

面向连续叠写的高精简中文手写识别方法研究

苏统华¹ 戴洪良¹ 张 健² 马培军¹ 邓胜春¹

(哈尔滨工业大学软件学院 哈尔滨 150001)¹ (哈尔滨工业大学材料科学与工程学院 哈尔滨 150001)²

摘 要 连续手写识别是中文手写输入技术的核心,自然、快捷地输入中文信息一直是模式识别乃至人工智能领域追求的目标。提出了一种有效克服小屏幕限制的连续叠写汉字识别方法。该方法基于切分-识别集成的解码框架,先使用过切分算法处理输入的书写轨迹;然后启用一种新颖的感知机算法判定字符的边界;随后采用来自字符分类模型、几何模型和语言模型的多种上下文信息进行路径解码。为适应不同类型的移动终端,特别提出了一种高效压缩字符分类模型的方法,以有效减少字符识别过程对存储和内存的占用。该识别方法已在 Android 平台上部署,并进行了大规模的测试实验。实验结果证实了该识别方法的性能和效率。

关键词 模式识别,连续中文叠写,笔画分类,分类器压缩,集束搜索

中图法分类号 TP319 文献标识码 A DOI 10.11896/j.issn.1002-137X.2015.7.064

Study on High Compact Recognition Method for Continuously Overlaid Chinese Handwriting

SU Tong-hua¹ DAI Hong-liang¹ ZHANG Jian² MA Pei-jun¹ DENG Sheng-chun¹

(School of Software, Harbin Institute of Technology, Harbin 150001, China)¹

(School of Materials Science and Engineering, Harbin Institute of Technology, Harbin 150001, China)²

Abstract Continuous Chinese handwriting recognition is the primary bottleneck for Chinese handwritten character input method. Naturally and quickly inputting Chinese text is the fundamental goal to the pattern recognition field even to the artificial intelligence. A novel recognition method was proposed for overlaid Chinese handwriting. It follows a segmentation-recognition integrated framework. Firstly, an over-segmentation algorithm is used to partition the handwriting trajectory. Then a perceptron algorithm is developed to locate the candidate character boundaries. Finally, multiple contexts including character recognition score, geometrical score and linguistic score, are utilized to decode the optimal recognition path. To match different mobile terminals, an appealing compression algorithm was proposed to make the character classifier compact, which reduces the storage consumption both in memory fingerprint and disk storage. The principled method is successfully ported to Android platform, enabling overlaid Chinese handwriting to be input on smart phones and further tested on large overlaid Chinese handwriting samples. Experimental results verify the effectiveness and efficiency of the method. It also works smoothly on smart phone, whose overlapped handwriting input function makes handwriting input remarkably efficient.

Keywords Pattern recognition, Overlaid Chinese handwriting, Stroke classification, Classifier compression, Beam search

1 概述

连续手写汉字识别是一种自然的智能交互方式。如果能够对连续输入的笔迹进行高效识别,将极大提高汉字输入的效率并提升用户的体验级别。随着智能手机、平板等各种新型智能终端的流行,通过手写方式录入文字的需求越来越普遍。从19世纪末的打孔卡和纸带输入,到盛行了数十年的键盘鼠标,再到现在流行的触摸输入,人机交互正在变得更直观、自然和人性化。连续手写作为最为自然的手写输入方式,如果能够达到实用层次,无疑会给使用者带来优质的用户

体验。同时,不容置疑,连续手写识别的水平也在某种程度上代表了人工智能技术的发展水平。

叠写是连续手写的一种,用来解决手机屏幕的尺寸限制。在2003年,就有研究人员提出在有限大小的空间内采用叠写的方式进行连续的手写输入的思想,即允许用户写完一个字后不停顿,仍然在这个有限大小的空间上继续写第二个字,直到用户写完一个短语或句子,最后对用户这次所写的字统一进行识别^[1]。叠写的效果如图1所示。实现叠写输入要解决的最主要问题就是对叠写的笔迹序列进行识别,并且由于叠写输入适合于在体积较小的手持设备上使用,因此对识别时

到稿日期:2014-06-22 返修日期:2014-08-16 本文受国家自然科学基金(61203260),黑龙江省博士后基金(LBH-Q13066),哈尔滨工业大学科研创新基金(HIT.NSRIF.2015083)资助。

