文章编号: 1001-0920(2008)06-0701-04

一种图像数据库聚类与归类方法研究

谢从华[1], 沈钧毅[2], 宋余庆[3], 常晋义[1]
(1. 常熟理工学院 计算机科学与工程系, 江苏 常熟 215500; 2. 西安交通大学 电子与信息工程学院, 西安 710049; 3. 江苏大学 计算机科学与通信工程学院, 江苏 镇江 212013)

摘　要: 为克服目前图像库浏览系统存在的效率低和交互性差等缺点,提出一种新的图像库聚类与归类方法. 采用随机抽取图像特征矩阵的列和网格划分矩阵的行,求图库特征的近似最优稀疏矩阵. 对此矩阵进行奇异值分解,以实现高维图像特征库降维;然后采用密度估计技术和爬山策略,定义和提取图像数据库的聚类以及归类. 实验表明,该方法比其他方法交互性好、速度快,具有对噪声数据不敏感等优点.

关键词: 图像库聚类; 图像库归类; 奇异值分解; 密度估计; 爬山算法
中图分类号: TP391　　**文献标识码**: A

Method of image database clustering and categorization

XIE Cong-hua[1], *SHEN Jun-yi*[2], *SONG Yu-qing*[3], *CHANG Jin-yi*[1]
(1. Department of Computer Science and Engineering, Changshu Institute of Technology, Changshu 215500, China; 2. School of Electronics and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China; 3. School of Computer Science and Telecommunication, Jiangsu University, Zhenjiang 212013, China. Correspondent: XIE Cong-hua, E-mail: x7c8h5@yahoo.com.cn)

Abstract: To overcome the disadvantages of low efficiency and bad interactive ability of image database browse system, a new method based on image database categorization and clustering is offered. By randomly sampling columns and grid segmenting rows of raw feature matrix, the approximate optimum sparse matrix of image database features is obtained. By the singular value decomposition (SVD) of this sparse matrix, the high dimensions of image database feature matrix can be reduced. Density estimation technique and hill-climbing strategy are used to define and extract image database clusters and categorization. Experimental results show that the method has interactive ability and good speed, and is not sensitive to noise data.

Key words: Image database clustering; Image database categorization; Singular value decomposition (SVD); Density estimation; Hill-climbing

1 引　　言

面对日益庞大的图像库,为用户提供一个简洁高效的图像搜索和浏览方案是一个充满挑战的课题. 目前,基于内容的图像检索是解决此类问题的主流技术[1],但很多系统都要求用户提供图例或草图,然后在图库中查找与图例相似的图像. 主要存在以下两方面的不足:

1) 交互性比较差. 因为要求用户对图库的内容比较熟悉,能提供图例,这对于大型图库是很难做到的,所以人机交互性比较差.

2) 检索的效率不高. 因为这些系统将图例与图库中的每一幅图进行比较,这对于大型图库而言,搜索代价比较大.

聚类分析将一个数据集中的元素按适当的相似性准则划分为若干个互不相交的子集. 先对图库进行聚类,再检索,则可以缩小检索范围,提高效率. 郑欣等人[2]在谱聚类的基础上,提出一种保局聚类的方法;Drineas 等人[3]提出一种基于奇异值分解的大型图库聚类方法;还有谱聚类[4]和基于学习的谱聚类[5]等图库聚类算法. 这些方法有效地对图库进行

收稿日期: 2007-04-30; **修回日期**: 2007-07-27.
基金项目: 国家自然科学基金项目(60572112); 江苏省软件与集成电路专项基金项目(苏信软[2005]196); 常熟理工学院青年启动基金项目(ky200657).
作者简介: 谢从华(1978—),男,重庆人,讲师,从事图像处理、图像挖掘的研究; 沈钧毅(1939—),男,江苏常熟人,教授,博士生导师,从事数据挖掘、智能信息控制等研究.

聚类,但存在以下两方面的不足:1) 只进行图库聚类,并没有选出该聚类中有代表性的图像进行归类,交互性比较差;2) 谱分解或奇异值分解后的聚类,存在对噪音敏感,不能处理任意形状等缺点.

