

一种基于多主体技术的分布式图像聚类算法*

路红¹, 陈立潮^{1†}, 潘理虎^{1,2}, 闫慧敏², 黄河清²

(1. 太原科技大学 计算机科学技术系, 太原 030024; 2. 中国科学院地理科学与资源研究所, 北京 100101)

摘要: 针对医学图像具有对比度较低,不同组织之间的模糊性较高的特点,给出一种基于多主体和数学形态学灰度形态运算的聚类算法。算法采用 agent 技术和多结构元素结合的模式,用结构元素做智能个体,每个不同类型的 agents 随机散布在离散空间格点上,在同时刻控制系统驱动下 agents 根据其自身结构元素的类型用给出的邻域平均算子自主选择作相应的运算进而实现图像聚类。算法无须先验知识和预处理操作,对初始聚类点不敏感,无须事先输入聚类簇数。算法具有分布式并行计算功能和自主分析能力。实验结果验证了该算法的可行性和可靠性。

关键词: 多主体; 图像聚类算法; 多结构元素; 邻域平均算子; 形态学运算

中图分类号: TP391.9

文献标志码: A

文章编号: 1001-3695(2012)11-4353-04

doi:10.3969/j.issn.1001-3695.2012.11.091

Distributed image clustering algorithm based on multi-agent technology

LU Hong¹, CHEN Li-chao^{1†}, PAN Li-hu^{1,2}, YAN Hui-min², HUANG He-qing²

(1. Dept. of Computer Science & Technology, Taiyuan University of Science & Technology, Taiyuan 030024, China; 2. Institute of Geographic Sciences & Natural Resources Research, Chinese Academy of Sciences, Beijing 100101, China)

Abstract: Medical images have the characteristics of lower contrast and higher fuzziness in different organizations. This paper proposed a clustering algorithm, which was based on multi-agent and gray-scale mathematical morphology operations. It combined agent technology and gray-scale mathematical morphology model in the algorithm. And it selected the structural element of the mathematical morphology as agent. It distributed each agents of different types randomly in the discrete spatial grid, and according to the type of its structural elements, it chose the appropriate operations with the given the neighborhood average operator to realize the image clustering in the same time control system driven. The algorithm did not require a priori knowledge and pre-processing operations, and it was not sensitive to the initial cluster points and need not prior input the number of clusters. This algorithm had distributed parallel computing capabilities and self-analysis capabilities. The experimental results show that the algorithm has the accuracy and reliability.

Key words: multi-agent; image clustering algorithm; multi-structure element; neighborhood average operator; morphology operation

0 引言

聚类分析是按照一定要求对数据集进行划分,使数据集内类内模式间的相似性尽量大,类间相似性尽量小。聚类在机器视觉、模式识别、医学图像处理等方面有广泛的应用^[1]。聚类算法的选择取决于待划分的数据类型、聚类的用途及目的等。由于不同领域的应用对聚类算法提出了特殊的要求,而不同的聚类算法也仅适用于某些特定的领域,实现聚类算法的智能性是目前研究的热点问题。

近年来研究者基于多主体技术相继提出了不同的聚类算法。Luo 等人^[2,3]多用 multi-agent 与 Han 等人^[1]研究的聚类算法结合实现了多主体聚类算法。Häkansson 等人^[4]用多智能体在网络和图像方面实现层次聚类不但提高了多主体聚类的效率,而且进一步减少了相邻智能体间的通信开销并加快了数

据的搜索。Moere 等人^[5]提出元胞蚁群相结合的方式论证了分散的多主体能自主地在多维数据集中甄别相似的数据模式并以可视化方式涌现。潘晓英等人^[6]采用密度敏感距离作为数据相似性度量,提出了一种密度敏感的多智能体进化聚类算法。

鉴于医学图像对比度较低,不同组织之间存在模糊性、不均匀性、数据差异性等特点使传统针对医学图像聚类的方法具有一定的局限性。在前期研究成果的基础上^[7],受到潘理虎^[8]、柯新利^[9]等人的启发,尝试用多智能体技术与数学形态学相结合的方法另辟蹊径。利用多主体系统具有的自组织性、独立性、健壮性、动态性、简单性和数学形态学对噪声不敏感、并行性等优点,给出一种智能聚类算法。采用 agent 技术与数学形态学结构元素结合的方式,将初始异构类型 agents 随机分布在离散空间格点上,在同一时钟驱动下并行同步用给出的邻