苏统华(1979—),男,博士,高级讲师,主要研究方向为模式识别、物联网智能信息处理,E-mail:thsu@hit.edu.cn;戴洪良(1990—),男,硕士生,主要研究方向为机器学习、数据挖掘;张 健(1986—),男,工程师,主要研究方向为材料科学与数据挖掘;马培军(1963—),男,博士,教授,主要研究方向为软件工程、数据挖掘;邓胜春(1971—),男,博士,教授,主要研究方向为数据挖掘。

的时间和空间复杂度要求都较高。



图1 叠写的“大石头”3个字

现已有一些针对识别叠写中文的研究工作。文献[2]提出了一种使用字符对的思想来对叠写中文进行识别的方法；文献[3]提出了一个基于过切分的叠写中文识别系统，并用该系统实现了一款手机上的手写输入应用。文献[2,3]都实现了对叠写汉字的识别，但它们用来进行训练和测试的样本都很少，且都没有在空间复杂度上进行优化。

最近几年，手写文本识别领域取得了一些新的研究成果。文献[4]提出了一种基于过切分，并结合了多种上下文模型进行识别的脱机中文手写文本识别方法，使用 CASIA-HWDB 脱机手写文本数据集^[5]进行训练和测试，得到了较好的识别性能。此后文献[6]使用类似文献[4]的识别方法对联机手写文本进行识别，也取得了较好的效果。

本文使用 CASIA-OLHWDB 联机手写中文文本数据集模拟叠写样本，实现了一个叠写识别系统，并用该系统在 Android 平台上实现了一款支持叠写中文输入功能的输入法应用。在实现叠写识别系统时，引入了文献[4]中提出的一些有效方法以提高叠写识别的性能。另外，考虑到手机平台的特殊性，本文在保持识别性能的前提下关注于优化系统的时间和空间复杂度，提出了一种针对单字符分类器的高效压缩方法，减少了输入法在外部存储设备和内存中占用的存储空间。

与一些现有的方法(如文献[2,3]中的方法)相比，本文使用的识别框架是新近证明可以对连续手写汉字得到高识别准确率的识别框架^[4]，其由于使用了几何模型，识别时利用到了更多几何方面的信息。另外，本文对字符分类模型的精简方法令实现时性能下降几乎可忽略不计而存储空间占用明显减小的连续手写汉字识别方法成为可能。

本文第2节介绍所实现的叠写识别系统，包括识别框架的总体流程；第3节介绍叠写识别的关键模型和算法设计，重点介绍字符分类器的压缩方法；第4节给出对识别系统及压缩字符分类器的效果进行测试的方法及测试结果；最后总结全文并给出未来的研究计划。

2 连续叠写识别框架

本文的叠写识别系统对一个叠写样本进行识别的总体过程如图2所示。

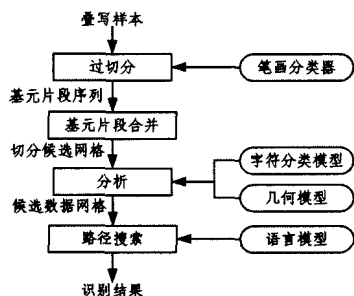


图2 识别叠写样本总体过程

识别时主要有过切分、基元片段合并、结合字符分类器和几何模型进行分析、结合语言模型进行路径搜索这些步骤。本节对整体路径搜索思路进行简要介绍，关于过切分模型、几何模型和精简字符模型的介绍见下一节。

路径搜索的目的是在候选数据网格中搜索出使路径评价函数的值最大的候选识别路径，本文使用集束搜索算法来进行路径搜索，使搜索出的结果最优或接近最优。搜索时使用的路径评价函数为：

$$f(P^n) = \sum_{i=1}^m [k_i \cdot \log p(c_i | x_i) + \lambda_1 \log p(c_i | h_i) + \lambda_2 \log p(z_i^g = 1 | g^h)] + \lambda_3 \cdot m \quad (1)$$

