汉字相似方式

序号	相似方式	相似汉字举例
1	笔划的有无	“汪注”、“乌鸟”、“拆折”
2	笔划的位置不同	“天夫”、“大丈”
3	笔划的长短不一	“日曰”、“土士”
4	存在某些笔划相似	“策笨”、“干于”
5	部首不同而字根相同	“决诀”、“准淮谁”
6	部首相同而字根不同	“迭选”、“治洽”

此外，还有许多汉字具有多种以上的相似结构。到底选择哪些汉字才最具有代表性，往往莫衷一是。

常见的相似汉字选择方法有两种，人为经验性地预定义相似汉字，或者通过探测汉字之间的距离选择相似汉字。然而，两种方法均存在不足之处，前者过于主观，后者过于客观，两种方法均没有与实际相联系，导致设计的相似汉字与实际中易于混淆的汉字存在一定偏差。归根到底，只有分类器才最有权力确定哪些汉字相似，因为最终的分类正是通过分类器进行的。本章以一级 3755 个汉字分类结果为导向，以第 3 章中 Net5-DS 网络错误分类结果为依据，设计了 15 组相似汉字，每组包括 10 个汉字。设计方法为：

1. 一级 3755 个汉字分类结果中，按照每个汉字识别错误次数从高到低进行排

列，得到识别错误次数最多的前 15 个汉字。把这 15 个汉字当作基准汉字，把每个基准汉字及其错误分类成的汉字归为一组相似手写汉字，得到 15 组相似手写汉字。为方便统一，将每组汉字数量固定为 10 个；

2. 前 15 个基准汉字中，如果后面某个基准汉字在前面其他基准汉字的错误分类结果中，则删除该基准汉字，依次向下顺延。例如“已”、“己”同为基准汉字，前面一组汉字中，基准汉字“已”被错误分类为“己”、“巳”…而后面又出现基准汉字“己”又被错误分类为“已”、“巳”…则删除“己”该组汉字，避免重复；

3. 对于分类错误类别超过 10 种的情况，删除错误次数最少的汉字。例如“汪”被误分类为“证”、“江”、“汗”、“注”、“汇”、“迁”、“迂”、“泛”、“记”，“风”，根据误分类次数，剔除错误分类次数最少的“风”；

4. 对于分类错误类别少于 10 种的情况，人为选择相似汉字进行补充。例如“请”被误分类为“清”、“墙”、“谓”、“倩”、“淆”、“情”、“晴”、“靖”，使用“睛”进行补充；

5. 对于多组之间出现重复分类错误的汉字情况，将错误次数较少的汉字从对应组移除。例如“曰”和“困”都被误识别为“田”，“曰”被误识别为“田”一次，“困”被误识别为“田”三次，故删除“曰”相似汉字组中的“田”。

最终设计的 15 组相似汉字如表 4.2 所示。

表 4.2 15 相似手写汉字

组号	相似手写汉字									
1	已	己	巳	乞	乙	忆	包	巴	记	厄
2	柬	束	来	更	吏	秉	枣	曳	果	隶
3	请	清	墙	谓	倩	淆	情	晴	靖	睛
4	鸣	呜	吗	鸟	乌	鸡	鸭	鸦	鸽	坞
5	淮	谁	渡	准	潍	唯	推	堆	难	惟
6	乎	手	伞	平	于	呼	苹	米	干	采
7	迭	选	送	返	版	造	迷	达	迸	远
8	汪	证	江	汗	注	汇	迁	迂	泛	记
9	昧	眯	味	肘	胖	胀	抹	沫	伴	拌
10	曰	目	日	四	回	同	月	白	口	旦
11	沮	诅	沿	泪	泌	泅	狙	油	迪	抽
12	困	囤	田	因	团	园	毋	囚	国	母
13	冶	治	沽	淌	活	沉	恰	洽	话	括
14	唁	哈	哼	嘻	倍	啥	哇	信	响	咱
15	扶	快	抉	挟	抹	决	诀	块	秧	殃

4.2 数据集扩增

卷积神经网络的性能很大程度上依赖于数据的多少，数据量越丰富，越有益于最终的分类结果。数据量过少，很容易出现过拟合问题。本章相似手写汉字识别任务中，每个相似手写汉字组包括 10 个种类汉字，总共包括约 2400 个训练样本，数据集过少，使用数据扩增技术增加训练样本数量。常见的数据扩增技术有弹性变换、仿射变换等，本文使用仿射变化对训练集进行扩充。

仿射变换是一种从二维坐标(x,y)到二维坐标(u,v)的线性变换，它可以保持原来的线共点、点共线的关系不变，保持原来相互平行的线仍然平行，保持原来在一条直线上的几段线段之间的比例关系不变。基本的仿射变换操作有平移、旋转、缩放等，其操作用矩阵进行表示为：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{4.1}$$

其中a、b、c为相关变换参数。本文中对相似汉字进行小幅度的旋转、平移、缩放操作，旋转范围在$\frac{\pi}{12}$之内，平移比例在 5%以内，缩放比例在 10%以内。仿射变换不仅可以扩充数据集，还可以模拟实际环境中的汉字倾斜，位置偏移，汉字大小不一问题。仿射变换效果如图 4.2 所示。每个样本进行 6 种仿射变换，仿射变换后，训练样本总数由 2400 扩增为 16800。

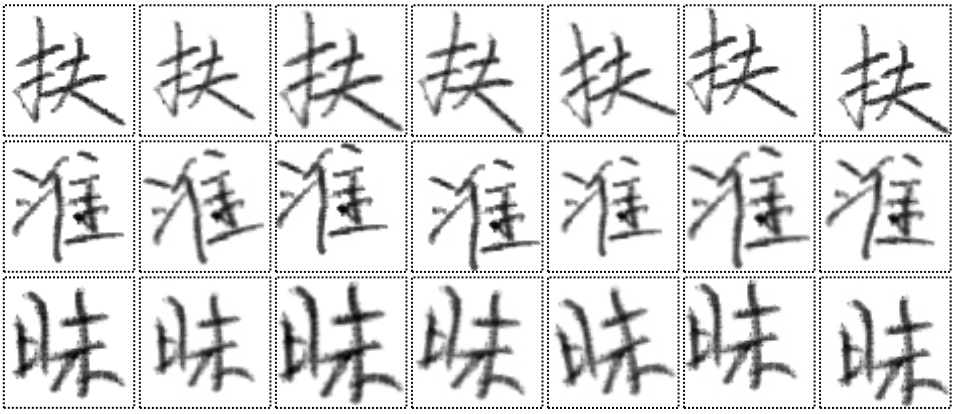


图 4.2 仿射变换效果图

4.3 基于 CNN 和 Center Loss 的相似手写汉字识别

4.3.1 Center Loss

Softmax 分类器只追求将不同种类样本分开，而不要求分开很多，导致形近字难以识别。为了使卷积神经网络能够学习到更加具有鉴别能力的特征，将相似手写汉字分离开，在传统 Softmax 损失的基础上，引入度量学习中的 Center Loss 函数到卷积神经网络。

