

# 武汉理工大学毕业设计（论文）

## 基于 BP 神经网络的手写字母识别

学院（系）： 信息工程学院

专业班级： 电信 1603 班

学生姓名： 刘云飞

指导教师： 黄龙庭

## 学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名：

年 月 日

## 学位论文版权使用授权书

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于 1、保密口，在 年解密后适用本授权书

2、不保密口。

（请在以上相应方框内打“√”）

作者签名：

年 月 日

导师签名：

年 月 日

## 摘 要

光学字符识别（OCR）在数字信息时代的今天的重要性不言而喻。本设计使用 MATLAB 设计手写字母识别模型，该模型使用 BP 神经网络学习字母样本和字母标签之间的关联。它的基本框架是图像获取、预处理、字符分割、归一化、特征提取、神经网络训练、手写字母识别和后处理部分。通过对 EMNIST 数据集的分析与处理形成标准化的特征向量（输入空间），将输入空间划分为训练集与测试集两部分，其中训练集数据用于输入神经网络以学习分类模型，测试集用于测试模型对新数据的泛化能力。本设计尝试多种特征提取方式并分析其优缺点，最后使用一种新的组合特征提取方式，与单一提取方式相比识别精确度更高，且特征向量维度较低，能降低分析和学习时间。最终的识别系统对小写字母的识别精确度达到最高 96.10% 和平均 85.10%。

**关键词：**机器学习；人工神经网络；光学字符识别；数字图像处理

## Abstract

The importance of optical character recognition (OCR) in today's digital information era is self-evident. This design uses MATLAB to design a handwritten letter recognition model. It uses an BP neural network to learn the association between letter samples and letter labels. The basic framework of the model is image acquisition, file preprocessing, letter segmentation, normalization, feature extraction and training the neural network, handwritten letter recognition and post-processing. By analyzing and processing the EMNIST dataset, a standardized feature vector (input space) is formed, and the input space is divided into a training set and a test set. The training set data is input to the neural network to learn the classification model. The test set is to exam the generalization capacity of new data of the model. This design attempts a variety of feature extraction methods and analyzes their advantages and disadvantages. Finally, a new combined feature extraction method is used. Compared with the single extraction method, the recognition accuracy is higher, and the feature vector dimension is lower, which can reduce analysis and learning time. The recognition accuracy of the final recognition system for lowercase letters reaches a maximum of 96.10% and an average of 86.10%.

**Keywords:** machine learning; artificial neural network; optical character recognition; digital image processing

## 目 录

第 1 章 绪论.....	1
1.1 手写字母识别的研究目的及意义.....	1
1.2 OCR 的发展与现状.....	1
1.2.1 OCR 的发展.....	1
1.2.2 OCR 的应用现状.....	2
1.3 本文的结构.....	3
第 2 章 手写字母分类模型.....	4
2.1 机器学习中的分类.....	4
2.2 神经网络.....	4
2.2.1 激活函数.....	5
2.2.2 反向传播算法.....	6
2.2.3 BP 神经网络.....	6
2.2.4 神经网络评估.....	9
2.3 本章小结.....	9
第 3 章 预处理和特征提取.....	10
3.1 EMNIST 数据集.....	10
3.2 预处理.....	11
3.2.1 灰度化与二值化.....	12
3.2.2 平滑去噪.....	13
3.2.3 字符分割.....	14
3.2.4 归一化处理.....	15
3.3 特征提取.....	16
3.3.1 区域密度特征.....	16
3.3.2 边缘距离特征.....	16
3.3.3 投影统计直方图特征.....	17
3.3.4 组合特征.....	18
3.4 本章小结.....	19
第 4 章 手写字母识别系统.....	20
4.1 识别系统流程.....	20
4.2 神经网络训练.....	21
4.2.1 数据集的选择与处理.....	21
4.2.2 Newff 函数.....	22

4.2.3 参数设定与训练结果 .....	22
4.3 文本图像处理 .....	24
4.3.1 字符分组 .....	24
4.3.2 行提取 .....	24
4.4 实验结果 .....	25
4.4.1 GUI 设计 .....	25
4.4.2 系统评估 .....	27
4.5 本章小结 .....	27
第 5 章 总结与展望 .....	28
5.1 总结 .....	28
5.2 展望 .....	28
参考文献 .....	29
致谢 .....	31

## 第1章 绪论

### 1.1 手写字母识别的研究目的及意义

将纸张上的文本符号转换成机器可处理格式的过程称为光学字符识别(Optical Character Recognition)，它是文档分析系统（File Analysis System）领域的核心，在基于纸张的社会向无纸化的电子信息世界的转变中起着重要的作用。手写字符识别已成为图像处理和模式识别领域中活跃而且富有挑战性的研究领域之一，它具有许多应用，早期的应用主要是邮件分拣，随后在很多自动化和商业领域应用广泛，主要包括盲人阅读辅助工具，银行支票识别，证件识别以及将手写文档转换为机读文本形式。手写体识别极大地促进了自动化过程的发展，同时它使许多应用中的人机接口变得灵活。

手写字符识别分为离线和在线识别两种类型，在离线识别中，通常通过扫描仪以光学方式捕获笔迹并用作图像；在线系统中连续点的二维坐标表示为时间的函数，系统读取笔划的顺序和形状，由于实时获取信息，在线方法已被证明在识别手写字符方面优于其离线方法<sup>[1]</sup>，但是，对于已经生成的文本或资料，离线识别的方法尤为重要。

手写字符的形状变化很大，很难为它们创建一套模板。因此，手写字符识别主要通过结构分析法实现，其结构可以分解为多个部分，因此可以通过这些部分的特征以及这些部分之间的关系来描述它，那么接下来的主要问题就是如何选择和描述这些特征和它们之间的这种关系，以便这种描述可以清楚地识别每个字符，因此，特征提取已成为模式识别研究的关键。从手写体识别开始被研究至今，人们一直在不断探索字符识别的新技术和新方法，它们旨在减少处理时间，同时提供更高的识别精度。

### 1.2 OCR 的发展与现状

#### 1.2.1 OCR 的发展

OCR 技术的提出者是 Tausheck，他于 1929 年德国获得了 OCR 的专利，其原理是模板/掩码匹配，该技术使用了光学和机械模板匹配，穿过机械掩模的光被光电探测器捕获并进行机械扫描，当发生完全匹配时，光线无法到达检测器，机器便能识别出打印在纸上的字符<sup>[2]</sup>，这在早期的邮件分拣系统中被广泛的使用。

事实上，19 世纪 70 年代，人们开发了一种称为 OCR-A 的特殊字体，可将其用于银行支票等，这种字体的每个字母的宽度完全相同（等宽字体）便于轻松区分每个字母，但这要求支票打印机全都使用该字体，OCR 设备也被设计为仅识别该字体。通过标准化一种简单的字体，OCR 在当时成为一个相对容易解决的问题，唯一的麻烦是，世界上大多数印刷品都不是用 OCR-A 印刷的，而且没有人使用这种字体来书写。很明显，该技术最大的制约

在于其仅适用于个别打印字符的识别，而对于手写字符或者其它字体的字符，模板匹配法就显得十分笨拙。因此，下一步就是使 OCR 程序识别多种非常常见的字体，如 Times, Helvetica, Courier 等打印字母和手写体字母。

在机器学习与人工智能迅速发展的时期，计算机通过学习经验识别字符成为可能。人们开始从字符中检测特征，然后基于这些特征区分不同的字符，这种方式亦被称为智能字符识别（Intelligent Character Recognition）。典型的 ICR 分类器有支持向量机模型，决策树模型和神经网络等。人工神经网络是一种可以模拟和适应未知系统并且可以同时处理定量和定性知识的系统，它的主要研究内容是如何使计算机模拟并实现人的自学和数学思维能力以从有限的样本中挖掘内部关系。神经网络主要是通过研究和存储已知样本的数据关系，推理规则和概率分布来推断和揭示未知数据样本中变量之间的潜在信息，神经网络存储包括定量和定性信息，它们均等地存储在神经网络的每个神经元中，因此具有非常强的鲁棒性和容错性<sup>[3]</sup>，神经网络的应用无疑给 OCR 的发展带来了新的技术突破。

### 1.2.2 OCR 的应用现状

尽管早期的 OCR 应用起源于海外，但近些年来，中国互联网行业发展十分迅猛，国内 OCR 技术与应用已处于世界前列。和很多其他技术不同，OCR 通常被用作“隐藏”技术，为我们日常生活中的许多知名系统和服务提供动力。如今十分重要的 OCR 技术用例包括纸质文档转成计算机文档，自动车牌识别以及为盲人和视障人士辅助阅读器。

OCR 的最著名的用例则是将打印的纸质文档转换为机器可读可编辑的文本文档形式。纸质文档经过 OCR 扫描处理后，即可使用 Microsoft Word 或 Google Docs 等文字处理器来编辑文本文档。在使用 OCR 技术之前，数字化打印纸质文档的唯一选择是手动重新键入文本，这不仅非常耗时，而且还带来了不准确和键入错误。事实证明，OCR 技术在对历史悠久的报纸和文本进行数字化方面非常有用，这些报纸和文本现在可以转换为可搜索的格式，并使得访问这些早期文本变得更加轻松快捷。

一旦打印出的页面以这种机器可读的文本格式显示，就可以按关键字搜索它，用文字处理程序进行编辑。机器可读的文本也可以通过屏幕阅读器进行解码，屏幕阅读器是使用语音合成器的工具来读取屏幕上的单词，以便盲人和视障人士可以阅读它们。早在上世纪 70 年代，OCR 的主要用途之一就是类似复读机的设备，称为 Kurzweil Reading Machine，该设备可以为盲人大声阅读印刷的书籍，如今这类设备在市场上需求量巨大，与此同时，相关技术也在不断被优化和创新，使得视力障碍人员对书面文字的需求成为可能，方便了他们的生活。

