

文章编号: 1001-2092(2010)01-2001-06

结构复杂数据的半监督聚类

黄添强¹, 余养强^{1a}, 秦小麟²

(1. 福建师范大学 a. 数学与计算机科学学院, b. 网络安全与密码技术福建省高校重点实验室, 福州 350007; 2. 南京航空航天大学 计算机科学与工程系, 南京 210016)

摘要: 基于成对限制, 提出一种半监督聚类算法(SCCD), 它能够处理存在多种密度结构复杂的数据且识别任意形状的簇. 利用成对限制反映的多密度分布信息计算基于密度的聚类算法(DBSCAN)的邻域半径参数 Eps, 并利用不同参数的 DBSCAN 算法处理复杂形状且密度变化的数据集. 实验结果表明, SCCD 算法能在噪声环境下发现任意形状且多密度的簇, 性能优于已有同类算法.

关键词: 半监督聚类; 多密度数据; 基于密度聚类; 成对限制

中图分类号: TP311.1; TP18

文献标识码: A

Semi-supervised clustering for complicated data

HUANG Tianqiang¹, YU Yangqiang^{1a}, QIN Xiaolin²

(1a. School of Mathematics and Computer Science, 1b. Key Lab of Network Security and Cryptology, Fujian Normal University, Fuzhou 350007, China; 2. Department of Computer Science and Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China. Correspondent: HUANG Tianqiang, E-mail: fjhtq@fjnu.edu.cn)

Abstract: A semi-supervised clustering for complicated data (SCCD) algorithm based on some pairwise constraints for multi-density and multi-shape data is proposed, which can deal with different densities complicated data and identify the cluster of arbitrary shapes. The parameter Eps of density-based algorithm for discovering clusters in large spatial database with noise (DBSCAN) is calculated automatically for each distribution of data. Then multi-stage DBSCAN is executed. Experimental results show that SCCD algorithm can find clusters of arbitrary shapes in multi-density data and outperforms the existed similar algorithm.

Key words: Semi-supervised clustering; Multi-density; DBSCAN; Pairwise constraints

1 引言

聚类是数据挖掘技术中应用最广泛与最活跃的技术之一. 它已经在生物学、地理学、遥感学、药学、人类学、市场营销与经济学等许多领域广泛应用. 现实世界中的事物纷繁复杂, 数据的产生与存在受到多种因素的影响, 同一个数据集中经常存在着多种密度的数据, 并且数据簇形状复杂. 已有的经典聚类算法通常考虑一种数据密度的情景, 例如在基于密度的聚类算法(DBSCAN)^[1]中, 隐含假定数据的簇密度是一致的. 由于决定密度阈值的参数 Eps(邻域半径)和 MinPts(邻域最少对象数)是全局唯一的, DBSCAN 只能发现密度近似的簇, 不能有效地处理多密度数据. 针对以上问题, 也有人提出了一些解决

方法, 如基于数据分区的密度聚类算法(PDBSCAN)^[2], 利用动态建模的层次聚类算法(Chameleon)^[3], 基于共享近邻度量的聚类算法(SNN)^[4]和基于网格的等密度线聚类算法(GDILC)^[5]等, 但它们在时间效率和处理噪声数据上都有待提高.

半监督聚类是一种新兴的机器学习方法, 它利用有限的信息(标签或者约束等)辅助聚类过程, 有效地提高了聚类质量. 其中基于约束的半监督聚类依靠用户提供的标号或约束来指导聚类过程, 可产生更合适的数据划分. 这包括基于约束修改目标函数, 或者基于标记对象初始化和约束聚类过程.

近年来, 许多学者先后研究了成对限制与概念

收稿日期: 2009201205; 修回日期: 2009205206.

基金项目: 福建省自然科学基金项目(2008J04004); 福建省青年人才创新基金项目(2006F3045).

