

国际音标图像字符细化方法

孙孝坤, 黄继风

(上海师范大学信息与机电工程学院, 上海 200234)

摘 要: 对图像文字进行细化有助于突出文字的形状特点和减少冗余的信息量, 在文字识别领域有着重要的应用。在分析研究传统细化算法后, 针对传统细化出现的畸变、细化不完全现象, 提出了一种对国际音标图像字符的细化方法。该算法通过对文字区域的边缘分类标记, 并判断被标记点是否满足可去除条件, 然后逐步去除边缘像素点, 最终能让国际音标图像字符的宽度细化到一个像素宽度。针对国际音标图像字符的实验表明, 该算法能够准确地对国际音标图像字符进行细化, 且简单高效。

关 键 词: 图像; 国际音标; 字符; 细化

中图分类号: TP 391

DOI: 10.11996/JGJ.2095-302X.2018020214

文献标识码: A

文 章 编 号: 2095-302X(2018)02-0214-07

A Thinning Method for International Phonetic Alphabet Characters

SUN Xiaokun, HUANG Jifeng

(The College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai 200234, China)

Abstract: To refine image characters can help to highlight the shape features of the character and reduce the amount of redundant information, which has important applications in the field of character recognition. After analyzing and studying the traditional thinning algorithm, aiming at the distortion and incomplete phenomenon of traditional refinement, this paper proposes a refinement method for the characters of international phonetic symbols. The algorithm makes it possible to refine the width of the international phonetic image character to a pixel width by marking the edge of the text area and judging whether the marked point satisfies the removable condition and then gradually removing the edge pixel. Experiment on the international phonetic alphabet character image shows that the algorithm proposed in this paper can accurately refine the international phonetic alphabet character image, simple and efficiently.

Keywords: image; international phonetic alphabet; character; thinning

随着国际化的发展, 各个国家之间的交流越来越密切, 为了更好地沟通, 需要一种统一的语音系统, 国际音标就是记录所有语音的统一音标。国际音标的产生是世界优秀文化的结晶, 闪烁着世界人民智慧的光芒。然而, 如何把大量传统纸质文献中的国际音标进行数字化, 是一个比较复杂而庞大的工程。对这些文献进行数字化最常见的手段是先对

文献进行扫描, 然后再对数字图片字符直接进行识别, 这种方法对文字字符不但不能准确地识别, 而且会大大降低识别的效率。对图像字符进行细化可以有效地解决这些问题, 因为细化的目的是得到图像文字的骨架, 去除图像上多余的像素, 从而突出图像文字的主要特征。对数量庞大的国际音标图像字符进行细化, 不但会提高识别准确度, 还会提高

收稿日期: 2017-06-17; 定稿日期: 2017-08-11

第一作者: 孙孝坤(1992-), 男, 河南驻马店人, 硕士研究生。主要研究方向为数字图像处理与模式识别。E-mail: 2323350313@qq.com

通信作者: 黄继风(1963-), 男, 河南郑州人, 教授, 博士。主要研究方向为模式识别与数字处理、生物信息学、视频图像识别等。

E-mail: jfhuang@shnu.edu.cn

识别速度, 进而提高工作效率。提取识别图像中的文字信息, 是把图像信息转化为文本信息的一个重要组成部分。而对图像文字细化又是提取识别图像中文字信息的首要工作, 并且对图像文字细化的好坏将直接影响图像文字识别的准确与否, 所以图像文字的细化是图像文字识别与再存储领域的一个重要的研究方向。

现有的细化算法如文献[1]采用了 ZS 算法; 文献[2]归纳和分析了多种经典细化方法。关于细化方面的算法最近几年有: 文献[3]使用的击中-击不中方法和文献[4-5]提出的细化方法, 但这些方法均是对指纹进行细化的, 而针对字符的新的细化算法却很少出现; 文献[6]提出了一种广义的草书和非草书文字细化算法; 文献[7]提出了一种手写体文字细化方法; 文献[8]对 15 种细化算法进行了测评, 但这些方法也均不是针对音标字符细化设计的。由于国际音标字符字迹的粗细不同, 弧线部分较多且弧度也不同, 字迹交叉、链接也是各种各样的, 用现有的一些细化算法进行细化, 如在实验部分所展示的结果会出现畸变、细化不完全现象。虽然人们可以对畸变进行修正, 但作用是有限的; 文献[9]使用的 EPTA 算法及类似的改进算法^[10], 这些算法对字符的斜线和弧线部分细化时会出现畸变。还有一些细化算法的抗干扰性差, 比如 Hilditch 算法^[11], 当图像中有噪声时, 细化结果不能达到很好的效果; 文献[12]提出及列举的一些细化算法虽然对文字的细化效果很好, 但是细化所用时间较长且效率低。