随着交通出行的电子化需求增大，机场，海关，车站都开始使用的护照等证件的 OCR 扫描技术极大地增加了安检和通行的效率。近几年来，移动智能手机端正以无法想象的速



度占据着人们的生活，屏幕捕捉文字，屏幕翻译等 OCR 技术也在不断的方便和丰富着人们的生活。

### 1.3 本文的结构

本次手写字母识别系统使用模块化设计，本文的主要结构如下：

第 1 章描述光学字符识别的研究背景和意义，并简单介绍其发展即应用现状。

第 2 章主要简单介绍了一些分类器的知识，结合机器学习阐述分类器的工作方式。然后重点介绍了神经元的结构以及神经网络的分类原理。然后系统的介绍了 BP 神经网络的学习算法与数学推导过程。

第 3 章主要介绍了本次实验使用的 EMNIST 手写字母数据集，讲述了数据集在输入模型训练之前的预处理和特征提取步骤。简单讲述了一些图像预处理方法，重点介绍了三种图片特征提取方式，最后通过实验决定最终的匹配神经网络的特征提取方法。

第 4 章讲述了手写字符识别模型的系统组成，包括 BP 神经网络的训练和文本图像处理。神经网络的训练包括数据集的选择和初步调整，神经网络的结构选择。文本图像处理包括文本文件可读性处理和复杂文本的行提取处理。

## 第2章 手写字母分类模型

手写字符与印刷体字符不同，由于习惯不同，人们的书写体存在诸多不同之处。正因为手写字符具有如此多变的形式，传统的模板匹配方式无法解决其分类问题。容易发现，尽管相同的字符具有多种手写形式，这些字符还是很容易被人眼识别出来，这是因为人能通过过去的学习积累经验，从而正确的做出区分。计算机通过自我学习建立这样的“经验”的过程称为机器学习，通过学习算法形成的系统称为模型，本章讲述用于分类的机器学习算法与分类器模型。

### 2.1 机器学习中的分类

机器学习的本质就是借助数学模型来理解数据与数据之间的关系，当模型的某些参数能适应这些数据变化时，该模型便具备了学习的能力，这是机器学习的认识阶段。一旦模型有了拟合旧的观测数据的能力，它必定能解释新的观测数据，这是机器学习的预测阶段。

机器学习的分类问题分为监督学习与非监督学习。类似于手写字母分类这种建立特征与真值（即标签）之间关联的有标签学习称为监督式学习，否则为非监督学习<sup>[4]</sup>。这种监督学习通过带标签的数据点建立输入与输出之间的关联从而对未知的，即不带标签的数据点进行预测和分类。机器学习中的分类学习模型有支持向量机模型（SVM），决策树模型和神经网络等。

### 2.2 神经网络

在不深究输入空间和输出空间之间的对应模型时，人们常常把他们之间极其复杂对应关系看作一个黑箱。现在本文用像图 2.1 这样的单元模拟该模型，由于结构类似人类神经元，也被称为神经元，这种模型的原理是将多个神经元按一定的层次结构关系组织成一个有机的整体，即神经网络。神经网络中每个神经元与其前面  $j$  个神经元的输入信号建立联系，这种联系的表现形式为神经元之间的权重大小，该神经元将其接收的所有输入求和与其阈值做对比（有时也称为加上偏置），最后通过激活函数产生该神经元的输出<sup>[3]</sup>。

事实上，神经元虽然可以看作是从人类神经元抽象出来的产物，但它与人类神经元工作原理有很大区别，所以从某种程度上讲，神经元更应该被成为“函数单元”，神经网络则是包含若干参数与函数的复合数学函数模型。神经网络的学习过程实质上就是通过训练数据来调整神经网络之间的权重和阈值。也就是说，神经网络的学习结果就蕴含在这些节点的阈值和节点间的权重中。

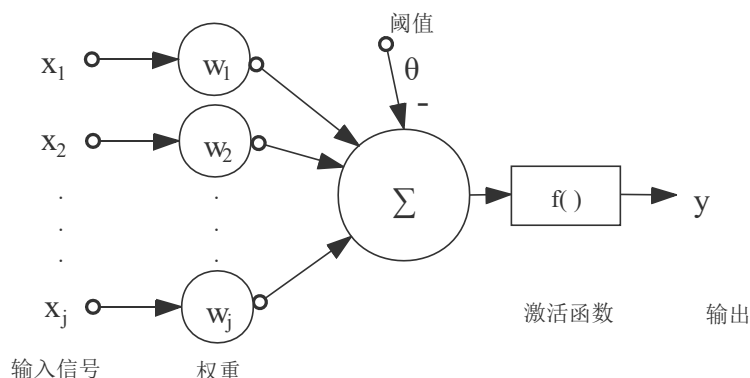


图 2.1 神经元模型

### 2.2.1 激活函数

有效的神经网络大多以数学证明做支撑，公式(2.1) 是上述模型的数学表达式表示：

$$y = f\left(\sum_{i=1}^j x_i w_i - \theta\right) \quad (2.1)$$

上述公式中  $f(\bullet)$  表示激活函数，是否可以不经过激活函数，直接将输入神经元与权重相乘在减去阈值直接向下一层传递结果呢？答案是否定的，因为激活函数为神经网络引入非线性属性，显然上述乘积与加和的图像都是平面或者高维的超平面，它们的特点是没有一点的弯曲，是线性的。线性模型虽然很容易求解和模拟，但神经网络的复杂性必然会受到限制。人们希望神经网络能模拟任意的函数，但如果不引入非线性属性，即便再怎么增加神经网络的层数，也达不到这种期望。激活函数还能决定是否激活该神经元使其对与其连接的下一层神经元起作用，换句话说，激活函数决定了本层神经元对下次一层神经元作用的程度大小。

最典型的激活函数是图 2.2(a)中的 sigmoid 函数，它的极端形式是图 2.2(b)所示的值为 0 或 1 的阶跃函数，其中函数值为 1 时神经元被激活，反之神经元不被激活。但是阶跃函数具有不连续，不可导等性质，不适用于作激活函数，sigmoid 函数把较大范围的值映射到 0 到 1 之间，因此也被称为“挤压函数”。

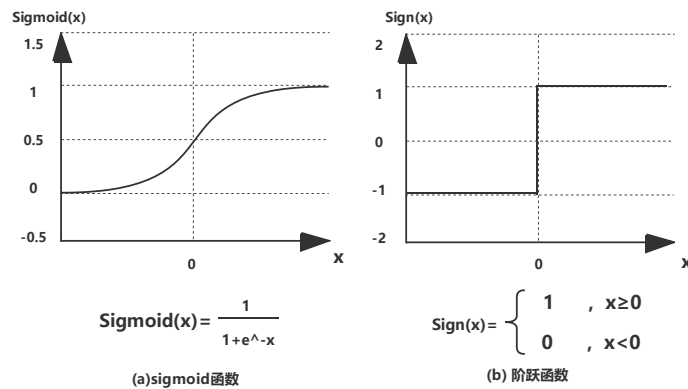


图 2.2 典型的神经元激活函数

### 2.2.2 反向传播算法

逻辑结构简单且由两层神经元组成的模型称为感知机，它是神经网络的前身，感知机的激活函数为阶跃函数。感知机的不足之处主要表现在三方面。第一，感知机只有最终输出时经过激活函数处理；第二，仅仅两层神经元限制了其复杂性；第三，感知机只能区分线性可分的对象，即可以用直线或超平面分开的对象。简单的感知机学习规则显然是不够的，因此需要更强大的学习算法。

反向传播算法（以下简称 BP 算法）是监督学习的一种方法，它使用梯度下降最快的方法实现对神经元连接权重的修改，对权重的修改旨在实现最小均方误差。起初这些神经元的权重和偏置被赋予随机值，即初始化过程，随后当一个训练样本被输入神经网络，可能产生的误差将反向传播以修改这些权重和偏置。即操作信号经过神经网络的预测值与实际值的差值作为反馈信号反过来改变神经元的权重和阈值，权重和阈值的不断改变使得输出值和实际值不断接近。

### 2.2.3 BP 神经网络

顾名思义，BP 神经网络就是利用了误差逆传播算法的神经网络，下面用数学推导描述图 2.3 所示的 BP 前馈神经网络。

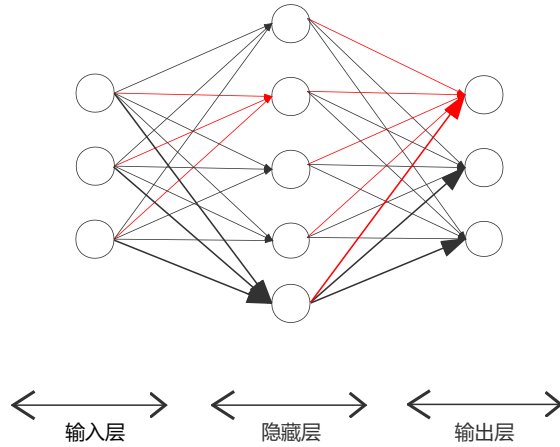


图 2.3 三层 BP 神经网络模型

BP 算法改变权重使基于梯度的，方向导数计算一个函数沿某方向的变化率，梯度值则是该点上方向导数最大值，沿梯方向就是沿着变化最快的方向。BP 神经网络学习算法的指导思想是沿着函数负梯度方向改变神经网络的权重，这个方向是误差函数下降最快的方向。