收稿日期: 2012-03-30; **修回日期:** 2012-04-26 **基金项目:** 国家自然科学基金资助项目(41071344);太原科技大学博士创新基金资助项目(20102030)

作者简介: 路红(1975-),女,江苏徐州人,硕士,主要研究方向为数据挖掘、人工智能;陈立潮(1961-),男(通信作者),教授,博士,主要研究方向为空间数据挖掘、人工智能、模式识别(chen_lc@263.net);潘理虎,男,副教授,博士,主要研究方向为人工智能、复杂系统建模;闫慧敏,女,副研究员,博士,主要研究方向为图像处理与分析、资源利用;黄河清,男,研究员,博士,主要研究方向为复杂地理系统模拟。

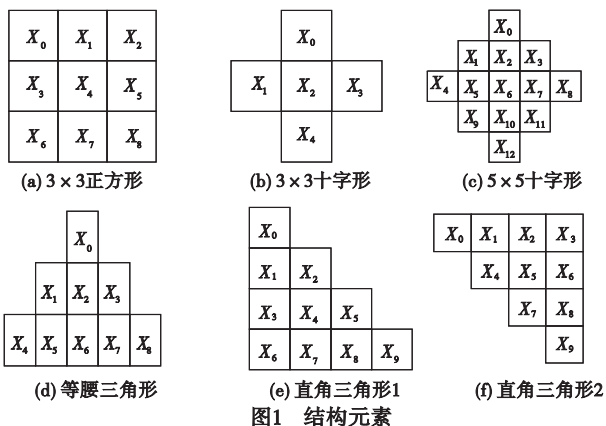
域平均算子进行运算,与此同时用不同颜色标记实现自动聚类,动态呈现聚类结果。该算法对医学图像准确快速地聚类将具有较高的实用价值。

1 基于 agent 和数学形态学结构元素的聚类算法

1.1 算法基本思想概述

ICAA (image clustering algorithm through intelligence agent and multi-structure element) 是对前期研究的 AMMC (clustering algorithm based on agent and mathematic morphology) 算法^[6]的改进。采用多结构元素类型 agents 并赋予其生命周期。算法对聚类数没有要求,实验对象是灰度医学图像。

算法的基本思想是将 agent 技术与结构元素结合起来,每一个 agent 对应一个异构结构元素(图1)。实验中依次分别用绿、黄、蓝、红、灰、洋红六种颜色的空心正方形表示。Agents 有不同的结构元素类型、目标和行为,agents 依据结构元素形状采用 ONA (element center based operator using neighborhood average) 算子自主感知图像局部信息由此进行决策,独立进行判断而后选择是用做何种操作(灰度膨胀运算或灰度腐蚀运算)与其他 agents 进行隐式交互实现自动聚类。Agents 用颜色信息标志热点可直观可视地表达热点位置所属的簇,并且并行、动态在全局层次产生固有的复杂数据聚类模式。由 Agents 局部自适应而涌现的智能求解复杂问题正是 agent 计算的核心^[10]。



ONA 算子的思想是:采用不同类型的 agents,其感知范围和邻域由结构元素的类型决定。选取结构元素的中心点作为热点。图1(a)正方形结构元素热点为 X_4 ,依此类推(b)~(f)热点分别为 X_2, X_6, X_2, X_4, X_5 ,用公式表示如下:

$$f(x, y) = \begin{cases} \alpha & \text{if } (\frac{1}{n} \sum_{i=0}^n x_i < \text{threshold}) \\ \beta & \text{else} \end{cases}$$