其中， p_n 表示第 n 条候选切分识别路径， m 为这条路径上的候选模式(由若干相邻基元片段组成)数， k_i 为该路径上的第 i 个候选模式中包含的基元片段数， c_i 表示第 i 个字， x_i 表示第 i 个候选模式， h_i 表示第 i 个字前面的所有字， g^h 表示路径上第 i 个和第 $i-1$ 个候选模式的几何信息， $z_i^g = 1$ 表示路径上第 i 个候选模式和第 $i-1$ 个候选模式之前的“间隙”是两个字之间的间隙。 λ_1 、 λ_2 、 λ_3 为正常量，可通过训练得到^[4]。最后一项 $\lambda_3 \cdot m$ 用来鼓励长路径，也即含候选模式多的路径，不加这一项则识别结果会偏向短路径。

式(1)中的 $p(c_i | x_i)$ 由字符分类模型中的字符分类器给出， $p(c_i | h_i)$ 由语言模型给出， $p(z_i^g = 1 | g^h)$ 可以由几何模型中的字符对分类器给出。本文对字符分类器和字符对分类器的直接输出结果都进行了置信度转换，以提高识别性能^[4,7]。

整个识别过程是在文献[4]提出的脱机连续手写汉字识别框架基础上改进而来的。由于文献[4]是针对普通的脱机连续手写汉字，与本文要解决的联机连续叠写汉字识别相差较大，因此在过切分、字符分类模型和几何模型上都做了改进。在过切分和几何模型上，更多地利用了脱机手写所没有的轨迹信息，并结合叠写的特点，使分类结果更准确。由于要被用在移动终端上，对字符分类模型进行了精简。

3 模型与算法

本文的连续叠写识别方法基于第2节的切分-识别集成的解码框架。先使用过切分算法处理输入书写轨迹，然后采用一种新颖的感知机算法判定字符的边界，随后采用来自字符分类模型、几何模型和语言模型的多种上下文信息进行路径解码。考虑到移动终端的特殊性，本文在保持识别性能的前提下重点关注如何优化系统的时间和空间复杂度，提出了一种对单字符分类器的压缩方法，以有效减少输入方法在外部存储设备和内存中占用的存储空间。

3.1 过切分算法

过切分是将叠写样本切分为由许多基元片段构成的序列，尽可能地使叠写样本中的每个手写字都可以由一个或若干个基元片段合并而成，但也不能切分得太“碎”，使叠写样本中的手写字被切成太多的基元片段。如图3所示，叠写的“这里”二字被过切分为3个基元片段。

本文用一个笔画分类器来进行过切分。笔画分类器将叠写样本中的一个笔画分为“是一个字的最后一个笔画”和“不是一个字的最后一个笔画”两类。使用了感知机模型来实现

笔画分类器,根据下式的结果进行分类:

$$o = w_0 + w_1 x_1 + w_2 x_2 + \dots + w_n x_n \quad (2)$$

其中, $w = (w_0, w_1, \dots, w_n)$ 为参数,通过训练得到; $x = (x_1, x_2, \dots, x_n)$ 为输入的特征向量。

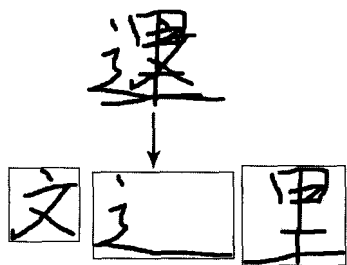


图3 过切分示意图

特征向量中的特征从当前判断的笔画和它的后一笔画中得到。使用的特征包括:两个笔画的点中 X 坐标的最小值和最大值、两个笔画的点中 Y 坐标的最小值和最大值、两个笔画几何中心的 X 坐标和 Y 坐标、当前笔画第一点和最后一点的 X 和 Y 坐标、后一笔画第一点和最后一点的 X 和 Y 坐标。从笔画中提取出这些几何信息后,将其中与 X 坐标相关和与 Y 坐标相关的值分别用叠写样本的宽和高进行归一化。得到了特征后,如果用式(2)得到的值 o 大于一个阈值,就认为当前笔画后的“间隙”是一个过切分点,否则认为不是。一般当阈值为 0 时分类的准确率最高,但这里要尽量将正确的切分点都找出来,所以应将阈值设置得稍微小一点,将更多的“间隙”判断为过切分点,同时又要尽量保持高的分类准确率,从而样本不会被切得太“碎”,具体设定的阈值和对应该阈值的笔画分类器性能将在第 4 节中给出。