本文提出的国际音标细化算法通过对文字区域的边缘分类标记, 并按标记对满足可去条件部分逐步细化, 针对性强, 能够有效地解决细化畸变、细化不完全的问题, 而且具有较强的鲁棒性, 计算复杂度底, 能够快速高效地对国际音标进行细化。本文提出的细化算法虽然是针对国际音标图像字符的, 但是也适用于对一般文字字符的细化, 可以为文字细化研究提供一种新思路。

提出的对国际音标图像字符细化的流程如图 1 所示。首先对国际音标图像进行二值化处理, 目标为 1, 背景为 0。然后用标记法对文字区域边缘分类进行标记, 再计算标记后的边缘像素点周围的邻域编码值, 并判断编码值是否满足文字连接可去点条件, 如果满足则删除, 即此边缘像素点的像素值赋为 0; 否则保留, 即此边缘像素点的像素值赋为 1。最后保存处理后的图像, 再反复调用算法对保存的图像处理, 直到与最近一次保存的图

像相比没有像素点变化, 即完成了对国际音标图像字符细化, 并且此时图像字符的宽度为一个像素。

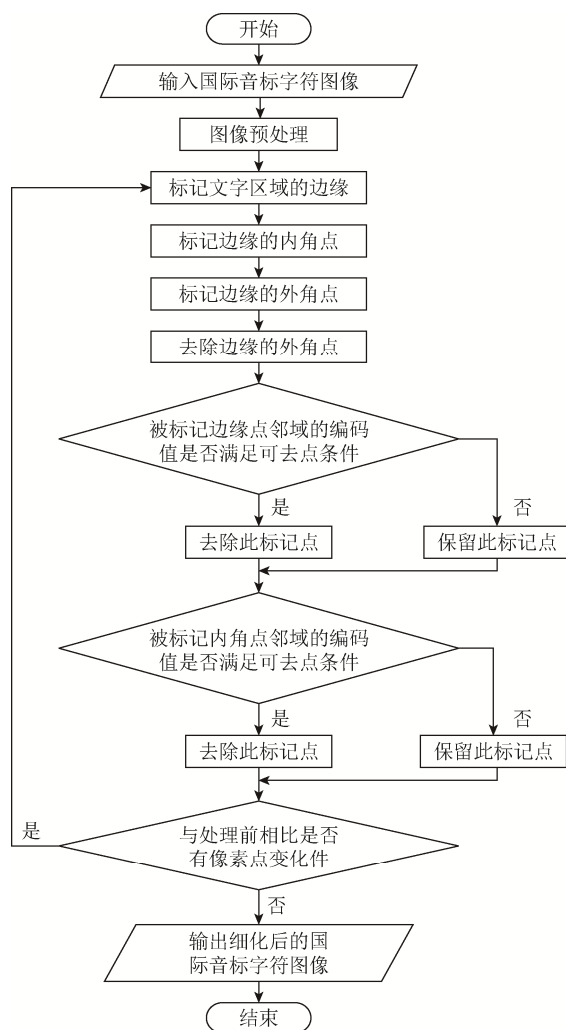


图 1 国际音标图像字符细化流程图

1 图像的预处理

对国际音标图像字符细化之前, 首先要对图像进行预处理。图像预处理是实现图像文字高精度细化的基础和前提, 其中包括对图像的滤波和二值化处理^[13]。

输入的字符图像中往往有噪声存在, 图像去噪是图像预处理中一个基本且重要的环节, 其可以从复杂的信息中提取所需的信息, 并抑制干扰信息。对国际音标字符图像去噪后可得到一个对比度比较高, 比较清楚干净的灰度图像。常用的去噪声方法有高斯滤波、均值滤波、中值滤波和最小均方差滤波等。本文采用的是 3×3 邻域的中值滤波。