其中: α, β 为给定簇的标定值,threshold 为用于调整聚类形状的阈值参数。上述情形是聚类簇数为2的情况将其推广,如聚类簇数是3则用公式表示为

$$f(x, y) = \begin{cases} \alpha & \text{if } (\frac{1}{n} \sum_{i=0}^n x_i < \text{threshold}_1) \\ \beta & \text{if } (\frac{1}{n} \sum_{i=0}^n x_i \geq \text{threshold}_1 \& \& \frac{1}{n} \sum_{i=0}^n x_i \leq \text{threshold}_2) \\ \gamma & \text{if } (\frac{1}{n} \sum_{i=0}^n x_i \geq \text{threshold}_2) \end{cases}$$

各参数解释同上,当对大数据集进行聚类分析时聚类簇数

可能多于三类使用大尺度结构元素可类推上述公式实现,此处不再赘述。

1.2 参数的设定

智能体 A 的属性参数包括如下定义:

$$A(\text{pos}, \text{value}, \text{mark}, \text{sense}, \text{type}, \text{color})$$

其中:pos 描述 agent 在图像环境中的位置信息,一般表示为 $\langle i, j \rangle$ (在边界位置或图像内部);value 表示 agent 所在空间位置的当前图像灰度值;mark 表示已被标志的图像空间位置信息;sense 是 agent 的感知范围;type 表示智能体所用的结构元素类型(图1中的六种形状);color 表示智能结构元素的颜色,与相应的结构元素类型对应。智能体总数量用 numAgents 表示,threshold_k 表示聚类簇的阈值参数,k 的取值是根据划分的簇数来确定的。

在某时刻 t,智能体的行为用下面的行为参数来定义:

$$A(t) = (\theta, \text{steptolive}, \text{move}, \text{breed}, \text{dead})$$

其中: θ 对应于所选择的操作(膨胀或腐蚀);steptolive 代表智能体的存活时间步(agent 在图像边界位置时生命期为1,这样可避免无效的计算,在图像内部时在最大和最小生命期内随机生成);move 表示智能体从当前位置随机移动至其 Moore 邻域;breed 表示繁殖新的智能体;dead 说明满足某种条件智能体从图像中删除。

智能体的行为包括:采用不同类型结构元素感知局部空间像素信息在热点处做标记;做完标记后,智能体在热点的 Moore 邻域图像位置区域的移动行为;智能体生命期终结或超过图像边界的死亡行为;智能体死亡后如果图像仍有未标记的区域则在此区域重生新的智能体的行为。

1.3 ICAA 算法描述过程

ICAA 算法的步骤如下:

- 输入一个大小为 $U_1 \times U_2$ 的灰度数字图像并显示。
- 异构 agents 随机分布到未做过操作的空间位置。
- Agents 在当前位置用变结构元素作灰度形态学运算:若 agent 刚好在图像的边界上用 1×1 的结构元素作灰度形态学运算而后令其死亡;否则用 agent 自身携带的结构元素类型(图1)作运算,并显示其所属的簇。
- 若 steptolive 等于0,则令此 agent 死亡;如果 stepto-live 不等于0并且 agent 存在未做过操作的 Moore 邻域,则 agent 移动到此位置,返回到 c)。
- 若 agent 所在空间位置其 Moore 邻域全做过操作,则此 agent 的任务完成令其死亡,并重新在未做过操作的位置上生成新的 agent,返回到 b)。
- 当没有新的 agents 生成时,聚类结束。
- 显示最终聚类结果。

1.4 边界处理和 agent 死亡的界定

Agents 在执行边界运算时作以下处理:图像的灰度取值区间是 $[0, N]$,图像背景信息值为 -9 999。若 agent 即结构元素的热点所处位置为图像的边界即结构元素非热点位置恰好存在有 -9 999 数值的空间位置信息,此时不论当前 agent 为何种类型均应变为 1×1 结构元素。Agent 在图像边界位置时要用

其自身空间位置信息决定它所属簇,这样就解决了用 ONA 算子做操作而产生地址越界的问题。

agent 死亡的界定:a)在图像边界处的 agent 做完操作后判为死亡,否则 agent 在边界位置上随机游走有可能出界;b) Agent 所在空间位置 Moore 邻域全做过操作,当前 agent 无法移动被困住令其死亡;c) agent 每移动一步导致生命期相应减 1,当 agent 生命周期 $steptolive < 1$ 自然衰退而死亡。以上三种情况令 agent 死亡在未做过操作的空间位置用蒙特卡罗方法随机生成新类型的 agent 进行相应运算直至聚类结束。