3.2 几何模型

本文的叠写识别方法使用了一个二元与类别无关的几何模型,用来评估两个相邻的候选模式是两个字的可能性,主要使用一个字符对分类器来实现。字符对分类器将两个相邻的候选模式分为“是相邻的两个字”和“不是相邻的两个字”两类,与笔画分类器一样,它也使用感知机来实现。

实现字符对分类器时使用的特征包括:两个候选模式各自的最上端和最下端 Y 坐标、最左端和最右端 X 坐标;两候选模式各自第一个笔画和最后一个笔画的最上端和最下端 Y 坐标、最左端和最右端 X 坐标、第一个点及最后一个点的 X 和 Y 坐标、几何中心的 X 和 Y 坐标。通过将根据式(2)得到的值 o 进行置信度转换后得到两个相邻的候选模式是两个字的可能性,也即式(1)中的 $p(z_i^f = 1 | g^h)$ 。

3.3 精简的字符分类模型

字符分类器的工作是对单个手写字进行识别,过去的一些压缩字符分类器的工作有针对 MQDF 分类器的^[8,9],也有针对基于 PCGM 的分类器的^[9]。文献[8,9]在压缩字符分类器的过程中都用到了 split VQ 方法。而文献[10]则用聚类的思想识别手写汉字,对聚类得到的簇中心使用 split VQ 方法也达到了压缩的目的。本文对所使用的 LVQ 分类器进行了压缩,与 split VQ 方法对被压缩的向量进行统一处理的方式不同,本文在压缩时对被压缩向量的每一维分别进行处理。

本文使用的 LVQ 分类器可处理 7356 个类别,其中包括

汉字、标点符号和一些特殊符号。该分类器用 2 个 160 维的向量来代表每个类别,于是,共有 14712 个 160 维的向量需被存储。在识别时,先从待识别的手写字中提取一个 160 维的特征向量,用这个特征向量与 14712 个向量求距离,与一个向量的距离越近就认为越有可能属于这个向量代表的类。

根据 LVQ 字符分类器的识别方法,假设中心数有 2 个,需要存储 14712 个 160 维的向量,向量每一维的数据使用 float 类型,那么占用的存储空间为 $14712 * 160 * 4 = 9415680$ 字节,约为 8.98MB,再加上一些其他的必需数据,LVQ 字符分类器的大小就超过 9MB,用在手机上就显得过大;而且为保证识别速度,输入法运行时必须将这些数据都存放在内存中,手机的内存资源也很宝贵。由于需要存储的这 14712 个 160 维向量是 LVQ 分类器需要存储的数据中的最主要部分,实现了对它们的压缩,也就把整个 LVQ 分类器所占的存储空间大幅度减小了。

在对 LVQ 字符分类器进行压缩时,对 14712 个 160 维向量的每一维分别进行处理,下面以第一维为例进行说明。取出所有 14712 个向量的第一维分量,则共有 14712 个浮点数。压缩时,先将这 14712 个浮点数使用 k-means 聚类算法聚类为 256 类,每个类用其类中心来表示,这里的类中心为一个浮点数,于是聚类后得到 256 个浮点数,并将它们保存。然后再用一个含 14712 个 byte 类型元素的数组记下原来的 14712 个浮点数分别所属的类。

这样,记录 14712 个向量第一维分量值的 14712 个浮点数被 256 个浮点数和 14712 个 byte 类型值所代替。若识别时需要使用第一个向量的第一维分量值,则先从含有 14712 个 byte 类型元素的数组中取出它属于第一维中的那一类,然后再从 256 个浮点数中取出表示这一类的浮点数代替原来未压缩时的浮点数并返回。任一向量的任一分量值都可以通过类似方法得到。

对 160 维都分别进行处理后,原来的 $14712 * 160$ 个浮点数由 14712 个 byte 类型值和 $160 * 256$ 个浮点数所代替,所占用的存储空间变为 $14712 * 160 + 160 * 256 = 2394880$ 字节,约为 2.28MB,几乎是原来的 1/4。这一精简方案,对于更复杂的模型会更有效。

4 性能测试实验

4.1 叠写实验数据集

由于缺少真实的叠写样本,本文使用 CASIA-OLHWDB 2.0—2.2 联机手文本数据集中的连续手写文本来模拟大量的叠写样本以进行训练和测试。CASIA-OLHWDB 2.0—2.2 数据集被分为训练集和测试集,训练集包含由 815 个人所写的 4072 个页面,测试集包含由 204 个人所写的 1020 个页面。模拟叠写样本通过将连续手写文本中的字进行位移,让它们的几何中心重叠在一起来实现。