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \mathbf{g}_k \quad (2.2)$$

上述公式中， $\mathbf{x}_k$  表示当前层神经元的权重矩阵， $\mathbf{x}_{k+1}$  表示下一层神经元的权重， $\mathbf{g}_k$  表示当前误差函数的梯度， $\eta$  表示学习率，该公式表述了权重的变化过程。下面以三层 BP 神经网络模型为例，详细描述 BP 算法，设输入神经元为  $x_i$ ，隐藏层神经元为  $y_j$ ，输出层神经元为  $z_l$ ，输入层与隐藏层之间的权重为  $w_{ji}$ ，隐藏层与输出层之间的权重为  $v_{lj}$ ，样本对应的真值为  $t_l$ ， $f(\bullet)$  是激活函数，推导过程如下：

在操作信号的正向传递中，隐藏层的输出可以表示为：

$$y_j = f(\text{net}_j) = f\left(\sum_i w_{ji} x_i - \theta_j\right) \quad (2.3)$$

其中隐藏层神经元激活函数输入  $\text{net}_j = \sum_i w_{ji} x_i - \theta_j$

输出层的输出可以表示为

$$z_l = f(\text{net}_l) = f\left(\sum_j v_{lj} y_j - \theta_l\right) \quad (2.4)$$

其中输出层神经元激活函数输入  $\text{net}_l = \sum_j v_{lj} y_j - \theta_l$

接下来计算误差  $E$ :

$$\begin{aligned} E &= \frac{1}{2} \sum_l (t_l - z_l)^2 = \frac{1}{2} \sum_l (t_l - f(\sum_j v_{lj} y_j - \theta_l))^2 \\ &= \frac{1}{2} \sum_l (t_l - f(\sum_j v_{lj} f(\sum_i w_{ji} x_i - \theta_j) - \theta_l))^2 \end{aligned} \quad (2.5)$$

上面提到，BP 算法利用误差的梯度反向调整神经网络的阈值，下面推导误差信号的反向传播，权重的改变：

输出层：

$$\frac{\partial E}{\partial v_{lj}} = \sum_{k=1}^n \frac{\partial E}{\partial z_k} \cdot \frac{\partial z_k}{\partial v_{lj}} = \frac{\partial E}{\partial z_l} \cdot \frac{\partial z_l}{\partial v_{lj}} \quad (2.6)$$

误差函数  $E$  包含多个  $z_k$  但其中只有一个  $z_l$  与  $v_{lj}$  相关，而且所有的  $z_k$  相互独立，其中

$$\frac{\partial E}{\partial z_l} = \frac{1}{2} \cdot \sum_k \left[ -2(t_k - z_k) \cdot \frac{\partial z_k}{\partial z_l} \right] = -(t_l - z_l) \quad (2.7)$$

$$\frac{\partial z_l}{\partial v_{lj}} = \frac{\partial z_l}{\partial net_l} \cdot \frac{\partial net_l}{\partial v_{lj}} = f'(net_l) \cdot y_j \quad (2.8)$$

将(2.7)，(2.8)代入(2.6)得

$$\frac{\partial E}{\partial v_{lj}} = -(t_l - z_l) \cdot f'(net_l) \cdot y_j \quad (2.9)$$

设输入层节点误差为

$$\delta_l = (t_l - z_l) \cdot f'(net_l) \quad (2.10)$$

那么

$$\frac{\partial E}{\partial v_{lj}} = -\delta_l \cdot y_j \quad (2.11)$$

隐藏层

$$\frac{\partial E}{\partial w_{ji}} = \sum_l \sum_j \frac{\partial E}{\partial z_l} \cdot \frac{\partial z_l}{\partial y_j} \cdot \frac{\partial y_j}{\partial w_{ji}} \quad (2.12)$$

误差函数  $E$  包含多个  $z_l$  且与每个  $z_l$  相关，它对应一个  $y_j$ ，只与部分  $w_{ji}$  相关，将(2.7)代入(2.12)得

$$\frac{\partial E}{\partial w_{ji}} = -\sum_l (t_l - z_l) \cdot f'(net_l) \cdot v_{lj} \cdot f'(net_j) \cdot x_j = -\sum_l \delta_l v_{lj} f'(net_j) \cdot x_j \quad (2.13)$$

设隐藏层节点误差为

$$\delta'_j = f'(net_j) \cdot \sum_l \delta_l v_{lj} \quad (2.14)$$

那么

$$\frac{\partial E}{\partial w_{ji}} = -\delta'_j \cdot x_i \quad (2.15)$$

由于权重的修正值  $\Delta v_{lj}$  和  $\Delta w_{ji}$  和误差函数成正比并且沿着误差函数梯度下降，因此修正最后两层神经元之间的权重值的表达式为

$$\Delta v_{lj} = -\eta \frac{\partial E}{\partial v_{lj}} = \eta \delta_l y_j \quad (2.16)$$

其中  $\eta$  代表学习率，表示隐藏层和输出层之间的权重修改公式为

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}} = \eta \delta'_j x_i \quad (2.17)$$

其中  $\eta$  表示学习率， $\delta'_j$  表示输出节点  $z_l$  的误差。BP 算法的思想不仅应用于多层前馈神经网络，也被应用于很多其他形式的神经网络中。

## 2.2.4 神经网络评估

BP 神经网络是迄今为止使用最多的人工神经网络模型，BP 算法同样也在被应用到各种神经网络的学习算法中，同时，BP 神经网络还具有各种变化形式也都被广泛的使用，但是这并不代表 BP 神经网络是完美的，在训练过程中，该神经网络仍然存在一些不足之处，比如训练可能落入局部梯度最小点（局部最优），或者收敛速度慢延长训练时间，还存在训练次数与网络性能负相关的情况，即增加训练样本可能较大的影响原神经网络效果。研究人员通常会做的改进也是针对这些现存的问题的，第一是增加神经网络的训练速度，二是提高神经网络的准确性，最后要避免陷入局部最小点。

## 2.3 本章小结

本章讲述了机器学习的相关知识，主要讨论了其中的分类问题，并讲述了神经元的结构和神经网络的分类原理，最后系统讲述了 BP 神经网络的原理与数学推导。

## 第3章 预处理和特征提取

本章介绍数据集预处理和特征提取过程，它们是字符图片输入字母识别模型前重要步骤，其中图片预处理是进行形态学处理和字符精准分割的必要条件；特征提取进一步浓缩输入图片的有效且具有较高区分度的信息，优秀的特征提取算法能使分类模型更好的发挥性能。图 3.1 为输入分类模型前的预处理和特征提取过程。

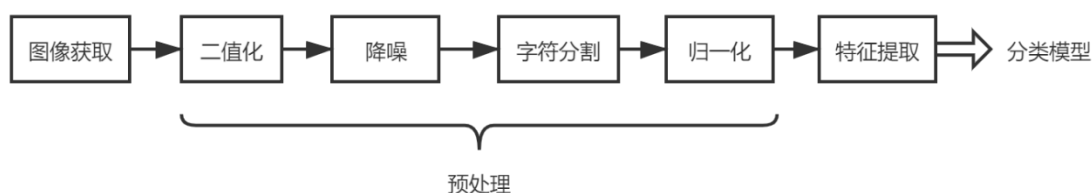


图 3.1 预处理与特征提取

### 3.1 EMNIST 数据集

MNIST 数据集应用十分广泛，可以说是机器学习，分类器，计算机视觉系统用来检测其效果最常用的数据集。本文选择的是 EMNIST 数据集，该数据集是对 MNIST 手写数字数据集进行了拓展，增加了手写字母部分。其特点在于将手写字母进行多种分类，根据不同的需求选择不同的分类方式，图 3.2 为 EMNIST 数据集的分类情况。

第一种为均衡分类，即每个字母数据集和训练集的字符个数相同，但是对大小写较相似的字母模糊处理，也就是做了归并处理；第二类为归并分类，归并方式与均衡分类方式相同，区别在于其每个字符按书写频率不同出现的次数不同；第三类为简单的类别分类，它不对大小写进行模糊处理，但字母出现次数不同；第四类为字母分类，这是一种极端的方式，将所用的字母大小写全部归并然后做均匀处理。