2 算法的复杂度分析

2.1 空间复杂度分析

该算法只对原始数据点进行一次扫描,提高了 I/O 的处理速度。算法所需主要内存空间由存储当前活着的智能体及存储图像数据两部分决定。Agents 所在空间位置下一时刻的状态值取决于当前时刻 agents 自身所采用的结构元素的类型。假设六种不同类型 agent 的规模均为 $line \times col$,即每个 agent 所占的存储空间记为 $grid$ 。当代 agents 所需存储空间为 $grid \times n$,其中 n 为当代存活 agents 的数量。而 $grid \times n$ 远小于 N ,故对于数据量为 N 的图像,其空间复杂度即为 $O(N)$,即空间复杂度与图像的大小有关。

2.2 时间复杂度分析

由于 agents 间是隐式交互故只需考虑每个智能体同其直接邻域的计算时间,计算范围被限制在局部空间可以大幅削减计算量。结构元素为 3×3 正方形需要计算每一空间单元的 9 个邻域,结构元素是 3×3 十字形则需要计算每一空间单元的 5 个邻域,依此类推。假设异构 agents 的数量相应记为 n_i ,整个算法的时间复杂度为 $O(\sum_{i=1}^6 \varphi_i n_i A_i)$ 。其中: A_i 是 agent 对应所用结构元素热点邻域计算量; φ_i 为选用此种结构元素所占的比例, $\sum_{i=1}^6 \varphi_i = 1$ 。在硬件环境允许的情况下,时间复杂度与 agents 采用的结构元素形状有关。

3 实验结果及分析

本实验的硬件环境是:2 GHz CPU、主存 2 GB、硬盘 250 GB;软件环境是:Microsoft Windows XP 操作系统,RePast3.0 平台,应用 Java 语言编写。为了验证算法的准确性采用图 1(a)~(f)六种形状结构元素进行实验。

3.1 算法的性能分析

实验所用的测试数据集是 256×256 的脑部医学图像。算法仅需知道图像的最大灰度值和图像背景信息值。对全局模式没有其他已知参数,即无须其他先验知识也无须预处理操作。由于实验选择栅格数据图像故无须做离散化的处理操作,用渐变的方式读图并显示,如图 2(a)所示。

图 2(a)所示的原始图像通过目视分析可知应划分为三类。为了衡量聚类的效果,采用图 1 中的六种结构元素按 0.1、0.1、0.2、0.2、0.2、0.2 比例分配 300 个智能体。使用冗余的 agents 能得到鲁棒的效果,并且能够降低计算时间,克服时

空限制的影响。图 2(b)是初始时散布 agents 的快照,依次用绿色、黄色、蓝色、红色、灰色和洋红代表正方形、十字形、钻石形、等腰三角形、下直角三角形、上直角三角形智能结构元素的类型。Threshold 用来调整簇的形状。首次运行程序后,如果不满意聚类效果可实时调整阈值参数,可把期望聚类到特定簇内的图像位置上的属性信息囊括进去。图 2(b)~(h)是阈值为 40、100 的 ICAA 算法提取聚类结果动态过程。用空心正方形表示 agents 的好处是可以即时观察聚类过程。在初始分配 agents 时,agents 是用不同颜色的空心正方形表示,而在下一时刻 agents 处的位置就自动划归所属的簇,并用簇的颜色呈现。Agents 状态有两种,或由于生命周期终止而死亡,或移到相应 Moore 邻域位置。实时实验观察得知,黄色十字形 agents 跑得较快而蓝色钻石形结构元素运算相对慢一些,这是因为钻石形结构元素计算量比十字形大一些。进一步验证了时间复杂度分析的准确性。实验表明:ICAA 算法能够发现隐含在数据集中的簇并予以识别,同时也准确地反映了原有图像数据的空间几何特征且能快速并行处理数据。本算法采用随机重启的局部搜索方式并不受初始点的影响亦不会掉入搜索空间的局部最优解陷阱。如遇到图像数据信息较少的点而导致 ICAA 算法失效时,可调整为单一类型的 agents 继续运行。