Center Loss，即中心损失，最早由 Wen 等人提出，应用在人脸识别中[54]。Center Loss 的思想是减小同类样本特征之间的距离，使得不同类样本之间的特征具有明显的区别。在 Center Loss 中，每个类别样本都维护一个类中心，Center Loss 即为每类样本与其对应类中心之间的距离，其函数表达式为：

$$L_C = \frac{1}{2}\sum_{i=1}^{m}\left\|\boldsymbol{x}_i - \boldsymbol{c}_{y_i}\right\|_2^2 \tag{4.2}$$

其中，$\boldsymbol{x}_i$ 表示第 i 个样本对应的全连接层输出特征，y_i 表示第 i 个样本对应的类别，$\boldsymbol{c}_{y_i}$ 表示第 y_i 个种类样本所对应的特征中心，维度和 $\boldsymbol{x}_i$ 一致，m 表示一个 batch 样本的数量。

联合损失示意图如图 4.3 所示。在最后一个全连接层 FC2 层后得到的特征 x_i，一路送入 Softmax 分类器，用于计算 Softmax 损失，另一路送入 Center Loss 函数，用于计算 Center Loss。计算中心损失时，样本特征 $\boldsymbol{x}_i$ 与特征中心 $\boldsymbol{c}_{y_i}$，既可以来自于 FC1 层的输出，也可以来自于 FC2 的输出，本文中来自于 FC2 层。

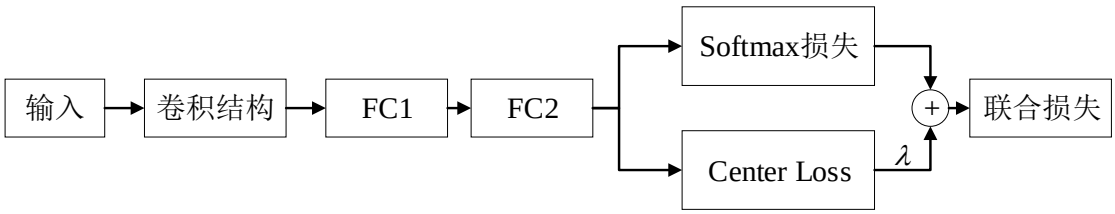


图 4.3 联合损失示意图

加入 Center Loss 后，联合损失包括 Softmax 损失和 Center Loss 两部分，函数表达式为：

$$\begin{aligned} L &= L_S + \lambda L_C \\ &= -\sum_{i=1}^{m} \log \frac{e^{x_i(y_i)}}{\sum_{j=1}^{n} e^{x_i(y_j)}} + \frac{\lambda}{2} \sum_{i=1}^{m} \left\| \boldsymbol{x}_i - \boldsymbol{c}_{y_i} \right\|_2^2 \end{aligned} \tag{4.3}$$

其中，L_S 表示 Softmax 损失，L_C 表示 Center Loss，n 表示 $\boldsymbol{x}_i$ 的维度，$\boldsymbol{x}_i\left(y_i\right)$ 表示特征 $\boldsymbol{x}_i$ 在对应类别 y_i 处的值。参数 λ 用于平衡两个损失函数，λ 过小，中心损失不起作用，λ 过大，模型可能会不收敛。Softmax 损失函数可以看作为联合损失函数的特例，当 λ 为 0 时，联合损失函数就为 Softmax 损失函数。

使用联合损失函数时，同样可以基于梯度下降的策略更新网络中参数。网络中参数可以分为两个部分，每个类的类中心 $\boldsymbol{c}_j$ 和卷积神经网络中原有的参数 W。$\boldsymbol{c}_j$ 只与中心损失函数有关，在每次迭代时，每个类别的特征中心根据该类样本特征 $\boldsymbol{x}_i$ 的平均值变化，并且使用参数 α 控制特征中心 $\boldsymbol{c}_j$ 变化的快慢，α 的作用类似于卷积神经网络中的学习率。特征中心 $\boldsymbol{c}_j$ 的更新方式如式(4.4)、式(4.5)所示：

$$\Delta \boldsymbol{c}_j = \frac{\sum_{i=1}^{m} \delta\left(y_i = j\right) \cdot \left(\boldsymbol{c}_j - \boldsymbol{x}_i\right)}{1 + \sum_{i=1}^{m} \delta\left(y_i = j\right)} \tag{4.4}$$

$$\boldsymbol{c}_j = \boldsymbol{c}_j - \alpha \cdot \Delta \boldsymbol{c}_j \tag{4.5}$$

式中，j 为样本对应的种类，$y_i = j$ 时，$\delta\left(y_i = j\right)$=1，$y_i \neq j$ 时，$\delta\left(y_i = j\right)$=0。

参数 W 的更新受到 Softmax 损失函数和 Center Loss 函数的共同影响，联合损失对参数 W 的偏导数为：

$$\frac{\partial L}{\partial W} = \sum_{i=1}^{m} \frac{\partial L}{\partial \boldsymbol{x}_i} \frac{\partial \boldsymbol{x}_i}{\partial W} \tag{4.6}$$

联合损失对特征 $\boldsymbol{x}_i$ 的偏导为：

$$\frac{\partial L}{\partial \boldsymbol{x}_i} = \frac{\partial L_S}{\partial \boldsymbol{x}_i} + \lambda \frac{\partial L_C}{\partial \boldsymbol{x}_i} \tag{4.7}$$

式(4.6)中 $\frac{\partial \boldsymbol{x}_i}{\partial W}$ 的计算、(4.7)中 $\frac{\partial L_S}{\partial \boldsymbol{x}_i}$ 的计算与传统卷积神经网络中一致，而，

$$\frac{\partial L_C}{\partial \boldsymbol{x}_i} = \boldsymbol{x}_i - \boldsymbol{c}_{y_i} \tag{4.8}$$

$\frac{\partial L}{\partial W}$ 即可由式(4.6)-(4.8)求出。

归纳而言，卷积神经网络中加入 Center Loss 函数后，整个网络的训练算法步骤如表 4.3 所示。

表 4.3 加入 Center Loss 后的 CNN 训练算法

加入 Center Loss 后的 CNN 训练算法：
1. 初始化卷积神经网络参数 W，类中心 $\boldsymbol{c}_j$，超参数 λ、α；
2. 迭代次数是否到达最大迭代次数，是则结束，否则执行步骤 3；
3. 计算联合损失 L，$L^t = L_S + L_C$；
4. 计算反向传播误差 $\frac{\partial L}{\partial \boldsymbol{x}_i}$，$\frac{\partial L}{\partial \boldsymbol{x}_i} = \frac{\partial L_S}{\partial \boldsymbol{x}_i} + \lambda \frac{\partial L_C}{\partial \boldsymbol{x}_i}$；
5. 更新类中心 $\boldsymbol{c}_j$，$\boldsymbol{c}_j = \boldsymbol{c}_j - \alpha \Delta \boldsymbol{c}_j$；
6. 更新参数 W，$\frac{\partial L}{\partial W} = \sum_{i=1}^{m} \frac{\partial L}{\partial \boldsymbol{x}_i} \frac{\partial \boldsymbol{x}_i}{\partial W}$；
7. 返回步骤 2。