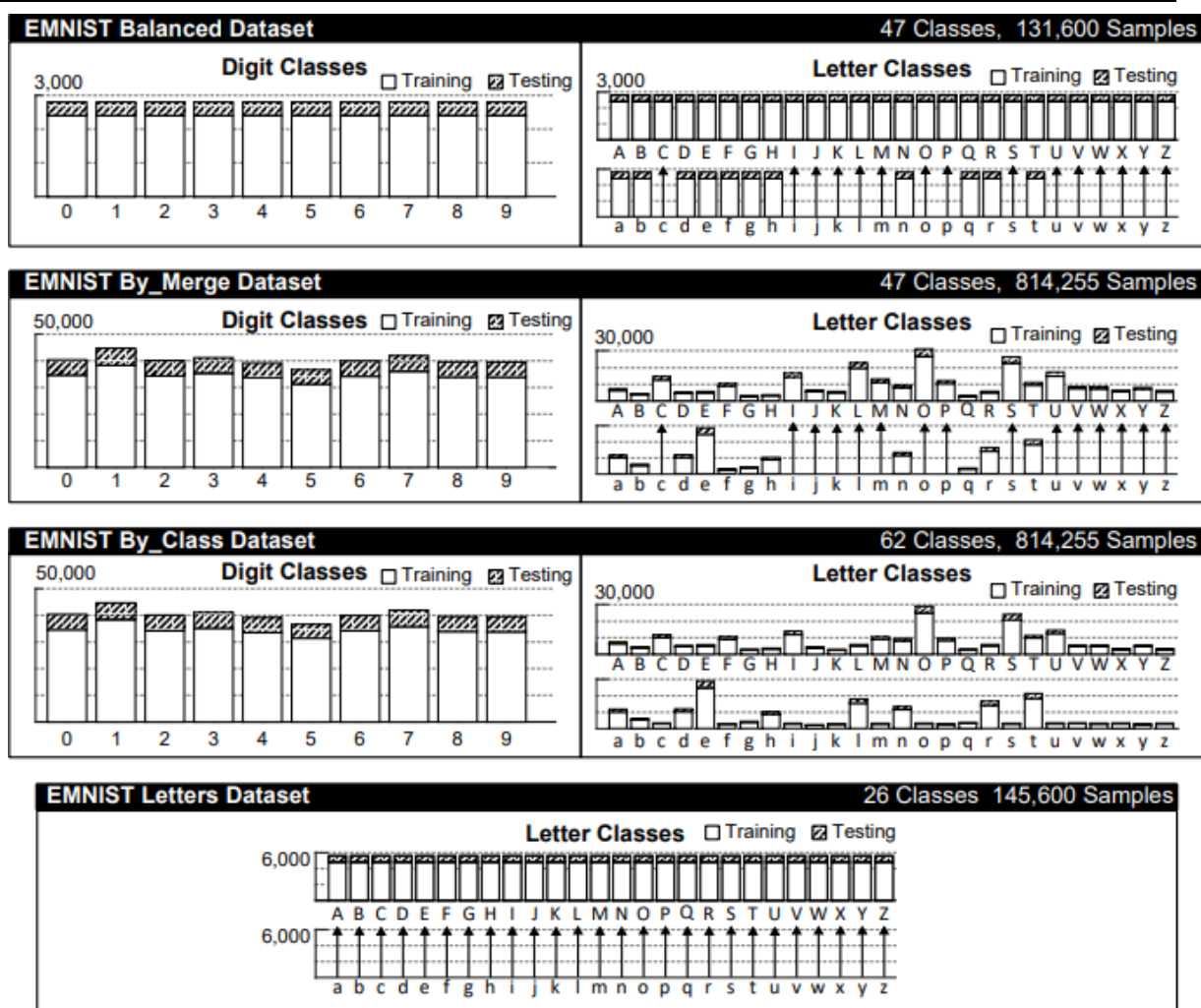


图 3.2 EMNIST 数据集分类方式

该数据集的每个分组中都有数以十万计的字符图片和字符标签，为了减少神经网络的训练时间，同时为了防止过拟合现象的发生，本次实验从该数据集中选取 8000 个样本用于实验，其中 7000 个样本作为训练集，1000 个样本作为测试集。

## 3.2 预处理

英文手写字母识别首先要获取字符文件，例如本设计中采用的手写字母图片的形式，该文件通常以扫描件的形式作为系统的输入，当含有字符的文件被扫描出，或者说原始文件被输入系统，它需要进行一系列的预处理才能被训练好的模型（将在第四章讨论）处理和识别。预处理过程首先是为了获得与数据集相同形式的图片序列，再者，它能为特征提取过程提供归一化的输入，二者都能提高手写字母识别的精确度百分比。最后，该过程去除了图片中的无用信息，显著降低图片的冗余度。预处理过程包括图片的基本预处理，字符提取和归一化处理。

### 3.2.1 灰度化与二值化

手写字符识别的原理是拾取字符的形状和骨架，因此图像的颜色与灰度等级对于该过程为无用信息。通常的扫描文件与获取的字符图片为 RGB 格式，即每个像素由 R，G，B 三原色合成，所以首先需要经过灰度化处理，灰度化处理是对每个像素做公式(3.1)的灰度值变换。

$$f(x, y) = 0.2989 * R + 0.5870 * G + 0.1140 * B \quad (3.1)$$

另外，由于识别过程仅对字符的形状敏感，灰度等级作为冗余信息可被投影到 0, 1 空间，即二值化过程，从某种程度上讲二值化过程也是降维的过程，能提高图像对比度，显著提高图片处理速度，二值化也是图像做形态学处理的前提，例如字符连续检测，图像膨胀等。二值化的原理是将 0~255 的灰度值映射为 0, 1 两个值，这种映射通常需要一个阈值，当灰度值大于该阈值灰度值设置为 1，反之设置为 0，如公式(3.2)所示， $T$  表示图像的阈值，这个阈值有时是可以变化的，因此二值化算法有多种，需要根据不同的需求选择二值化算法。

$$g(x, y) = \begin{cases} 1, & f(x, y) \geq T \\ 0, & \text{otherwise} \end{cases} \quad (3.2)$$

阈值的选择也会对处理效果产生很明显影响，本次设计针对的使文本文档，图像中大部分像素被期望为白的背景，少部分是黑色的笔迹，图 3.3 显示的就是情形，第一个波峰显示的像素数目较少的是较暗的笔迹，第二个波峰显示的是像素数目较多的较亮的背景，理论上讲在这种情况下，选择两个波峰之间的波谷作为阈值能够很好的区分字迹和背景。

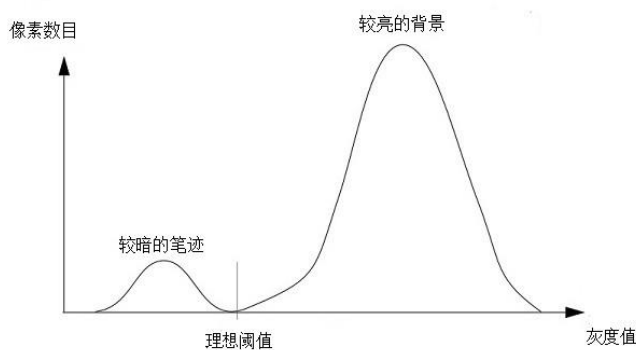


图 3.3 浅色背景深色字迹图像灰度统计

目前较成熟的二值化算法均可分为全局阈值法和局部阈值法两种。

#### 1 全局阈值法

全局阈值法包括固定阈值方式（公式(3.2)）和 Otsu 法即最大类间法。该算法返回一个全局阈值，该阈值是通过最小化类别间灰度值方差来确定的，它将像素分为前景和背景两

种，当图 3.3 那样有两个明显的波峰时，最大类间法确定的阈值就接近波谷的位置。

## 2 局部阈值法

因为图像经常会出现亮度在不同区域有明显不同的现象，为了解决二值化带来的图像存在全黑或全白区域问题，可以自适应的在颜色深浅不同的区域设定不同的阈值。

显然，手写字符图像是典型的前景背景模型，用基于灰度统计的最大类间算法最为合适。图 3.4 所示的是用最大类间法二值化的手写字符图像处理结果。

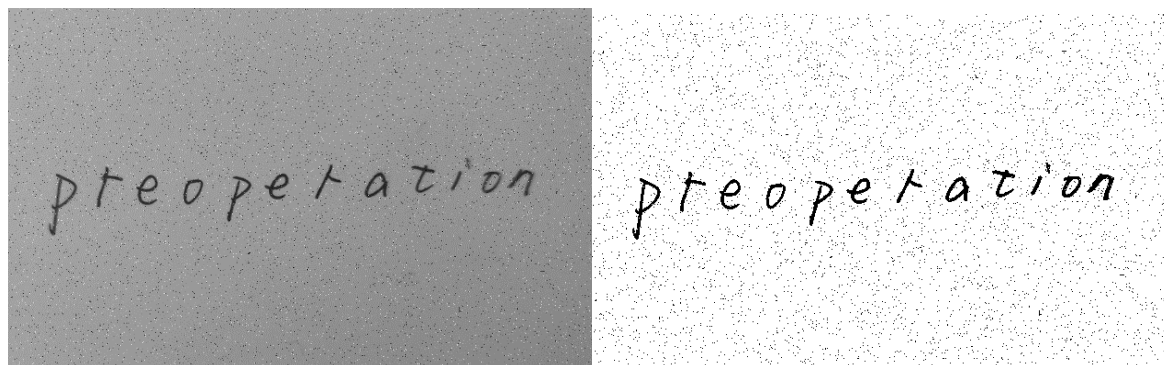


图 3.4 手写字符的二值化

### 3.2.2 平滑去噪

图像的二值化是基于阈值的，本次设计使用的最大类间法是一种全局阈值二值化方法，当阈值选定后，灰度小于该阈值的设置为字迹（黑色），在这个过程可能存在一些较小的暗区域被误当成字迹被转换成了黑色。此外，更重要的是，手写字符图片在扫描的时候不可避免会引入很多噪声，同样的在二值化过程中部分噪声会被转化黑色误当作字符。如图 3.4 所示的二值化图像不能进行下一步的处理，必须做去噪。噪声像素区域的存在会影响连续区域检测，进而使字符个数的判断出现错误，这无疑使得图像分割的结果以及最终的识别结果出现错误。

二值化后的图像有“非黑即白”的特征，往往噪声都以离散的小像素区域形式呈现，所以只要去除像素数明显小于字符本身像素数的小区域就能达到简单的去噪处理。图 3.5 所示为去噪效果，MATLAB 中形态学处理对象是白色区域（只对白色敏感），故在降噪前做了反色处理。



图 3.5 手写字符去噪

此外，不难发现去噪后的字符中仍含有黑色噪点，这些内部噪点可以通过膨胀的方法处理，膨胀的原理是基于结构单元将白色像素附近的黑色像素“同化”，其中结构单元用来控制膨胀的强度。膨胀最简单的应用之一是连接孔隙，图 3.6 显示了这种连接效果。膨胀除了能去除字符内部的噪声点之外，还能对图像做平滑处理，这同样会增加识别结果的精确度。



