

中北大学

硕士学位论文

基于BP神经网络的手写英文字母识别

姓名：高靓

申请学位级别：硕士

专业：应用数学

指导教师：白艳萍

20090416

基于 BP 神经网络的手写英文字母识别

摘要

随着社会的发展，英语作为国际通用语言得到了日益广泛的应用，因此有大量的英文文档整理、查询、统计的工作需要完成，而英文文档识别系统可以轻而易举地完成很多以前难以想象的工作。

本文所设计的英文字母识别系统包括输入图片、归一化、识别三个部分，本文所使用的英文字母样本都是在 Windows 自带的画图板中手写输入获得的。

首先采用 MATLAB 的 `imread.m` 函数读取手写体英文字母图像，因为字母图像还没有达到提取特征和输入识别的要求，本文中采用自定义的归一化算法进行图像预处理，将 28×28 像素图像归一化成 10×14 像素的图像，再用逐像素特征提取法提取字母特征。

预处理后将提取到的字母特征进行编码作为神经网络的输入。抽取输入样本的特征向量，并选取 520 幅样本（20 组）对 BP 神经网络进行训练，然后将另外的 520 幅样本输入到训练好的网络中，就可以根据网络的输出得到识别结果，并且本文的识别系统在 MATLAB 环境下进行了仿真模拟。

本文的研究表明，基于 BP 神经网络的手写英文字母识别系统可以高效的识别与训练样本字体相同的字符，同时具有一定的抗干扰和形变的能力，还可以很好的应用于与字符识别相关的其他领域。

本文给出了英文字母识别的 BP 网络设计过程、方案、及部分源程序。

关键词：图像处理，模式识别，特征提取，手写体字符识别，BP 神经网络

Handwritten English Letters Recognition Based on BP neural network

Abstract

With the development of society, English as an international common language has been increasingly widely used, so a large number of documents in English need to be tidied up, queried and statistic, and the English Document Recognition system can easily complete many unimaginable work before.

Handwriting Recognition System for English letters can be divided into three main parts: input image, normalization and recognition. The English letter samples used in the paper have been got from the paint board belongs to Windows system.

This paper uses the `imread.m` function of MATLAB to read the handwritten letter images, because the letter images have not yet reached the feature extraction request, we need use the self-defined normalization method to preprocess 28×28 pixel images to 10×14 pixel images, then we extract features pixel by pixel.

After pre-processing we encode the gained letters characteristic which used to be input of the neural network. By cramping out the characteristic vector from samples, we select 520 samples(20 groups) to train the BP Neural Network, and then put the other 520 samples into the trained Neural Network, the distinguished digital scripts are obtained according to the output of the neural network, and we carried on simulation in the environment of MATLAB.

This paper shows that Handwritten English Letters recognition system based on BP neural network is able to effectively recognize characters in the same font of training samples. In addition, it has certain capacity of anti-interference and anti-deformation, it can also be used in other fields related with OCR.

The designing principle, process and the program of BP net and some source program will be shown in this paper.

Keywords: Image processing, pattern recognition, character extraction, handwriting character recognition, BP Neural Network

原创性声明

本人郑重声明：所呈交的学位论文，是本人在指导教师的指导下，独立进行研究所取得的成果。除文中已经注明引用的内容外，本论文不包含其他个人或集体已经发表或撰写过的科研成果。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律责任由本人承担。

论文作者签名： 王颖 日期： 2009.6.2

关于学位论文使用权的说明

本人完全了解中北大学有关保管、使用学位论文的规定，其中包括：
①学校有权保管、并向有关部门送交学位论文的原件与复印件；②学校可以采用影印、缩印或其它复制手段复制并保存学位论文；③学校可允许学位论文被查阅或借阅；④学校可以学术交流为目的，复制赠送和交换学位论文；⑤学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

签 名： 王颖 日期： 2009.6.2
导师签名： 王颖 日期： 2009.6.2

第一章 绪论

1.1 目的和意义

随着计算机技术的推广应用,尤其是互联网的日益普及,人类越来越多地依赖计算机获得各种信息,大量的信息处理工作也都转移到计算机上进行。在日常生活和工作中,存在着大量的文字信息处理问题,因而将文字信息快速输入计算机的要求就变得非常迫切,输入速度低已经成为信息进入计算机系统的主要瓶颈,影响着整个系统的效率。因此,要求有一种能将文字信息高速、自动地输入计算机的方法。

目前,文字输入方法主要有键盘输入、手写识别、语音输入和机器自动识别输入等。人工键盘输入方法需要经过一定时间的学习训练才能掌握,手写识别和语音输入虽然简单便捷,但其输入速度不高,对于大量已有的文档资料,采用这些方法要花费大量的人力和时间。因此,能够实现文字信息高速、自动输入的只能是计算机自动识别技术,即光学字符识别技术。光学字符识别技术(OCR)即(Optical Character Recognition)是计算机自动、高速地辨别纸上的文字,并将其转化为可编辑的文本的一项实用技术。它是新一代计算机智能接口的一个重要组成部分,也是模式识别领域的一个重要分支。

手写字母识别一直是热门的研究课题,字母识别要解决数据的采集、处理及选择、输入样本表达的选择、模式识别分类器的选择以及用样本集对识别器的有指导的训练等典型问题。字符识别和处理的信息可分为两大类:一类是文字信息,处理的主要是用各国家、各民族的文字(如:汉字,英文等)书写或印刷的文本信息;另一类是数据信息,主要是由阿拉伯数字及少量特殊符号组成的各种编号和统计数据,如:邮政编码、统计报表、财务报表、银行票据等等^[1]。目前文字识别技术已经广泛应用到了各个领域,它作为计算机智能接口的重要组成部分,在信息处理领域中可以大大提高计算机的使用效率,是办公自动化、新闻出版、计算机翻译等领域中最理想的输入方式,将庞大的文本图像压缩成机器内码可以节省大量的存储空间。

1.2 光学字符识别技术的现状及展望

OCR 由于更容易被人们接受、掌握,近几年来,它同语音识别、行为识别等一起日益成为人们研究的焦点。IBM、Motorola、HP 和 Microsoft 等公司都陆续展开了这方

面的研究，并推出了很多的字符识别产品。字符识别产品的广泛应用推动了模式识别等相关理论的发展^[2]。

自从 1929 年 Dausheck 取得光学字符识别的专利以来，经过近一个世纪的发展，OCR 已经成为当今模式识别领域中最活跃的研究内容之一^[3]。它综合了图像处理、计算机图形学和人工智能等多方面的知识，并在计算机及其相关领域中得到了广泛应用。经过这几十年的发展，OCR 技术取得了很大进展。无论是国内还是国外都有了相对成熟的 OCR 的产品。传统的 OCR 产品主要解决文字自动录入的问题，现代的 OCR 产品实际上已经成为了沟通现实世界和计算机世界的一座桥梁。

我国在 OCR 技术方面的研究工作起步较晚，20 世纪 70 年代才由中科院自动化所的戴汝为院士牵头进行手写体字符识别，并在 1974 年将所研究的手写体数字识别系统应用到邮政信件的自动分拣中^[4,5]。70 年代末开始进行汉字识别的研究，到 1986 年汉字识别的研究进入一个实质性阶段，取得了较大的成果，不少研究单位相继推出了中文 OCR 产品。国家的“863 计划”对 OCR 技术的研究给予了很大的资助，促进了 OCR 的重大成果的取得。目前，联机手写汉字识别技术已经相当成熟，市场上已经出现了多项具有代表性的产品。脱机印刷体识别也有成熟产品出现，识别率也基本能够达到实用的要求。脱机手写汉字识别的研究也取得了很大的进展，小字符集单字识别技术相对更为成熟，例如北京邮电大学在 1998 年开发的金融汉字识别系统在国家 863 测试中获得 99.7% 的识别正确率；同时大字符集汉字识别方法，如余弦整形变换方法等也达到了很高的精度^[6]。文字识别理论与技术在强烈的社会需求推动下产生并不断向前发展。然而可以说还没有哪个识别器达到完美的识别效果。在过去的数十年中，研究者们提出了许许多多的识别方法。字母识别属于模式识别的范畴，是模式识别中的一个具体问题，因而，对于字母识别有着不同于其它模式识别的特殊要求。字母识别除了要求识别精度高和工作可靠外，还要求其识别效率高，识别速度快，这就需要精确的建立数学模型^[7,8,9,10]。

随着技术的不断进步，在可预计的将来下面几种新的识别技术将得到迅速的发展，推动光学字符识别系统不断趋于完善。

(1) 基于模糊技术的识别方法

由于字符本身，特别是手写字符，在字型字体上的变化很大，导致在文字识别中存在着很大的不确定性，因此模糊数学的观念便很自然地引用到了模式识别领域中。

1976 年, A.Rosenfeld 等人提出了一种景物标识松弛算法, 1977 年, R.Jain 等人运用模糊集的理论对复杂图像进行了分析, 实现了运动目标的检测, 同时开始了模糊数学在图像识别中的应用^[11]。

(2)结合语义理解的后处理技术

与识别前的预处理相对, 可以对识别的结果进行后期处理, 提高识别的正确率。分析人类在识别文字的过程, 文字一般都是结合上下文进行理解的, 因此, 计算机在识别文字时也可以在识别单字的基础上, 结合单字的上下文信息对识别结果进行校正, 以单词甚至句子作为一次识别的结果。依据对语言文字的统计信息, 可以确定某个文字后可能跟随的候选字符集, 达到缩小搜索范围、简化计算的目的。这种结合上下文信息识别的技术存在的问题主要集中在如何高效组织候选字符子集、实现候选字符的快速定位上。

(3)多种策略的综合集成

在 OCR 领域, 虽然新的算法思路不断涌现, 但是在一个高效的 OCR 系统中仅仅采用一种识别方式是无法满足现实要求的。单个识别策略的能力有限, 因此采用多种策略实现优势互补, 多角度利用字符的特征信息是 OCR 发展的方向。在这个方向上经常采用的集成策略有投票法、概率法、Dempster-Shafer 法以及行为知识空间法等多种综合方法。以投票法为例, 顾名思义, 每种识别策略都拥有一张选票, 对于同一个字符不同策略各自产生自己的结果即投票, 所有策略投票之后, 得票最多的识别结果就是最终的识别结果。显然, 在这种综合方式中需要大量资源。一方面, 要使各种算法都能够完成; 另一方面, 如果各种算法间的并行性不好, 总的执行时间就要成倍增长。

1.3 光学字符识别方法

文字的模式表达形式和相应的字典形成方法有多种, 每种形式又可以选择不同的特征, 每种特征又有不同的抽取方法, 这就使得判别方法和准则以及所用的数学工具不同, 形成了种类繁多、形式特别的文字识别方法。用于文字识别的模式识别方法可以大致分为统计模式识别、结构模式识别和人工神经网络识别^[12]。

1.3.1 基于统计特征的字符识别技术

一般选取同一类字符中共有的、相对稳定的并且分类性能好的统计特征作为特征向

量。常用的统计特征有字符二维平面的位置特征、字符在水平或者垂直方向投影的直方图特征、矩特征和字符经过频域变换或其它形式变换后的特征等。

大量字符的统计特征经过提取、学习、分类形成关于字符原型知识构成识别字符的模板信息，这些模板信息存储在识别系统中。未知图像在识别时首先提取相同的统计特征，然后与识别系统存储的字符原型知识匹配比较，根据比较结果确定字符最终分类，达到识别的目的。其中，基于字符像素点平面分布的识别算法，因为算法简单、实现方便的特点而成为最常用的匹配方法。这种算法一般先将字符图像归一化为模板的几何维数，然后根据像素点的位置逐个匹配，求出模板和图像的某种距离指标。由于要对每个像素点逐个匹配，造成算法实现计算量大，且对噪音、字符的偏移和变形非常敏感，因此对输入的待识别图像要求较高。Kellner 和 Glauberman 在 1956 年提出用二维图像的投影代替图像点阵信息的思路。二维的图像被一维的投影代替，计算量减少，同时也消除了文字在投影方向偏移的影响，但是对于字符的旋转变形却无能为力。基于统计特征的字符识别技术对于形近字符区分能力弱。因此，通常应用于字符的粗分类。

1.3.2 基于结构特征的字符识别技术

实际应用中，更一般的情况是相近字符的识别和像手写字符那样字型变化很大的字符的识别，因此，发展出了基于结构的字符识别技术。这种技术首先要提取字符的结构。根据识别策略的不同，结构的选择也有所不同。可以选择字根、笔划，也可以选择比笔划更小的笔段。提取出的结构又称作字符的子模式、部件、基元，所有基元按照某种序排列起来就成了字符的特征。

识别过程是在提取基元的基础上，利用形式语言和自动机理论，采取词法分析、树匹配、图匹配和知识推理的方法分析字符结构的过程。常用的结构特征有：笔划的走向、孤立的点，以及是否含有闭合笔画等由于汉字自身具有很强的结构性，利用汉字的结构特点进行识别同样可以达到很好的效果。在对汉字笔划进行统计的基础上，提出了一种利用汉字横、竖、撇、捺的结构信息进行识别的方法-弹性网格特征提取方法。这种方法首先将汉字按照前景色的投影划分成维数不同的矩形区域然后分别提取各个区域中的横、竖、撇、捺的结构信息，最后结合统计的结果进行分类。在试验中达到 94%—99% 的识别率，算法效果令人满意。