4.3.2 基于 CNN 和 Center Loss 的相似手写汉字识别

本章用于相似手写汉字识别的基础 CNN 模型结构为：64×64×8Input-50C3-MP2-100C3-MP2-200C3-MP2-400C3-MP2-200FC-10FC-Softmax-Output，记为 Net-S。因为每组相似汉字只有 10 个汉字，训练样本总数不是很大，故这里采用的卷积神经网络不是很复杂，总体和第 3 章中 Net2-DS 网络结构类似。区别在于全连接层 FC1 神经元个数由 1000 个降低为 200 个，FC2 神经元个数由 3755 个改为 10 个。根据第 3 章实验结果，Net-S 网络依然采用八方向梯度特征图像作为输入，使用深度可分离的卷积方式。

基于 CNN 和 Center Loss 的相似手写汉字识别网络结构如图 4.4 所示，卷积神经网络部分和 Net-S 相同，第二个全连接层 FC-10 输出的特征用于计算 Softmax 损失和 Center Loss。

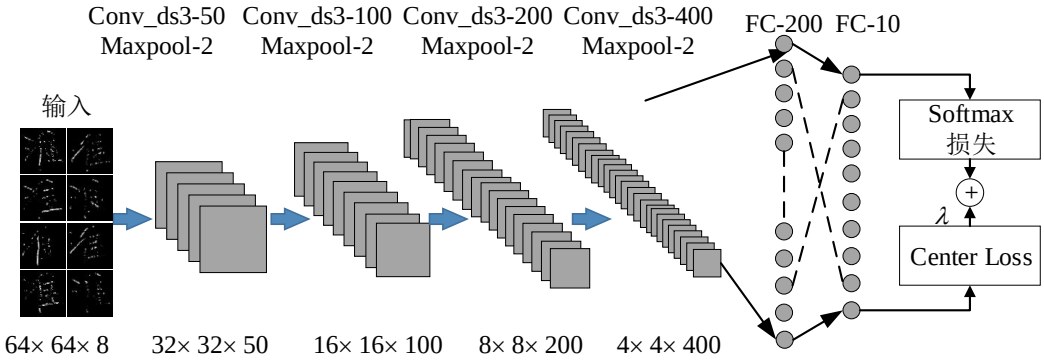


图 4.4 CNN 与 Center Loss 结合后的网络模型结构图

实验环境，优化器、权重W、激活函数、Dropout 等超参数设置均与第 3 章相同，batch size 设置有所不同，由 128 调整为 64。数据集分为训练集、验证集与测试集三个部分，训练集与验证集分别为 CASIA-HWDB1.1 数据集中训练集部分、验证集部分对应的相似手写汉字样本，测试集为 ICDAR-2013 竞赛数据集中对应的相似手写汉字样本。训练集上使用仿射变换进行数据集扩充后，每组相似汉字约包括 16800 个训练样本，600 个验证样本，600 个测试样本，对应每个种类的汉字数量分别为 1680、60、60 个。

模型训练时，每个类的中心$\boldsymbol{c}_j$初始化为 0。Center Loss 函数的性能受到超参数λ和α的影响。λ控制 Center Loss 对特征$\boldsymbol{x}_i$的影响程度，α控制特征中心$\boldsymbol{c}_j$的变化快慢。分析不同λ、α下，相似手写汉字识别准确率情况。文献[54]中的研究表名，α为 0.5 时，能够取得较高的人脸识别准确率。因此，本文首先固定α为 0.5，让λ在区间[0.0001,0.1]取值，获得测试集上每组相似手写汉字识别准确率，求取其平均值。不同λ下，15 组相似手写汉字识别平均正确率如图 4.5 所示。

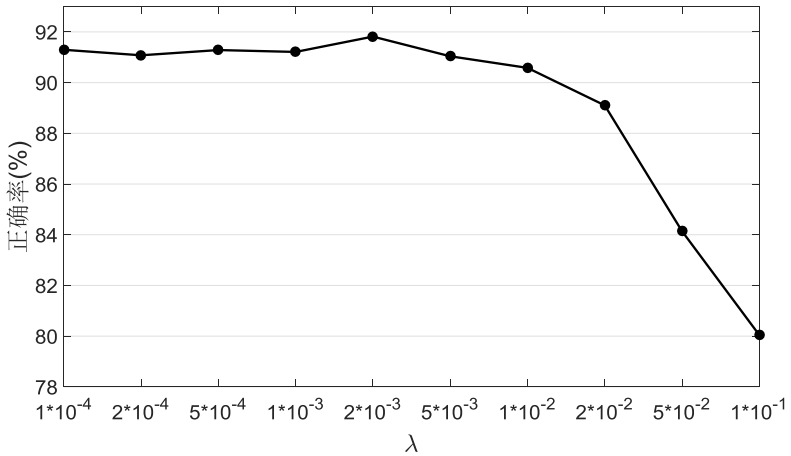


图 4.5 $\alpha=0.5$时，不同λ下 15 组相似手写汉字识别平均正确率

从图 4.5 中可以看出，λ取值比较小时，网络取得较高的识别准确率，并且波动幅度不大，整体准确率在 91%左右。λ取值较大，特别当$\lambda > 0.01$时，模型识别准确率急剧下降，性能比较恶劣，不再收敛。当λ为 0.002 时，模型取得最高的识别准确率。

然后固定$\lambda = 0.002$，α在区间[0.01,1]之间取值。不同α下，15 组相似手写汉字识别平均正确率如图 4.6 所示。

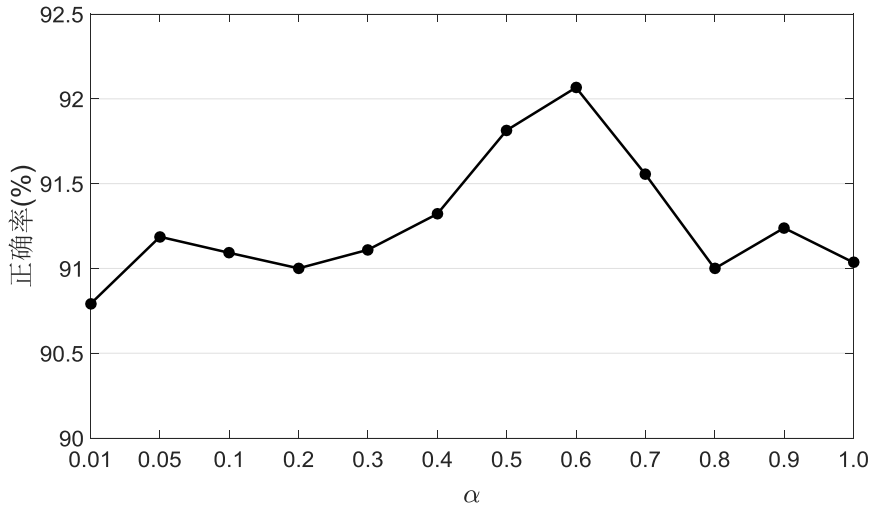


图 4.6 $\lambda = 0.002$时，不同α下 15 组相似手写汉字识别平均正确率

从图 4.6 中可以看出，模型识别准确率对α不是非常敏感，在一个比较广的范围内，α取值不同时，模型识别准确率比较稳定。最终当$\lambda = 0.002$，$\alpha = 0.6$时，模型取得了最好的平均识别准确率 92.07%。