对国际音标图像字符细化需要先将前景和背景分离, 并且重点是对文字进行细化, 所以只需要提取图像中的文字, 可将文字以外的部分都当

作背景。对图像进行二值化处理,即图像中文字区域标记为 1(像素点值置为 1),其余区域标记为 0。本文采用最大类间方差法,即 OTSU^[14](大津法)求取二值化处理的阈值 T ,该方法基于图像的灰度直方图,以目标和背景类间方差最大或最小为阈值选取准则,计算简单,可以满足实时性要求。

假设原始图像为 $g(x,y)$,阈值为 T ,对图像进行二值化处理,即

$$g(x,y)=\begin{cases} 1, & g(x,y) \leq T \\ 0, & g(x,y) > T \end{cases} \quad (1)$$

2 文字的细化

细化是一种简化图像的方法,在图像处理和模式识别中得到了广泛的研究和应用。细化的好处主要是能够保持图形的连通性和拓扑关系的不变性,细化后的骨架信息比轮廓线和游程编码更直观,图形特征易提取。对被处理的文字图像进行细化有助于减少冗余的信息量,而突出主要特征。本文提出的针对国际音标图像字符细化方法,与传统的细化算法相比,该方法设计灵活,便于实现,是一种非常有效的细化算法,能避免图像细化的毛刺现象,保持图像连通性。

该算法的细化部分可分为两大步骤:①分类识别标记文字区域的边缘;②对标记过的文字区域边缘按查表法进行取舍。

2.1 识别标记文字区域边缘

(1) 识别并标记文字区域的边缘。将经过预处理后的原国际音标字符图像矩阵 g 另存为 g_1 ,如果原图像矩阵 g 中像素点 $g(x,y)$ 的值是 1,并且其邻域上下左右的 4 个像素点值中至少有一个是 0,如图 2(1)~(14)所示,则像素点 $g(x,y)$ 为文字区域的边缘(此处包括边缘的内角点和外角点),将图像矩阵 g_1 中对应的边缘像素点 $g_1(x,y)$ 的像素值赋值为 2,即

$$g_1(x,y)=\begin{cases} 2, & g(x,y)=1 \text{ \& } k_1=1 \\ g_1(x,y), & \text{else} \end{cases} \quad (2)$$

其中, k_1 为判断文字区域边缘的条件,即在像素值为 1 的像素点 $g(x,y)$ 处,其邻域上下左右的 4 个像素点值中至少有一个是 0 为真。

(2) 识别并标记文字区域边缘的内角点。将图像矩阵 g_1 另存为 g_2 ,如果像素点 $g_1(x,y)$ 的值是 1,且其 3×3 邻域 4 个角点中任意一个角点的像素值是 0,且该角点在此邻域内紧邻的另两个像素点的像素值都不是 0,如图 3(1)~(4)所示(其中 p 代表 1 或

2),则 $g_1(x,y)$ 为文字区域边缘的内角点,把图像矩阵 g_2 中对应的像素点 $g_2(x,y)$ 的像素值赋值为 3,即

$$g_2(x,y)=\begin{cases} 3, & g_1(x,y)=1 \text{ \& } k_2=1 \\ g_2(x,y), & \text{else} \end{cases} \quad (3)$$

其中, k_2 是判断文字区域边缘内角点的条件,即在像素值为 1 的像素点 $g_1(x,y)$ 处,其 3×3 邻域 4 个角点中任意一个角点的像素值是 0,且该角点在此邻域内紧邻的另两个像素点的像素值都不是 0 为真。

