

纵横投影的七段码“8”字型手写数字识别

马碧燕, 郑胜林, 赵全友

(广东工业大学 信息工程学院, 广东 广州 510640)

摘要: 根据电子屏幕只需七段码“8”字型就可将 0~9 这 10 个数字表示出来的情况, 设想把数字作为一个整体放到七段码“8”字型中进行识别. 本文提出了一种纵横投影的七段码“8”字型手写数字识别, 其做法是根据数字的外形特征将数字对七段码“8”字的各框上进行投影, 将手写符转化为标准符进行识别, 从而简化字符特征的提取. 与传统识别方法相比, 更容易实现. 实验证明了这种方法的有效性.

关键词: 数字识别; 七段码“8”字型; 投影

中图分类号: TP391.41

文献标识码: A

文章编号: 1007-7162(2006)03-0113-05

手写体数字识别是多年的研究热点. 目前, 一些手写体数字识别产品已在许多实用系统应用, 如在邮政编码、税务报表、统计报表等需要处理大量字符信息录入的场合^[1]. 在过去的数十年中, 研究者们对识别方法进行了大量的研究, 提出了许许多多的识别方法. 有人根据电子屏幕只需七段码“8”字型就可将 0~9 这 10 个数字表示出来这种思路提出了一种新颖的方法——基于“8”字型的手写体数字识别方法. 常用的是斜投影七段码“8”字型手写数字识别, 这种方法首先对字体进行初步判断, 确定需要一个还是两个光源再进行投影. 例如数字“1”、“7”就只需一个光源向上及向右投影, 而“8”则需要两个光源, 一个分配在上半圈向四面八方投影, 另一个分配在下半圈向四面八方投影. 这种方法在投影前要对字符进行预判断, 而这种预判断存在不确定性. 另外, 这种投影方法用的是斜线投影, 模拟上有交点的两条线段在数字图像中会缺失.

本文对传统的投影方法进行了大量研究, 提出了纵横投影的七段码“8”字型手写数字识别. 这种方法不需要确定光源, 对任一数字均向七个框上投影, 投影线段是水平或垂直的直线段, 不会造成交点的缺失. 这种方法判决简单, 而且纵横直线段投影克服了斜线段投影需要确定光源及缺失交点的不足.

1 横投影的七段码“8”字型手写数字投影

在对各框进行投影前, 先对数字进行预处理, 把数字变成一定规格的、合乎要求的形式. 预处理包括 3 步, 投影也分 3 步, 如图 1 所示.

1.1 预处理

预处理包括细化、切割及规划这几步^[2].

细化也称骨架化, 其实是对原图进行处理, 达到从粗到细的过程. 细化处理能去除字符上不影响连通性的轮廓像素, 获得单位宽度的中心骨架, 从而方便对字符的轮廓跟踪. 本系统使用的

收稿日期: 2005-07-06

作者简介: 马碧燕 (1982-), 女, 在读研究生, 主要研究方向为电力系统及其自动化.

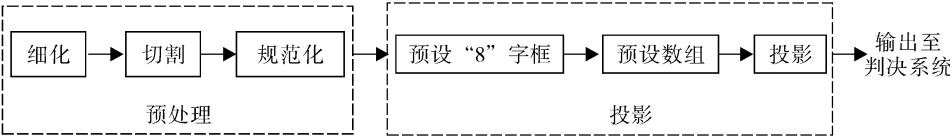


图 1 纵横投影实现方法

是基于膨胀和腐蚀的形态操作来进行细化处理的^[1-3]。

切割即从一幅有白边的图中把数字部分切下来, 去除边框。切割的基本思想如下: 从上往下逐行搜索, 碰到 $f(x, y) = 1$ 则将此行定为起始行, 将该行的值赋给一个变量, 如 $up = x$; 同理, 从左往右逐列搜索, 碰到白点后将该列定为起始列, 将该列的值赋给一个变量, 如: $left = y$; 从下往上逐行搜索, 碰到 $f(x, y) = 0$ 则将此行定为末行, 令 $down = x$; 从右往左逐列搜索, 碰到白点后将该列定为末列, 令 $right = y$ 。最后以这 4 条边为界将数字切下来^[4]。