4.4 结合 CNN 与 SVM 的相似手写汉字识别

CNN 中常用的分类器为 Softmax 分类器，它的输出含义明显，为对应类别的概率。但是 Softmax 分类的结果只取决于输入特征的最大值，没有有效利用其他特征。例如输入特征为一个n维向量$\boldsymbol{a} = (a_1, a_2, \cdots a_n)$，向量$\boldsymbol{a}$的最大值元素$a_i$完全决定分类结果为第$i$个输出对应的类别。当存在相似汉字的影响时，假设第$i$、$j$个输出对应的汉字很相似，样本正确标签为$i$。经常会出现相似汉字对应的特征值$a_j$大

于正确类别对应的特征值 a_i 的情况，导致 Softmax 分类器分类错误、无法有效地对相似汉字进行分类。

考虑使用其他分类器对 CNN 提取到的特征进行分类。SVM 是一种性能优异的分类器，广泛应用在各种分类任务中，因此结合 CNN 与 SVM 识别相似手写汉字。首先使用 CNN 提取特征向量，再使用 SVM 对特征向量进行分类，识别相似手写汉字。

4.4.1 SVM

SVM 是一种常用的分类器，它根据结构风险最小化准则，在使训练样本分类误差极小化的前提下，尽量提高分类器的泛化推广能力[55, 56]。从实施的角度，训练支持向量机等价于解一个线性约束的二次规划问题，使得分隔特征空间中两类模式点的两个超平面之间距离最大，而且它能保证得到的解为全局最优点。

SVM 最初运用于二分类问题，它的思想是在不同类别样本之间寻找一个超平面，使得两类模式样本到超平面的最近距离之和最大[57]。超平面的位置示意图如图 4.7 所示。

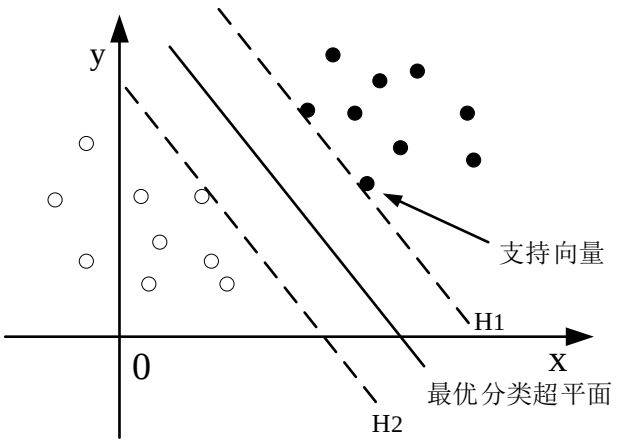


图 4.7 超平面位置示意图

线性可分时，以二分类数据分类为例，讨论 SVM 分类器超平面计算方式。设训练样本 $D_i=(x_i,y_i),i=1,\cdots,n$，$y_i\in\{+1,-1\}$，其中 x_i 为输入样本，y_i 为样本类别，n 为样本数，超平面方程为 $\boldsymbol{wx}+b=0$。为使超平面对所有样本正确分类，又要保证间隔最大，这个二分类问题可以转换成一个带约束的最小值问题：

$$\min \frac{1}{2}\|\boldsymbol{w}\|^2,\ s.t.\ y_i\left[\boldsymbol{w}x_i+b\right]\geq 1\ \ i=1,2,\cdots n \tag{4.9}$$

为解决这个带约束的优化问题，引入拉格朗日函数：

$$L\left(\boldsymbol{w},b,\alpha\right)=\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}-\sum_{i=1}^{n}\alpha_i\left[y_i\left(\boldsymbol{w}x_i\right)-1\right] \tag{4.10}$$

其中$\alpha_i>0$为拉格朗日系数。最优化问题的解满足对$\boldsymbol{w}$和b的偏导数为 0，将问题转换为对应的对偶问题：

$$\max Q\left(\alpha\right)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j x_i^T x_j\ ,\ \ s.t.\sum_{i=1}^{n}\alpha_i y_i=0\ \ \alpha_i\geq 0, i=1,2,\cdots n \tag{4.11}$$

若$\boldsymbol{\alpha}^*=\left(\alpha_1^*,\alpha_2^*,\cdots,\alpha_n^*\right)$为最优解，则权值向量：

$$\boldsymbol{w}^*=\sum_{j=1}^{n}\alpha_j^* y_j x_j \tag{4.12}$$

$$b^*=y_i-\sum_{j=1}^{n}y_j\alpha_j^* x_j^T x_i \tag{4.13}$$

其中$\left(x_i,y_i\right)$满足$y_i\left(\boldsymbol{w}x_i\right)-1=0$，因此最优分类超平面函数为：

$$f\left(\boldsymbol{x}\right)=\operatorname{sgn}\left\{\left(\boldsymbol{w}^*\boldsymbol{x}\right)+b^*\right\}=\operatorname{sgn}\left\{\left(\sum_{j=1}^{n}\alpha_j^* y_j x_j^T x_i\right)+b^*\right\} \tag{4.14}$$

不为 0 的α_i所对应的样本即为支持向量，最优分类超平面就由支持向量所确定。

当样本数据为线性不可分时，SVM 的解决思路有两种。当只是因为少数点的影响导致无法线性可分时，或原始数据是线性可分的，但是存在少量“异常点”导致分类超平面差强人意；可以加入松弛变量和惩罚因子，找到相对“最好”的分割平面，尽可能地将数据分类正确。大多数情况下，当数据线性不可分时，需要使用核函数，将低维的数据映射到更高维的空间，使得低维空间中的线性不可分数据在高维空间是线性可分的，那么在高维空间使用线性分类模型即可。常用的核函数有：

1 线性核函数：$K\left(x_i,x_j\right)=x_i^T x_j$；

2 多项式核函数：$K\left(x_i,x_j\right)=\left(x_i^T x_j+\gamma\right)^d$；

3 径向基核函数：$K\left(x_i,x_j\right)=\exp\left(-\frac{\left\|x_i-x_j\right\|^2}{2\sigma^2}\right)$；

4 Sigmod 核函数：$K\left(x_i,x_j\right)=\tanh\left(\alpha x_i^T x_j+c\right)$。

SVM 最初运用于二分类，当使用 SVM 进行多分类时，基础的方法有一对一法（One Versus One，OVO）和一对多法（One Versus Rest，OVR）。一对一法是在任意两类样本之间设计一个 SVM，所以 k 个类别需要设计 $\frac{k\left(k+1\right)}{2}$ 个分类器，分类时，计算所有分类器分得的类别，进行投票，选择票数最多的类别作为待分类种类。一对多法将 k 分类问题转换为 k 个二分类问题，训练时依次把某个类别样本归为一类，其他剩余样本归为另一类，总共需设计 k 个 SVM 分类器。