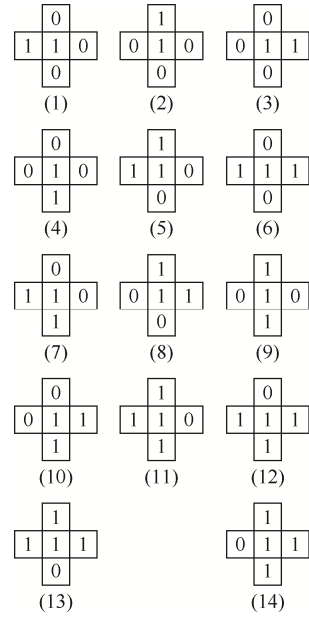


图 2 判断文字区域的边缘

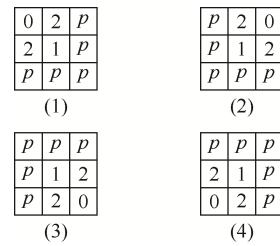


图 3 判断文字区域边缘的角点

(3) 识别并标记文字区域边缘的外角点。将图像矩阵 g_2 另存为 g_3 ,如果像素点 $g_2(x,y)$ 的值是 2,即文字区域的边缘。然后计算其 3×3 邻域的编码值,并对其编码值进行查表,判断是否属于文字区域边缘外角点,如果是则把图像矩阵 g_3 中对应点的像素值赋值为 4,即

$$g_3(x,y)=\begin{cases} 4, & g_2(x,y)=2 \text{ \& } Wsum \in k_3 \\ g_3(x,y), & \text{else} \end{cases} \quad (4)$$

其中, $k_3=[3 \ 6 \ 12 \ 24 \ 48 \ 96 \ 192 \ 129 \ 7 \ 14 \ 28 \ 56 \ 112 \ 224 \ 193 \ 131 \ 15 \ 30 \ 60 \ 120 \ 240 \ 225 \ 195 \ 135]$ 是判断文字区域边缘为外角点的编码值集合,即由所有

文字区域边缘的外角点 3×3 邻域编码值的种类组成的集合; $Wsum$ 为像素点 $g_2(x, y)$ 处 3×3 邻域计算的编码值, 其中所用的编码矩阵 $W = [128 \ 1 \ 2; 64 \ 0 \ 4; 32 \ 16 \ 8]$, 其目的是可以对文字区域边缘外角点

的每一种情况作唯一编码。编码值计算如图 4 所示, 先把邻域像素点值大于 0 的转变成 1, 即二值化处理, 再把二化后的邻域矩阵与编码矩阵对应相乘相加, 最后得到编码值 $Wsum$ 。

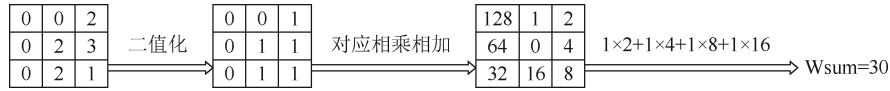


图 4 编码值计算

经过对文字区域的边缘分类识别标记后, 可进行下一步的分类查表法对文字区域取舍, 也即是细化。其目的不但是能够保证只对文字区域的边缘一步步做细化操作, 而且能够准确地判断出文字区域的边缘哪些可以去掉或保留。使最终细化的结果能够保障字符字迹的长度不变, 字迹的宽度细化成一个像素, 并且不会出现毛刺, 细化不完全或过度细化的现象。

2.2 分类删除文字区域边缘

(1) 删除文字区域边缘外角角点。将图像矩阵 g_3 另存为 g_4 , 在图像矩阵 g_4 中找出像素值为 4 的点, 即文字区域边缘外角角点, 然后进行删除操作, 即将此点的像素值赋值为 0。见式(5), 图 5 为对应删除文字区域边缘外角角点操作前后的图示。

$$g_4(x, y) = \begin{cases} 0, & g_4(x, y) == 4 \\ g_4(x, y), & \text{else} \end{cases} \quad (5)$$

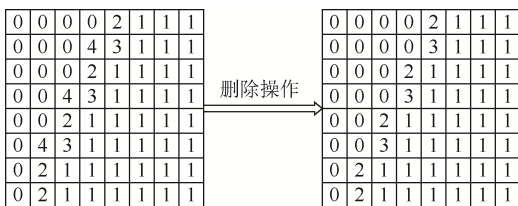


图 5 删除文字区域边缘外角角点

(2) 删除文字区域边缘点。将图像矩阵 g_4 另存为 g_5 , 在矩阵 g_5 中找出像素值为 2 的点, 即文字区域边缘点。然后计算其 3×3 邻域的编码值, 并对其编码值进行查表, 判断是否属于可去点, 如果是则做删除操作, 即将此点的像素值赋值为 0; 否则作保留操作, 即像素值赋值为 1, 式(6), 图 6 是对应删除文字区域边缘点操作前后的图示。

$$g_5(x, y) = \begin{cases} 0, & g_5(x, y) == 2 \ \& \ Wsum \in N \\ 1, & g_5(x, y) == 2 \ \& \ Wsum \notin N \\ g_5(x, y), & \text{else} \end{cases} \quad (6)$$