本文使用 CASIA-OLHWDB 2.0—2.2 数据集模拟出了两类叠写样本,一类过滤了原样本中除中文、数字和英文外的符号,另一类没有过滤。模拟过滤了符号的叠写样本的原因是:最终实现的输入法在手写输入界面上设置了直接输入常

用标点符号的按键,所以用户可以更直接、快捷地通过按键输入这些标点符号,而不用手写。模拟过滤了符号的叠写样本时,将需过滤的符号作为两个连续叠写样本间的分割点,也即将两个需被过滤的符号间的中文、数字和英文模拟为一个叠写样本。模拟不过滤符号的叠写样本时,将原手写文本样本中的每一行模拟为一个叠写样本。对于这两类,都模拟了训练集和测试集两个样本数据集,模拟它们时分别使用 CASIA-OLHWDB 2.0—2.2 的训练集和测试集。模拟出的叠写样本数据集信息见表 1,其中训练集 A 和测试集 A 分别表示过滤了符号这一类的训练集和测试集,训练集 B 和测试集 B 分别表示未过滤符号这一类的训练集和测试集。

表 1 叠写样本集信息

样本集	样本数	样本字数范围	样本平均字数
训练集 A	101938	1 到 74	9.6
测试集 A	25354	1 到 90	9.6
训练集 B	41710	1 到 52	25.9
测试集 B	10510	2 到 47	25.7

4.2 实验设定

在过切分时,本文将用式(2)得到的值来判断过切分点的阈值设定为-0.5,以保证绝大部分的正确字间切分点能被找出。实际在测试后,97.4%的正确切分点都被找了出来,且此时笔画分类器的分类准确率为 92.3%。合并基元片段时,综合考虑识别速度和识别性能,本文设定最多将连续的 3 个基元片段合并为一个候选模式。在使用字符分类器对候选模式进行识别时,取前 10 个候选。最后使用集束搜索算法搜索路径时,设定集束搜索的宽度为 10。所使用的语言模型为二元语言模型,由人民日报新闻语料训练而来,语料中有 3 亿多字符,综合考虑识别的速度、性能以及空间复杂度,对语言模型进行了剪枝处理,最后得到的语言模型大小为 3.97MB。

4.3 字符分类器压缩性能测试

对字符分类器压缩方法的性能测试通过对比未压缩字符分类器和压缩后字符分类器的识别准确率得到。由于训练字符分类器时使用了 CASIA-OLHWDB 2.0—2.2 数据集中训练集的单字样本,因此测试时只使用从其测试集的样本里提取到的单字样本,共包括由 204 个人所写的 268510 个单字样本。测试的结果如表 2 所列。可见,压缩的效果良好,压缩后的字符分类器大小不到原来的 1/3。而第一候选的准确率只降低了不到 0.01%,前 10 候选的准确率甚至还提升了。另外,由于压缩后识别的时间复杂度不变,识别用时的变化也不大。

表 2 字符分类器压缩性能的测试结果

分类器	准确率		用时 (min)	大小 (MB)
	第一候选	前 10 候选		
压缩前	87.824%	97.051%	7.7	9.57
压缩后	87.815%	97.054%	8.6	2.99

4.4 叠写识别系统识别准确率测试

测试叠写识别系统识别准确率时使用了生成的过滤了符号和未过滤符号这两类叠写样本的测试集,测试的结果如表 3 所列。其中 AR 和 CR 分别表示识别精确率和识别准确率^[8]。对过滤了符号的样本,识别准确率达 89.3%,接近单字识别的准确率。从表 3 还可以看出,识别系统对过滤了符

号的叠写样本和不过滤符号的叠写样本的识别准确率差别较大,这说明识别系统对除中文汉字、英文和数字外的符号的识别能力较低。这主要是因为这些符号一般较小,与通常的中文汉字差别很大,在分割识别时难以将它们区分出来。