4.4.2 结合 CNN 与 SVM 的相似手写汉字识别

改进后的网络结构如图 4.8 所示。CNN 部分与 Net-S 网络结构相同，在全连接层 FC-200 后接 SVM 分类器进行分类。模型将 Net-S 网络全连接层 FC-200 的输出作为特征向量，使用 SVM 分类器替代 Softmax 分类器进行分类，识别相似手写汉字。

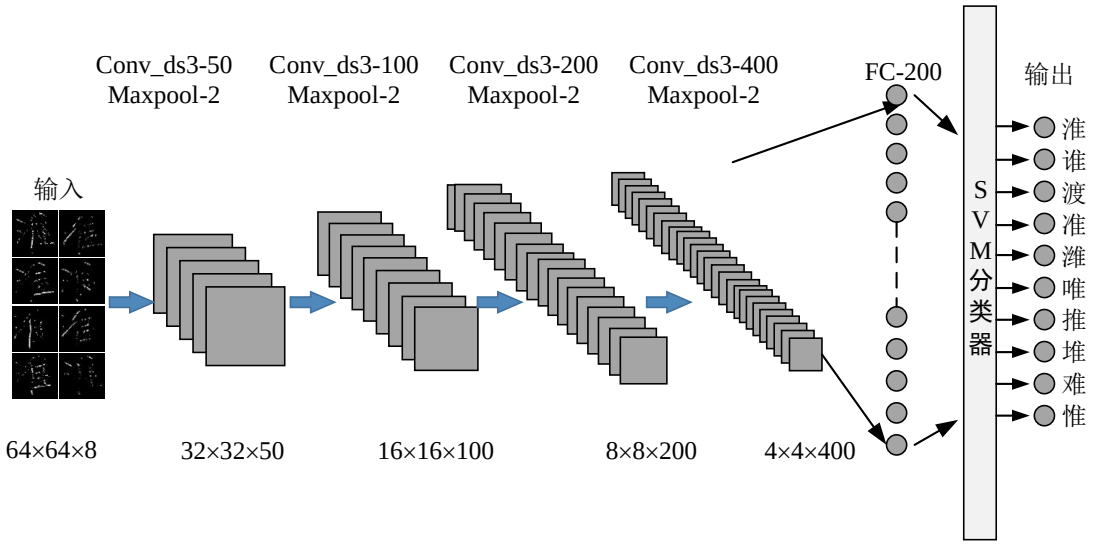


图 4.8 结合 CNN 与 SVM 的相似手写汉字识别网络结构图

模型训练分为 CNN 训练和 SVM 训练两部分。实验环境、数据集、CNN 超参数设置与 4.3.2 中均保持一致。首先训练 CNN 用于提取手写汉字 200 维特征向量。训练时，使用每组汉字训练样本分别训练 Net-S，得到训练好的 15 个 CNN 模型，并保存对应的模型权重。CNN 训练结束后，将训练集样本、验证集样本通过训练好的 CNN 获取全连接层 FC-200 的输出，得到 200 维特征向量，用于训练 SVM。

然后训练 SVM 分类器，用于分类相似手写汉字。本文采用的核函数为径向基核函数，使用网格搜索法寻找最佳的惩罚因子 C 与核函数参数 σ。针对每组参数 (C,σ)，在训练集上使用 15 组相似汉字特征向量训练 SVM，在验证集上进行验证，获取 15 组相似手写汉字在验证集上的准确率，求取平均值，取验证集上平均准确率最高的一组参数作为 SVM 模型参数。最终获得的最优模型参数为(0.4,10)。

测试时，同样先利用训练好的 CNN 获得测试样本的 200 维特征向量，再使用 SVM 进行分类。最终在测试集上取得的平均准确率为 90.98%。

4.5 实验结果分析

4.5.1 实验结果分析

对两种相似手写汉字识别结果进行分析，并单独使用 CNN、特征提取+SVM 的方法对相似手写汉字进行识别作为对比。表 4.4 给出了四种方法在每组汉字上的识别准确率，其中 CNN 为单独使用基础卷积神经网络 Net-S 对 15 组相似手写汉字进行识别的结果，SVM 为单独使用支持向量机对相似手写汉字进行识别的结果。单独使用 SVM 识别相似手写汉字时，先提取图像的 512 维梯度特征，再降维到 200 维进行识别。

从表 4.4 中可以看出，单独使用 CNN 识别相似手写汉字的基础准确率为 89.39%。CNN+Center Loss 的方法取得了最好的识别效果，平均准确率为 92.07%，相对于单独使用 CNN 的方法平均准确率提升了 2.68%。CNN+SVM 的方法效果次之，平均识别准确率为 90.98%，相对于单独使用 CNN 的方法平均准确率提升了 1.59%。而传统特征提取+SVM 方法识别效果较差，平均准确率仅为 86.72%。

表 4.4 相似手写汉字识别准确率结果

组号	CNN(%)	SVM(%)	CNN+SVM(%)	CNN+Center Loss(%)
1	89.11	87.44	92.45	90.48
2	90.25	89.52	93.76	92.29
3	84.38	81.41	85.82	85.83
4	88.33	89.50	92.83	90.86
5	87.33	85.33	90.33	90.69
6	95.47	91.46	94.13	96.84
7	93.03	88.58	93.21	94.89
8	88.96	87.62	92.30	93.84
9	91.97	93.10	94.30	95.85
10	85.47	83.30	88.81	89.68
11	92.15	90.65	93.82	95.02
12	86.97	83.13	87.81	91.51
13	87.73	80.33	86.53	88.60
14	91.20	87.47	92.35	95.11
15	88.48	81.91	86.30	89.51
平均值	89.39	86.72	90.98	92.07

为了更加清晰直观地观察每种方法在不同组汉字上的识别率变化情况，图 4.9 给出了对应的折线图。可以发现，四种方法识别准确率具有相同的变化规律，对于同一组相似汉字，四种方法识别结果要么都偏低，要么都偏高。使用 CNN+Center Loss 的分类方法，在所有组的分类结果上均高于单独使用 CNN 以及 SVM 的分类方法。CNN+SVM 的方法，性能比较稳定，最好的一组识别准确率为 94.30%，最差的一组识别准确率为 85.82%，波动为 8.48%，而 CNN、SVM、CNN+Center Loss 的方法准确率波动分别为 11.09%、12.77%、11.01%。