传统识别方法中，对输入图像采取统一分辨率进行识别，分辨率由系统的计算和存

储资源先期决定,造成了系统资源的浪费和识别效率的降低。J.Park 分析了传统结构识别方法中的这些弊端,提出主动字符识别(Active Character Recognition)的思想,主动依据输入图像,动态确定结构特征的选取,实验证明,达到了节省资源,加速识别的目的。

与统计识别方法相对应,字符的结构识别技术更加便于区分字型变化大的字符和字型相近的字符。但是由于对结构特征的描述和比较要占用大量的存储和计算资源,因此算法在实现上相对复杂、识别速度慢。

1.3.3 基于神经网络的字符识别技术

迄今为止,人类识别文字的能力远远胜于计算机,无论是变形的字符、模糊的字符,甚至是破损的字符,人类都能很好地识别。基于人工神经网络的字符识别技术目的就是力图通过对人脑功能和结构的模拟来实现字符的高效识别。

经过近几年的迅速发展,人工神经网络在字符识别方面得到了广泛的应用。在 OCR 系统中,人工神经网络主要充当分类器的功能。网络的输入是字符的特征向量,输出是字符的分类结果,即识别结果。由于识别策略的不同和对问题理解水平的限制,输入的特征向量所包含的信息常常是冗余的,甚至是矛盾的。经过反复学习,神经网络可以智能地将特征向量优化,去除冗余、矛盾的信息,强化类间的差异。其次,由于神经网络采用分布式的网络结构,本身具备可以并行的条件,可以加快大规模问题的求解速度。Krezyak 和 Le Cun 主要研究了 BP(Back-Propagation)神经网络在文字识别方面的应用,针对 BP 网络学习速度慢、泛化能力弱的缺点,在 BP 网络的基础上产生了竞争监督学习的策略。为减小神经网络的规模,赵跃龙提出了一种基于 BP 网络的手写数字识别方案,提高了神经网络的工作效率。

由于人工神经网络是对生物的神经网络的一种极端的简化,以及人们对大脑活动的认知还停留在初级阶段,人工神经网络在学习效率和算法收敛性等方面还存在很多亟待解决的问题。

1.4 问题和困难

光学文字识别技术(OCR)可分为印刷体文字识别和手写体文字识别两大类,后者又可分为联机(on-line)手写文字识别和脱机(off-line)手写文字识别。从识别的难度来看,多体印刷体文字识别难于单体印刷体文字识别,手写体识别难于印刷体识别,而脱机手写

识别又远远难于联机手写体识别。可喜的是,经过科研人员的努力,我国已有印刷体文字识别和联机手写文字识别的商品出售,目前已形成百家争鸣、百花齐放的局面,但是脱机手写文字识别还处于实验室研究阶段。在脱机手写文字识别领域,非特定人脱机手写文字识别又难于特定人手写文字识别^[3]。

1.4.1 印刷体字符识别

1. 印刷体的字体

印刷体字符由于不存在变形,或者说变形不会像手写体字符那么大,所以识别起来相对容易一点,不过也不像想象中的那么简单。印刷体字符有很多种字体,而且新的字体还在不断地产生。不同字体的同一字符的结构约 80%相同,其主要差异在于:

- (1)同字体的字符之间有较大的差异;
- (2)笔划装饰及方向角度不同;
- (3)笔划长短、位置有变化;
- (4)笔划形态变化:如黑体笔划粗而平直,宽度一致;楷体笔划有较大曲率。

2. 印刷品质的高低

由于字符笔划的不同、纸张质量的高低以及印刷版面的影响,印刷体字符在品质上有高品质和低品质之分。印刷品质越高,正确识别率也越高,对于低品质的印刷体字符识别起来有一定的困难。

3. 采样设备质量差异

采集的待识别图像的质量与采集图像的设备质量(如摄像头,电源等)和采样外界环境电磁干扰有直接的关系,受外界因素的影响,图像的质量会出现各种各样的情况。

1.4.2 手写体字符识别

我们所研究的手写体字符的类别有三十几种,并且笔画简单,其识别问题似乎不是很困难。但事实上,一些测试结果表明,字符的正确识别率并不如印刷体汉字识别正确率高。这其中的主要原因是:

- (1)笔画简单,其笔画差别相对较小,其中一些字符的字形相差不大;

(2)数字和英文字母虽然只有三十多种,但是同一字符的写法千差万别。全世界许多国家和地区都在使用,其书写带有明显的区域特性,这样就很难做到兼顾全世界各种写法的极高识别率的通用性字符识别系统。

(3)在实际应用中,对数字识别正确率的要求要比文字的要苛刻的多。这是因为,数字没有上下文关系,每个数字识别的结果都至关重要,因此,研究高性能的手写体字符识别系统是一个相当有挑战性的任务^[14]。

1.5 本文的内容安排

第一章 介绍了光学字符识别技术(OCR)的发展现状和各种识别方法,以及光学字符识别技术存在的问题和困难。

第二章 介绍了人工神经网络的基本概念,如神经元、激活函数、学习算法等。重点介绍了 BP 神经网络模型及其改进的 BP 神经网络学习算法和误差反向传播的原理。本文采用 BP 神经网络进行手写体大写字母英文字母的识别。

第三章 介绍了字符识别和图像处理的基本概念和基本方法,以及 MATLAB 图像处理工具箱的基本函数。

第四章 详细介绍了本文字母样本图像的预处理的过程,包括字母图像的保存、批量读取和归一化。

第五章 详细介绍了基于 BP 神经网络的手写体大写字母英文字母的识别模型,包括网络的选择、输入层和输出层节点数目的确定、隐含层节点数的确定、学习速率的选择、期望误差的选择等。

第六章 详细介绍了基于 BP 神经网络模型的训练过程,说明了该系统程序实现的过程,并且分析了该模型的准确性。

1.6 本文的创新点

1. 本文将逐像素特征提取方法应用到手写体大写字母英文字母识别中。
2. 提出了统一归一化英文字母图像的算法,用于将不同像素的图像归一化到同一尺寸。
3. 采用 BP 神经网络建立了手写体大写字母英文字母的识别系统,由于隐含层神经元数目的确定没有统一标准,本文经过实验最终确定隐含层神经元数目为 30,这样既保证网络收敛又使得网络获得比较理想的测试结果。

第二章 BP 神经网络

2.1 人工神经网络

神经科学研究表明,生物神经系统是由大量的神经细胞或神经元广泛相互连接组成的,一个神经元与另一个神经元密切接触,传递神经冲动,实现信息传递。人的大脑皮层是由六个不同的功能区组成的,上面布满了大约 1.4×10^{11} 个神经细胞,它相当于一万台大型计算机并行运行。所以说,人脑是一个广泛相互连接的复杂非线性系统,这种系统具有特殊的信息处理功能。人工神经网络(简称神经网络)是神经系统的模拟,包括了大脑神经系统的许多特征^[15]。研究人的大脑的目的就是试图模拟人脑信息处理机制来设计新一代智能型计算机,所以,在工程上我们所研究的都是人工神经网络(Artificial Neural Network, 简称 ANN)的范畴。为此,美国的神经网络学家 Hecht Nielsen 给出人工神经网络的一般定义:神经网络是由多个非常简单的处理单元彼此按某种方式相互连接而形成的计算机系统,该系统是靠其状态对外部输入信息的动态响应来处理信息的。因此,我们所指的神经网络都是指人工神经网络^[16]。

2.1.1 发展简史

最早用数学模型对神经系统中的神经元径向理论建模的是美国心理学家麦卡洛克(W. McCulloch)和数学家皮茨(W. Pitts)。1943年,美国心理学家 W.S. McCulloch 和数学家 W.A. Pitts 提出了一个非常简单的神经元模型——MP 模型^[17]。模型将神经元当作一个功能逻辑器件来对待,从而开创了神经网络模型的理论研究。

1957年, Rosenblatt 在 MP 模型的基础上,提出了感知器(Perceptron)的概念^[18],并第一次把神经网络的研究从纯理论的探讨付诸于工程实现。感知器是第一个真正意义上的神经网络,包括了许多现代神经网络的基本原理,整个模型的结构大体上符合神经生理学知识。Rosenblatt 给出了两层感知器的收敛定理^[19],并提出了引入隐层处理单元的三层感知器。感知器模型的提出吸引了众多学者加入到神经网络的研究中了。

1959年,美国工程师威德罗(B. Widrow)和霍夫(M. Hoff)提出了自适应线性元件和被称为 Widrow-Hoff 学习规则(又称 δ 学习规则)的神经网络训练方法^[20-22]。它是感知器的变化形式,尤其在修正权矢量的算法上进行了改进,不仅提高了训练收敛速度,而且还做成了硬件,并将训练厚的人工神经网络成功地用于抵消通讯中的回波和噪声,

成为第一个用于解决实际问题的人工神经网络。

1969 年, 人工智能创始人之一明斯基 (M. Minsky) 和帕伯特 (S. Papert) 在合著的《感知器》一书中对以单层感知器为代表的简单人工神经网络的功能及其局限性从数学上进行了深入的分析, 使得当时许多神经网络研究者感到前途渺茫, 客观上对神经网络理论的发展起了一定的消极作用。

1982 年, 美国学者霍普菲尔德 (Hopfield) 提出了一种反馈神经网络^[23], 用于联想记忆和优化计算。1984 年, Hopfield 又提出了连续的 Hopfield 神经网络模型^[24], 将神经元的响应函数由离散的二值改为连续的模拟值。

1986 年, 儒默哈特 (D. E. Rumelhart) 等人提出了解决多层神经网络权值修正的算法——误差反向传播法 (BP 算法), 成为当前应用最为广泛的神经网络模型^[25], 找到了解决明斯基和帕伯特提出的问题的办法, 从而给人工神经网络增添了活力, 使其得以全面迅速地恢复发展起来^[26]。

1987 年 6 月在美国圣地亚哥召开了第一届世界神经网络会议, 标志着神经网络研究在世界范围内形成了高潮。进入 90 年代后, 神经网络的国际会议接连不断。1989 年我国在广州召开了全国第一届神经网络信号处理会议 1991 年在南京召开了中国第二届神经网络学术大会。此后, 在 1992、1993、1994、1995 年又分别召开了第三、四、五、六届 C²N² 大会。

2.1.2 主要特点

1. 高度的并行性

人工神经网络是由许多相同的简单处理单元并联组合而成, 虽然每个单元的功能简单, 但大量简单单元的并行活动, 但其对信息的处理能力与效果惊人。

2. 高度的非线性全局作用

神经网络系统是由大量简单神经元构成的, 每个神经元接受大量其他神经元的输入, 通过非线性输入、输出关系, 产生输出影响其它神经元。网络就是这样互相制约相互影响, 实现从输入状态空间到输出状态空间非线性映射的。网络的演化遵从全局性作用原则, 从输入状态演化到终态而输出。从全局观点来看, 网络整体性能不是网络局部性能的简单迭加, 而表现某种集体性行为; 而电脑遵从串行式局域性操作原则, 每一步计算与上一步计算紧密相关, 并对下一步产生影响, 问题是通过算法逐步进行处理的。

3. 良好的容错性与联想记忆功能

人工神经网络通过自身的网络结构能够实现对信息的记忆，而所记忆的信息是存储在神经元之间的权值中。从单个权值中看不出所储存的信息内容，因而是分布式的存储方式。这使得网络具有良好的容错性，并能进行聚类分析、特征提取、缺损模式复原等模式信息处理工作。

4. 十分强的自适应、自学习功能

人工神经网络可以通过训练和学习来获得网络的权值与结构，呈现出很强的自学习能力和对环境的自适应能力。

2.1.3 基本结构与模型

人工神经网络是从生物学神经系统的信号传递抽象出的一门学科。在神经网络中，最基本的是神经元，神经元由三部分组成：细胞体、树突和轴突。树突是树状的神经纤维接收网络，它将电信号传递给细胞体，细胞体对这些数据进行整合并进行阈值处理。轴突是单根长纤维，它把细胞体的输出信号导向其他神经元。神经元的排列拓扑结构和突触的链接强度决定了神经网络的功能。