表 3 叠写识别系统的识别测试结果

叠写样本	AR	CR
过滤符号	87.8%	89.3%
不过滤符号	82.2%	83.4%

另外,还使用过滤了符号的叠写样本测试了不使用置信度转换和不使用几何模型这两种情况,测试时其他情况都相同,结果如表 4 所列。使用置信度转换令 AR 提高了 2.1%,使用几何模型令 AR 提高了 1.4%,这说明使用置信度转换和使用几何模型都能让识别性能有一定的提高。

表 4 不同识别方案的识别测试结果

识别方案	AR	CR
不使用置信度转换	85.7%	88.2%
不使用几何模型	86.4%	88.7%

4.5 实际运行效果

将识别方法应用到 Android 系统上,制作了一款 Android 平台上支持叠写的手写中文输入法,并对所实现的输入法进行了简单的测试。令两个人作为用户分别使用叠写输入功能输入本文中文摘要的内容,其中有 255 个汉字,两个用户通过多次叠写输入,除汉字外的标点和英文字母则使用普通的按键输入。图 4 为测试时的两张输入法运行效果图。

缩行分类模型的方法,以有效减少字符识别过程对存储和内存的占用。识别方法已在 android 平台上部署,并进行了大规模测试实验。实验结果证实了识别方法的性能和效率

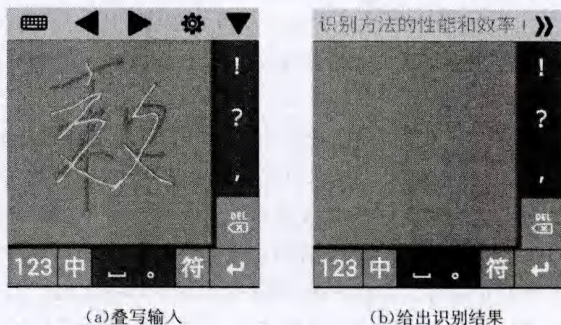


图 4 输入法的运行效果

将测试的两个用户分别叫做用户 A 和用户 B,测试的结果如表 5 所列。其中平均每次输入字数为输入的总汉字数除以用户进行叠写输入的次数。统计时,将用户手写错误而导致的错误识别结果也作为识别错误。可见,输入法对两位用户的叠写输入都给出了较好的识别结果,识别方法的实际应用是成功的。

表 5 实际输入法输入效果的测试

用户	平均每次输入字数(个)	AR	CR
A	10.2	96.5%	96.5%
B	8.8	97.6%	97.6%

结束语 本文在实现叠写中文识别系统时,引入了一些近年来手写文本识别领域研究出的有效方法,取得了较好的效果。最终实现的叠写中文识别系统在所使用的叠写样本测

数据集上达到了 89.3% 的识别准确率,置信度和几何模型的使用对识别性能都有一定程度的贡献。另外,所提出的压缩字符分类器方法效果尤为明显,压缩后字符分类器存储在外部存储设备上时所占用的空间和运行时在内存中所占用的存储空间都大幅度减小,且压缩后对识别准确率和识别速度的影响都不大。本文所研究的叠写识别系统在 Android 平台上予以部署,实现了一个支持叠写中文输入功能的输入法应用,软件运行流畅,效果良好。

为了使叠写识别系统有更高的识别准确率,进一步的研究工作可以从获取真实的叠写样本和在实现几何模型时使用更多有效的几何信息这两个方面来考虑。

参考文献

- [1] Shimodaira H, Sudo T, Nakai M, et al. On-line Overlaid-Handwriting Recognition Based on Substroke HMMs[C]// Seventh International Conference on Document Analysis and Recognition. Washington DC: IEEE, 2003: 1043-1047
- [2] Wan Xiang, Liu Chang-song, Zou Yan-ming. On-line Chinese Character Recognition System for Overlapping Samples[C]// 2011 International Conference on Document Analysis and Recognition. Washington DC: IEEE, 2011: 799-803
- [3] Zou Yan-ming, Liu Ying-fei, Liu Ying, et al. Overlapped handwriting input on mobile phones[C]// 2011 International Conference on Document Analysis and Recognition. Washington DC: IEEE, 2011: 369-373