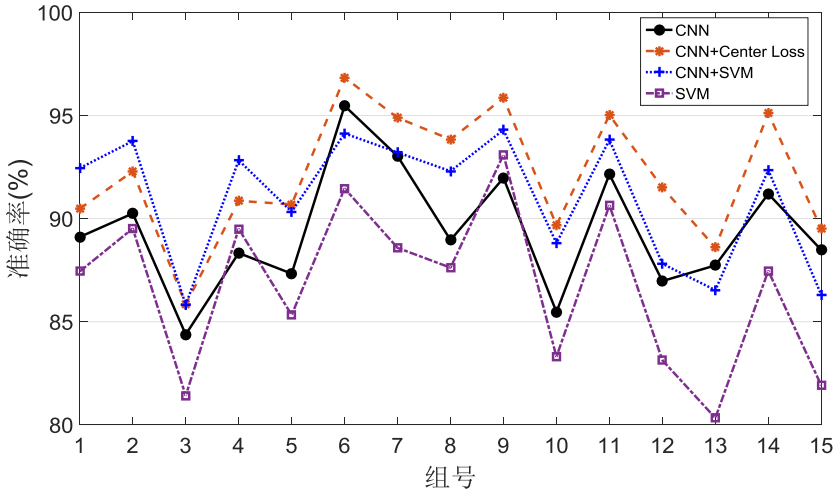


图 4.9 15 组相似手写汉字识别准确率折线图

实验结果表明，相似手写汉字识别中，相对于传统的特征提取加 SVM 的分类方法，CNN 具有更好的识别效果。SVM 性能较差的原因是在特征提取过程中丢失了细微的鉴别信息，而卷积神经网络具有强大的特征提取能力，刚好可以弥补这一缺陷。通过对 CNN 的 Softmax 层进行改进，使用 CNN+Center Loss，CNN+SVM 的方式，均能够在 CNN 的基础上进一步提升识别准确率。

4.5.2 相关讨论

本文第 3 章和第 4 章分别对 3755 类汉字及相似手写汉字进行了识别，在实际应用中，可以采用二级分类的结构，先对汉字进行 3755 类分类，判断汉字是否属于相似汉字组中汉字，如果不属于，则直接输出汉字分类结果，如果属于，则使用对应的相似手写汉字分类器模型进行二次分类，输出最终的分类结果。使用 CNN+SVM 的相似手写汉字识别方法时，二级分类示意图如图 4.10 所示。

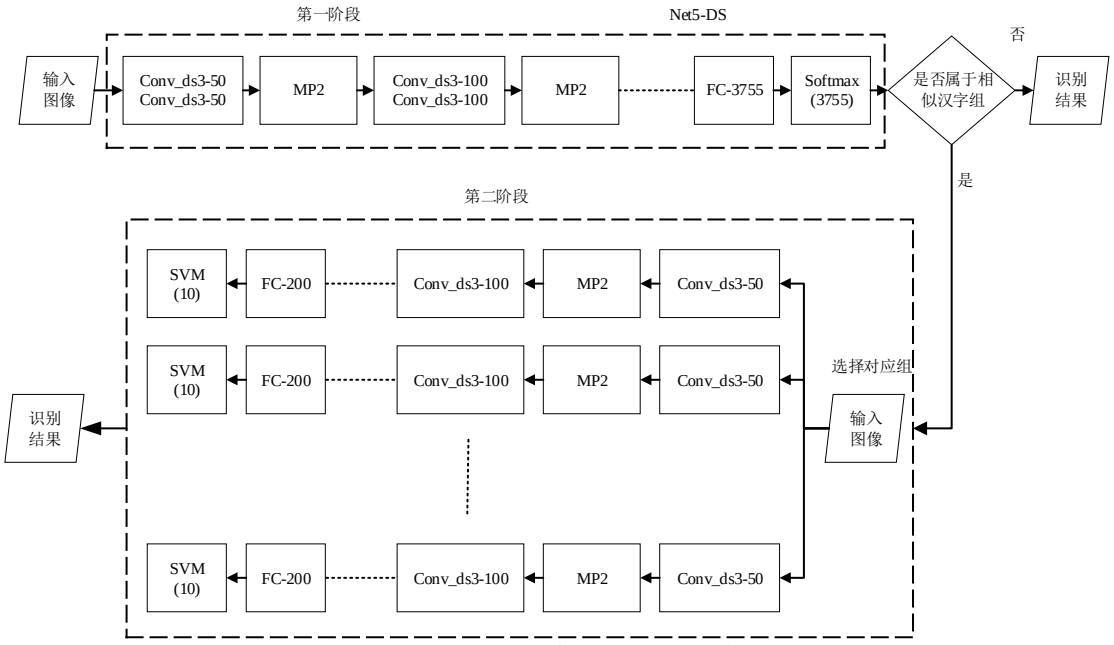


图 4.10 手写汉字二级分类示意图

4.6 本章小结

本章对相似手写汉字识别问题进行了专项研究。首先根据一级 3755 类汉字识别结果，设计了具有代表性的 15 组相似汉字。并且针对数据集过小问题，使用仿

射变换进行数据集扩充。接着介绍了基于 CNN 和 Center Loss 的相似手写汉字识别方法，及结合 CNN 与 SVM 的相似手写汉字识别方法。实验结果表明，两种方法均能够提升相似手写汉字的识别效果，取得较高的识别准确率。

第 5 章 总结与展望

5.1 全文总结

汉字个数众多、书写随意、而且汉字中形近字较多，使得汉字识别一直是模式识别中的热点与难点。近年来，深度学习发展迅速，在包括图像分类在内的多个领域取得了巨大成功，特别是卷积神经网络在图像分类中显现出良好的性能。在此背景下，本文使用卷积神经网络对手写汉字进行识别，具体包括以下几个部分：

1. 调研了脱机手写汉字识别及深度学习的国内外研究现状，明确了汉字识别的难点，阐述了本文的主要研究内容。

2. 分析对比了基于卷积神经网络的手写汉字识别方法与传统手写汉字识别方法，详细讲解了全连接神经网络、卷积神经网络的相关原理，并指出本文所使用的减轻网络过拟合、加速网络训练的方法。

3. 基于卷积神经网络，对 GB2312-80 规定的一级 3755 个汉字进行识别。对卷积神经网络输入，卷积方式进行改进，使用汉字图像八方向梯度特征作为卷积神经网络输入，使用深度可分离卷积方式进行卷积。分析对比了卷积神经网络模型复杂度、输入数据、卷积方式对汉字识别准确率的影响。实验结果表明，复杂的网络模型易于获得更好的识别效果，但性能增益递减，汉字的八方向梯度特征图像输入及深度可分离卷积方式，能够在不同程度上提升汉字的识别效果。

4. 对一级 3755 个汉字错误识别结果进行统计，分析错误原因，发现错误主要来源于形近字的影响。在此基础上，进一步研究相似手写汉字识别方法。根据一级 3755 个汉字识别中的错误识别数据，设计出 15 组相似汉字，每组包括 10 个汉字。使用两种不同的方法对相似手写汉字进行识别。方法一，使用 Softmax 损失函数和 Center Loss 函数作为卷积神经网络的联合损失函数，对相似手写汉字进行识别。方法二，将卷积神经网络看成特征提取器，使用 SVM 分类器替代 Softmax 进行分类。实验结果表明，CNN+Center Loss，及 CNN+SVM 的方法均能够提升相似手写汉字的识别准确率。

5.2 展望

尽管本文提出的基于深度学习的脱机手写汉字识别方法取得了不错的效果，但汉字识别的道路远没有结束。由于研究时间和研究条件有限，本文仍然存在一些不足之处，未来可以在以下几个方面进行进一步的研究：

1. 汉字数量巨大，超过 5 万个，本文只对 GB2312-80 规定的最常用的一级 3755 个汉字进行了识别，未来可以考虑进行更大类别的汉字分类任务，例如 GB2312-80 标准中的所有一级和二级 6763 个汉字。

2. 深度学习的训练往往需要很长时间，本文在训练卷积神经网络时面临着同样的问题，虽然采用 GPU 训练相对 CPU 速度大幅提升，但远远不够，未来可以研究加速卷积神经网络训练的相关算法，提升训练效率，节省训练时间。

3. 本文只对脱机手写单字进行了识别，下一步可以在此基础上结合语言模型进行文本行识别，还可以进行自然场景下的文字识别。

4. CASIA-HWDB 数据集中，存在一定的“脏数据”及错误标签的样本，如果能够快速准确地剔除这些样本，能够使实验更加准确，进一步提高识别准确率。如何科学有效地定位这些样本，也是一件充满意义且具有挑战性的事。

参考文献

[1] Liu Chenglin, Yin Fei, Wang Dahan, et al. Online and offline handwritten Chinese character recognition: benchmarking on new databases[J]. Pattern Recognition, 2013, 46(1): 155-162.