图 3.6 字符内黑色噪点与膨胀处理

### 3.2.3 字符分割

手写字符识别模型的输入是单个字符的特征，即特征提取是针对单字符的，这意味着必须将手写字符图片中的内容分割成单个字符，也就是字符分割。字符分割首先要对连续字符进行检测，MATLAB 提供了相关函数可框选连续的像素区域从而进行分割。使用 MATLAB 进行膨胀后的图像字符分割的具体步骤如下：

- 1) 使用 `bwlabel` 函数确定二值图像中连续区域数目  $N$ ，并用罗马数字 1- $N$  对每个区域编号并生成新的带编号的图像矩阵  $L$ ；
- 2) 对图像  $L$  使用 `rectangle` 函数进行连续区域框选，这个过程中框取的可能不是完整的字母，比如  $i$ ,  $j$  等非完全连通字母，会被分成两个部分进行框取，如图 3.7(a)所示，字母  $i$  被分成了两个部分，另外，这时候的通过 1) 计算的  $N$  并不是图像中真

实字符数。

- 3) 上一步中误将 i, j 的点当作另一个字符会造成错误，因此可以采用不框取占用像素数较少的点的方式，如图 3.7(b)所示。
- 4) 此时 L 表示的就是图像中字符的真实数目，但不能作为分割的元素，因为部分字母不完整，只含有下半部分。为了获取完整的单个字符，将每个矩形框向上扩展字符的 50%，再与原来的图像对比，如果矩形框包含的字符与原来的矩形框不同，则将取新的矩形框作为图像分割的元素进行分割。

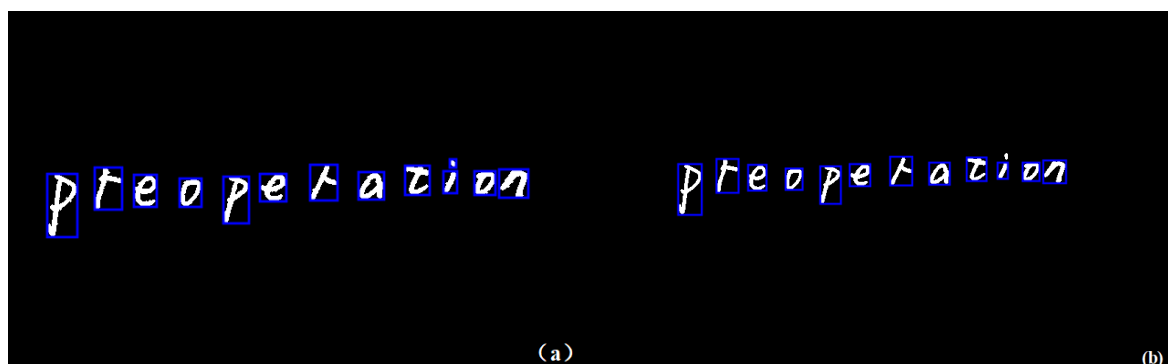


图 3.7 (a)框选所有连续区域；(b)不框选不连接的点区域

### 3.2.4 归一化处理

需要指出，手写字母识别模型是通过训练集学习的，特别是特征提取（将在下一节讲述）环节，对数据集的形式依赖程度大。本次设计使用的是 EMNIST 数据集，该数据集的字母数据是 784 维的向量形式，转化成 28\*28 的矩阵图片后带有黑色边框且做了笔迹等宽处理，为了提高模型的泛化能力，应对新的字符图像做归一化处理，保证每次输入系统的数据形式与训练模型时的输入数据类型保持一致。归一化就是对被分割的各个子图像做形状上的变化，这个过程中使用了图像插值算法形态学的骨架提取和膨胀算法，然后对每个图像做边框化处理。如图 3.8(a)为骨架提取结果，(b)为字符归一化的字符序列。

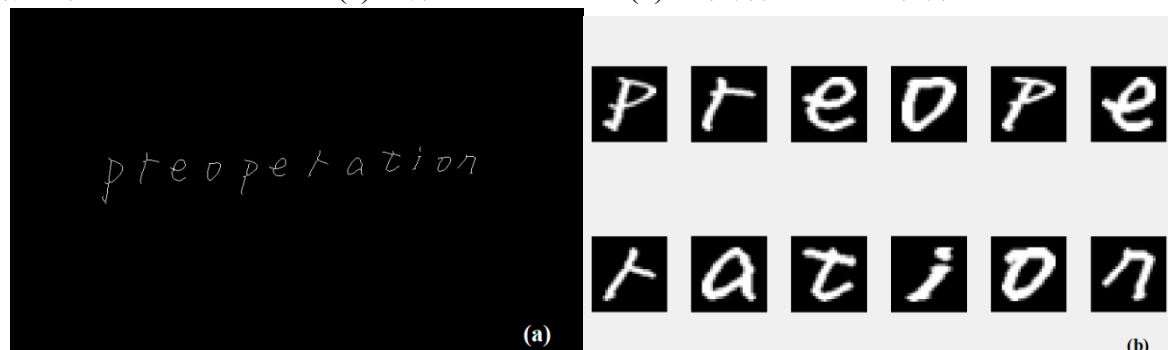


图 3.8 (a)骨架提取；(b)归一化处理

### 3.3 特征提取

在二值图像的识别过程中，特征提取不是必要的，直接将图像列化成一维向量作为模型的输入对于较少分类（十分类以下）的问题来说也能达到一定的区分效果。但这种极端的方式的缺点在于不加处理的原始数据的计算量过大。此外，如此大的数据量并不会增加识别的精度，而且大概率会降低精度。逐像素法提取特征性能差，对于手写字母这种多分类问题分类效果不好，因此，特征提取在该设计中是必不可少的。特征提取又是降维的过程，通过该过程可将原始数据缩减为更易于管理的数据以进行处理。大型数据集的一个特点是需要大量计算资源来处理如此多的变量，特征提取可有效减少必须处理的数据量，同时仍能准确，完整地描述原始数据集。

当需要减少处理数据所需的资源数量而又不丢失重要或相关信息时，特征提取过程就十分有用。特征提取还可以减少给定数据的冗余量，此外，数据的减少提高了机器学习过程中学习和泛化步骤的速度。当然，要想提高学习和泛化能力，合适且优秀的特征提取方法是十分重要的，本次设计主要使用了区域密度特征，边缘距离特征和投影统计特征。下面分别进行介绍其原理并对其效果加以分析。

#### 3.3.1 区域密度特征

区域密度特征是将图片用网格分成  $6 \times 6$  的小区间，分别提取每个区间的特征，最终形成一个 36 维的列向量，如图 3.9，这种方法也叫粗网格提取法。

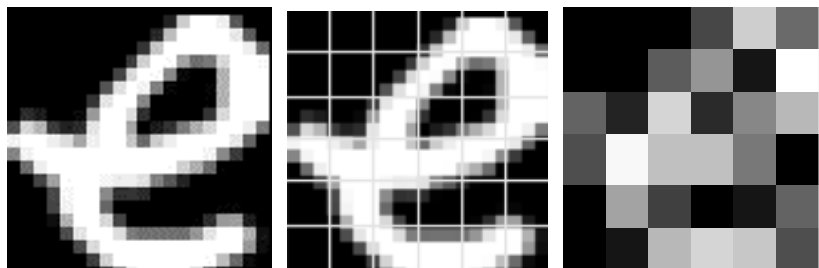


图 3.9 区域密度提取特征

这种特征提取的原理是统计每个小区域内字符像素占当前单元格所有像素的比例，这种方式比较直观的显示出了字符的书写范围和形状，图 3.9 分别显示了原始图片，网格提取和密度值计算过程。这种特征提取方式的缺点在于方法过于简单单一，对于多分类模型的效果不是很友好。

#### 3.3.2 边缘距离特征

使用边缘距离特征作为神经网络的输入特征时，获取一个尺寸为  $28 \times 28$  的字符的图像，因此图像中存在的像素总数为 784，将每个像素值都看作变量，因此一个特征向量包

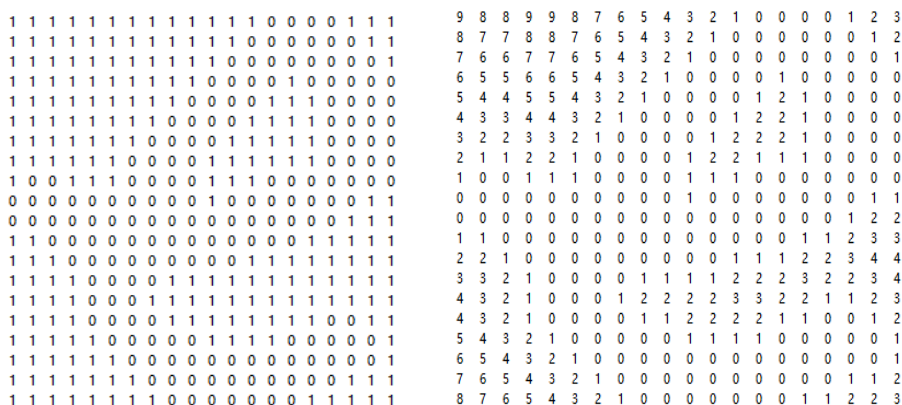


图 3.10 边缘距离提取特征

### 3.3.3 投影统计直方图特征



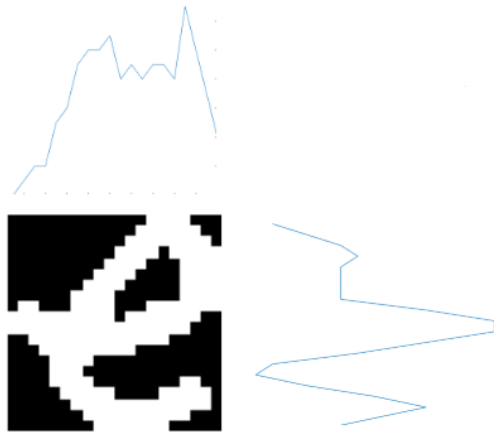


图 3.11 统计投影特征原理

3.3.4组合特征

除上述特征外，还可以从图像中获取很多种特征，只要能代表图像的特征且具有一定的区分度就可以用于字符分类，例如骨架图像的交叉点数，尖点数，膨胀图像的区域数。和上面介绍的特征类似，这些特征都有一个共同的缺点：存在两个或多个字符之间的特征相似度高，这样不利于区分它们。所以一种高效的特征提取方式是组合多个相关度低的特征形成一个新的特征，这些特征可以很简单，当然最好维度也较低，因为特征提取是要对训练集中的每一个元素进行处理的，训练集有大量的数据，维度高的特征组合会增加相当多的计算量。选择简单的特征相组合可以相互弥补区分度较差的部分，也就是互补，互补性的好坏取决于这些特征之间的相关性，即相关性越低的特征组合互补性越强，这种组合的区分能力越高。