规范化就是将切割好的图变成一幅 24×16 像素大小的图。规范化的思想是从原图中取其中一些点, 而筛去另外一些点, 取点的规则如下: 假设从一幅大小为 $[m, n]$ 的图变为 24×16 像素的图, 则 $f(x, y)$ 中的点 (x, y) 满足 $x \times 24 \div m$ 及 $y \times 16 \div n$ 都为整数的点留下, 其它点删去, 即 $\text{mod}(i \times 24 \div m, 1) = 0$ 和 $\text{mod}(j \times 16 \div n, 1) = 0$ 该自编程序效果不好, 所以最后改用库函数^[5]。

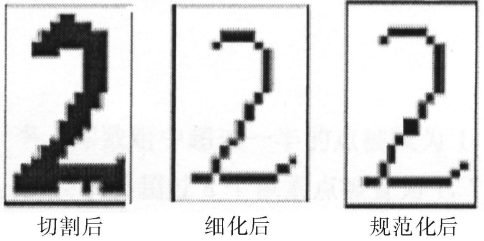


图 2 对数字“2”进行预处理后的效果图

预处理效果如图 2 表所示

1.2 投影

纵横投影的七段码“8”字型手写数字识别法对任一数字均向七个框上投影, 七框分别为上框、中框、下框、左上框、右上框、左下框、右下框。投影线段是水平或垂直的直线段, 七段码“8”字的七个框都投影完后每条框的值不是 0 就是 1, 由 $2^7 = 128$ 可知投影后各边框的值的组合只有 128 种。纵横线段投影法是分块进行横竖扫描的一种方法, 各步骤解释如下。

1.2.1 预设空白“八”段码“8”字框

预设一个空白的大小为 24×16 像素的空间 $u(x, y)$ 来放投影后的七段码“8”字型的数字。其中上框为第 1 行, 即 $u(1, y)$; 中框为第 12 行, 即 $u(12, y)$; 下框为最后一行, 即 $u(24, y)$; 其中 $y \in [1, 16]$ 。左上框为第 1 列的前 12 行, 即 $u(x, 1)$; 右上框为第 16 列的前 12 行, 即 $u(x, 16)$; 其中 $x \in [1, 12]$ 。左下框为第 1 列的 12 行至 24 行, 即 $u(i, 1)$; 右下框为第 16 列的 12 行至 24 行, 即 $u(i, 16)$; 其中, $i \in [12, 24]$ 。

1.2.2 预设数组

预设 7 个空白数组 $u_1 \sim u_7$ 分别用来放 7 个框每点的投影结果, 分别给它们一个与上述各框大小一致的零空间。

1.2.3 向各框投影

1) 向上框投影: 从第 1 列到第 16 列逐列扫描 1 至 8 行, 若第 1 列的 1 至 8 行任一行有黑点, 则存储上框投影结果的数组 $u_1(1, y)$ 的第一点置 1, 即 $u_1(1, 1) = 1$ 否则置 0 $u_1(1, 1) = 0$ 。若第 2 列的 1 至 8 行任一行有黑点, 则存储上框投影结果的数组 $u_1(1, y)$ 的第二点置 1, 即 $u_1(1, 2) = 1$ 否则置 0 $u_1(1, 2) = 0$ 。重复上述操作至第 16 列。

2) 向中框投影: 从第 1 列到第 16 列逐列扫描 9 至 16 行, 若某列有黑点, 则存储中框投影结果的数组 $u_2(1, y)$ 的相应点置 1, 否则置 0