[2] 周星辰. 基于深度模型的脱机手写体汉字识别研究[D]. 杭州: 浙江大学, 2016.

[3] Wen Yandong, Zhang Kaipeng, Li Zhifeng, et al. A discriminative feature learning approach for deep face recognition[C]// European conference on computer vision (ECCV). Amsterdam, Netherlands: Springer Verlag, 2016: 499-515.

[4] Wang Hao, Wng Yitong, Zhouzheng, et al. CosFace: large margin cosine loss for deep face recognition[C]// 2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE Press, 2018: 5265-5274.

[5] Graves A, Jaitly N. Towards End-to-end speech recognition with recurrent neural networks[C]// Proceedings of the 31st International Conference on Machine Learning. Beijing, China: International Machine Learning Society, 2014: 1764-1772.

[6] Lee K, Park C, Kim N, et al. Accelerating recurrent neural network language model based online speech recognition system[C]// 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Calgary, AB, Canada: IEEE Press, 2018: 5904-5908.

[7] Mulder W D, Bethard S, Moens M F. A survey onthe application of recurrent neural networks to statistical language modeling[J]. Computer Speech & Language, 2015, 30(1): 61-98.

[8] Gehring J, Auli M Grangier D, et al. Convolutional sequence to sequence learning[C]// Proceedings of the 34th International Conference on Machine Learning. Sydney, NSW, Australia: International Machine Learning Society, 2017: 1243-1252.

[9] Chen Chenyi, Seff A, Kornhauser A, et al. Deepdriving: learning affordance for direct perception in autonomous driving[C]// 2015 IEEE International Conference on Computer Vision (ICCV). Santiago, Chile: IEEE, 2015: 2722-2730.

[10] Sun Shaohui, Sarukkai R, Kwok J, et al. Accurate deep direct geo-localization from ground imagery and Phone-Grade GPS[C]// 2018 IEEE/CVF Conference on

Computer Vision and Pattern Recognition Workshops (CVPRW). Salt Lake City, UT, USA: IEEE, 2018: 1129-11297.

[11] Casey R, Nagy G. Recognition of printed Chinese characters[J] IEEE Transactions on Electronic Computers, 1966, 15(1): 91-101.

[12] 刘全升. 深度学习及其在脱机手写汉字识别领域的应用研究[D]. 广州: 华南理工大学, 2016.

[13] 国家标准总局. GB/T 2312-1980. 信息交换用汉字编码字符集 基本集[S].

[14] 高灿. 基于卷积神经网络的脱机手写汉字识别系统研究[D]. 合肥: 安徽理工大学, 2017.

[15] 金连文, 钟卓耀, 杨钊, 等. 深度学习在手写汉字识别中的应用综述[J]. 自动化学报, 2016, 42(08): 1125-1141.

[16] Liu Chenlin, Marukawa K. Pseudo two-dimensional shape normalization methods for handwritten Chinese character recognition. Pattern Recognition, 2005, 38(12): 2242-2255.

[17] 王有旺. 深度学习及其在手写汉字识别中的应用研究[D]. 广州: 华南理工大学, 2014.

[18] Liu Chenlin. Normalization-cooperated gradient feature extraction for handwritten character recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(8): 1465-1469.

[19] Kimura F, Takashina K, Tsuruoka S, Miyake Y. Modied quadratic discriminant functions and the application to Chinese character recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1987, 9(1): 149-153.

[20] Mangasarian O L, Musicant D R. Data discrimination via nonlinear generalized support vector machines[J]. Complementarity: Applications, Algorithms and Extensions, 2001, 50: 233-251.

[21] Kim H J, Kim K H, Kim S K, Lee J K. On-line recognition of handwritten Chinese characters based on hidden Markov models[J]. Pattern Recognition, 1997, 30(9): 1489-1500.

[22] Liu Chenlin, Sako H, Fujisawa H. Discriminative learning quadratic discriminant function for handwriting recognition. IEEE Transactions on Neural Networks[J], 2004, 15(2): 430-444.

[23] Jin Xiaobo, Liu Chenlin, Hou Xinwen. Regularized margin-based conditional

log-likelihood loss for prototype learning[J]. Pattern Recognition, 2010, 43(7): 2428-2438.

[24] Lin Chenglin, Yin Fei, Wng Qiufeng, et al. ICDAR 2011 Chinese handwriting recognition competition[C]// 2011 11th International Conference on Document Analysis and Recognition (ICDAR). Beijing, China: IEEE Press, 2011: 1464-1469.

[25] Yin Fei, Wang Qiufeng, Zhang Xuyao, et al. ICDAR 2013 Chinese handwriting recognition competition[C]// 2013 12th International Conference on Document Analysis and Recognition (ICDAR). Washington, DC, USA: IEEE Press, 2013: 1464-1470.

[26] Liu Chenlin, Yin Fei, Wang Dahan, et al. Online and offline handwritten Chinese character recognition: benchmarking on new databases[J]. Pattern Recognition, 2013, 46(1): 155-162.

[27] Wang Yanwei, Liu Changsong, Ding Xiaoqing. Similar pattern discriminant analysis for improving chinese character recognition accuracy[C]// 2013 12th International Conference on Document Analysis and Recognition (ICDAR). Washington, DC, USA: IEEE Press, 2013: 1056-1060.

[28] 杨钊, 陶大鹏, 张树业, 等. 大数据下的基于深度神经网的相似汉字识别[J]. 通信学报, 2014, 35(9): 184-189.

[29] 高学, 王有旺. 基于CNN和随机弹性形变的相似手写汉字识别[J]. 华南理工大学学报(自然科学版), 2014, 42(1): 72-76.

[30] Wang Qingqing, Lu Yue. Similar handwritten Chinese character recognition using hierarchical CNN model[C]// 2017 14th International Conference on Document Analysis and Recognition (ICDAR). Kyoto, Japan: IEEE, 2017: 603-608.

[31] LeCun Y, Bose B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition[J]. Neural Computation, 1989, 1(4): 541-551.

[32] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[33] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. Science, 2006, 313(5786): 504-507.

[34] Hinton G E, Osindero S, Teh Y. A fast learning algorithm for deep belief nets[J]. Neural Computation, 2006, 18(7): 1527-1554.