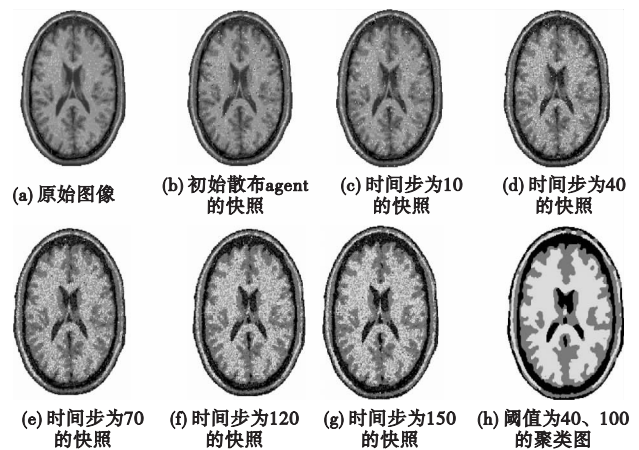


图2 阈值为40、100图像聚类过程演示

3.2 输入参数调整

调整参数实验如下:

在聚类簇数固定的情况下,单独对阈值参数进行调整,令 $threshold_1$ 、 $threshold_2$ 分别为 50、110、60、110、40、100、36、80,发现算法执行时间不受其影响,阈值的变动仅对簇的形状有控制作用,如图 3 所示。其中 $threshold_1$ 为调整灰质簇的形状, $threshold_1$ 越大灰质部分也相应增大(聚类结果图上黑色部分表示); $threshold_2$ 、 $threshold_1$ 之差为调整脑脊液的形状,其大小随着差的增加而增加(聚类结果图上较深色部分表示);图像最大灰度值与 $threshold_2$ 之差决定白质的大小(聚类结果图上较浅色部分表示)。从图 3 得知阈值分别为 50、110 时能较好地提取图像的特征聚类,效果比较理想。

单独调整 $steptolive$ 智能体生命期,初始时在小于 30 的整数间随机生成生命期,同样可以使 $steptolive$ 为 1,这样 agents 做完操作就死亡,不必在 Moore 邻域移动。参数 $steptolive$ 的改变对执行时间也不构成任何影响。

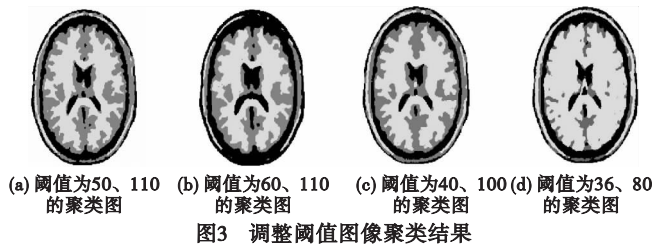
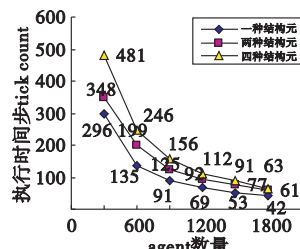
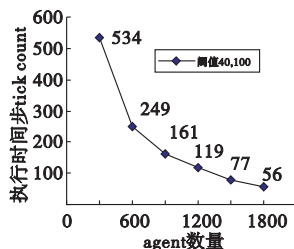


图4是阈值固定为40、100, *steptolive* 为2~11, *num Agents* 从300起,以步长300变化到1800的性能曲线。当 *agents* 数量为300时聚类完成所需时间步为534, *agents* 数量为1500时聚类完成所需时间步为77。由图可知,执行时间随 *agents* 数量的增加成反比变化。由于 *agents* 间隐式交互,故不会发生一个位置被两个 *agents* 占据的情况,调整 *agents* 的数量没有负面影响即没有其他额外开销,随着 *agents* 数量的增加能更快地减少算法执行时间(本实验环境下 *numAgents* 小于7000)。