表 3.1 各特征提取效果

提取方式	维度	识别率(%)	训练耗时(s)	识别率低字母
区域密度特征	36	76.90	29	g,i,j,q(70 以下)
投影统计特征	56	63.60	32	k,n,q,r,s(50 以下)
边缘距离特征	784	45.10	134	a,c,e,o,u,v(30 以下)
组合特征	93	86.20	37	r(70 以下)

表 3.1 所示的是以小写字母为例，使用各种特征提取方法的维度，最佳识别率，最低训练消耗时间和这种情形下识别率较低的字母。其中每次训练除了神经网络结果改变（最佳识别率对应的结构），其余条件均不变：每次训练集选择 8000 个字符图片，测试集 1000 个字符图片，最高迭代次数 1000，学习率 0.1。



可以看出组合特征的识别效果最好，组合特征选择三种属性作为最终特征进行分类，包括区域密度特征（主），投影统计特征和一个连续区域数特征（提高 i, j 的识别率）。图 3.12 显示了使用组合特征时各字母的识别准确率。

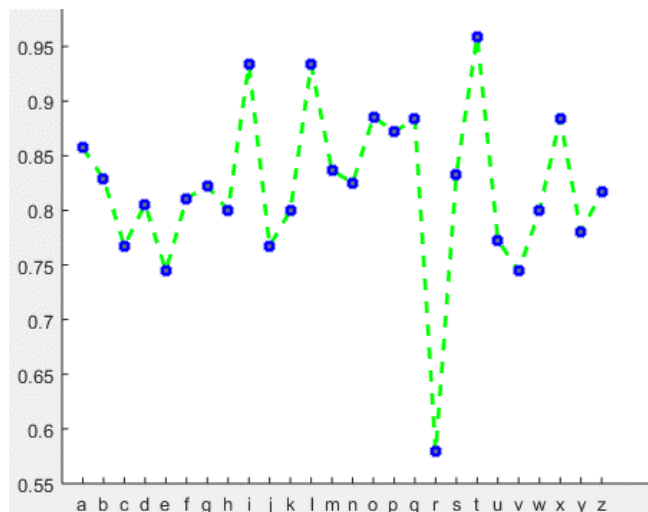


图 3.12 组合特征各字母识别率

### 3.4 本章小结

本章系统介绍了字符图片输入系统前需要做的准备工作，图像的二值化，去噪，归一化等预处理操作和特征提取操作。这两个步骤前者直接决定能否识别出正确的字符，保证输入字符识别系统的图像与数据集中的图像尽可能相似，后者选择合适的特征作为区分的依据。本次设计使用一种新的组合特征提取方式，经多次调整参数，识别率达到了较高的水平，也是使用在最终识别系统的特征提取方式。

## 第4章 手写字母识别系统

在上一章研究了图像输入系统前的准备工作，预处理和特征提取，此时数据集中的图像数据已经不再是一张图像或者一个图像直接转换成的列向量，它已经转化成较完整的描述数据集原始数据的特征向量，神经网络从这些字符特征和字符标签中学习两者的联系，形成手写字母识别模型。本章将介绍该模型的学习过程以及一般文本文件如何通过分类模型形成可读性强的计算机可识别的文本。

### 4.1 识别系统流程

手写数字从文本图像的输入到识别成文本文件的流程如图 4.1 所示。

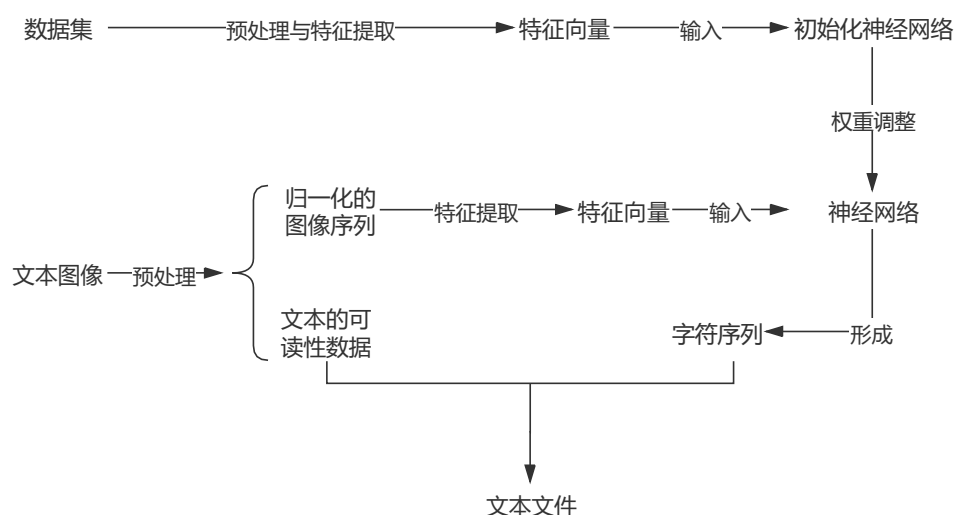


图 4.1 首写字母识别全框图

图中对数据集的预处理比较简单，因为每个数据集图像中只含有一个字符，只需要简单的二值化和去噪就能进行特征提取形成特征向量，然后连同标签一同输入神经网络。此时的神经网络仍是一个刚初始化的“生”的神经网络，因为初始化的神经网络的神经元权重和阈值都是随机的，经过训练集的训练才能形成一个“成熟”的 BP 神经网络。至此，神经网络识别模型已经形成，只需将其导出，此后的识别便只需要调用它，不用再花大量的时间训练。

与数据集的预处理不同，真正的手写字符图像的预处理并非如此简单，因为现实中的手写字符图像需要进行多行分割和字符的单词合并，以将识别出的分离的字符合并成一个可读的文本文件。

## 4.2 神经网络训练

本次实验使用 MATLAB 设计程序，其中 BP 神经网络的训练部分使用的是 MATLAB 集成好的函数，实验中不需要进行 BP 神经网络的模型建立，只需要进行调用，如此一来就会减少很多工作量。实验使用不同的数据集和不同的数据集处理方式对神经网络进行训练，其效果也存在很大的差异，因此，对数据集的分析是必须的。

### 4.2.1 数据集的选择与处理

本次实验选择的是 EMNIST 数据集，为了达到更高的识别精度，本次实验对各种分类的数据集分别做了不同的处理，以从中选择最佳的处理方式用于识别模型。

表 4.1 EMNIST 数据集分类方式

分类方式	类别数	大小写模糊	字母频率	适用性
Balanced	37	部分模糊	均衡	较差
By_Merge	37	部分模糊	书写频率	较差
By_Class	52	无	书写频率	可用
Letters	26	全部模糊	均衡	不可用

表 4.1 对这些分类做了分析，可以看到上述数据并没有包含数据量的统计，因为每个数据集中的样本数目都是以十万计的，实际处理中并不会全部使用，因为训练集样本过多可能会导致训练过程出现过拟合现象。此外，从表中可以看出均衡方式和归并方式都对字母大小写做了部分模糊，字母分类方式则直接全部大小写归并，无法对其进行分离，这对本次手写字母识别系统特征提取算法不友好，故采用无大小写模糊的按类别方式。特征提取采用上文的组合特征提取。

表 4.2 By\_Class 组处理方式

处理方式	样本总数	每组样本数	平均识别率	特点
不处理（按频率）	20800	不定	80.20	字频高正确率高
均衡处理	20800	800	86.10	正确率较均匀

通过大量实验得出如果对按类别分类的数据集不加处理的选择出一些数据进行训练，由于每个字符的出现频率不同导致选择的样本中单词常用字母比单词不常用字母训练次数多，这样一来，字频高的字母的识别准确度要比字频低的字母高一些，这对按字频书写的段落更友好，尽管平均识别率略低，如表 4.2 所示。做均衡处理是为了减少训练次数对训练结果的影响，尽可能使每个字母对于神经网络是平等的，而且这种处理方式降低字符间出错率方差，这也是最终使用的处理方式。

### 4.2.2 Newff 函数

Newff 函数是 MATLAB 中集成的函数，该函数的输入为神经网络的输入向量集合和输入标签以及神经网络结构，并返回一个前馈误差反向传播的初始化神经网络，该函数会自动调用 `initnw` 函数初始化每个神经元的权重和阈值为神经网络的训练做准备，如图 4.2 所示为最终的神经网络结构，该网络的输入层有 93 个神经元，隐藏层有 140 个神经元，输出层为 26 个神经元。训练神经网络使用 `train` 函数，该函数的输入为初始化的神经网络，输入样本集合和输入标签，返回训练好的神经网络。如果不对训练参数加以调整，`train` 函数会自动使用默认参数训练神经网络，例如学习率，最高迭代次数，训练算法和训练目标。一旦训练函数被触发，神经网络就开始被训练，直到达到预定的训练迭代次数或者最小均方误差，或者校验错误连续出现 6 次。本实验设定的学习率为 0.1，训练算法使用速度较快的改进的带弹性的 BP 训练算法。