作者简介: 黄添强(1971), 男, 福建莆田人, 副教授, 博士, 从事人工智能、数据挖掘的研究; 秦小麟(1953), 男, 南京人, 教授, 博士生导师, 从事安全数据库、空间数据库等研究.

聚类算法 (COBWEB)^[6], 全连接聚类算法 (Complete2link)^[7] 和期望最大化算法 (EM)^[8] 等算法的结合, 其中 Kiri Wagstaff^[9] 等将对限制融合到 K 均值 (K2Means) 算法, 使得在聚类过程中无条件地满足这些限制. 由于该方法对成对限制过于苛求, Sugato Basu^[10] 等提出基于成对限制的 K 均值半监督聚类算法 (PCK2Means). 该算法利用成对限制修改聚类目标函数, 使得聚类过程受到一定程度的约束, 约束的程度通过权重来设定, 权重越小约束的作用越小, 但其聚类过程需要依赖参数和大量的约束来达到好的结果. 针对该问题, Nizar Grira^[11] 等提出成对约束下的竞争凝聚的半监督聚类算法 (PCCA), 并将其运用在区分图像上. Davidson^[12] 等将对限制应用到层次聚类上, 同时提出了成对限制在聚类过程中的可行性和困难性问题^[13, 14]. Wei Tang^[15] 等阐述了一种成对限制引导的特征映射, 有效地对高维数据进行降维.

本文把半监督学习的方法融入经典的聚类算法 DBSCAN 中, 解决了聚类算法 DBSCAN 参数难以确定的问题, 并且可以在存在多种不同密度的数据集中发现任意形状的簇.

2 结构复杂数据的半监督聚类

在真实世界里的数据可能包含多种密度, 数据簇形状复杂, 因而传统算法遇到了困难. 本文根据成对约束在多密度数据集上的特性, 提出一种面向结构复杂的数据半监督聚类算法 (SCCD). SCCD 通过用户提供的 must2link 限制和 cannot2link 限制, 求得各个密度区域的数据分布密度, 计算各个密度区域上的 DBSCAN 算法的参数 Eps; 然后利用这些参数对数据集多次执行 DBSCAN 算法, 依次获得各个不同密度区域的聚类结果. SCCD 算法由 3 个部分组成: 1) 利用 must2link 初步获得参考 Eps 集合; 2) 在参考 Eps 集合上根据数据本身的特性, 同时利用 cannot2link 包含的信息进行筛选各密度级别上的代表 Eps; 3) 利用代表 Eps 执行多步 DBSCAN 聚类算法.

2.1 相关定义

定义 1 成对限制 must2link 约束 M : 对于 $(x_1, x_2) \in M$, 数据对象 x_1 和 x_2 具有相同的类标签; Cannot2link 约束 C : 对于 $(x_1, x_2) \in C$, 数据对象 x_1 和 x_2 具有不同的类标签.

定义 2 (对象的 Eps 邻域^[1]) 给定对象在其半径 Eps 区域内所包含的数据对象的集合.

定义 3 (密度相连) 如果存在对象 $o \in D$, 使得对象 p 和 q 都是从 o 关于 Eps 和 MinPts 的密度可达^[1], 则对象 p 到对象 q 是关于 Eps 和 MinPts 的密

度相连.

定义 4 (核心对象)^[1] 对象 $p \in D$, 如果对象 p 的 Eps 邻域内包含对象数目不小于给定的参数 MinPts, 则称对象 p 是核心对象.

定义 5 (核对象) 如果对象 p 是核心对象, 并且以 p 为起始点执行 DBSCAN 算法 (簇扩展函数 ExpandCluster^[1]) 所生成的簇的元素个数不小于给定参数 Den, 则称对象 p 为核对象.

定义 6 (边界对象) 如果对象 p 不是核心对象, 但可从其他核心对象直接密度可达^[1], 则称对象 p 为边界对象.

定义 7 (噪声对象) 对象 p 既不是核心对象, 也不是边界对象, 那么 p 是噪声对象.