图像数据库"归类"就是选出每一个聚类中最具代表性的图例所组成的集合. Bertrand Le Saux 等人[6]提出一种自适应健壮竞争(Adaptative Robust Competition,ARC)的图库归类方法;Frigui 等人[7]提出了自组织振荡网络(Self-Organization of Oscillators Network,SOON)的图库归类方法. 这些方法选出了每一个聚类的代表性图像对图库进行归类. 但直接在高维空间上归类,速度和效率比较差,而且所选出的图库样本太多,需要用户多次选择图例.

图像数据库的聚类和归类问题可描述为:给定 m 个 n 维向量的图像样本集合,其样本矩阵 $X_{m\times n}=[X_1,X_2,\cdots,X_m]^{\mathrm{T}}$,采用一定的方法聚类可得到 r 个聚类$\{\beta_1,\beta_2,\cdots,\beta_r\}$,选出每一个聚类的最具有代表性图例得到图像数据库归类$\{X_1^*,X_2^*,\cdots,X_r^*\}$. 为此,本文提出一种基于奇异值分解和密度估计的图库聚类与归类方法.

2 基于奇异值分解的图像特征库降维

如何选择有效的特征是提高高维图库聚类和归类效率的关键技术之一. 通常情况下,对高维小样本问题采用主成分分析法,需要较大的计算量. 而奇异值分解可以将高维矩阵转化为低维的特征向量,从而减少了计算量. 本文在奇异值分解的基础上,提出一种简单快速有效的降维方法. 与文献[3-6]相比,本文和文献[3]采用了降维方法,减少了聚类和归类的计算量. 与文献[3]相比,本文在减少样本矩阵列数的同时,采用了不等宽的网格划分,将稠密矩阵变成稀疏矩阵.

定理1 设 $X_{m\times n}$ 的秩为 $\mathrm{rank}(X)=r$,其奇异值为 $\lambda_1,\lambda_2,\cdots,\lambda_r$,那么存在 m 阶酉矩阵 U 和 n 的酉矩阵 V,使得

$$X_{m\times n}=U\sum V^{\mathrm{T}}. \tag{1}$$

其中 $U_{m\times m}$ 和 $V_{n\times n}$ 分别为 X 的左奇异特征向量和右奇异特征向量,即 XX^H 和 X^HX 的标准正交特征向量,对应的特征值 $\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_r>0$,构成 $\sum=\mathrm{diag}(\lambda_1,\lambda_2,\cdots,\lambda_r,\cdots,\lambda_{\min(m,n)})$ 对角阵.

小规模低维图库的奇异值分解比较容易,但大型高维图库(如 $m\geq10^5$,$n\geq100$)的精确奇异值分解则比较困难. 文献[3]根据概率分布,采用随机抽取技术,只提取 X 的 n 维空间的 k 维子空间 D 作为 X 的近似(k 远远小于 m 和 n),设计了一种简单有效的随机抽取列技术,并证明了由此产生的是 X 的最优近似矩阵 X_k. 虽然此方法减少了列数,但 X 是稠密矩阵,奇异值分解运算量仍很大. 为此,本文在不同的维上采用不同的网格宽度进行划分,将某一列的同一个区间的不同数都用区间的中心点代替,以便化简为稀疏矩阵. 这里用 X_i 表示样本 i,X^i 表示矩阵的第 i 列.

首先,求出每一个 X 在第 j 维上的最大值 $\max(X^j)$ 和最小值 $\min(X^j)$;其次,第 j 维上的数据按宽度 $\delta_j=(\max(X^j)-\min(X^j))/m$ 的网格划分,同一个区间的不同数据用该区间的中心点代替;最后进行奇异值分解. 这样,在第 j 维上就产生了很多相同的数据块,将稠密矩阵变成稀疏矩阵,可以加快矩阵数据分解的速度.

综上所述,本文设计的近似快速奇异值特征分解(Approximation Fast Singular Value Decomposition,AFSVD)描述如下:

算法1 AFSVD

输入:$X_{m\times n}=[X_1,X_2,\cdots,X_m]^{\mathrm{T}}$ // X 为特征向量;

输出:$H_{m\times k},\lambda_1,\lambda_2,\cdots,\lambda_k$ // λ 为特征值.