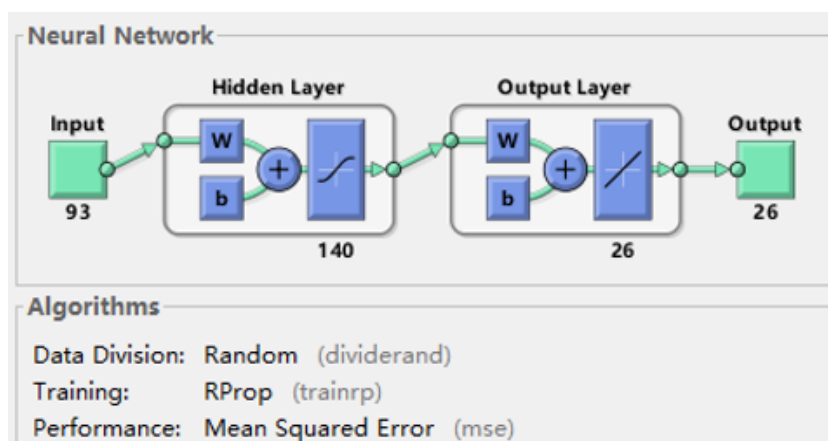


图 4.2 Newff 的神经网络结构与参数

### 4.2.3 参数设定与训练结果

上面说过使用 `train` 函数对初始化的神经网络进行训练并返回一个能较准确的对测试集进行识别的神经网络。而且已经知道，神经网络最终的效果主要取决于其结构，即层数和神经元个数。由于一般的分类问题使用 1 到 2 层神经网络就可以满足需求，本次实验重点研究了一层和两层神经网络中节点数对训练结果的影响。

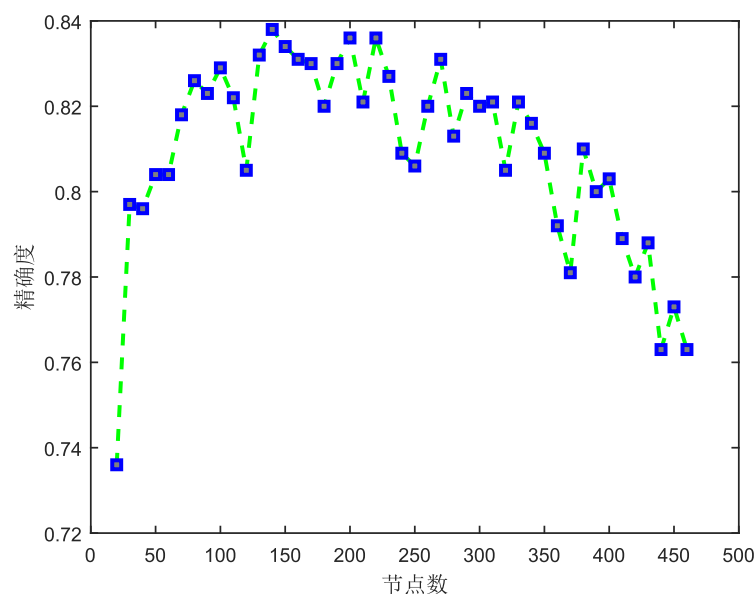


图 4.3 单层神经网络结构对训练效果的影响

图 4.3 为单层神经网络的训练结果，显然单层的情况下，神经网络在结点数为 148 时其识别平均精度最大。当神经元个数小于 100 时识别率偏低，而且神经元个数超过 300 时精确度也显著下降。由此看来，神经网络的效果与复杂度并不是正相关的。单层的神经网络如此，多层的神经网络亦然，如图 4.4 所示为双层神经网络的结构与精确度的关系，很明显，在两层神经元节点数均在 70 左右时达到最好的效果，再继续增加节点数反而会使准确度降低。

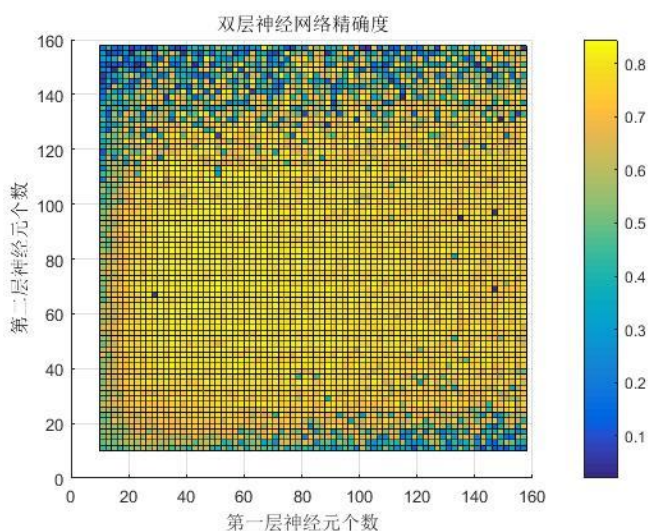


图 4.4 双层神经网络结构对训练效果的影响

## 4.3 文本图像处理

本次设计的神经网络训练是基于单个字母的训练集进行的，也就是说神经网络模型只对单个字母敏感。本次设计需要对复杂的文字段落进行识别，为了增加文本可读性，本设计加入了段落行提取和字符分组处理。

### 4.3.1 字符分组

所谓字符分组，就是将模型识别出的一串字母分组形成单词以增加其可读性。在数据集的预处理部分已经讲到，对手写字符框选时需要对图片做连续检测，检测后生成的文本图片中每个字符都用一种标号表示。字符框选时获取能包含每个字符的最小矩形框，同样，MATLAB 可以获取更多连续字符的区域特性。在字符分组实验中，考虑到单词字母间距离小于单词间距离。可以获取各个字母的中心坐标，然后对中心横坐标进行差分计算得出任意两个字符间的距离。经过大量实验验证得取 1.5 倍平均间距作为是否插入空格的阈值能基本准确的对字符进行分组。

### 4.3.2 行提取

同样的，由输入系统用于识别的手写字符图像复杂性不同，必须引入适用于多行字符的识别系统。本次实验使用基于统计的方法对图像进行行分割，使图像中的单行字符序列被分离然后逐行进行识别，最后将识别出的结果组成，如图 4.5 所示。

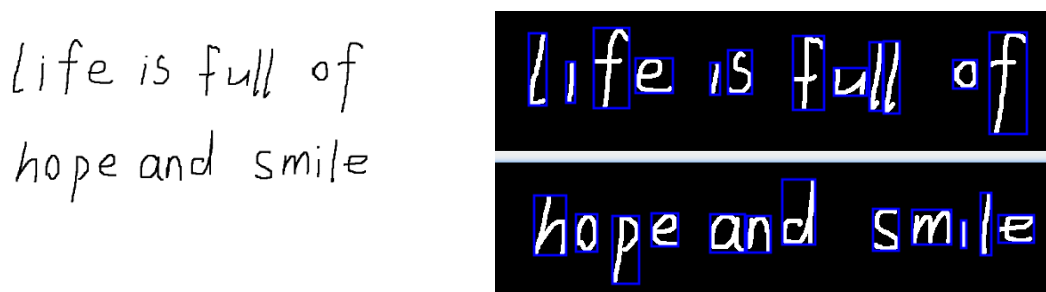


图 4.5 行分割效果

行提取的方法将文本图片进行行投影计算每行的字符元素个数，从而形成一个列向量。绘制该列向量的坐标图可以发现图形呈现明显的波峰波谷式，波峰显示的是字符区域，波谷显示的是字符行之间的空白区域，如图 4.6，因此只需要以波谷作为行与行之间的分割线就能完美的分隔开。

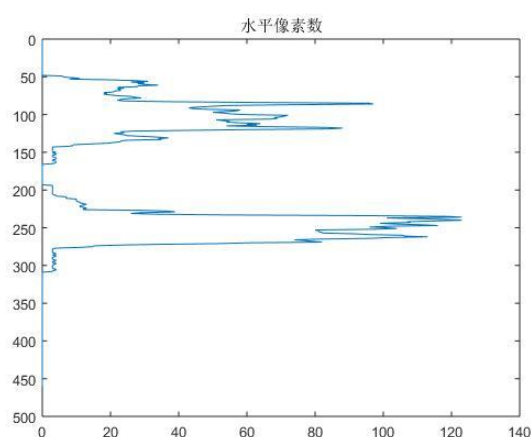


图 4.6 行投影结果

## 4.4 实验结果

### 4.4.1 GUI 设计

该设计最终以 GUI（Graphical User Interface）用户界面呈现，设计 GUI 界面的用意是使该识别系统的输入输出更加直观，对原始图片的识别更加方便。编辑页面如图 4.7 所示，其中坐标图用于显示预处理后的文本图像，文本框用于显示识别结构。两个按钮分别触发选择原始图像和开始识别指令。