定义 8 (第 k 最近邻距离) 对于一个数据对象 $p \in D$, p 点到它的第 k 个最近的邻居的距离为对象 p 的第 k 最近邻距离, 记为 p_kd .

定义 9 (k 邻居) 对于一个数据对象 $p \in D$, 它的第 k 个最近邻居的集合称为对象 p 的第 k 邻居, 记为 p_kn .

2.2 参数 Eps 初选算法

在多密度数据集中, 对象的第 k 最近邻距离反映了该对象邻近区域的密度分布, 第 k 最近邻距离越大, 密度越小, 数据分布稀疏; 第 k 最近邻距离越小, 密度越大, 数据分布稠密. 在存在不同密度的数据集中, 对于一个相同密度区域内的数据集 D_1 , $p, q \in D_1$, 则 p 和 q 的第 k 最近邻距离相差很小, 同时也反映了 D_1 数据区域的密度分布情况, 而对于不同密度区域的数据点之间的第 k 最近邻距离相差则很大.

对于 must2link 约束集合 M , 若 $(a, b) \in M$, 表示对象 a 和 b 处于同一个簇中, 根据上面的分析, 数据 a 和 b 的第 k 最近邻距离相差很小. 这里利用 M 集合中的元素确定不同密度区域的数据分布的参考半径 Eps, 具体见如下算法.

算法 1 RefEps (参数 Eps 初选算法)

输入: must2link 限制对 (p, q) , 数据集 D , 邻居数 k ;

输出: 一个参考邻域半径 Eps.

1) 计算数据 p, q 的 k 最近邻距离 p_kd 和 q_kd , 以及它们的 k 邻居集合 p_kn 和 q_kn .

2) $bp = \text{ExpandCPQ}(p, D, p_kd, k, p, q)$,

$bq = \text{ExpandCPQ}(q, D, q_kd, k, p, q)$;

3) If bq and bp

4) Then $Eps = (p_kd + q_kd)/2$; Return;

5) Else If bp

6) Then $Eps = p_kd$; Return;

- 7) Else If bq
- 8) Then Eps = q_kd ; Return;
- 9) Else
- 10) 邻居列表 link = $p_kn \ G \ q_kn$;
- 11) Repeat
- 12) 对于每个 point I link, 计算其 k 最近邻距离 pointkd;
- 13) 统计 ExpandCPQ(point, D, pointkd, k, p, q) 为真的数量 sum;
- 14) if sum > 0 Then Eps = 所有函数值为真的数据点的 k 最近邻距离 pointkd 的均值; Return;
- 15) Else
- 16) Update 邻居列表 link = 所有的 point 的 k 邻居总的集合(point I link)
- End if
- 17) Until 得到 Eps 的值
- 18) End If
- 19) End

注 1 函数 ExpandCPQ(Point, Dataset, Eps, MinPts, p, q) 是指在数据集 Dataset 上, 计算以数据 Point 为起点, 查询关于半径为 Eps 和数量参数为 MinPts 的邻域, 并不断向外扩张所形成的簇. 最后, 判断簇中是否包含数据 p, q, 包含则返回真, 否则返回假. 算法中 k 的值为 DBSCAN 算法中的 MinPts 参数. 文献[1]表明 MinPts 的值超过 4 以后对算法影响不大, 所以 RefEps 算法对 k 的取值是不敏感的.

图 1 中显示了 RefEps 算法初选参数 Eps 的一种情况: 图 1(a) 中 p 和 q 是 must2link 限制对; 图 1(b) 是以 p 为核心点, 以它的第 k(k=6) 最近距离 p_kd 为邻域半径向外延伸产生的簇 B(连线部分); 图 1(c) 是以 q 为核心点, 以它的第 k(k=6) 最近距离 q_kd 为邻域半径向外延伸产生的簇 C(连线部分). 显然簇 B 包含 p 和 q, 而簇 C 不同时包含 p 和 q. 于是选则 p_kd 作为一个候选的参数 Eps.