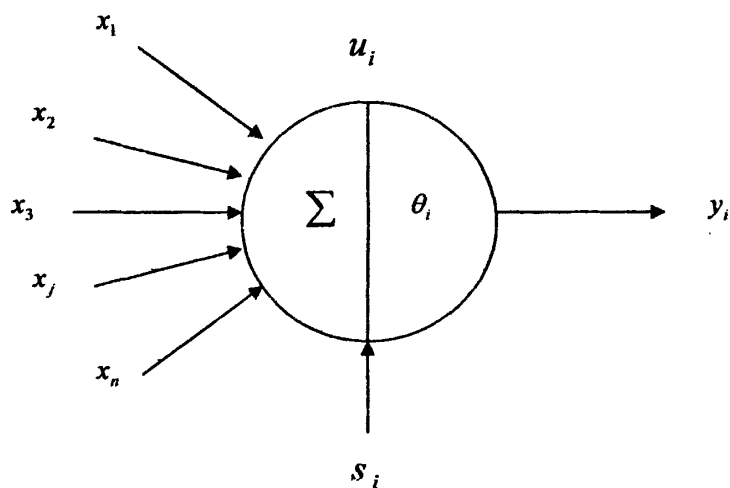


图 2.1 神经元结构模型

神经元是神经网络的基本处理单元，一般表现为一个多输入、单输出的非线性器件，通用的结构模型如图 2.1 所示。

其中， u_i 为神经元 i 的内部状态， θ_i 为阈值， x_j 为输入信号， w_{ij} 表示与神经元 x_j 连

接的权值， s_j 表示某一外部输入的控制信号。

通过大量神经元的并行计算和分布存储，神经网络具有很强的计算能力和存储能力，因此神经网络是一种基于大规模并行的分布式处理系统^[27]。神经元模型常用一阶微分方程来描述，它可以模拟生物神经网络突触膜电位随时间变化的规律^[28]。

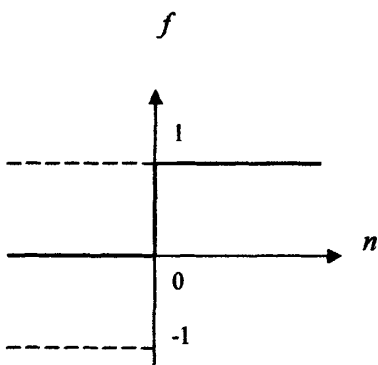
$$\begin{cases} \tau \frac{du_i}{dt} = -u_i(t) + \sum w_{ij} x_j(t) - \theta_i \\ y_i(t) = f[u_i(t)] \end{cases} \quad (2.1)$$

激活函数是一个神经元及网络的核心。神经元的输出由函数 f 表示，一般利用以下函数表达式来表现网络的非线性特征。

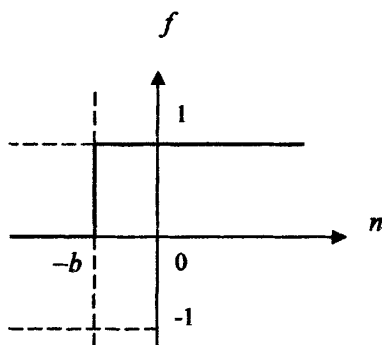
(1) 阈值型

$$f(u_i) = \begin{cases} 1 & u_i \geq 0 \\ 0 & u_i < 0 \end{cases} \quad (2.2)$$

这种激活函数将任意输入转化为 0 或 1 的输出， f 为阶跃函数，如图 2.2。



(a) 没有偏差的阈值型激活函数



(b) 带有偏差的阈值型激活函数

图 2.2 阈值型激活函数

(2) 分段线性型

其中，在运算的线性区域内放大因子置为 1。这种形式的激活函数是对非线性放大器的近似，如图 2.3。

$$f(u_i) = \begin{cases} 1 & u_i \geq u_2 \\ au_i + b & u_i \leq 0 < u_2 \\ 0 & u_i < u_1 \end{cases} \quad (2.3)$$

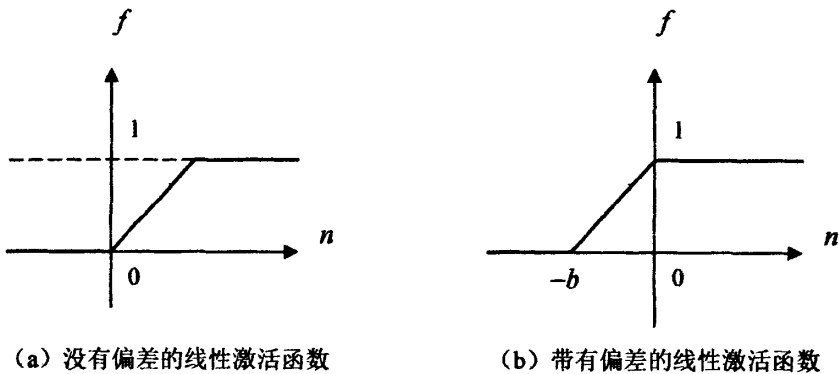


图 2.3 线性激活函数

(3) S 型函数 (Sigmoid)

S 型激活函数反映了神经元的饱和特性，由于其函数连续可导，调节曲线的参数可以得到类似阈值函数的功能。它是严格的递增函数，在线性与非线性之间显现出较好的平衡，常见的 S 型激活函数有如下两种，如图 2.4。

$$\begin{aligned} \text{tansig}(x) &= \frac{1-e^{-x}}{1+e^{-x}} & \text{logsig}(x) &= \frac{1}{1+e^{-x}} \end{aligned} \quad (2.4)$$

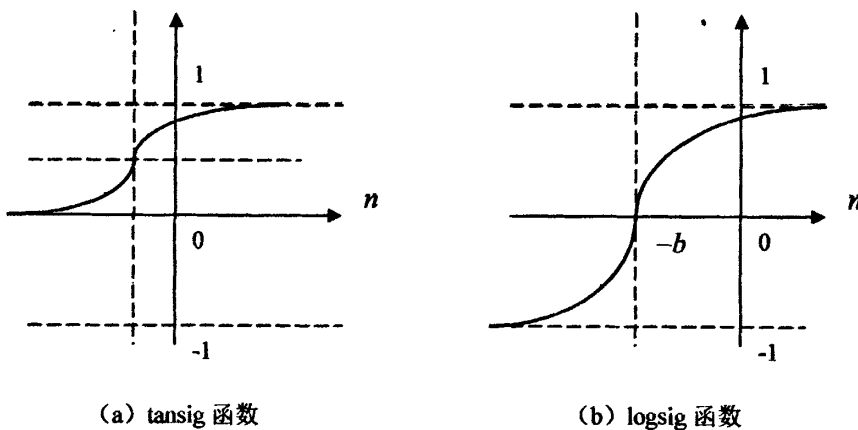


图 2.4 S 型激活函数

人工神经网络的拓扑结构一般分两种：前向网络和反馈网络。前向网络的信号流向是从输入通向输出，而反馈网络的主要不同点表现在它的输出信号通过与输入连接而返回到输入端，从而形成一个回路。前向网络的输出只是与当前输入及其连接权值有关，而在反馈网络中，由于将输出循环返回到输入，所以每一时刻的网络输出不仅取决于当前的输入，而且还取决于上一时刻的输出。其输出的初始状态由输入矢量设定后，随着

网络的不断运行,从输出反馈到输入的信号不断改变,也使得输出不断变化,从而使网络表现出暂态特性,这使得反馈网络表现出前向网络所不具有的震荡或收敛特性。

经过对大脑的深入研究,人们经过简化、抽象和模拟,提出了若干人工神经网络基本模型。这些模型反映人脑功能的若干基本特征,但并不是人脑的逼真描写,它们从不同侧面模拟了大脑的功能,并已被实验所证实。此外,近年来随着神经网络研究的深入,新的模型不断出现,估计已达近百种。其中主要包括 Hopfield 网络、前馈网络(如 MADA-LINES、多层感知器网络、径向基函数网络、函数链网络等)、反向网络(CPN)、脑中盒状态(BSB)模型、自组织特征映射(SOFM)、Hamming 网络、自适应共振理论(ART)理论、Boltzmann 机、学习矢量量化(LVQ)等等。

2.2 BP 神经网络模型

反向传播神经网络(Back-Propagation Network,简称 BP 网络)是对非线性可微分函数进行权值训练的多层网络,是前向神经网络的一种。

BP 网络主要用于:

- 1) 函数逼近与预测分析:用输入矢量和相应的输出矢量训练一个网络,逼近一个函数或预测未知信息;
- 2) 模式识别:用一个特定的输出矢量将它与输入矢量联系起来;
- 3) 分类:把输入矢量以所定义的合适方式进行分类;
- 4) 数据压缩:减少输出矢量维数以便于传输与存储。

在人工神经网络的实际应用中,80%—90%的神经网络模型是采用 BP 网络或它的变化形式,它也是前向网络的核心部分,体现了人工神经网络最精华的部分。

1986 年 Rumelhart, Hinton and Williams 报告了反向传播算法的发展^[29]。BP 神经网络的模型结构如图 2.5 所示。

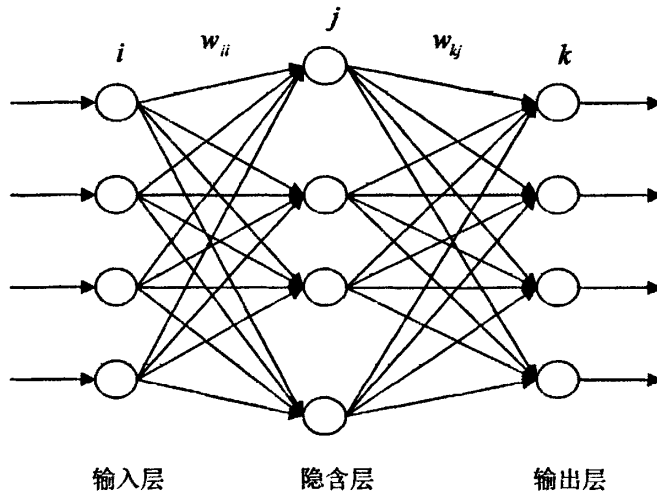


图 2.5 具有一个隐含层和输出层的 BP 神经网络模型

BP 网络的激活函数必须是处处可微的，所以它就不能采用二值型的阈值函数 $\{0, 1\}$ 或符号函数 $\{-1, 1\}$ ，BP 网络经常使用的是 Sigmoid 函数、正切函数或线性函数。Sigmoid 型激活函数具有非线性放大系统功能，它可以将输入从负无穷大到正无穷大的信号，转换成 0 到 1 或 -1 到 1 之间的输出，对较大的输入信号，放大系数较小；而对较小的输入信号放大系数较大，所以采用 Sigmoid 型激活函数可以去处理和逼近非线性的输入/输出关系。不过，如果在输出层采用 Sigmoid 型激活函数，输出则被限制在一个很小的范围内，若采用线性激活函数，则可使网络输出任意值。所以只有当希望对网络的输出进行限制，如限制在 0 和 1 之间，那么在输出层应当包含 Sigmoid 型激活函数。

2.3 BP 学习算法及规则

BP 网络是一种具有三层或三层以上的神经网络，上下层之间实现全连接，而每层神经元之间无连接。当一对学习样本提供给网络后，神经元的激活值从输入层经各中间层向输出层传播，在输出层的各神经元获得网络的输入响应。接下来，按照减少目标输出与实际误差的方向，从输出层经过各中间层逐层修正各连接权值，最后回到输入层，这种算法称为“误差逆传播算法”，即 BP 算法。

从大的方面来说，学习算法可分为两类：有监督的学习算法和无监督的学习算法。对于有监督的学习算法，学习和应用是分开的两个过程。学习过程是一个外加的对神经

网络的连接强度进行系统调整的过程，而应用过程则是一个自治的动力学过程。通俗的说，这种神经网络就是“先学习，再使用”。而无监督的学习算法中的学习过程和应用过程是一个统一的过程。它不需要单独的学习过程，而是在与外界环境的不断交互作用中达到自组织学习，也就是“边干边学，干中学，学中干”。

BP 算法属于一种有监督的学习算法。其主要思想是：对 n 个输入学习样本： X^1, X^2, \dots, X^n ，已知其对应的输出样本为： T^1, T^2, \dots, T^m 。学习的目的是用网络的实际输出 Y^1, Y^2, \dots, Y^m 与目标矢量 T^1, T^2, \dots, T^m 之间的误差来修正其权值，使实际输出与目标矢量尽可能地接近：即：使网络输出层的误差平方和达到最小。它是通过连续不断地在相对于误差函数斜率下降的方向上计算网络权值和偏差的变化而逐渐逼近目标的。

BP 算法由两部分组成：信息的正向传递与误差的反向传播。在正向传递过程中，输入信息从输入层经隐含层逐层计算传向输出层，每一层神经元的状态只影响下一层神经元的状态。如果在输出层没有得到期望的输出，则计算输出层的误差变化值，然后反向传播，通过网络将误差信号沿原来的连接通路反传回来，修改各层神经元的权值直至达到期望目标。

现以图 2.6 所示的具有一个隐含层的简化网络图为例进行 BP 算法推导。

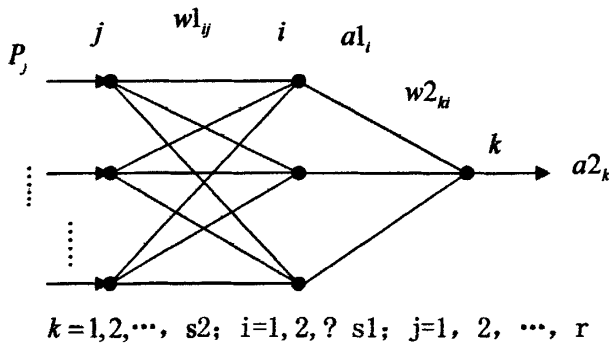


图 2.6 具有一个隐含层的简化网络图

设输入为 P ，输入神经元有 r 个，隐含层内有 $s1$ 个神经元，激活函数为 $F1$ ，输出层内有 $s2$ 个神经元，对应的激活函数为 $F2$ ，输出为 A ，目标矢量为 T 。