3) 向下框投影: 从第 1 列到第 16 列逐列扫描 17 至 24 行, 若某列有黑点, 则 $u_3(1, y)$ 中的相应点置 1, 否则置 0

4) 向左上框投影: 从第 1 行到第 12 行逐行扫描 1 至 8 列, 若某行有黑点, 则 $u_4(x, 1)$ 中对应的点置 1, 否则置 0

5) 向右上框投影: 从第 1 行到第 12 行逐行扫描 9 至 16 列, 若某行有黑点, 则 $u_5(x, 1)$ 中对应的点置 1, 否则置 0

6) 向右下框投影: 从第 13 行到第 24 行逐行扫描 9 至 16 列, 若某行有黑点, 则 $u_6(x, 1)$ 中对应的点置 1, 否则置 0

7) 向左下框投影: 从第 13 行到第 24 行逐行扫描 1 至 8 列, 若某行有黑点, 则 $u_7(x, 1)$ 中对应的点置 1, 否则置 0

1.3 对数组加工

对数组 $u_1 \sim u_7$ 进行统计, 将统计的阈值定为一半, 即数组中超过一半的点被赋为 1, 则与此数组相对应的“日”字框就置 1, 否则置 0 $u_1 \sim u_3$ 要求要超过 8 个像素点被置为 1, “日”字框上对应的框就整条置 1, 否则置 0 $u_4 \sim u_7$ 要求要超过 6 个像素点被置为 1, “日”字框上对应的框就整条置 1, 否则置 0

对各框逐步投影的效果图如图 3 所示. 对各数字投影的图如图 4 所示.

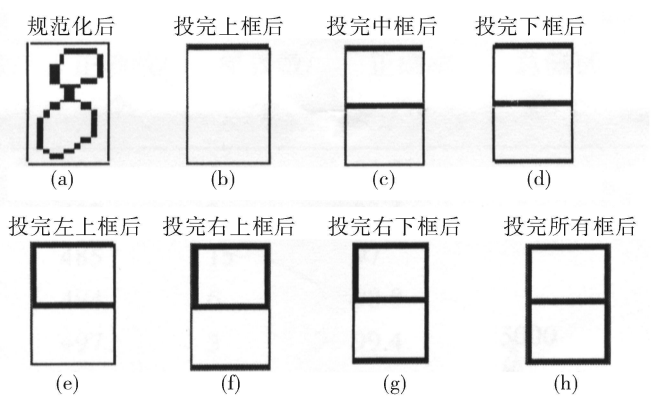


图 3 对数字“8”进行投影时各步骤效果图

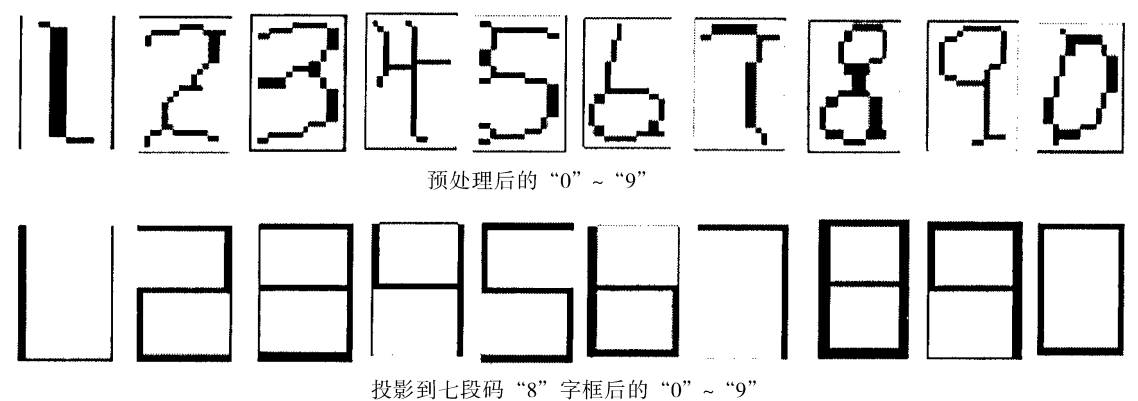


图 4 各数字的投影图

2 投影后的识别处理

对数字投影后就要对这些七段码的“8”字进行判决, 本文提供了一种可供参考的判决分类树, 树干如下:

从上框是否为 0 分开两类.

上框为 0 时, 又再继续看下框是否为 0 又再分两类, 一直分下去.