其中, $N = [3 \ 5 \ 7 \ 12 \ 13 \ 14 \ 15 \ 20 \ 21 \ 22 \ 23 \ 28 \ 29 \ 30 \ 31 \ 48 \ 52 \ 53 \ 54 \ 55 \ 56 \ 60 \ 61 \ 62 \ 63 \ 65 \ 67 \ 69 \ 71 \ 77 \ 79 \ 80 \ 81 \ 83 \ 84 \ 85 \ 86 \ 87 \ 88 \ 89 \ 91 \ 92 \ 93 \ 94 \ 95 \ 97 \ 99 \ 101 \ 103 \ 109 \ 111 \ 112 \ 113 \ 115 \ 116 \ 117 \ 118 \ 119 \ 120 \ 121 \ 123 \ 124 \ 125 \ 126 \ 127 \ 131 \ 133 \ 135 \ 141 \ 143 \ 149 \ 151 \ 157 \ 159 \ 181 \ 183 \ 189 \ 191 \ 192 \ 193 \ 195 \ 197 \ 199 \ 205 \ 207 \ 208 \ 209 \ 211 \ 212 \ 213 \ 214 \ 215 \ 216 \ 217 \ 219 \ 220 \ 221 \ 222 \ 223 \ 224 \ 225 \ 227 \ 229 \ 231 \ 237 \ 239 \ 240 \ 241 \ 243 \ 244 \ 245 \ 246 \ 247 \ 248 \ 249 \ 251 \ 252 \ 253 \ 254 \ 255]$ 为判断文字区域边缘点邻域编码值满足可去点的取值集合, 即由所有满足可去点的文字区域边缘 3×3 邻域的编码值种类组成的集合; $Wsum$ 是像素点 $g_5(x, y)$ 处 3×3 邻域计算的编码值, 其中所用的编码矩阵 $W = [128 \ 1 \ 2 \ 64 \ 0 \ 4 \ 32 \ 16 \ 8]$, 其目的为可以对文字区域边缘的每一种情况作唯一编码。

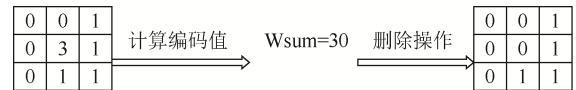


图 6 删除文字区域边缘点

(3) 删除文字区域内角角点。在矩阵 g_5 中找出像素值为 3 的点, 即文字区域边缘内角角点。然后计算其 3×3 邻域编码值, 并对其编码值进行查表, 判断是否是可去点, 如果是则做删除操作, 即把此点的像素值赋值为 0; 否则作保留操作, 即像素值赋值为 1。式(7), 图 7 是对应删除文字区域边缘内角角点操作前后的图示。

$$g_5(x, y) = \begin{cases} 0, & g_5(x, y) == 3 \ \& \ Wsum \in N \\ 1, & g_5(x, y) == 3 \ \& \ Wsum \notin N \\ g_5(x, y), & \text{else} \end{cases} \quad (7)$$

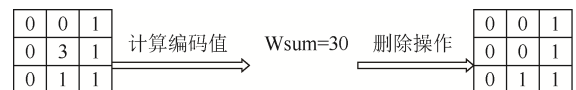


图 7 删除文字区域边缘内角角点

另创建像素值全为 0 的图像矩阵 g_6 ，在图像矩阵 g_5 中找出像素值大于零的点，并把图像矩阵 g_6 中对应的点像素值赋值为 1。然后对图像矩阵 g_6 重复第 2 章节步骤，直到与上一次保存的图像相比没有像素点变化，则此时图像字符的宽度为一个像素。