2.3.1 信息的正向传递

(1) 隐含层中第 i 个神经元的输出为:

$$a1_i = f1(\sum_{j=1}^r w1_{ij} P_j + b1_i), \quad i = 1, 2, \dots, s1 \quad (2.5)$$

(2) 输出层第 k 个神经元的输出为:

$$a2_k = f2(\sum_{i=1}^{s1} w2_{ki} a1_i + b2_k), \quad k = 1, 2, \dots, s2 \quad (2.6)$$

(3) 定义误差函数为:

$$E(W, B) = \frac{1}{2} \sum_{k=1}^{s2} (t_k - a2_k)^2 \quad (2.7)$$

2.3.2 利用梯度下降法求权值变化及误差的反向传播

(1) 输出层的权值变化。

对从第 i 个输入到第 k 个输出的权值有:

$$\Delta w2_{ki} = -\eta \frac{\partial E}{\partial w2_{ki}} = -\eta \frac{\partial E}{\partial a2_k} \frac{\partial a2_k}{\partial w2_{ki}} = \eta (t_k - a2_k) \cdot f2' \cdot a1_i = \eta \cdot \delta_{ki} \cdot a1_i \quad (2.8)$$

其中

$$\delta_{ki} = (t_k - a2_k) \cdot f2' = e_k \cdot f2' \quad e_k = (t_k - a2_k) \quad (2.9)$$

同理可得:

$$\Delta b2_k = -\eta \frac{\partial E}{\partial b2_k} = -\eta \frac{\partial E}{\partial a2_k} \frac{\partial a2_k}{\partial b2_k} = \eta (t_k - a2_k) \cdot f2' = \eta \cdot \delta_{ki} \quad (2.10)$$

(2) 隐含层的权值变化。

对从第 j 个输入到第 i 个输出的权值, 有:

$$\begin{aligned} \Delta w1_{ij} &= -\eta \frac{\partial E}{\partial w1_{ij}} = -\eta \frac{\partial E}{\partial a2_k} \frac{\partial a2_k}{\partial a1_i} \frac{\partial a1_i}{\partial w1_{ij}} \\ &= \eta \left(\sum_{k=1}^{s2} (t_k - a2_k) \cdot f2' \cdot w2_{ki} f1' \cdot p_j \right) = \eta \cdot \delta_{ij} \cdot p_j \end{aligned} \quad (2.11)$$

其中:

$$\delta_{ij} = e_i \cdot f1', \quad e_i = \sum_{k=1}^{s2} \delta_{ki} w2_{ki}, \quad \delta_{ki} = e_k \cdot f2', \quad e_k = t_k - a2_k \quad (2.12)$$

同理可得:

$$\Delta b1_i = \eta \delta_{ij} \quad (2.13)$$

2.3.3 误差反向传播的流程图

误差反向传播过程实际上是通过计算输出层的误差 e_k ，然后将其与输出层激活函数的一阶导数 f_2' 相乘来求得 δ_k 。由于隐含层中没有直接给出目标矢量，所以利用输出层的 δ_k 进行误差反向传递来求出隐含层权值的变化量 Δw_{ki} 。然后计算 $e_i = \sum_{k=1}^{s_2} \delta_k \cdot w_{ki}$ ，并同时通过将 e_i 与该层激活函数的一阶导数 f_1' 相乘，而求得 δ_i ，以此求出前层权值的变化量 Δw_{ji} 。图 2.4 给出了形象的解释。

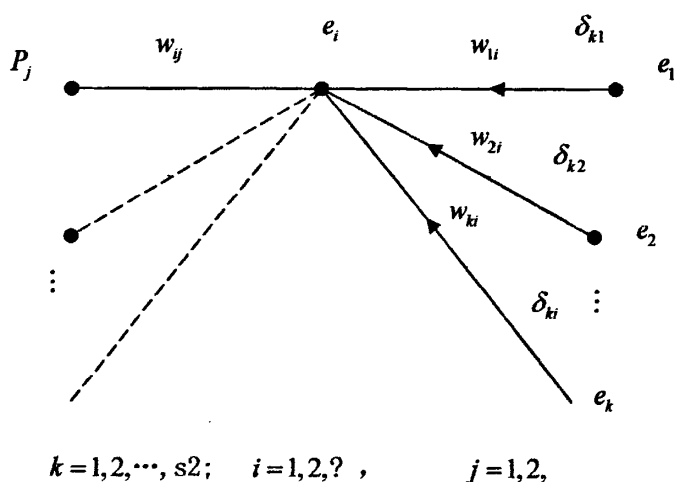


图 2.7 误差反向传播法的图形解释

BP 算法要用到各层激活函数的一阶导数，所以要求其激活函数处处可微。对数 Sigmoid 型激活函数 $f(n) = \frac{1}{1+e^{-n}}$ ，其导数为：

$$\begin{aligned} f'(n) &= \frac{0 - e^{-n}(-1)}{(1+e^{-n})^2} = \frac{1}{(1+e^{-n})^2} (1+e^{-n}-1) \\ &= \frac{1}{1+e^{-n}} \left(1 - \frac{1}{1+e^{-n}}\right) = f(n)[1-f(n)] \end{aligned} \quad (2.14)$$

$$\text{对于线性函数的导数有: } f'(n) = n' = 1 \quad (2.15)$$

2.4 改进的 BP 网络学习算法

虽然反向传播法得到广泛的应用，但它也存在自身的限制与不足。对于一些复杂的

问题, 由于初始的学习速率太小, BP 网络可能要进行很长的时间训练。在网络的训练过程中, 当初始权值调的过大, 可能使得所有的或大部分神经元的中权值总和偏大, 从而使得对网络的调节过程几乎停顿下来。通常为了避免这种现象的发生, 一是选取较小的初始权值, 二是采用较小的学习速率, 但这又增加了训练的时间。BP 算法可以使网络收敛到一个解, 但它并不能保证所求为误差超平面的全局最小解, 很可能是一个局部极小解。但是网络有时无法逃出这一局部最小值。因此, 为了加快训练速度, 避免陷入局部极小值和改善其他能力, 提出了对反向传播法的改进方法。

2.4.1 附加动量法

附加动量法使网络在修正其权值时, 不仅考虑误差在梯度上的作用, 而且考虑在误差曲面上变化趋势的影响, 其作用如同一个低通滤波器, 它允许网络忽略网络上的微小变化特性。在没有附加动量的作用下, 网络可能陷入浅的局部极小值。

该方法是在反向传播法的基础上在每一个权值的变化上加上一项正比于前次权值变化量的值, 并根据反向传播法来产生新的权值变化。带有附加动量因子的权值调节公式为如下, 其中 k 为训练次数, mc 为动量因子。

$$\Delta w_{ij}(k+1) = (1-mc)\eta\delta_i p_j + mc\Delta w_{ij}(k) \quad (2.16)$$

$$\Delta b_i(k+1) = (1-mc)\eta\delta_i + mc\Delta b_i(k) \quad (2.17)$$

附加动量法的实质是将最后一次权值变化的影响, 通过一个动量因子来传递。当动量因子取值为零时, 权值的变化仅是根据梯度下降法产生; 当动量因子取值为 1 时, 新的权值变化则是设置为最后一次权值的变化, 而依梯度法产生的变化部分则被忽略了。

根据附加动量法的设计原则, 当修正的权值在误差中导致太大的增长结果时, 新的权值应被取消而不被采用, 并使动量作用停止下来, 以使网络不进入较大的误差曲面; 当新的误差变化率对其旧值超过一个事先设定的最大误差变化率时, 也得取消所计算的权值变化。其最大误差变化率可以是任何大于或等于 1 的值。典型的值取 1.04。所以在进行附加动量法的训练程序设计时, 必须加进条件判断以正确使用其权值修正公式。

训练程序中对采用动量法的判断条件为:

$$mc = \begin{cases} 0 & \text{当 } SSE(k) > SSE(k-1) \cdot 1.04 \\ 0.95 & \text{当 } SSE(k) < SSE(k-1) \\ mc & \text{其他} \end{cases} \quad (2.18)$$

2.4.2 自适应学习速率

对于一个特定的问题,要选择适当的学习速率不是一件容易的事情,通常是凭经验或实验获取,但即使这样,对训练初期功效较好的学习速率,不一定对后来的训练合适。因此,采用在训练过程中自动调整学习速率,其准则为:检查权值的修正值是否真正降低了误差函数,如果确实如此,则说明所选取的学习效率小了,可以对其增加一个量;若不是这样,而产生了过调,那么就应该减少学习速率的值。此方法可以保证网络总是以最大的可接受的学习速率进行训练。当一个较大的学习速率仍能够使网络稳定学习,使其误差继续下降,则增加学习速率,使其以更大的学习速率进行学习。一旦学习速率调得过大,而不能保证误差继续减少,则减少学习速率直到使其学习过程稳定为止。

自适应学习速率的调整公式为:

$$\eta(k+1) = \begin{cases} 1.05\eta(k) & SSE(k+1) < SSE(k) \\ 0.7\eta(k) & SSE(k+1) > 1.04 * SSE(k) \\ \eta(k) & \text{其他} \end{cases} \quad (2.19)$$

2.4.3 L-M 法

Levenberg-Marquardt 法,简称 L-M 法,实际上是梯度下降法和牛顿法的结合。梯度下降法在开始几步下降较快,但随着接近最优值时,由于梯度趋于零,使得目标下降缓慢;而牛顿法可以在最优值附近产生一个理想的搜索方向。L-M 法的搜索方向定为:

$$S(X^{(k)}) = -(H^{(k)} + \lambda^{(k)}I)^{-1} \nabla f(X^{(k)}) \quad (2.20)$$

令 $\eta^{(k)} = 1$, 则 $X^{(k+1)} = X^{(k)} + S(X^{(k)})$ 。

起始时, λ 取一个很大的数(如 10^4),此时相当于步长很小的梯度下降法;随着最优点的接近, λ 减小到零,则 $S(X^{(k)})$ 从负梯度方向转向牛顿法的方向。通常,当 $f(X^{(k+1)}) < f(X^{(k)})$, 减小 λ (如 $\lambda^{(k+1)} = 0.5\lambda^{(k)}$); 否则,增大 λ (如 $\lambda^{(k+1)} = 2\lambda^{(k)}$) [30]。

第三章 字符识别与图像处理

3.1 字符识别

人的思维可以对初次见到的事物进行分类。比如，即使人们初到一个城市，也可以轻易地辨认出“街道”、“房屋”、“汽车”这样的事物。正是人脑的这种推广能力，使得人们利用见到过的有限事物和现象，形成各种事物类别的概念。这些有限的、个别的事物和现象就可以称为模式，而整个类别的现象和事物则可以称为模式类，或者简称为类。人们根据所见模式的特性，将其划归为某一类的过程，实际上就是模式识别的过程。

字符识别是模式识别的一个传统研究领域。从 50 年代开始，许多的研究者就在这一研究领域开展了广泛的探索，并为模式识别的发展产生了积极的影响。字符识别一般可以分为两类：联机手写字符识别和光学字符识别(Optical Character Recognition OCR)或称离线字符识别。在联机手写字符识别中，计算机能够通过与计算机相连的手写输入设备获得输入字符笔划的顺序、笔划的方向以及字符的形状，所以相对 OCR 来说它更容易识别一些。但联机字符识别有一个重要的不足就是要求输入者必须在指定的设备上书写，然而人们在生活中大部分的书写情况是不满足这一要求的，比如人们填写各种表格资料，开具支票等。如果需要计算机去认识这些已经成为文字的东西，就需要 OCR 技术。常见的模式识别的方法有两种：

1. 决策理论方法

又称统计方法，是发展较早也比较成熟的一种方法。被识别对象首先数字化，变换为适于计算机处理的数字信息。许多模式识别系统在数字化环节之后还进行预处理，用于除去混入的干扰信息并减少某些变形和失真。随后是进行特征抽取，即从数字化后或预处理后的输入模式中抽取一组特征。所谓特征是选定的一种度量，它对于一般的变形和失真保持不变或几乎不变，并且只含尽可能少的冗余信息。特征抽取过程将输入模式从对象空间映射到特征空间。这时，模式可用特征空间中的一个点或一个特征矢量表示。这种映射不仅压缩了信息量，而且易于分类。在决策理论方法中，特征抽取占有重要的地位，但尚无通用的理论指导，只能通过分析具体识别对象决定选取何种特征。特征抽取后可进行分类，即从特征空间再映射到决策空间。为此而引入鉴别函数，由特征矢量