上框为 1 时, 从中框是否为 0 分为两类: 中框为 0 时, 以左上框是否为 0 再继续分下去. 中框为 1 时, 以左上框是否为 0 分为两类.

判决法则因人而异, 是非常灵活的, 有时只要修改一个子叶的判决, 都会提升较大的识别率. 这需要研究人员更多地尝试与修改.

3 实验结果

上述方法已在 MATLAB 上编程实现^[6], 并对该方法进行了大量实验, 实验中采用广东工业大学信息工程学院图像研究室提供的数字采集样本, 表 1 显示出了其中一次的实验结果. 实验中的样本为 0~9 的手写数字, 共 5 000 个, 每个数字 500 个. 实验结果表明识别率精度为 98.68%.

4 总结

本论文提出了基于纵横投影的七段码“8”字型手写数字识别系统, 与传统的方法相比, 它不需要进行初次判断, 且对数字的初始化要求低, 即使是对书写不好的数字也有一定的识别效果. 例如, 对没闭合的数字“0”或“8”, 若不闭合的空间很少, 不足以影响投影结果的话, 完全不需任何的加工就能判别出来.

该系统能达到 98.7% 的总识别率, 只是对数字“2”和“7”的识别率较低, 分别为 97% 和 96.6%, 这是因为“7”没有写成垂直, 本该投到右下框的变成投到左下框了, 被误判为“2”. 因此对“2”和“7”的改善是最为迫切的, 对“7”的改善是通过投影前进行判断, 把“1”和“7”分别开来, 单独只对上框投影. 对“2”的改善是通过增强预处理或者通过加一些字符征来进行识别. 只要数字“2”和“7”的识别率上去了, 整个系统的识别率就会上升到一个更为实用的层次.

参考文献:

[1]柳回春, 马树元. 基于结构特征的手写体数字识别算法[J]. 计算机工程, 2002, 28(11): 28-60

[2]向华, 李天牧. 手写体数字分割的研究[J]. 云南大学学报, 1995, 17(1): 74-80

[3]章毓晋. 图象处理和分析[M]. 北京: 清华大学出版社, 2003: 26-40

[4]柳回春, 马树元. UK 心理测试自动分析系统的手写体数字识别[J]. 北京理工大学学报, 2002, 22(5): 599-603

[5]徐飞, 施晓红. MATLAB 应用图像处理[M]. 西安: 西安电子科技大学出版社, 2003: 38-47

[6]楼顺天, 陈生潭, 雷虎民. MATLAB 5.x 程序设计语言[M]. 西安: 西安电子科技大学出版社, 2001: 95-102

Writing Numbers Identification by Projection in Seven Segments Frame “8”

MA Bi-yan, ZHENG Sheng-lin, ZHAO Quan-you

(Faculty of Information Engineering Guangdong University of Technology, Guangzhou 510640, China)

Abstract According to the fact that the electronic screen just needs a frame “8” to denote the 10 numbers from 0 to 9, it is attempted to take the numbers as a whole to be put to the frame “8” and then recognized. This text is put forward according to the traditional recognition method, the baseline is used to project while bias is used in traditional recognition method. Projection is made to each frame of the frame “8” according to the shell character of the digit, and hand-writing numbers are changed to standard seven segments frame numbers, which can predigest the distilling of the feature of the number. Compare to the traditional recognition method, this one can be carried out much more easily. Its validity is proved in our experiment.

Key words digit recognition, frame “8”; projection

(上接第 112 页)

Neural Network Face Recognition Based on Length Extension Technique

ZHANG Chen, PAN Bao-chang, ZHENG Sheng-lin, LU Yi-tao

(Faculty of Information Engineering Guangdong University of Technology, Guangzhou 510640, China)

Abstract After a great deal of investigations on the principle of BP model and the structure of the neural network, this paper manages to demonstrate how to increase the convergent speed of the BP neural network by optimizing the input of the network. Furthermore, a creative optimization technique—length extension technique, is presented here to improve the performance of neural network. After lots of experiments, this paper successfully increases the convergent speed of the BP neural network.

Key words face recognition, K-L expansion, BP neural network, length extension technique (LET)