3 实验结果

为验证本文提出的国际音标图像字符细化方法的实际效果，在 CPU 为 Intel(R) Core(TM)i5 @ 3.20 GHz 的电脑上和 Matlab 2015b 环境下，对 100 幅国际音标字符图像进行实验，结果见表 1，其中每幅图像分别用 3 种算法细化用时对比，并以部分实验结果为例进行分析，如图 8~13 所示。

文献中评价细化算法通常从连通性、对噪声的鲁棒性和速度等方面考虑。在速度上，本文算法的时间复杂度为 $T(n)=O(n^2)$ 。处理时间与字体笔迹的形态密切相关，笔迹粗的处理时间相对长。从表 1 和图 8 的实验结果可看出，本文算法无论在单幅图像细化用时上，还是在多幅图像平均用时上都少于 ZS^[1]和 EPTA^[9]算法，且能够对字符完全细化，没有冗余。

表 1 实验结果分析比较

算法	字符图像数	细化每幅图像平均时间(s)	细化效果
ZS	100	1.05	斜线不连续
EPTA	100	1.06	部分有冗余
本文	100	0.92	完全细化

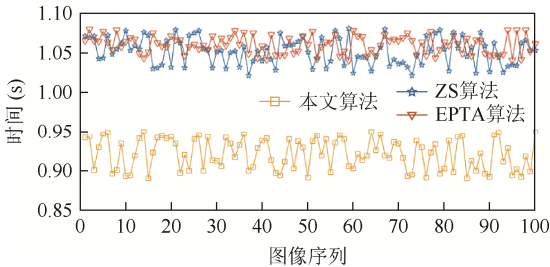


图 8 3 种算法对每幅图像细化用时对比

图 9~11 为细化实验结果对比图，ZS 算法细化的结果斜线部分不连续、交叉点部分有冗余，且有时会出现畸变情况，如图 10(b)中的第一个字符。EPTA 算法细化结果有冗余，不能完全细化，且会出现断点畸变情况，如图 11(c)中第 5 个字符。而本文算法细化结果没有冗余，字迹宽度也都为一个像素，字符具有很好地连通性，不会出现畸变情况。



图 9 细化实验结果图 1



图 10 细化实验结果图 2



图 11 细化实验结果图 3

为了验证本文算法的鲁棒性, 又分别对手写毛笔中文字图像、斜体英文和罗马数字图像进行了细化实验, 分别如图 12~13 所示。图 12(a)是手写毛笔汉字字符原图像, 字符笔迹的边缘有凹凸不平的地方, 但是经过本文算法细化时并没有产生畸变或细化不完

全的情况。最终的细化结果是字符笔迹的宽度为一个像素, 而长度仍然是原字符的长度, 如图 12(b)所示。图 13(a)为斜体英文和罗马数字原图像, 字符虽然是斜体, 并且笔迹宽度也不尽相同, 但是丝毫不影响本文算法的细化, 细化结果仍然很出色, 如图 13(b)所示。