1) $p(l)=\|X^l\|^2/\|X\|_F^2$; // $1\leq l\leq n$;

2) 随机抽取 Y 中的 c 列特征,构造 $X_{m\times n}$ 的近似矩阵 $C_{m\times c}$,第 i 列为 $X^i/\sqrt{cp_i}$,$1\leq i\leq c$;

3) 计算近似矩阵 $C_{m\times c}$ 各列的划分宽度 δ_j;

4) 构造稀疏矩阵 D,If $C_{ij}\in[\min(C^j)+(t-1)\delta_j,\min(C^j)+t\delta_j]$,Then C_{ij} 映射为 $D_{ij}=\min(C^{(j)})+(t+1/2)\delta_j$,$1\leq t\leq m$;

5) 计算 D 的最大 k 个左奇异特征向量($h^{(1)},h^{(2)},\cdots,h^{(k)}$),返回 $H=(h^{(1)},h^{(2)},\cdots,h^{(k)})$.

3 基于密度估计和爬山算法的图像库聚类和归类

首先用 $H_{m\times k}$ 代替 $X_{m\times n}$ 进行降维;然后对 H 聚类和归类. 文献[2,4]在 H 上进行 k-均值聚类,存在对噪音敏感,不能处理任意形状等缺点. 为此,本文在密度聚类[8]的基础上,给出一种新的图库聚类和归类的方法. 与文献[2,4]相比,本文方法可有效克服上述缺点. 与文献[8]相比,本文方法有以下两个优点:1) 本文是在奇异值分解后的低秩近似矩阵上进行的,所以时间和空间性能比较好;2) 本文在聚类的同时对图像数据库进行归类,可以提供所有具有代表性图例,供用户浏览大型图像数据库.

1) 密度函数:矩阵 $H_{m\times k}$ 空间上的密度函数

$$f_{\mathrm{Gauss}}^{H}(H_j)=\sum_{i=1}^{m}\mathrm{e}^{-\frac{d(H_i,H_j)}{2\sigma}}. \tag{2}$$

其中:$j=1,2,\cdots,m$;σ 为预先给定的正数,称为平

滑参数. 但是,大规模图库的全局密度函数计算量太大. 可构造一个局部密度函数表示相邻数据对它的影响,即

$$f_{\mathrm{Gauss}}^{\mathrm{Near}(H_j)}(H_j) = \sum_{H_i \in \mathrm{Near}(H_j)} \mathrm{e}^{-\frac{d(H_i,H_j)}{2\sigma}}, \qquad (3)$$

其中 Near:$\mathrm{Near}(H) = \{H^* \mid d(H,H^*) < k\sigma, k \in Z\}$. 实验表明,$k = 4$ 足够表示相邻数据的影响.

2) 聚类代表性图例: 矩阵 $H_{m\times k}$ 空间上的点 H^* 称为聚类代表性图例,当且仅当密度函数 $f_{\mathrm{Gauss}}^{\mathrm{Near}}$ 在 H^* 处取得局部极大值.

3) 图库聚类与噪声图: 给定 $\xi > 0$, H^* 为矩阵 $H_{m\times k}$ 的聚类代表,且 $f(H^*) \geq \xi$,则由 H^* 所代表的全体图像所构成的集合称为以 H^* 为中心的图库聚类. 如果 $f(H^*) < \xi$,则称由 H^* 所代表的数据点为噪声图.

4) 图库归类: $H_{m\times k}$ 中所有聚类代表性图例 H^*,且满足 $f(H^*) \geq \xi$ 的集合是图库归类.

在图库的近似密度函数上,本文采用爬山策略进行聚类和归类. 基于核密度估计的图像库聚类(Kernel Density Estimation Based Image Database Clustering, KDEIDC) 的形式化算法如下:

算法 2　KDEIDC

输入: 矩阵 $H_{m\times k}$, 平滑参数 σ 和密度阈值 ξ;

输出: 图库聚类 C, 图库归类 CA 和噪声图集 N.