[35] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep

convolutional neural networks[C]// 26th Annual Conference on Neural Information Processing Systems. New York, USA: Neural information processing systems foundation, 2012: 1097-1105.

[36] Szegedy C, Liu Wei, Jia Yang Qing, et al. A. Going deeper with convolutions[C]// 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, MA, USA: IEEE Press, 2015: 1-9.

[37] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556, 2014.

[38] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus, OH, USA: IEEE Press, 2014: 580-587.

[39] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. Nature, 2015, 518(7540): 529-533.

[40] Vinyals O, Fortunato M, Jaitly N. Pointer networks[C]// 29th Annual Conference on Neural Information Processing Systems. Montreal, Quebec, Canada: Neural information processing systems foundation, 2015: 2674-2682.

[41] He Kaiming, Zhang Xiangyu, Ren Shaoqing, et al. Deep residual learning for image recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE Press, 2016: 770-778.

[42] Sabour S, Frosst N, Hinton G E. Dynamic routing between capsules[C]// 31th Annual Conference on Neural Information Processing Systems: Neural information processing systems foundation, 2017: 3859-3869.

[43] 孙巍巍. 基于深度学习的手写汉字识别技术研究[D]. 哈尔滨, 哈尔滨理工大学, 2017.

[44] 郭鹏. 深度卷积神经网络及其在手写体汉字识别中的应用研究[D]. 成都: 四川师范大学, 2016.

[45] Zhang Xuyao, Bengio Y, Liu Chenglin. Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark[J]. Pattern Recognition, 2017, 61(SI): 348-360.

[46] Heiden R, Gren f. The Box-Cox metric for nearest neighbor classification improvement[J]. Pattern Recognition, 1997, 30(2): 273-279.

[47] Ioffe S, Szegedy C. Batch Normalization: Accelerating deep network training by reducing internal covariate shift[C]// Proceedings of the 32nd International

Conference on Machine Learning. Lille, France: International Machine Learning Society, 2015: 448-456.

[48] 冷玉龙, 韦一心. 中华字海[M]. 北京: 中国友谊出版公司, 1994: 6.

[49] Zhang Honggang, Guo Jun, Chen Guang, et al. HCL2000 - A large-scale handwritten Chinese character database for handwritten character recognition[C]// 2009 10th International Conference on Document Analysis and Recognition (ICDAR), 2009: 286-290.

[50] Liu Chenglin, Yin Fei, Wang Dahan, et al. CASIA online and offline Chinese handwriting databases[C]// 2011 International Conference on Document Analysis and Recognition. Barcelona, Spain: IEEE Press, 2011: 37-41.

[51] Wu Chunpeng, Fan Wei, He Yuan, et al. Handwritten character recognition by alternately trained relaxation convolutional neural network[C]// 2014 14th International Conference on Frontiers in Handwriting Recognition. Heraklion, Greece: IEEE Press, 2014: 291-296.

[52] Zhong Zhuoyao, Jin Lianwen, Xie Zecheng. High performance offline handwritten Chinese character recognition using GoogLeNet and directional feature maps[C]// 2015 13th International Conference on Document Analysis and Recognition (ICDAR). Tunis, Tunisia: IEEE Press, 2015: 846-850.

[53] 封筠. 基于支持向量机的脱机手写相似汉字识别的研究[D]. 北京: 北京科技大学, 2005.

[54] Wen Yandong, Zhang Kaipeng, Li Zhifeng, et al. A discriminative feature learning approach for deep face recognition[C]// European conference on computer vision (ECCV). Amsterdam, Netherlands: Springer Verlag, 2016: 499-515.

[55] 丁世飞, 齐丙娟, 谭红艳. 支持向量机理论与算法研究综述[J]. 电子科技大学学报, 2011, 40(1): 2-10.

[56] 高学, 金连文, 尹俊勋, 等. 一种基于支持向量机的手写汉字识别方法[J]. 电子学报, 2002, 30(5): 651-654.

[57] 奉国和. SVM 分类核函数及参数选择比较[J]. 计算机工程与应用, 2011, 47(03): 123-124, 128.

致谢

光阴荏苒，一晃研究生三年生活即将结束，我也即将离开母校步入社会。在这三年里，我成长了很多，不仅收获了宝贵的知识与技能，而且学到了许多为人处世的道理，心态也更加成熟。一路走来，感慨万千。在这毕业之际，我想真诚地向研究生期间给予我帮助与陪伴的人表示感谢。

首先感谢我的导师谭钦红老师，谭老师知识渊博、耐心负责、严谨求实，在研究生学习期间及论文写作过程中，给予我一次又一次的帮助，每当我遇到困难坎坷时，她都能帮助我共同解决，时刻为我们着想。同时，感谢实验室黄俊老师，三年来，黄老师兢兢业业，带领我们做项目，极大地提升了我们解决实际问题的能力。这里，还要感谢信号处理与片上系统团队的代少升、梁燕老师，在日常学习生活中给予我们的关心与指导。

感谢实验室同届的刘灿、余忠永、胡嘉豪、袁梅、胡丹、许二敏、钟琳倩、胡煦、吴正同学，三年来，大家相互陪伴，共同成长，使我的研究生生活丰富多彩。感谢施新岚同学，大家一起参加比赛，互相探讨学术问题。感谢王君龙师兄研究生期间对我的关心与照顾，感谢刘武启师弟和郑小楠师妹，在生活上给我的鼓励与帮助，感谢朝夕相处的室友给我营造良好的生活学习环境。

特别感谢我的父母，几十年来，他们任劳任怨，含辛茹苦抚养我长大成人，在背后默默地支持我。他们就是我最坚强的后盾，是他们激励着我前行，每当我有些许懈怠时，想起他们，我就明白了拼搏的意义。

最后，感谢各位评审老师百忙之中抽出时间来评阅我的论文，正是你们的建议与意见，使我不断提高、进步、做更好的自己。

攻读硕士学位期间从事的科研工作及取得的成果

参与项目情况：

[1] 基于 AR/VR（增强现实/虚拟现实）技术的学生认知平台研发（CY170603），纵向，2017.10-2018.6

[2] 智能家居安防监控系统，横向，2018.04-2018.07

发表论文情况：

[1] **黄洋**, 谭钦红, 施新岚. 结合八方向梯度特征和 CNN 的相似手写汉字识别[J]. 信息通信, 2019, (4): 5-8.

发表专利情况：

[1] **黄洋**, 黄俊, 谭钦红, 等. 一种基于卷积神经网络的脱机手写汉字识别方法. 中国, 201811502762.6[P]. 2018.

获奖：

[1] “华为杯”中国研究生数学建模竞赛全国三等奖

[2] 重庆市科慧杯研究生创新创业大赛一等奖

[3] “华为杯”全国研究生电子设计竞赛西南赛区二等奖

[4] 蓝桥杯全国软件和信息技术专业人才大赛重庆赛区 C/C++程序设计三等奖

[5] “兆易创新杯”全国研究生电子设计竞赛西南赛区三等奖

[6] 研究生一等学业奖学金（2 次）