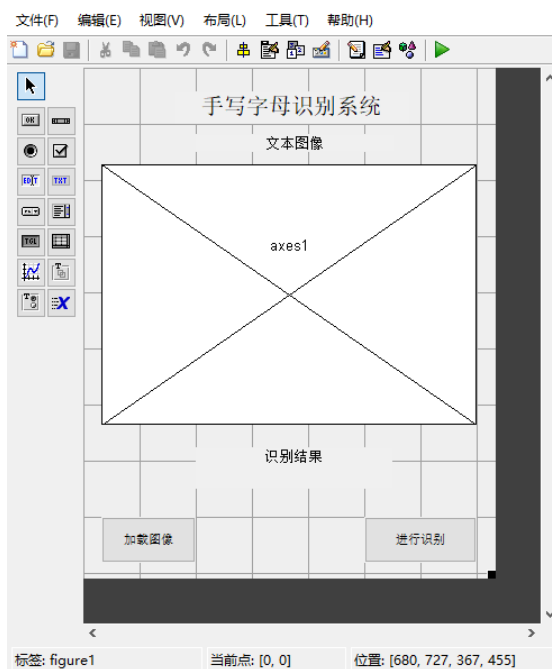


图 4.7 GUI 编辑窗口

GUI 图形界面创建好之后，需要对每个控件编写代码，例如 Axes 控件的生成函数和 Push Button 的回调函数，它们分别控制控件的初始化形式和点击执行的命令，设计好的

GUI 界面如图 4.8 所示。



图 4.8 系统 GUI 界面

点击“加载图像”按钮选择原始图像，此时原始字符图像被显示在文本图像的图像框里，再点击“进行识别”按钮，标记原始图像中的字符，识别处理之后的结果在“识别结果”文本框显示。如图 4.9 为识别结果。

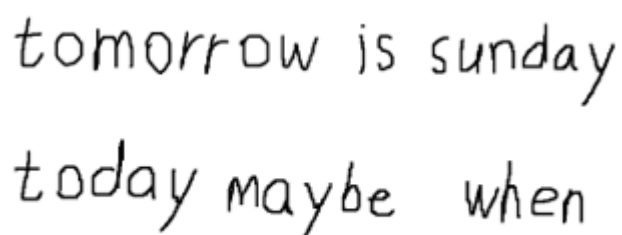


图 4.9 手写字母识别结果

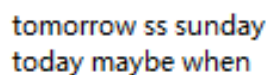


#### 4.4.2 系统评估

该系统在识别手写字母的过程中，能以较高的正确率识别出手写字母文本图像，图 4.9 的识别结果与文本图像完全一致。但该系统也在识别时也会出现一些错误，如图 4.10 所示的多行文字中，系统将第一行的字母 *i* 错误的判断为 *s*，出现错误的主要原因是 BP 神经网络的输入特征区分度还不够。



tomorrow is sunday  
today maybe when



tomorrow ss sunday  
today maybe when

图 4.10 系统识别结果

#### 4.5 本章小结

本章介绍的手写字母识别系统的总流程，首先讲解了数据集的选择和预处理过程，然后介绍了实验中神经网络的结构选择和训练结果，最后阐述用于提高文本可读性的字符分组和针对复杂文本文件的行提取处理并展示了系统的设计及其识别结果。

## 第5章 总结与展望

### 5.1 总结

本文主要讲述了 OCR 的发展现状，然后使用一种模型实现了手写字符识别系统：基于 BP 神经网络的手写字母识别。实验总共分为四个部分，第一部分为数据集选择和处理，第二部分为预处理和特征提取，第三部分为神经网络的训练，第四部分为手写字符文本文件的识别和后处理。

在数据集处理部分将原始的数据集处理并提取出适合训练分类模型的子数据集，子训练集的另一个功能是它不使用原数据集数以十万计的字符数量进行训练，减少了模型训练出现过拟合现象的可能性。文本图像预处理过程是将原始文本图像转化成适用于分类模型的单字符形式，主要的步骤依次是灰度化，二值化，降噪，行提取，字符分割和归一化处理。然后本实验通过多种方式对单字符进行特征提取，包括区域密度特征，边缘距离特征和投影统计特征，最终实验选择的是一种组合的特征提取方式，它结合几种相关度低的特征形成一个新的特征，能增加模型的泛化程度。本设计通过大量实验选择最合适的神经网络结构，使用组合特征提取方式最终达到单字符识别率最高 96.10%，平均 85.10% 的精度。最后，为了使文本的可读性增加，实验还增加了后处理部分，包括字符分组和单行识别结果的组合，较精确地完成了实验的要求。

### 5.2 展望

手写字符识别的重要性不言而喻，本次设计较精确地实现了手写字母识别的功能，并且对于较复杂的文本文件也能完成识别，除了能较精准地预测字符之外，本设计将识别结果的字符序列合并成完整可读的字符文件，保留了原文本的所用信息。

本实验将大小写字母分别训练，所以只能单独地识别小写字母或大写字母，如果有更充分的时间，本实验能将大小写字母包括数字和部分字符加入识别模型。另外，本次设计最大地难点在于特征提取步骤，如果选择合适的特征提取方式能增加文本识别的精确度，下一步考虑使用更多更复杂的特征提取方式。最后，可以通过不同模型对数据集进行学习与训练。

## 参考文献

- [1] R. Plamondon. On-line and off-line handwriting recognition : A comprehensive survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000, 22(1): 63-84.
- [2] S. Mori, C. Suen, K. Yamamoto. Historical review of OCR research and development[J]. Proceedings of the IEEE, 1992, 80(7): 0-1058.
- [3] B. Xu, H. Zhang, Z. Wang, H. Wang, Y. Zhang. Model and Algorithm of BP Neural Network Based on Expanded Multichain Quantum Optimization[J]. Mathematical Problems in Engineering, 2015, 1-11.
- [4] 宋晓茹, 吴雪, 高嵩. 基于深度神经网络的手写数字识别模拟研究[J]. 科学技术与工程, 2019, 19(05): 198-201.
- [5] J. Pradeep, E. Srinivasan, S. Himavathi. Neural network based handwritten character recognition system without feature extraction[C]. International Conference on Computer, IEEE, 2011.
- [6] R. Vaidya, D. Trivedi, S. Satra, P. Pimpale. Handwritten Character Recognition Using Deep-Learning[C]. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, 2018, 772-775.
- [7] B. Koyuncu. Handwritten Character Recognition by using Convolutional Deep Neural Network[C]. International Journal of Engineering Technologies IJET 5, 2019, 1-5.
- [8] 许宜申, 顾济华, 陶智. 基于改进 BP 神经网络的手写字符识别[J]. 通信技术, 2011, 44(5): 106-109.
- [9] Wang Y, Ban X, Chen J, et al. License plate recognition based on SIFT feature[J]. Optik - International Journal for Light and Electron Optics, 2015, 126(21): 2895-2901.
- [10] 黄巧巧. 基于 BP 神经网络的手写数字识别系统研究[D]. : 华中师范大学, 2009.
- [11] 松尾丰, 了解人工智慧的第一本书[M]. 台湾: 经济新朝社, 2016.
- [12] Zhihua Zhou. Ensemble Methods Foundations and Algorithms[M]. CRC Press, 2012.
- [13] P. Kevin. Machine learning: A probabilistic perspective[M]. London: The MIT Press, 2012.
- [14] Christopher M. Bishop, Pattern recognition and machine learning[M]. Springer Science+ Business Media, LLC, 2011.
- [15] C. Liu, K. Nakashima, H. Sako, et al. Handwritten digit recognition: investigation of normalization and feature extraction techniques[J]. Pattern Recognition, 2004, 37(2): 265-279.
- [16] 耿西伟, 张猛, 沈建京. 基于结构特征分类 BP 网络的手写数字识别 [J]. 计算机技术与发展, 2007, 017(001): 130-132.
- [17] C. Rafael, Gonzalez. Digital Image Processing, Third Edition [M]. 北京: 电子工业出版社, 2017.
- [18] Kevin P. Murphy. Machine learning: A probabilistic perspective[M]. London: The MIT Press, 2012.

- [19] 宋曰聪, 胡伟. 手写体数字识别系统中一种新的特征提取方案[J]. 计算机科学, 2007, (9).
- [20] Pal U, Belaïd A, Chaudhuri B. A System for Bangla Handwritten Numeral Recognition[J]. Iete Journal of Research, 2015, 52(1).
- [21] 李新良. 基于模板匹配法的字符识别研究算法[J]. 计算机科学与自动化, 2012, 31(2): 90-93.
- [22] 管琳. 基于 BP 神经网络的脱机手写体数字识别[D].: 中北大学,2007.
- [23] 张魁. 基于遗传-BP 神经网络的手写数字的识别方法[D].: 西安科技大学,2012.
- [24] 魏振义. 基于 BP 神经网络的手写数字识别系统的设计与实现[D].河北科技大学,2013.
- [25] U. Pal, A. Belaïd, B. Chaudhuri. A System for Bangla Handwritten Numeral Recognition[J]. Iete Journal of Research, 2015, 52(1).

## 致谢

本次设计在黄龙庭老师的指导下完成，在这里我郑重地感谢这几个月来黄老师给我推荐的每一本书，提供的每一种方法路径。同样我也特别感动在我有困难的时候老师给予的各种指导帮助和我在有所突破的时候老师给予的肯定。从接到这个题目到中间的不断完善到最终的实现，我遇到诸多问题与难以解决的麻烦，多亏了黄老师不厌其烦地帮我解决。由于今年毕业设计地特殊性，在疫情期间导师与学生无法见面交流，老师还是在百忙之中每周与我们开会讨论问题，挨个解决每个同学地难点疑问，这保障了我们的毕业设计的顺利进行，黄老师严谨的治学态度、高尚的敬业精神，以及平易近人、诲人不倦的作风让我敬佩的同时更是让我受益匪浅。借此完成论文之际，谨向辛勤指导我的导师致以最诚挚的谢意。

另外，在毕业设计完成之际，我的大学本科生涯也即将结束，在这里感谢教我专业知识的所有老师，你们是我未来学习和工作的圆梦人。

最后感谢父母的一路陪伴和同学们的帮助支持，感谢有你们。