1) Scan $H_{m\times k}$ and set all as unmarked;

2) $C = \varnothing$; $CA = \varnothing$; $N = \varnothing$;

3) 对未标记点 H_j 计算其局部密度函数 $f_{\mathrm{Gauss}}^{\mathrm{Near}}$ 和梯度, $j = 1,2,\cdots,m$;

4) 沿梯度方向,根据 HILL_Climbing(H_j) 找到局部最大的 H^*;

5) If H^* 不在 CA 中, Then $CA = CA \cup H^*$;

6) If $(f(H^*) \geq \xi)$, Then $C[i] = H_j \cup \mathrm{near}(H_j) \cup H^* \cup \{P \mid H_j \rightarrow H^*\}$, 并标记这些点;

7) Else $N = N \cup H_j \cup H^* \cup \{P \mid H_j \overset{P}{\rightarrow} H^*\}$, 并标记这些点;

其中, HILL-Climbing(H_j) 过程如下:

1) $B = H_j$; $R = B + \delta \frac{\nabla f_{\mathrm{Gauss}}(B)}{|\nabla f_{\mathrm{Gauss}}(B)|}$;

2) while $f(R) \geq f(B)$;

3) $\{B = R; R = B + \delta \frac{\nabla f_{\mathrm{Gauss}}(B)}{|\nabla f_{\mathrm{Gauss}}(B)|}\}$;

4) $H^* = R$.

在算法 KDEIDC 中,步骤 1) 扫描一遍数据库,所以时间复杂度为 $O(\|H\|)$,步骤 2) 为 $O(1)$. 因为爬山算法最坏的时间复杂度为 $O(\log(\|H\|))$,所以步骤 3) ～ 步骤 7) 的最坏时间复杂度为 $O(\|H\|\log(\|H\|))$,故 KDEIDC 的最坏复杂度为 $O(\|H\| + \|H\|\log(\|H\|))$.

4　实验结果

本文用 Vc ++. Net 在内存 512 M, CPU 为 Pentium Ⅳ1.73 GHz, Windows 2 000 的机器上实现 ARC[6], SOON[7] 和本文方法. 采用的数据集为哥伦比亚 COIL100[9] 图库,其中有 100 个对象,每 1 个对象有 72 幅图片,共计 7 200 幅. 提取的特征有基于颜色直方图 (32 维)、基于颜色直方图布局 (32 维)、基于颜色矩 (9 维) 和基于纹理共生矩阵 (16 维) 4 大类,共计 89 维. 为了简化实验,本文选取其中 20 个物体,每一个物体随机选取 45 幅图. 从其余 80 个物体中每个物体选取 1 或 2 幅,共计 100 幅的噪音图像,如图 1 和图 2 所示.

图 1　同一个物体不同角度的照片样例

图 2　图库中不同物体的照片样例

SOON, ARC 和本文方法的实验结果如表 1 所示. SOON 和 ARC 方法会将一些聚类分裂出几个聚类,所以图库归类类别数目比较多,错误率比较高. 从表 1 可以看出,本文方法最接近合理的归类数目,图库归类错误率比较低. 本文方法对噪声图像识别率远远大于 SOON 和 ARC 方法. 本文采用了快速

表 1　不同聚类方法的结果比较

	ARC	SOON	本文方法
图库归类类别数	31	42	23
图像归类的错误率 / %	20	35	10
噪声图像识别率 / %	89	82	98
时间 / s	212	223	98
聚类维数	89	89	10

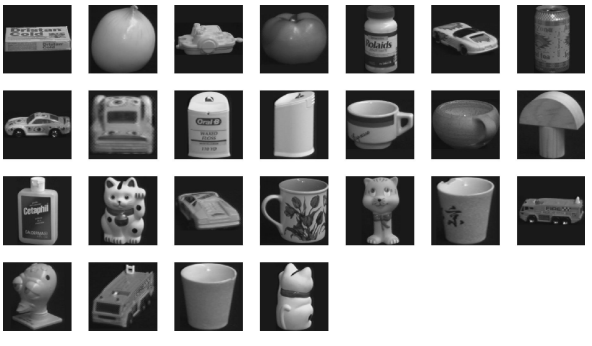

图 3　本文方法聚类中的代表性图例

的奇异值分解方法,维数降低到 SOON 和 ARC 的 1/9,所以速度明显要快几倍. 本文方法所得到的图库归类如图 3 所示.

5 结 论

本文首先改进奇异值分解,用于高维空间的降维;然后构造低维空间的密度函数,通过爬山策略实现图像数据库的聚类和归类. 一方面,利用图库聚类不需要固定 k 值以及对噪音不敏感等优点,可以加快基于内容的图像检索的速度. 另一方面,在聚类的基础上提取每一个聚类中最具有代表性的图例供用户使用,改善了基于内容的图像检索的人机交互性能和界面.