在阈值固定为40、100调整 *agents* 类型的比例分别用一种十字形结构元素,十字形和正方形两种结构元素按1:1分配及十字形、正方形、下直角三角形、上直角三角形四种结构元素按1:1:1:1分配同样对图2(a)进行实验。上述三个实验均能在聚类结束时呈现图2(h)的最终结果,亦得出 *agents* 数量与执行时间成负线性相关即随着 *agent* 数量的增加执行时间相应缩减(图5)。这三个实验相比较而言,单一十字形结构元素运行时间最短,两种类型结构元素次之;四种类型结构元素最慢,验证了算法的时空复杂度。



虽然在 *numAgents* 相同的条件下多种结构元素执行时间稍稍逊色,但采用异构类型结构元素为进一步满足后续处理空间图像的异质性作了充分的准备,并且体现了 *agents* 类型的多样性和灵活性。同时这一不足可通过增加 *numAgents* 进行弥补。

4 结束语

ICAA 算法可快速处理任意、复杂的聚类形状,克服了需要

预先输入聚类簇数的缺陷,并解决现有聚类技术对聚类初始点敏感及易陷入局部簇的限制,易于实现高性能并行运算。从实验结果看,ICAA 算法可保证聚类的质量和实时性、合理性、准确性和灵活性。本文先尝试用六种不同类型的 *agent* 而后希望对图像不同的区域用特定的结构元素类型及转换规则来实现自动聚类。结果表明,本算法对于快速准确聚类,定位医学图像中的病灶具有明显效果,有较高的应用价值。下一步将继续与应用相结合,针对同场景连续图像的快速聚类进行算法改进。

参考文献:

- [1] HAN Jia-wei, MICHELINE K. Data mining: concepts and techniques [M]. [S. l.]: Simon Fraser University, 2006:405-410.
- [2] CHAITMONTREE S, ATKINSON K, COENEN F. Multi-agent based clustering: towards generic multi-agent data mining [C]//Proc of the 10th Industrial Conference on Advances in Data Mining: Applications and Theoretical Aspects. Berlin: Springer Verlag, 2010:115-127.
- [3] LUO Chao, ZHAO Yan-chang, LUO Dan, et al. Agent-based subspace clustering [C]//Proc of the 15th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining. Berlin: Springer Verlag, 2011:370-381.
- [4] HÅKANSSON A, HARTUNG R L. Autonomously creating a hierarchy of intelligent agents using clustering in a multi-agent system [C]//Proc of International Conference on Artificial Intelligence. 2008:89-95.
- [5] MOERE X A, CLAYDEN J J, DONG A. Data clustering and visualization using cellular automata ants [C]//Proc of the 19th Australian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence. Berlin: Springer Verlag, 2006:826-836.
- [6] 潘晓英, 刘芳, 焦李成. 密度敏感的多智能体进化聚类算法[J]. 软件学报, 2010, 21(10):2420-2431.
- [7] 路红, 陈立潮, 潘理虎, 等. 基于多主体技术和数学形态学的空间聚类算法[J]. 计算机工程, 2012, 38(18):158-161.
- [8] 潘理虎, 黄河清. 农业土地利用变化的人工社会模型研究[J]. 系统仿真学报, 2010, 22(8):1964-1969.
- [9] 柯新利, 邓祥征. 内嵌空间聚类算法的分区地理元胞自动机建模与应用[J]. 地球信息科学学报, 2010, 12(3):365-371.
- [10] 钟伟才, 刘静, 刘芳, 等. 组合优化多智能体进化算法[J]. 计算机学报, 2004, 27(10):1341-1353.

(上接第4333页)

- [8] NANNI L, MAIO D. Weighted sub-Gabor for face recognition [J]. Pattern Recognition Letters, 2007, 28(4):487-492.
- [9] WANG Jian-zhong, MA Zhi-qiang, ZHANG Bao-xue, et al. A structure-preserved local matching approach for face recognition [J]. Pattern Recognition Letters, 2011, 32(3):494-504.
- [10] 倪雪蕾. 局部型子模式特征提取方法探讨及人脸识别应用[D]. 南京:南京航空航天大学, 2006.
- [11] WANG Jian-zhong, ZHANG Bao-xue, WANG Shu-yan, et al. An adaptively weighted sub-pattern locality preserving projection for face recognition [J]. Journal of Network and Computer Applications, 2010, 33(3):323-332.
- [12] XUE Hui, ZHU Yu-lian, CHEN Song-can. Local ridge regression for

face recognition [J]. Neurocomputing, 2009, 72(4-6):1342-1346.