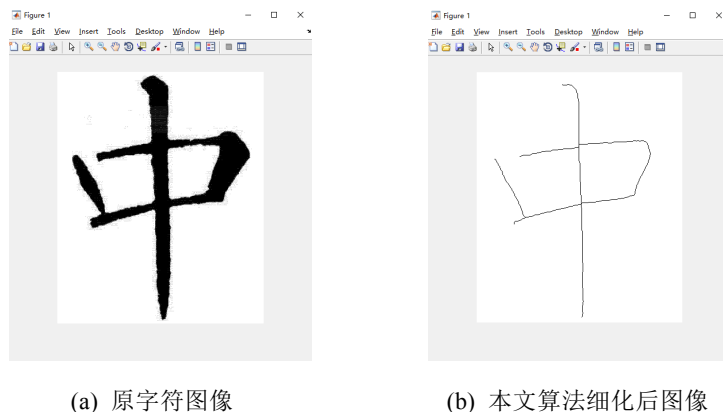


图 12 手写毛笔中文汉字图像字符实验图

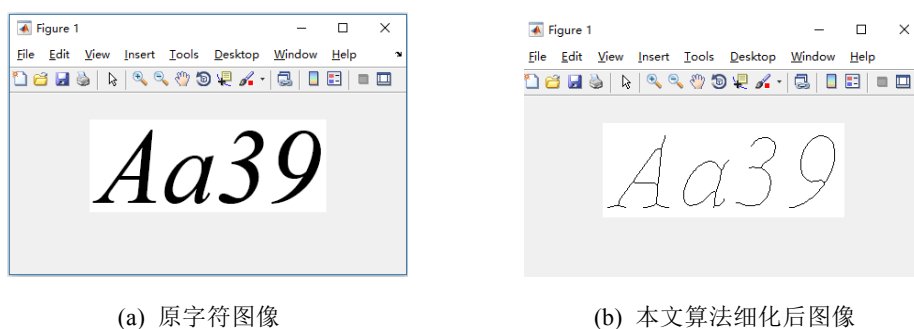


图 13 斜体英文和罗马数字图像字符实验图

4 结 束 语

实验结果表明,本算法非常出色地实现了对国际音标字符图像的细化,细化后的字符图像不但没有产生畸变,而且平均细化时间较短,细化效果满足一个像素宽度,没有冗余,鲁棒性强。该算法有效地解决了国际音标字符图像的细化问题,为进一步对国际音标字符图像的识别起到了重要作用。但是对图像文字的细化仅仅是提取识别图像中文字信息的第一步,而提取识别图像中的文字信息还有很多困难要克服,并且也是图像处理领域的一个重要研究方向,这也是下一步的工作方向。

参 考 文 献

- [1] ZHANG Y T, SUEN Y C. A fast parallel algorithm for thinning digital patterns [J]. Communications of the ACM, 1984, 27(3): 236-239.
- [2] LAM L, LEE S W, SUEN C Y. Thinning methodologies a comprehensive survey [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2002, 14(9): 869-885.
- [3] WEN C, AO G, TIAN Y. A thinning method for fingerprint image based on Hit-miss transformation [C]//IEEE International Conference on Computer Science and Automation Engineering (CSAE). New York: IEEE Press, 2011: 225-228.
- [4] VADVADGI G K, SANDEEPP P H. A modified thinning algorithm for minutiae feature extraction of fingerprint images on FPGA [J]. International Journal of Electrical, Electronics and Data Communication, 2015, 3(3): 42-47.
- [5] ZHOU Z, WANG L. A method of thinning interference fringes based on the characteristic of interference fringes [C]//2011 International Conference on Computer Science and Service System (CSSS 2011). New York: IEEE Press, 2011: 2885-2887.
- [6] SHAIKH N A, SHAIKH Z A. A generalized thinning algorithm for cursive and Non-Cursive language scripts [C]//9th International Multitopic Conference. New York: IEEE Press, 2007: 1-4.
- [7] 王会英, 张有会, 张静, 等. 一种基于离散 Voronoi 图的手写体文字细化方法[J]. 计算机工程与应用, 2008, 44(15): 178-181.
- [8] VINCZE M, KOVARI B. Comparative survey of thinning algorithms [EB/OL]. (2009-11-14) [2012-02-17]. <http://uni-obuda.hu/conference/cinti2009>.
- [9] BAO J, FAN J. Robust parallel thinning algorithm for binary images [J]. Computer Aided Engineering, 2006, 15(4): 43-46.
- [10] 韩建峰, 宋丽丽. 改进的字符图像细化算法[J]. 计算机辅助设计与图形学学报, 2013, 25(1): 62-66.
- [11] 贾瑜, 饶建辉. 一种对文字图像细化的改进 Hilditch 算法研究[J]. 武汉工业学院学报, 2006, 25(3): 37-39.
- [12] ZHU X F, ZHANG S Y. A shape-adaptive thinning method for binary images [C]//IEEE International Conference on Cyberworlds. New York: IEEE Press, 2008, 721-724.
- [13] ZHANG S H. Study and realization of algorithms for Chinese characters image's preprocessing [J]. Microcomputer Development, 2003, 4(4): 53-57.
- [14] OTSU N. A threshold selection method from graylevel histograms [J]. IEEE Transactions on Systems Man and Cybernetics, 1979, 9(1): 62-66.