- [4] Wang Qiu-feng, Yin Fei, Liu Cheng-lin. Handwritten Chinese Text Recognition by Integrating Multiple Contexts[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(8): 1469-1481
- [5] Yin Fei, Wang Da-han, Wang Qiu-feng. CASIA Online and Offline Chinese Handwriting Databases[C]// 2011 International Conference on Document Analysis and Recognition. Washington DC: IEEE, 2011: 37-41
- [6] Wang Da-han, Liu Cheng-lin, Zhou Xiang-dong. An approach for real-time recognition of online Chinese handwritten sentences[J]. Pattern Recognition, 2012, 45(10): 3661-3675
- [7] Liu Cheng-lin. Classifier Combination Based on Confidence Transformation[J]. Pattern Recognition, 2005, 38(1): 11-28
- [8] Teng Long, Jin Lian-wen. Building compact MQDF classifier for large character set recognition by subspace distribution sharing[J]. Pattern Recognition, 2008, 41(9): 2916-2925
- [9] Wang Yong-qiang, Huo Qiang. Building compact recognizers of handwritten Chinese characters using precision constrained Gaussian model, minimum classification error training and parameter compression[J]. International Journal on Document Analysis and Recognition, 2011, 14(3): 255-262
- [10] 杨军. 聚类分析及其在大类汉字识别中的应用[D]. 广州: 华南理工大学, 2007
- Yang Jun. Application of the Clustering Analysis in the Large vocabulary Chinese Character Recognition[D]. Guangzhou: South China University of Technology, 2007

(上接第 284 页)

实验中 ORL 和 PIE_pose27 的类数皆取 36。稀疏后的基图像如图 6 和图 7 所示。由图 6 和图 7 可知, NMF 算法的稀疏度较差, CNMF 的稀疏度也不高, CNMFS 算法的稀疏度最高。由此表明, CNMFS 算法可以更好地得到图像的局部表示。

结束语 本文提出了稀疏约束的半监督非负矩阵分解, 并给出了相应的迭代公式以及收敛性的证明。以流行的人脸数据库为实验对象, 应用本文提出的新方法进行了实验, 利用两个评价指标(聚类准确率和归一化互信息)来衡量所提算法识别精度的好坏。实验表明, 算法得出了更好的识别率和聚类性能。进一步考察算法的稀疏度, 结果显示所提出的算法的稀疏度最高。该算法能够得到图像的最佳局部表示, 使得基图像具有更好的判别能力。

参考文献

- [1] Lee D D, Seung H S. Learning the parts of objects by non-negative matrix factorization[J]. Nature, 1999, 401(6755): 788-791
- [2] 杜世强, 石玉清, 王维兰, 等. 基于图正则化的半监督非负矩阵分解[J]. 计算机工程与应用, 2012, 48(36): 194-200
- Du Shi-qiang, Shi Yu-qing, Wang Wei-lan, et al. Graph regularized-based semi-supervised non-negative matrix factorization[J]. Computer Engineering and Applications, 2012, 48(36): 194-200

- [3] Cai Deng, He Xiao-fei, Han Jia-wei, et al. Graph regularized non-negative matrix factorization for data representation[J]. IEEE Trans on Pattern Anal Mach Intell, 2011, 33(8): 1548-1560
- [4] Hoyer P O. Non-negative matrix factorization with sparseness constraints[J]. Journal of Machine Learning Research, 2004, 5(9): 1457-1469
- [5] Sun Fu-ming, Tang Jin-hui, Li Hao-jie, et al. Multi-label image categorization with sparse factor representation[J]. IEEE Transaction on Image Processing, 2014, 23(3): 1028-1037
- [6] Li S Z, Hou Xin-wen, Zhang Hong-jiang, et al. Learning spatially localized, parts-based representation[C]// Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Los Alamitos, California, USA, 2001(1): 207-212
- [7] Liu Hai-feng, Wu Zhao-hui, Li Xue-long, et al. Constrained non-negative matrix factorization for image representation[J]. IEEE Trans on Pattern Anal Mach Intell, 2012, 34(7): 1299-1311
- [8] Shahnaza F, Berry M W, Paucab V, et al. Plemmons. Document clustering using nonnegative matrix factorization[J]. Information Processing Management, 2006, 42(2): 373-386
- [9] Lovasz L, Plummer M. Matching Theory[M]. North Holland, 1986
- [10] Michael W, Shakhina A, Stewart G W. Computing sparse reduced-rank approximations to sparse matrices[J]. ACM Transactions on mathematical software, 2004, 19(3): 231-235