- [13] ZHU Y li-lian. Sub-pattern non-negative matrix factorization based on random subspace for face recognition [C]//Proc of International Conference on Wavelet Analysis and Pattern Recognition. 2007:1356-1360.
- [14] SIM T, BAKER S, BSAT M. The CMU pose, illumination, and expression (PIE) database [C]//Proc of IEEE international conference on automatic face and gesture recognition. Washington DC: IEEE Computer Society, 2002.
- [15] LEE K C, HO J, KRIEGMAN D. Acquiring linear subspaces for face recognition under variable lighting [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2005, 27(5):684-698.

BspLPP、SpLPP、SpPCA 的运行时间低于标准 LPP 和 PCA,其中,随着分块数目的增多,SpLPP 的运行时间不断上升,当分块数目为 32 时,运行时间最多。BspLPP 的运行时间相对较短且最稳定,相对于 SpLPP、LPP 和 PCA,至少分别节省 60%、80%、80% 的运行时间。因此,本文提出的算法可以应用于维数较大的数据集,随着分块数目的增多,运行时间基本不变。

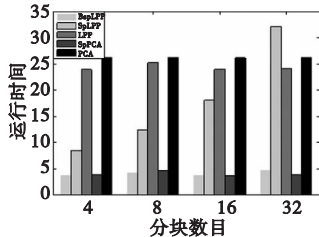


图2 不同图像划分方式下的运行时间

本文对式(2)中的参数 t 和 k (近邻个数) 的选择进行实验对比。经过对表 1 的解释说明,将每幅图像平均划分成 8 个子块,并且使降维后每幅图像的维数为 600 维左右。固定参数 $t=600$,分别令 $k=3, 5, 7, 9$ 和 11,其分类识别率如表 2 所示。由表可知,当 $k=7$ 时该算法的分类识别率最高, k 值过小可能造成局部信息不完整, k 值过大可能会引起降维后局部流形结构失真。然后,固定参数 $k=7$,分别令 t 取不同的参数,如表 3 所示,分类准确率随着 t 值的增大而增大,当 $t=600$ 时其分类准确率最高,随后有下降趋势。

表 2 不同 k 值下的分类识别率

k 值	识别率
3	97.02 ± 1.77
5	97.18 ± 1.17
7	97.30 ± 1.49
9	96.81 ± 2.73
11	97.25 ± 4.96

表 3 不同 t 值下的分类识别率

t 值	识别率
800	97.05 ± 1.20
600	97.55 ± 1.55
400	97.30 ± 1.08
200	97.16 ± 1.79
100	96.63 ± 1.92
10	94.83 ± 2.80

2.2 数据集 YaleB

人脸数据库 YaleB 包含 38 个人,每人的图像数目约为 64 幅,其中 32 幅作为训练数据,剩余图像作为测试数据,图像拍摄于不同的光照、时间、角度、表情以及姿态等条件,所有图像都被调整为 32 × 32 像素。实验中,将每幅图像划分成 16 块,即 8 × 8 维的子模块。

对于子模块运用 LPP 降维到不同维数后的分类识别率如图 3 所示。首先使用子模块思想的算法 BspLPP、SpLPP 和 SpPCA 的分类精度远远高于标准的 LPP 和 PCA,充分说明基于子模块思想的算法能更精确地保持数据的局部信息结构。其次,随着维数的变化,BspLPP 与 SpLPP 的分类精确度趋于平稳,而 SpPCA 的分类精度不稳定,说明 BspLPP 与 SpLPP 的性能更加稳定。相对于其他算法,BspLPP 即使将子模块维数降到很小(如 8 维)依然具有较高的分类精度,这是其他几种算法所不具备的优势,同时,该算法的分类准确率略高于 SpLPP,当子模块降维到 30 维左右时达到最高。