计算出相应于各类别的鉴别函数值，通过鉴别函数值的比较实行分类。

决策理论方法的模式识别系统一般由四部分构成：数据获取、常规处理、特征提取和选择、分类决策。如图 3.1 所示：

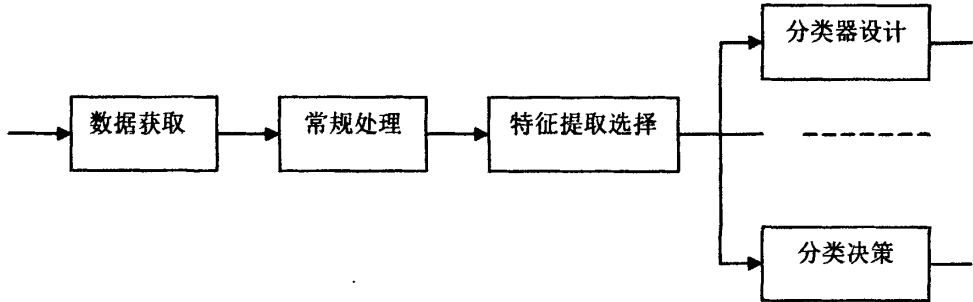


图 3.1 决策理论方法模式识别系统的构成图

2. 句法方法

又称结构方法或语言学方法。其基本思想是把一个模式描述为较简单的子模式的组合，子模式又可描述为更简单的子模式的组合，最终得到一个树形的结构描述，在底层的最简单的子模式称为模式基元。在句法方法中选取基元的问题相当于在决策理论方法中选取特征的问题。通常要求所选的基元能对模式提供一个紧凑的反映其结构关系的描述，又要易于用非句法方法加以抽取。显然，基元本身不应该含有重要的结构信息。模式以一组基元和它们的组合关系来描述，称为模式描述语句。基元组合模式的规则由所谓语法来指定。一旦基元被鉴别，识别过程可通过句法分析进行，即分析给定的模式语句是否符合指定的语法，满足某类语法的即被分入该类。

一般的句法模式识别系统可以分为两个部分：识别部分和分析部分，其中识别部分由预处理、基元提取和句法(或结构)分析组成，而分析部分包括基元选择和文法推断^[31]。如图 3.2 所示：

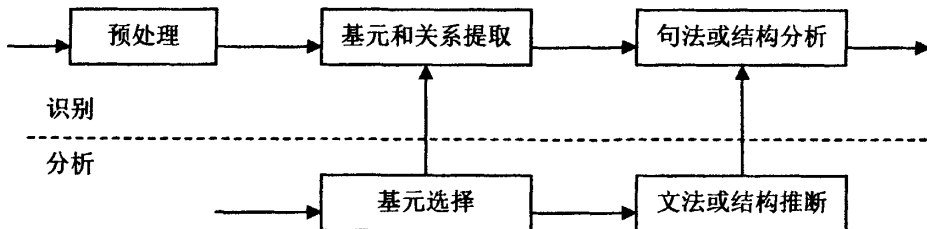


图 3.2 句法模式识别系统的构成图

3.2 图像处理

图像信号的数字处理是指将图像作为图像信号的数学处理技术,按照人们通常的习惯,也称为数字图像处理技术。最常见的是用计算机对图像进行处理,它是在以计算机为中心的包括各种输入、输出、存储及显示设备在内的数学图像处理系统上进行的。

3.2.1 图像分类

1. 按照图像的存在形式分类:

(1) 实际图像:通常为二维分布,又分为可见图像和不可见图像。可见图像指人眼能够看到并能接受的图像,包括图片、照片、图、画、光图像等。不可见图像如温度、压力、高度和人口密度分布图等。

(2) 抽象图像:如数学函数图形,包括连续函数和离散函数。

2. 按照图像亮度等级分类

(1) 二值图像:只有黑白两种亮度等级的图像。

(2) 灰度图像:有多重亮度等级的图像。

3. 按照图像的光谱分类

(1) 彩色图像:图像上的每个点有多于1个的局部特性,如在彩色摄影和彩色电视中重现的3基色(红、绿、蓝)图像,每个像点就有分别对应3个基色的3个亮度值。

(2) 黑白图像:每个像点只有一个亮度值分量,如黑白照片、黑白电视画面等。

4. 按照图像是否随时间变换分类

(1) 静止图像:不随时间而变换的图像,如各类图片等。

(2) 活动图像:随时间而变换的图像,如电影和电视画面等。

5. 按照图像所占空间和维数分类

(1) 二维图像:平面图像,如照片等。其数学表示为 $f(x,y)$, f 为光强, x,y 为二维空间坐标。

(2) 三维图像:空间分布的图像,即立体图像。其数学表示为 $f(x,y,z)$, f 为光强, x,y,z 为三维空间坐标^[32]。

3.2.2 图像存储原理

从结构上讲,图形文件分为两种,即位图和矢量图。在位图中,图像由许多的屏幕

小点（我们通常说的像素）组成，这些小点对应显存中的“位”，而就是这些“位”决定了像素的图形属性，如像素的颜色、灰度、明暗对比度等。当一个像素所占的位数多时，它所能表现的颜色就更多、更丰富，从整体上看，图像的色彩就更艳丽，分辨率就更高。位图中所分的二位图、八位图等正是指像素所占的位数。当位图被放大或缩小时，由于像素的数量没有改变，图像的分辨率就会降低，图像的外观自然就大打折扣。相信朋友们在 Word 或 WPS 当中插入图像，安排图文结构时，或多或少都会遇到这种情况。就这方面来看，位图的缺点是显而易见的——分辨率的固定导致大分辨率的清晰图像占用大量空间；像素的分散性使动态图像的表达显得困难，例如看 VCD 时出现马赛克现象，就是像素丢失造成的。

因此，研究人员开发出一种新的图形格式——矢量图。顾名思义，矢量图就是用矢量代替位图中的“位”。简单说来，矢量图不再给图的全部像素作统一的标记，而是用矢量给图的几何部分作标记。可以说矢量图的优点是很多的：能无限放大、缩小而不失真；不需要将图像每一点的状态记录下来，因而比相同质量和大小的位图占用的空间少得多。

3.2.3 数字图像的表达

从广义上说，图像是自然界景物的客观反映。以照片形式或初级记录介质保存的图像是连续的，计算机接受和处理数字图像，无法接受和处理这种空间分布和亮度取值均连续分布的图像。

一副 $m \times n$ 的数字图像可用矩阵表示为

$$F = \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,n-1) \\ f(1,0) & f(1,1) & \cdots & f(1,n-1) \\ \vdots & \vdots & & \vdots \\ f(m-1,0) & f(m-1,1) & \cdots & f(m-1,n-1) \end{bmatrix} \quad (3.1)$$

数字图像中的每个像素都对应于矩阵中相应的元素。把数字图像表示成矩阵的优点在于，能应用矩阵理论对图像进行分析处理。在表示数字图像的能量、相关等特性时，采用图像的矢量表示比用矩阵表示方便。若按行的顺序排列像素，使该图像后一行第一个像素紧接前一行最后一个像素，可以将该幅图像表示成 $1 \times mn$ 的列向量 f ，即

$$f = (f_0, f_1, \cdots, f_{mn-1})^T \quad (3.2)$$

式中, $f_i = (f(i,0), f(i,1), \dots, f(i,n-1))^T, i=0,1,\dots,m-1$ 。这种表示方法的优点在于, 对图像进行处理时, 可以直接利用向量分析的有关理论和方法。构成向量时, 既可以按行的顺序, 也可以按列的顺序。选定一种顺序以后, 后面的处理都要与之保持一致。

3.2.4 图像信号数字化

1. 数字化

数字化就是对模拟图像信号离散化的过程, 包括空间离散化和幅值的离散化, 如图 3.3 所示为将一幅图像从其原来的模拟形式转换为数字形式的处理过程, “转换”是非破坏性的。

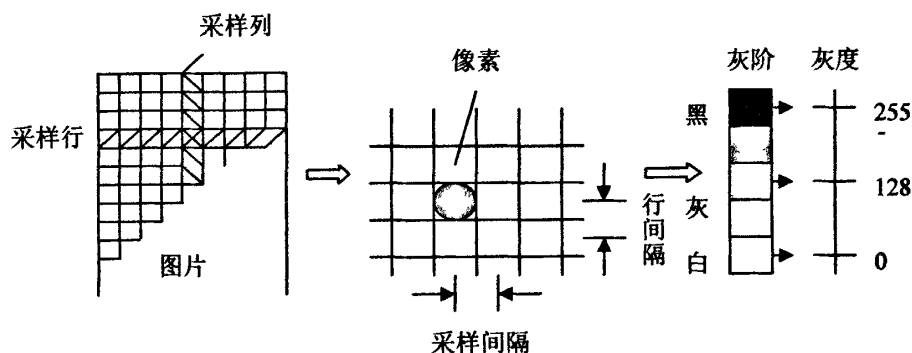


图 3.3 图像数字化

2. 扫描

扫描是指对一幅图像内给定位置的寻址。在扫描过程中被寻址的最小单元是图像元素即像素, 随摄影图像的数字化就是对胶片上一个一个小斑点的顺序扫描。

3. 采样

图像的采样是将在空间上连续的图像转换成离散的采样点 (即像素) 集的操作, 就是把一幅连续图像在空间上分割成 $M \times N$ 个网格, 每个网格对应为一个像素点, 用一个亮度值表示。由于图像是二维分布的信息, 所以采样是在 x 轴和 y 轴两个方向上进行的。

一般采样的方法有两类: 一类是直接对表示图像的二维函数值进行采样, 即读取各离散点上的信号值, 所得结果就是一个样点值阵列, 所以也成为点阵采样。另一类是先将图像函数进行某种正交变换, 用其变换系数作为采样值, 故称为正

交系数采样。

设对 $f(x, y)$ 按网格均匀采样, x 、 y 方向上的采样间隔分别为 Δx 、 Δy , 则采样点的位置为 $x = m\Delta x, y = n\Delta y (m, n = 0, \pm 1, \pm 2, \dots)$ 。采样函数如图 3.4 所示。定义采样函数:

$$f(x, y) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \delta(x - m\Delta x, y - n\Delta y) \quad (3.3)$$

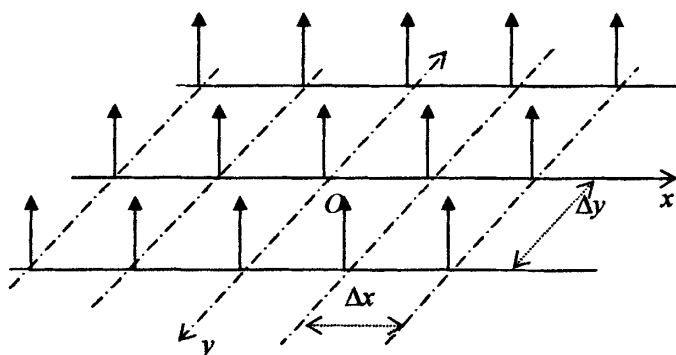


图 3.4 采样函数

4. 量化

经过采样的图像, 只是在空间上被离散为像素(样本)的阵列, 而每一个样本灰度值还是一个有无穷多个取值的连续变化量, 必须将其转化为有限个离散值, 赋予不同码字才能真正成为数字图像, 这样的转化过程称为量化。简单地说, 量化就是对图像灰度值的离散化处理, 也就是对每个样点值数值化, 使其只与有限个可能电平数中的一个对应。

量化也可以分为两种: 一种是将样点灰度值等间隔分档取整, 称为均匀量化。另一种是不等间隔分档取整, 称为非均匀量化。因为都要取整, 故量化也常称为整量或整量化过程^[33,34]。

3.2.5 图像特征提取

1. 逐像素特征提取法

这是一种最简单的特征提取法, 对图像进行逐行逐列的扫描当遇到黑色像素时取其特征值为 0, 遇到白色像素时取其特征值为 1, 这样当扫描结束以后就形成了一个维数与图像中像素点的个数相同的特征向量矩阵。

这种特征提取方法的特点是算法简单，运算速度快，可以使 BP 网络很快地收敛，训练效果好，缺点是适应性不强。但可以通过加大训练样本数目的方法来增强其适应性。

2. 骨架特征提取法

两幅图像由于它们的线条粗细不同，使得两幅图像差别很大，但是将它们的线条进行细化后，统一到相同的宽度，如一个像素的宽度时，这两幅图像的差距就不那么明显。利用图形的骨架作为特征进行数码识别，就使得识别有了一定的适应性。对细化后的图像利用 EveryPixel 函数进行处理就可以得到细化图像的特征向量矩阵。骨架特征提取的方法对于线条粗细不同的数码有一定的适应性，但是图像一旦出现偏移就难以识别。

3. 垂直方向数据统计特征提取法

这种特征提取方法的算法就是自左向右对图像进行逐列的扫描，统计每列黑色像素的个数，然后自上而下逐行扫描，统计每行的黑色像素的个数，将统计结果作为字符的特征向量，如果字符的宽度为 w ，长度为 h ，则特征向量的维数是 $w+h$ 。

4. 13 特征点提取法

即从每个字符中提取 13 个特征点。首先把字符平均分成 8 份，统计每一份内黑色像素点的个数作为 8 个特征，然后，统计水平方向中间两列和竖直方向中间两列的黑色像素点的个数作为 4 个特征，最后，统计所有黑色像素点的个数作为第 13 个特征。13 特征点提取法有着很好的适应性，但是由于特征点的数目太少所以在样本训练的时候比较难收敛^[35]。