参考文献(References)

[1] Datta Ritendrata, Li Jia, Wang James Z. Content-based image retrieval-approaches and trends of the new age [C]. Proc of the 7th Int Workshop on Multimedia Information Retrieval. Singapore: Conjunction with ACM Int Conf on Multimedia, 2005: 253-262.

[2] 郑欣, 林学闾. 图像数据库的保局聚类[J]. 计算机研究与发展, 2006, 43(3): 463-469.
(Zheng X, Lin X Y. Locality preserving clustering for image database [J]. J of Computer Research and Development, 2006, 43(3): 463-469.)

[3] Petros Drineas, Alan M Frieze, Ravi Kannan, et al. Clustering large graphs via the singular value decomposition[J]. Machine Learning, 2004, 56(3): 9-33.

[4] Andrew Y Ng, Michael I Jordan, Yair Weiss. On spectral clustering: Analysis and an algorithm [M]. Cambridge: MIT Press, 2001.

[5] Francis R Bach, Michael I Jordan. Learning spectral clustering[M]. Cambridge: MIT Press, 2004.

[6] Bertrand Le Saux, Nozha Boujemaa. Unsupervised robust clustering for image database categorization[C]. IEEE-IAPR Int Conf on Pattern Recognition (ICPR'2002). Quebec, 2002, (1): 259-262.

[7] Frigui Hichem, Nozha Boujemaa, Soon-Ann Lim. Unsupervised clustering and feature discrimination with application to image database categorization[C]. Proc of the IFSA World Congress and 20th NAFIPS Int Conf. Vancouver, 2001: 401-406.

[8] Alexander Hinneburg, Daniel A Keim. A general approach to clustering in large databases with noise[J]. Knowledge and Information Systems, 2003, 5(4): 387-415.

[9] Nene S A, Nayar S K, Murase H. Columbia object image library (coil-100) [R]. New York: Columbia University, 1996.

(上接第 700 页)

参考文献(References)

[1] Goldberg D E. Genetic algorithms in search, optimization and machine learning, reading[M]. MA: Addison-Wesley, 1989.

[2] Emily G, Jessie B. Parallel genetic algorithms: An exploration of weather prediction through clustered computing[J]. J of Computing Sciences in Colleges, 2003, 18(5): 272-273.

[3] Enrique A, Francisco L, Antonio J N. Parallel heterogeneous genetic algorithms for continuous optimization[J]. Parallel Computing, 2004, 30(5): 699-719.

[4] 刘立芳, 霍红卫, 王宝树. PHGA-COFFEE:多序列比对问题的并行混合遗传算法求解[J]. 计算机学报, 2006, 29(5): 727-733.
(Liu L F, Huo H W, Wang B S. PHGA-COFFEE: Aligning multiple sequences by parallel hybrid genetic algorithm[J]. J of Computers, 2006, 29(5): 727-733.)

[5] Kohlmorgen U, Schmeck H, Haase K. Experiences with fine-grained parallel genetic algorithms[J]. Annals of Operations Research, 1999, 90: 203-219.

[6] Martin P, Prasanna P, Arun R. Fine-grained parallel genetic algorithms in Charm + + [J]. ACM Crossroads Magazine: Parallel Computing, 2002, 8(3).

[7] Fernando G L, Claudio F, Hugo M. Massive parallelization of the compact genetic algorithm [C]. Proc of the Int Conf on Adaptive and Natural Computing Algorithms. Coimbra, 2005: 530-533.

[8] Jowens J D, Luebke D, Govindaraju N. A survey of general purpose computation on graphics hardware[C]. Euro-Graphics 2005. Dublin, 2005: 21-51.

[9] Fok K L, Wong T T, Wong M L. Evolutionary computing on consumer-level graphics hardware [J]. IEEE Intelligent Systems, 2005, 22(2): 69-78.

[10] Qizhi Yu, Chongcheng Chen, Zhigeng Pan. Parallel genetic algorithms on programmable graphics hardware [J]. Lecture Notes in Computer Science, 2005, 36(12): 1051-1059.

[11] 吴恩华. 图形处理器用于通用计算的技术现状及其挑战[J]. 软件学报, 2004, 15(10): 1493-1504.
(Wu E H. State of the art and future challenge on general purpose computation by graphics processing unit[J]. J of Software, 2004, 15(10): 1493-1504.)