对于子模块运用 LPP 降维到不同维数后的运行时间如图 4 所示。随着降维后维数的增多,多数算法的运行时间都呈上升趋势,只有 BspLPP 的运行时间增长到一定程度时具有下降趋势。与 SpLPP、LPP 和 PCA 相比,BspLPP 的运行时间较低,但略高于 SpPCA。基于子模块思想的算法运行时间总体上低于标准 LPP 和 PCA。

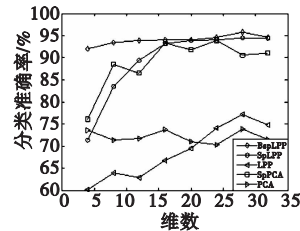


图3 子模块降到不同维数后的分类识别率

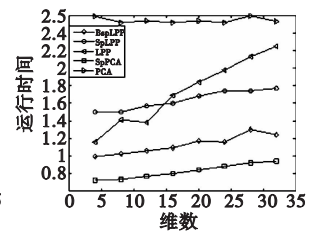


图4 子模块降到不同维数后的运行时间

在该数据集上,同样对算法中参数 k 和 t 的值进行筛选:

- a) 固定参数 $k=5$,令 t 分别取不同的值,如表 4 所示,当 $t=10$ 时该算法具有最高的分类准确率;b) 固定参数 $t=10$,分别令 k 取不同的值,如表 5 所示,随着 k 值的减小识别率逐渐升高,当 $k=5$ 时达到最高,然而当 $k=3$ 时,识别率急剧下降,原因可能是 k 值过小造成局部信息不准确,易受噪声干扰。

表 4 不同 t 值下的分类识别率

t 值	识别率
800	92.39 ± 1.51
500	92.47 ± 1.49
100	93.96 ± 0.02
10	95.94 ± 1.07
5	95.53 ± 1.29
1	94.04 ± 0.55

表 5 不同 k 值下的分类识别率

k 值	识别率
3	93.05 ± 0.50
5	95.16 ± 0.77
7	94.45 ± 2.73
9	93.96 ± 1.07
11	93.87 ± 0.08

3 结束语

人脸识别问题中,已存在的基于子模块的流形学习算法将所有个体的人脸图像作为一个整体,进而进行子图像的划分与子模块的组合,即假设所有类别的流形是统一的。而实际上不同类别以及同一类别的不同模块均具有其独特的流形结构,因此,本文提出 BspLPP 人脸识别算法,较好地保持了图像的局部信息特征和流形结构。实验表明,该算法提高了分类准确率,同时大幅减少了运行时间。下一步的工作是深入分析不同的模块之间是否具有某种相关性。

参考文献:

- [1] XU Yong, ZHONG Ai-ni, YANG Jian, et al. LPP solution schemes for use with face recognition[J]. *Pattern Recognition*, 2010, 43(12): 4165-4176.
- [2] XU Yong, SONG Feng-xi, FENG Ge, et al. A novel local preserving projection scheme for use with face recognition[J]. *Expert Systems with Applications*, 2010, 37(9): 6718-6721.
- [3] 张国印,楼宋江,程慧杰,等. 基于局部和全局的特征提取算法及在人脸识别中的应用[J]. *计算机科学*, 2009, 36(8): 285-287.
- [4] ZHAO W, CHELLAPPA R, PHILLIPS P J, et al. Face recognition: a literature survey[J]. *ACM Computing Surveys*, 2003, 35(4): 399-458.
- [5] ZOU Jie, JI Qiang, NAGY G. A comparative study of local matching approach for face recognition[C]. *IEEE Trans on Image Processing*, 2007, 16(10): 2617-2628.
- [6] HSIEH P C, TUNG P C. A novel hybrid approach based on sub-pattern technique and whitened PCA for face recognition[J]. *Pattern Recognition*, 2009, 42(5): 978-984.
- [7] KADAPPAGARI V K, ATUL N. SubXPCA and a generalized feature partitioning approach to principal component analysis[J]. *Pattern Recognition*, 2008, 41(4): 1398-1409.

(下转第 4356 页)