此外，还有梯度统计，弧度统计，角点提取方法等其他特征向量提取方法。

在本文中，经多次尝试，最终选用第一种特征提取方法即逐像素特征提取法。

第四章 字母识别预处理

目前,在神经网络模式识别中根据对输入样本的表达方式的选择有下面两大类:一类是直接将字母图像经数值化处理之后得到的像素点原始样本作为神经网络的输入;另一类则是对这种原始像素点构成的原始输入样本再作进一步的预处理或变换。虽然,在某些情况下,人们采用第一类样本表达获得了较好的识别结果,但这种输入方式,通常使得网络分类器结构的复杂程度大大增加,所以,对原始样本进行预处理,已成为神经网络模式识别成功的一个很重要的方面。

这一章主要讲述对字母图像进行预处理的步骤和各种方法的应用,预处理效果的好坏直接影响到字符识别的准确率和效率,对经过处理后的图像提取特征,再进行识别才能得到更大的识别率。

根据本文处理图像的需要又提出了批量读取图像的方法和自定义的归一化算法,最后对字母图像进行编码,使之成为神经网络可以接受的输入向量。

4.1 图像的保存和读取

本文要识别的是手写体大写英文字母 A 到 Z,共 26 个字母。本文用到的手写体英文字母是采用 Windows XP 操作系统自带的画图板手写输入的,保存为 BMP 格式单色位图,也就是二进制图像,也称二值图像。本文的样本图像共有 1040 幅(40 组),其中 520 幅为训练样本,520 幅为测试样本。

BMP(bitmap 的缩写)文件格式是 Windows 本身可以直接提供读取支持的位图文件格式。一个 BMP 格式的文件通常有 BMP 的扩展名,但有一些是以 rle 为扩展名的, rle 的意思是行程长度编码(run length encoding)。这样的文件意味着其使用的数据压缩方法是 BMP 格式文件支持的两种 rle 方法中的一种。BMP 文件可用每像素 1、4、8、16 或 24 位来编码颜色信息,这个位数称作图像的颜色深度,它决定了图像所含有的最大颜色数。一幅 1-bpp (bit per pixel) 的图像只能有两种颜色。而一幅 24-bpp 的图像可以有超过 16 兆种不同的颜色。本文采用的是 1-bpp 单色位图,图像只有黑和白两种颜色。在一幅二进制图像中,每一个像素将取两个离散数值(0 或 1)中的一个,从本质上说,这两个数值分别代表状态“开”(on)或“关”(off)。其中 1 表示白色,0 表示黑色。如图 4.1,

显示的是部分手写英文字母图片。

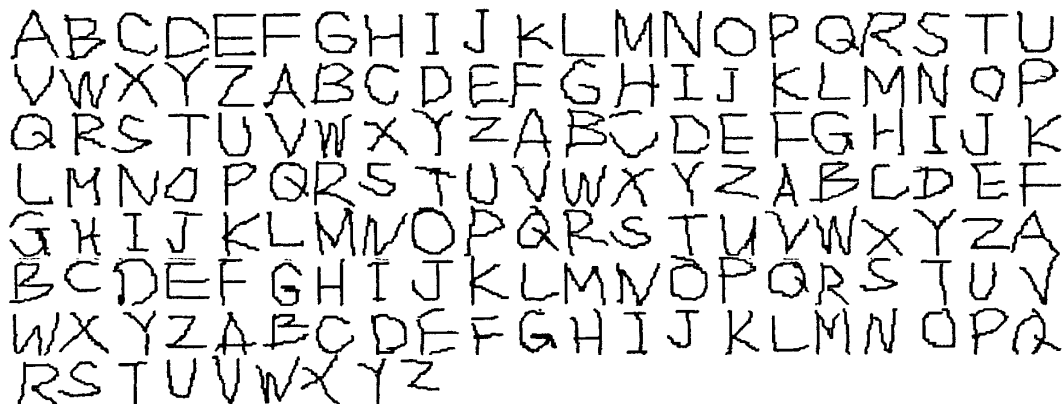


图 4.1 部分手写英文字母图片

在 MATLAB 中, 采用函数 `imread` 读取图像, 调用格式为 `imread('filename')`。由于图像的数量很多, 在本文中我们采用批量读取图像的办法: 在 MATLAB 安装目录的 `work` 文件夹下, 新建一个文件夹命名为 `trainsample` (即训练样本), 把图像保存在文件夹 `trainsample` 中, 用显示函数 `dir` 读入该文件夹, 保存到变量名为 `filename` 的数组中; 再用 `imread` 读取 `filename` 中的图像数据, 最后将图像保存到细胞型数组 `a` 中, 其中 `a` 是一个 1×260 的数组, 它的每一个元素保存的是一幅图像的逻辑型数组。

读取图像的 MATLAB 代码如下:

```
filename=dir('D:\Program Files\MATLAB\R2007b\work\trainsample\*.bmp');
t=520;
for i=1:t
    Q(:, :, i)=imread(filename(i).name);
end
for i=1:t
    a{ i}=Q(:, :, i);
end
```

4.2 图像的归一化

在本文中, 我们在画图板中直接将图像属性设置成 28×28 像素进行手写输入, 这样保存下来的图像都是 28×28 像素的图像, 以便我们接下来进行归一化的工作。

在 MATLAB 中, 采用函数 `imresize` 来改变图像的大小, 调用格式为 `I2=imresize`

(I,rate)。其中, I 表示原始图像的数组, I2 表示改变大小后图像的数组, rate 表示变化率。例如将原来的图像缩小为原来的一半, 即 $rate=0.5$ 。这一算法的缺点是只能将原始图像的高度和宽度同时变化相同的比例, 本文中我们希望将原始图像变成 10×14 像素的图像, 该算法就无法实现, 这种算法不适合解决我们面临的问题。

本文采用一种新的归一化处理算法, 就是要把原来各不相同的字符统一到同一尺寸, 在本文中是统一到同一高度, 然后根据高度来调整字符的宽度。具体算法如下: 先得到原来字符的高度, 并与系统要求的高度做比较, 得出要变换的系数, 然后根据得到的系数计算每一个原始像素点归一化后在新图像中的位置, 最后计算新图像中每个像素点的值, 方法是: 求出原始像素点及其周围三个点的像素值的平均值, 最后用函数 floor.m 取整, 调用格式为 $B=floor(A)$, B 取小于或者等于 A 的整数。

归一化算法程序代码如下:

```
ans=imread('A.bmp');
imshow(ans)
[r,c]=size(ans);
for i=2:(r-1)
    for j=2:(c-1)
        n=i/(r/14);
        n=ceil(n);
        m=j/(c/10);
        m=ceil(m);
        M(n,m)=(ans(i,j)+ans(i,j+1)+ans(i,j-1)+ans(i-1,j))/4;
        N(n,m)=floor(M(n,m));
    end
end
imshow(N)
```

图 4.2 和图 4.3 显示了部分 28×28 像素图像通过该归一化算法计算所得到的 10×14 像素图像。

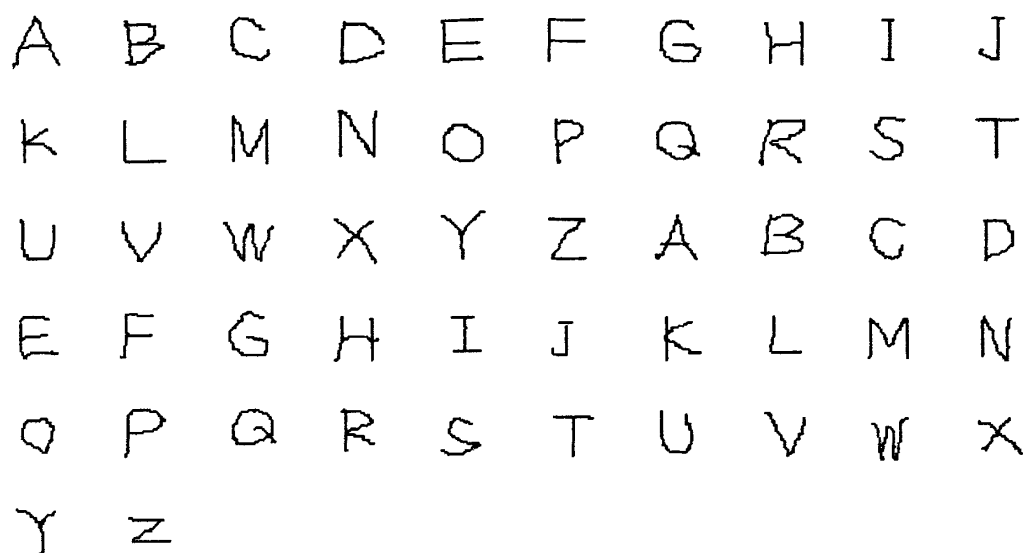


图 4.2 部分 28×28 像素图像

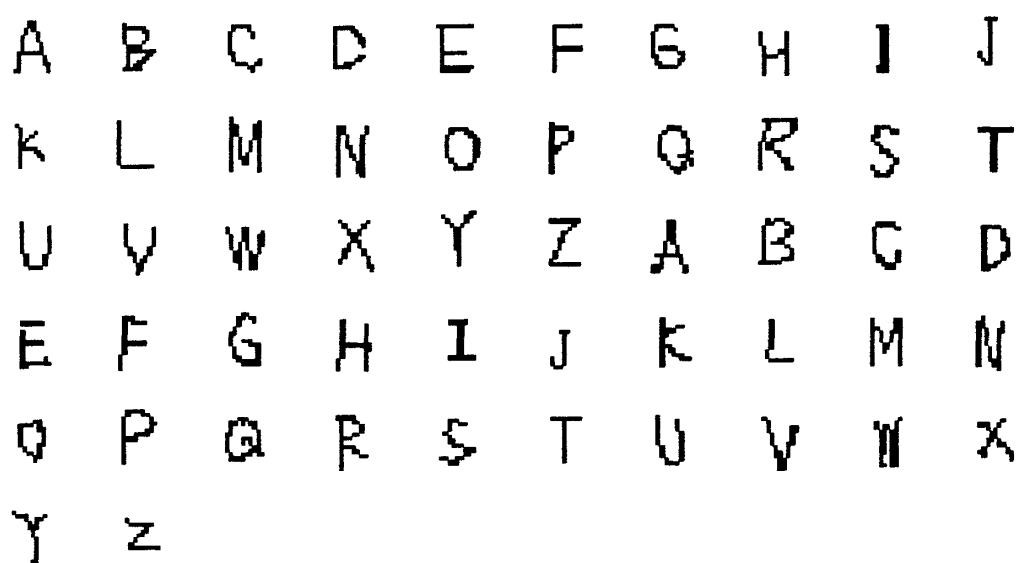


图 4.3 28×28 像素图像归一化成 10×14 像素图像

从以上图 4.2 和图 4.3 的对比我们可以看到，归一化后的图像几乎没有出现笔画断开、变形、缺失等现象，而是完整的保存了原有图像的所有特征，尤其是字母的骨架结构没有发生变化，因此对我们以后的识别工作不会造成额外的影响，能够使识别比较顺利的进行。

4.3 定义标准化字母

由于神经网络本身含有记忆的功能,为了使神经网络具有较好的训练效果,本文根据印刷体英文字母定义了 26 个手写大写英文字母的标准化模板。要获得大写英文字母的标准化模板,首先必须对每个英文字母进行处理,以便构造输入向量,如图 4.4 所示。

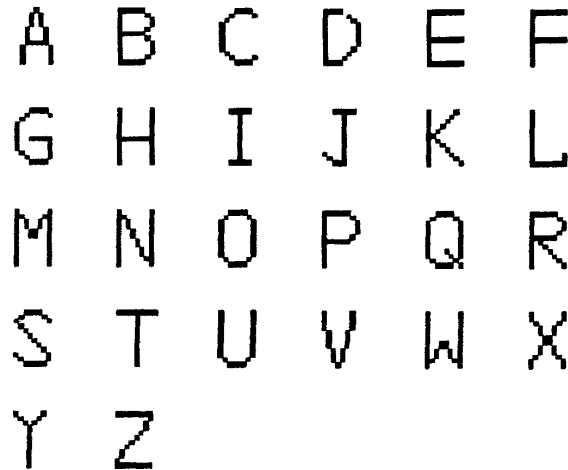


图 4.4 自定义的大写英文字母标准化模版

每个英文字母被分成 14 行 10 列的小块进行数字化,再用一个 $10 \times 14 = 140$ 个元素的列矩阵来表示。140 个元素的列矩阵被定义为一个输入向量矩阵,其相应数据的位置值为 0,显示为黑色,而其他位置值为 1,显示为白色。例如字母“A”和字母“I”用 10×14 矩阵可以表示为:

A = [1 1 1 1 1 1 1 1 1 1	I = [1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 0 1 1 1 1	1 1 1 0 0 0 0 0 1 1
1 1 1 1 0 1 0 1 1 1	1 1 1 1 1 0 1 1 1 1
1 1 1 1 0 1 0 1 1 1	1 1 1 1 1 0 1 1 1 1
1 1 1 1 0 1 0 1 1 1	1 1 1 1 1 0 1 1 1 1
1 1 1 0 1 1 1 0 1 1	1 1 1 1 1 0 1 1 1 1
1 1 1 0 1 1 1 0 1 1	1 1 1 1 1 0 1 1 1 1
1 1 1 0 1 1 1 0 1 1	1 1 1 1 1 0 1 1 1 1
1 1 1 0 0 0 0 0 1 1	1 1 1 1 1 0 1 1 1 1
1 1 0 1 1 1 1 1 0 1	1 1 1 1 1 0 1 1 1 1
1 1 0 1 1 1 1 1 0 1	1 1 1 1 1 0 1 1 1 1
1 1 0 1 1 1 1 1 0 1	1 1 1 1 1 0 1 1 1 1
1 1 0 1 1 1 1 1 0 1	1 1 1 0 0 0 0 0 1 1
1 1 1 1 1 1 1 1 1 1]	1 1 1 1 1 1 1 1 1 1]

在神经网络的训练中，将 14×10 的矩阵转化为一个 140×1 的列向量：

```
A=[1 1 1 1 1 1 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 0 1 0 1 1 1,1 1 1 1 0 1 0 1 1 1,1 1 1 1 0
1 0 1 1 1,1 1 1 0 1 1 1 0 1 1,1 1 1 0 1 1 1 0 1 1,1 1 1 0 1 1 1 0 1 1,1 1 1 0 0 0 0 0 1 1,1 1 0 1 1
1 1 1 0 1,1 1 0 1 1 1 1 1 0 1,1 1 0 1 1 1 1 1 0 1,1 1 0 1 1 1 1 1 0 1,1 1 1 1 1 1 1 1 1 1]'

I=[1 1 1 1 1 1 1 1 1 1,1 1 1 0 0 0 0 0 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1 0
1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1
0 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 1 1 0 1 1 1 1,1 1 1 0 0 0 0 0 1 1,1 1 1 1 1 1 1 1 1 1]'
```

那么，26 个大写英文字母则由 140×26 的输入矩阵表示。在 MATLAB 中，将这一输入矩阵保存在变量 `standard` 中，为下面的识别做准备，代码为：

```
standard=[A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z]
```

第五章 基于 BP 神经网络的手写大写英文字母识别模型

本章我们将依据第二章中所介绍的 BP 神经网络的结构与模型以及算法规则，详细介绍本文所用到的神经网络模型是如何建立的。

5.1 网络的选择

设计一个能够识别26个英文字母的神经网络,意味着每当给训练过的网络一个表示某一个字母的输入时,网络能够正确的在输出端指出该字母。神经网络训练应当是有监督的,训练出对输入端的26组分别表示字母A与Z的数组,能够对应输出端1到26的具体位置。理论已经证明:具有偏差和至少一个S型隐含层加上一个线性输出层的网络,能够逼近任何一个有理函数。虽然增加网络的层数可以更进一步的降低误差,提高精度,但同时使网络复杂化,从而增加了网络权值的训练时间。而误差精度的提高实际上也可以通过增加隐含层中的神经元数目来获得,其训练效果也比增加层数更容易观察和调整。所以,本文采用含有一个隐含层的三层BP神经网络来进行大写英文字母的识别。

5.2 输入层和输出层节点数目的确定

根据模式识别的知识,神经网络的每一个输入节点对应着一个特征值,如果按照统计特征提取和结构特征提取组合的方式,每个字符有成千上万个特征,如果把字符的所有特征直接作为神经网络的输入,可能会取得较好的识别结果,但同时也存在一个严重的问题,那就是神经网络的规模比较庞大。

在第四章中我们介绍了每个字母都可以用一个 10×14 的矩阵来表示,即每一个字母的编码都是一个 140×1 的矩阵,每一个矩阵就是一个输入向量,即输入层的节点数是140个。本文中所用到的样本共有40组,每组由A到Z共26个字母组成,于是输入向量就是一个 140×26 的矩阵,共有40组这样的输入向量。

网络的输出节点数一般是所识别对象的类别数,本文以26个大写英文字母作为类别数,共有26个输出可能,因此人工神经网络的输出层的节点数是26个。网络的设计是为了使其输出矢量在正确的位置上输出为1,而在其他位置上输出为0。然而噪声输入矢量可能导致网络的1或0输出不正确,或出现其他输出值。为了使网络具有抗干扰能力,在网络训练后,再将其输出经过一层竞争网络函数的处理(在MATLAB下为

compet.m),使网络的输出只在最接近输入值的输出位置输出为1,保证在其他位置输出均为0。因为BP算法是有监督的学习算法,因此我们可以指定输入向量分为哪一类,如:字母“A”分为第一类,字母“B”分为第二类,以此类推,数字“Z”分为第二十六类,因此对于从A到Z这二十六个字母,T的每一行代表一个类别,如T(1,1)代表第一类,T(2,2)代表第二类,以此类推。每一列对应一个输入向量,从A到Z,Z为第26个输入向量。其对应的输出如下所示。

$$T = \begin{bmatrix} 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 \end{bmatrix}$$

5.3 隐含层节点数的确定

确定隐含层节点数的最基本原则是:在满足精度要求的前提下取尽可能紧凑的结构,即取尽可能少的隐含层节点数。研究表明,隐含层节点数不仅与输入、输出层的节点数有关,更与需解决的问题的复杂程度和转换函数的形式以及样本数据的特性等因素有关。

若隐含层节点数太少,网络可能根本不能训练或网络性能很差;若隐含层节点数太多,虽然可使网络的系统误差减小,但一方面使网络训练时间延长,另一方面,训练容易陷入局部极小点而得不到最优点,也是训练时出现“过拟合”的内在原因。因此,通常情况下,网络隐含层神经元数目的选择原则是:在能够解决问题的前提下,再加上1到2个神经元以加快误差的下降速度。本文经过大量的试验,确定隐含层神经元的数目为30个。

5.4 改进的 BP 算法比较

在 MATLAB7.0 的神经网络工具箱中,提供了各种改进 BP 算法的函数,例如:Traingd,标准 BP 算法;Traingdm,带有动量项的梯度下降法;Traingda,自适应学习速率法;Traingdx,带有动量项的自适应学习速率法。本文采用 Traingdx 自适应学习速率法训练网络。选择两组训练样本和一组测试样本,分别用上述几种典型的改进 BP 算法训练网络,目标误差定为1,初始学习速率0.02,最大迭代次数5000,结果如表5.2所示。

表 5.1 几种典型 BP 算法训练效果比较

训练函数	BP 算法	训练次数	运行时间(s)	均方误差
Traingdx	自适应学习速率法	1637	102.6820	0.96372
Traingd	标准 BP 算法	5000	354.8640	7.40963
Traingdm	带动量项的梯度下降法	5000	258.5460	4.47358
Traingda	自适应学习速率法	5000	221.8970	4.72201

Traingd, Traingdm 和 Traingda 这三种算法都没有达到预定的目标误差,运算到最大迭代次数停止,只有 Traingdx 达到了目标误差,且迭代次数少,运行速度快。

5.5 学习速率的选择

学习速率决定每一次循环训练中产生的权值变化量。大的学习速率可能导致系统的不稳定;但小的学习速率导致较长的训练时间,可能收敛很慢,不过能保证网络的误差值不跳出误差表面的低谷而最终趋于最小误差值。一般情况下,倾向于选取较小的学习速率以保证系统的稳定性。学习速率的取值范围在 0.01-0.8 之间。可以将网络的学习过程分为两个阶段:第一阶段为粗学习阶段,在此阶段各随机权向量均随着输入模式方向调整,并能大致确定输入模式在竞争层中的映射位置,一般此阶段的学习速率 $\alpha > 0.5$;第二阶段为细调整阶段,此阶段学习集中在相对较小的范围内对连接权向量进行调整,

学习速率应该随学习不断深入而不断减小。这样可以使得学习的网络变得越来越稳定。另外,也可以使竞争获胜的神经元比其邻域内的其它神经元具有更大的学习速率。本文中经过试验,我们发现学习速率取值为 0.02 时收敛速度较快。

5.6 期望误差的选择

选择合适的期望误差值是相对于所需要的隐含层节点数来确定的,因为较小的期望误差值要靠增加隐含层的节点以及训练时间来获得的。本文同时对几个不同期望误差值的网络进行训练,最后考虑运行时间、隐含层神经元数和手写体数字测试结果正确率等综合因素的基础上取值为 0.5。

5.7 其他参数的设定

调用 MATLAB 中的 train.m 函数需要对以下几个参数赋值,如下:

```
net.trainParam.epochs=20000;    %最大训练步长
net.trainParam.lr_inc =1.05;    %增长的学习率
net.trainParam.lr_dec =0.7;    %减小的学习率
net.trainParam.mc =0.95;        %附加动量因子
net.trainParam.min_grad=1e-10   %最小执行梯度
```


第六章 英文字母识别系统的实现

本章将根据前面所给出的系统理论框架和算法讨论了大写英文字母识别系统的设计原则、结构与功能以及在 MATLAB 环境下仿真系统的程序实现，并对系统运行结果进行了初步分析。

6.1 BP 网络训练

本文的网络训练分为两步：

(1) 将已经定义好的标准大写英文字母模板作为输入向量进行训练，用变量 `net` 保存网络训练结果，MATLAB 代码为：

```
[net,tr] = train(net,standard,T);
```

其中 `standard` 是输入向量，`T` 是目标向量；

(2) 将 `net` 训练后得到的所有神经元的权值赋值给 `netn`，此时再将训练样本输入网络 `netn` 继续训练，MATLAB 代码：

```
netn = net;
```

```
[netn,tr] = train(netn,P,T);
```

其中 `P` 是训练样本构成的输入向量，`T` 是目标向量。

训练结束后，进入测试阶段。用训练好的网络 `netn` 测试样本。本文的测试样本共有 520 个，共二十组，每组包含 A 到 Z 二十六个字母。采用 MATLAB 仿真函数 `sim.m` 计算测试结果。

6.2 系统的结构与功能

本文的大写英文字母识别系统主要可分为三个部分：输入图片、归一化和识别。其具体的结构如下图 6.1 所示：

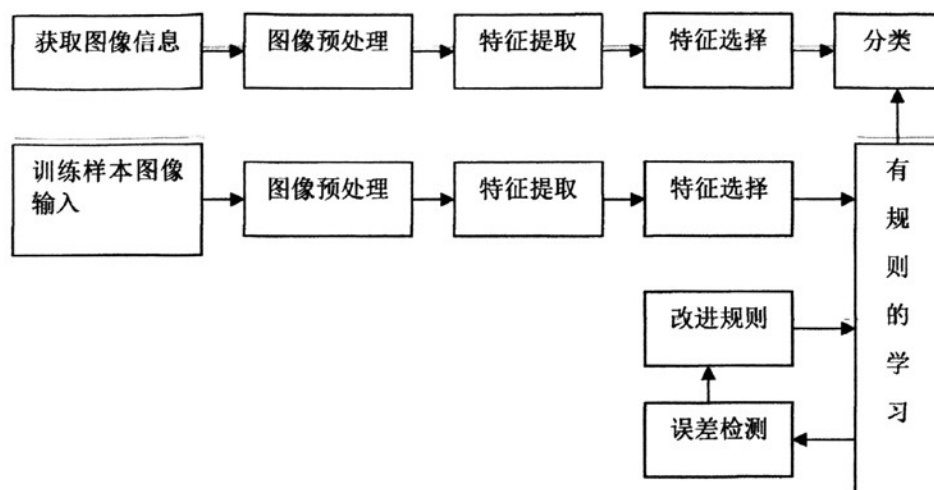


图 6.1 系统结构图

6.3 系统的程序实现

本系统是在 MATLAB 环境下进行的仿真模拟，为了使本系统更直观且易于操作，利用 MATLAB 自带的 GUI (Graphical User Interface)，即图形用户接口，来进行可视化编程并提供友好的交互界面。图 6.2 是本系统的初始化界面。

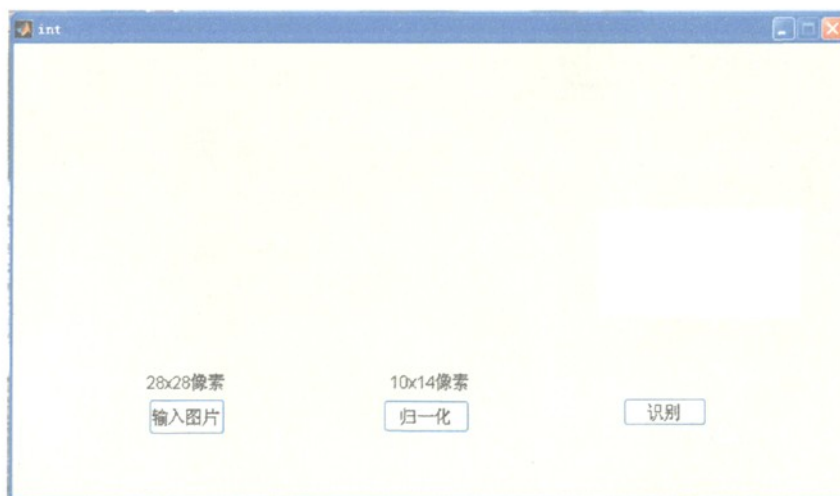


图 6.2 系统的初始化界面图

6.4 实验结果

本文所设计的仿真系统包括三个模块：输入图片、归一化以及识别，它们虽然在算

法设计上是相互独立的,但在功能上却是相互依存的,前一个模块的运行结果是后一个模块的输入,整个系统性能的好坏与每个模块都密切相关。下表 6.1 给出了运用该系统对 520 个英文字母测试样本进行识别的实验统计表。

表 6.1 BP 网络手写字母识别实验统计表

字母	测试样本数	正确识别数	正确率(%)
A	20	13	65.00
B	20	14	70.00
C	20	16	80.00
D	20	14	70.00
E	20	18	90.00
F	20	18	90.00
G	20	10	50.00
H	20	17	85.00
I	20	15	75.00
J	20	14	70.00
K	20	13	65.00
L	20	17	85.00
M	20	11	55.00
N	20	12	60.00
O	20	16	80.00
P	20	16	80.00
Q	20	10	50.00
R	20	12	60.00
S	20	12	60.00
T	20	17	85.00
U	20	15	75.00
V	20	14	70.00
W	20	13	65.00
X	20	16	80.00
Y	20	15	75.00

中北大学学位论文

Z	20	16	80.00
总计/平均	520	374	71.92

结论

本文提出了用 BP 神经网络进行手写体字母识别的方法。实验结果表明, BP 神经网络既能完成特征提取又能完成分类功能, 经过足够多的原始样本训练后的 BP 神经网络, 其容错性和识别性比较好。

通过本文的识别系统可以看到, 字母“E”和“F”的识别结果最好, 达到了 90%; 字母“G”和“Q”的识别率不高仅为 50%, “G”被误识为“C”, “O”和“Q”, “Q”被误识为“B”, “G”和“R”。所有测试样本的平均识别正确率为 71.92%。

今后工作展望: 手写体字母的特征提取是一个非常复杂的问题, 可以考虑先用比较适合于做特征提取的自组织特征映射网络提取样本特征, 然后再用 BP 神经网络完成分类, 充分利用两种网络的优点可能达到更好的识别效果。

附录

输入:

```
clear all
global fname fpath see
fname='3G.bmp';
[fname,fpath]=uigetfile('*.*bmp', '打开');
see=fname;
I=imread(see);
subplot('position',[-0.05,0.3,0.5 0.5]);
imshow(I,'notruesize');
```

归一化:

```
ans=imread(fname);
for i=2:27
    for j=2:27
        n=i/(28/14);
        n=ceil(n);
        m=j/(28/10);
        m=ceil(m);
        M(n,m)=(ans(i,j)+ans(i,j+1)+ans(i,j-1)+ans(i-1,j))/4;
    end
end
subplot('position',[0.45,0.3,0.15 0.45]);
imshow(M,'notruesize');
```

自定义标准字母:

```
A = [1 1 1 1 1 1 1 1 1 1 ... B = [1 1 1 1 1 1 1 1 1 1 ... C = [1 1 1 1 1 1 1 1 1 1 ...
      1 1 1 1 1 0 1 1 1 1 ...      1 0 0 0 0 0 0 1 1 1 ...      1 1 1 1 0 0 0 0 1 1 ...
      1 1 1 1 0 1 0 1 1 1 ...      1 0 1 1 1 1 1 0 1 1 ...      1 1 1 0 1 1 1 1 0 1 ...
      1 1 1 1 0 1 0 1 1 1 ...      1 0 1 1 1 1 1 1 0 1 ...      1 1 0 1 1 1 1 1 1 1 ...
      1 1 1 1 0 1 0 1 1 1 ...      1 0 1 1 1 1 1 1 0 1 ...      1 0 1 1 1 1 1 1 1 1 ...
      1 1 1 0 1 1 1 0 1 1 ...      1 0 1 1 1 1 1 0 1 1 ...      1 0 1 1 1 1 1 1 1 1 ...
      1 1 1 0 1 1 1 0 1 1 ...      1 0 0 0 0 0 0 1 1 1 ...      1 0 1 1 1 1 1 1 1 1 ...
      1 1 1 0 1 1 1 0 1 1 ...      1 0 1 1 1 1 1 0 1 1 ...      1 0 1 1 1 1 1 1 1 1 ...
      1 1 1 0 0 0 0 0 1 1 ...      1 0 1 1 1 1 1 1 0 1 ...      1 0 1 1 1 1 1 1 1 1 ...
      1 1 0 1 1 1 1 1 0 1 ...      1 0 1 1 1 1 1 1 0 1 ...      1 0 1 1 1 1 1 1 1 1 ...
      1 1 0 1 1 1 1 1 0 1 ...      1 0 1 1 1 1 1 1 0 1 ...      1 1 0 1 1 1 1 1 1 1 ...
      1 1 0 1 1 1 1 1 0 1 ...      1 0 1 1 1 1 1 0 1 1 ...      1 1 1 0 1 1 1 1 0 1 ...
      1 1 0 1 1 1 1 1 0 1 ...      1 0 0 0 0 0 0 1 1 1 ...      1 1 1 1 0 0 0 0 1 1 ...
      1 1 1 1 1 1 1 1 1 1]';      1 1 1 1 1 1 1 1 1 1]';      1 1 1 1 1 1 1 1 1 1]';
```

[illegible]

43

[illegible]

[illegible]

参考文献

- [1] 翟海金.手写体数字识别方法的研究与实现.硕士学位论文.南京理工大学,2005.
- [2] 荆涛,王仲.光学字符识别技术与展望[J]. 计算机工程.2003,2:1-2.
- [3] Mori S.Historical Review of OCR Research and Development. Proceedings of IEEE,1992,80(7): 1029-1058 W.K.
- [4] 刘迎建,戴汝为.识别在线手写汉字的字型结构排序法.自动化学报.1988,14.3:198-202.
- [5] 费越,汪力新,戴汝为. 竞争监督学习法在集成型识别系统中的应用. 自动化学报. 1999,25.3:303-308.
- [6] 徐蔚然. 统计分类器的银行票据自动处理系统若干关键技术研究. 博士学位论文. 北京邮电大学,2003.
- [7] Govindan V.K.,Shivaprasad A.P.,Character Recognition-A Review,Pattern Recognition, 1990.
- [8] C.Y.Suen,M.Bertoh,and S.Mori,Automatic recognition of hand-printed characters-The state of the art,Proc.IEEE,vol.68,pp.469-487,1980.
- [9] C.Downton and S.Impedovo eds.,Progress in Handwriting Recognition,World Scientific Publishing Co.Pte.Ltd.,1997.
- [10] S.Impedovo,P.S.P.Wang and H.Bunke(eds), Automatic Bankcheck Processing,World Scientific Publishing Co.Pte.Ltd.,1997.
- [11] 朱学芳. 模糊集在手写数字识别中的应用. 模糊系统与数学.1996-02: 70-75.
- [12] 胡小锋,赵辉. 图像处理与识别实用案例精选[M]. 北京:人民邮电出版社, 2004.264=384.
- [13] 利达. 神经网络集成识别手写体数字研究[D]. 硕士学位论文. 北京邮电大学, 2007.
- [14] 李冰. 基于多神经网络集成的手写体字符识别[D]. 硕士学位论文. 华中科技大学, 2005.
- [15] Amari, S., and Nagaoka H., Information Geometry, New York: AMS and Oxford University Press. 2000.
- [16] 黄德双. 神经网络模式识别系统理论. 电子工业出版社. 1996. 319-323.
- [17] Taylor, J.G., "Neural computation: The historical background," in E. Fiesler and R. Beale, eds., Handbook of Neural Computation, New York: Oxford University Press, 1997

- [18] Rosenblatt, F., "The Perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*. 1958, Vol. 65. P 386-408.
- [19] Widrow, B., "An adaptive adaline neuron using chemical memistors. Technical report, Stanford Electronics Laboratory. 1960.
- [20] Widrow, B., "Generalization and information storage in networks of adaline neurons," in M.C. Yovitz, G.T. Jacobi, and G.D. Goldstein, eds, *Self-Organizing Systems*, 1962.P435-461, Washington, DC: Spartan Books.
- [21] Widrow, B. and M.E. Hoff, Jr., "Adaline switching circuits," *IRE WESCON Convention Record*, 1960.P 96-104.
- [22] Hopfield, J.J., "Neural networks and physical systems with emergent collective computational abilities," *Proceedings National Academy of Science*, 1982.vol. 79, P2554-2558.
- [23] Hopfield, J.J., "Neurons with graded response have collective computational properties like those of two-state neurons," *Proceedings National Academy of Science*, 1984. vol. 81, P3088-3092.
- [24] 丛爽, 面向MATLAB工具箱的神经网络理论与应用[M]. 合肥:中国科学技术大学出版社, 2003。
- [25] Rumelhart D,Hinton G, Williams R.Learning Representation by Back Propagation Errors[J].*Nature*,1986,(323)P533-536.
- [26] 李士勇. 模糊控制-神经控制和智能控制论. 哈尔滨工业大学出版社, 1998.9.
- [27] 郭晶, 孙伟娟. 神经网络理论与 MATLAB7 实现. 电子工业出版社. 2006. 20-21.
- [28] Widrow, B. and M.A. Lehr, "30 years of adaptive neural networks: Perceptron, madaline, and back-propagation," *Proceeding of the Institute of Electrical and Electronic Engineers*, 1990. vol. 78, pp.1415-1442.
- [29] Rumelhart, D.E., G.E. Hinton, and R.J. Williams, "Learning representations of back-propagation errors," *Nature (London)*, 1986a.vol. 323, pp. 533-536.
- [30] 焦李成,神经网络系统理论[M],西安电子科技大学出版社,1991.
- [31] 西奥多里德斯(希)等著,李晶皎等译. 模式识别. 电子工业出版社, 2006.
- [32] 余成波. 数字图像处理及 MATLAB 实现. 重庆大学出版社, 2003. 5.
- [33] 刘直芳. 数字图像处理与分析. 清华大学出版社, 2006. 9.

[34] 王慧琴. 数字图像处理. 北京邮电大学出版社, 2006. 2-15.

[35] 四维科技, 胡小峰, 赵辉, Visual C++/MATLAB 图像处理与识别实用案例精选. 北京:人民邮电出版社,2004.

攻读硕士期间发表的论文及所取得的科研成果

1. 高靛, 白艳萍. 微加速度计产品分类器数学模型. 中北大学学报(自然科学版)(增刊), 2008(29)增刊(2): 68-71.
2. 高靛, 白艳萍. 基于BP神经网络的英文字母识别. 中北大学学报(自然科学版)(增刊), 2008(29)增刊:128-130.

致谢

在硕士研究生阶段三年的学习中，在导师白艳萍教授的精心指导下，我一直从事图里与模式识别模式分类的相关领域，特别是字符识别方面的研究。在最后完成论文

回首往事，展望未来，忍不住感恩之情溢满胸怀，感谢老师的谆谆教诲，同学和的热情关心和帮助，感谢所有诚挚帮助我的人。

在此要感谢白艳萍老师多年孜孜不倦的教诲以及生活上的悉心关怀，我的每一份收获源于导师的辛勤汗水，特别在论文撰写阶段，导师花费了大量的时间和精力详细全文，并提出了很多宝贵的意见。导师严谨的治学态度、诲人不倦的精神永远值得习，并必将对我将来的工作和生活产生深刻的影响。

此外，还要感谢审阅本文的教授、专家和学者，感谢你们对本文的审阅和悉心指导。

基于BP神经网络的手写英文字母识别

作者：[高靓](#)
学位授予单位：[中北大学](#)

本文读者也读过(5条)

1. [李永江](#) [基于LVQ神经网络的手写英文字母识别](#)[学位论文]2008
2. [吴迪](#), [赵鹤鸣](#), [陶智](#), [Wu Di](#), [Zhao Heming](#), [Tao Zhi](#) [基于改进BP算法的英文字母识别](#)[期刊论文]-[计算机工程与应用](#)2006, 42(23)
3. [谢丽娟](#), [丁正生](#), [陈俏](#), [XIE Li Juan](#), [DING Zheng Sheng](#), [CHEN Qiao](#) [基于BP网络的字母识别](#)[期刊论文]-[微型机与应用](#)2009, 28(18)
4. [邓铭辉](#), [孙枫](#), [张志](#) [一种基于MATLAB的手写字母的神经网络识别方法](#)[期刊论文]-[应用科技](#)2001, 28(10)
5. [丛爽](#), [陆婷婷](#), [Cong Shuang](#), [Lu Tingting](#) [用于英文字母识别的三种人工神经网络的设计](#)[期刊论文]-[仪器仪表学报](#)2006, 27(z3)

引证文献(1条)

1. [段明秀](#), [何迎生](#) [基于LVQ神经网络的手写字母识别](#)[期刊论文]-[吉首大学学报（自然科学版）](#) 2010(2)

本文链接：http://d.g.wanfangdata.com.cn/Thesis_Y1509334.aspx