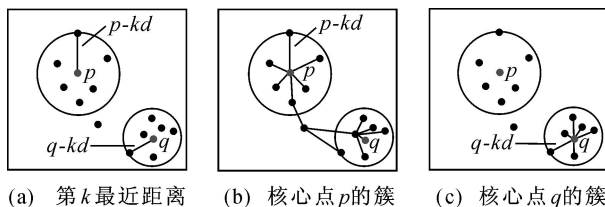


图 1 参数 Eps 的计算

多个数据密度分布的采样 must2link 限制数据对, 经由 RefEps 算法便可得到参考邻域半径 Eps 的集合 E.

2.3 参数 Eps 精选算法

在同一个密度区域可能存在多个 must2link 限制产生多个局部的参考半径 Eps, 这样同一个密度区域就存在多个参考半径 Eps. 现在对参考半径 Eps 集合 E 按密度区域进行划分, 即按大小进行划分, 以便对数据集中每一个密度分布选择最佳反映其密度分布的邻域半径 Eps. 划分的标准是不同密度区域的邻域半径差异大, 同一个密度区域的邻域半径差异小. 具体见如下算法.

算法 2 RepEps(参数 Eps 精选算法)

输入: 参考邻域半径 Eps 的集合 E, cannot2link 集合 C, 邻居数 k;

输出: 各个密度分布对应的邻域半径 Eps 的集合 Rep.

- 1) 对集合 E 进行升序排列; $Rep = \langle, D = \langle,$
- 2) bound = +] // 计算分割阈值 bound;
- 3) For each cannot2link(p, q) I C
- 4) 计算数据 p 和 q 的 k 最近邻距离 p_kd 和 q_kd ;
- 5) 计算 $d = |p_kd - q_kd|$;
- 6) $D = D \ G \ \{d_i\}$; // d_i 为 (d, p_kd, q_kd) 三元组
- 7) End For;

- 8) $e = \frac{1}{|D|} \sum_{i=0}^{|D|} d_i \# d$, // $d_i \in D$
- 9) For each $d_i \in D$
- 10) If $d_i \# d > e$ Then
- 11) 统计集合 E 中介于 $d_i.p_kd$ 与 $d_i.q_kd$ 之间的元素的个数, 设为 num
- 12) $Min = d_i.d / (num + 1)$
- 13) If $Min < bound$
- 14) Then bound = Min;
- 15) End If
- 16) End If
- 17) End For;
- 18) 设置初始检索起始点为 $s = 0$
- 19) Repeat
- 20) If $E[s+1] - E[s] < bound$ Then $s = s + 1$;
- 21) Else
- 22) 将 $E[s]$ 添加到集合 Rep; $s = s + 1$;
- 23) End If
- 24) Until $s >$ 集合 E 的大小
- 25) End

注 2 算法 2) 中步骤 2) ~ 15) 为计算分割阈值 bound 的过程. 阈值 bound 反映的是不同密度区

域之间的差异界限. 而 cannot2link 限制产生的 k 最近邻距离差异大, 反映了不同密度分布的差异, 可根据 cannot2link 的这种性质来获得阈值 bound. 值得注意的是, cannot2link 限制对 p 和 q 属于不同的簇, 但如果同处于一个密度区域, 则它们的 k 最近邻距离差异很小, 所以在步骤 10) 剔除这些限制对, 保留属于不同的簇, 同时处于不同密度区域的限制对产生分割阈值 bound.

2.4 多步 DBSCAN 聚类

在 2.3 中获得代表 Eps , 进行从小到大的排序. 设置初始数据集 D_s 为整个数据集 D . 从小到大依次选取 Eps , 对数据集 D_s 执行核对象定义基础上的 DBSCAN 算法, 获得在该邻域半径 Eps 所反映的密度级别上的簇, 并更新 D_s 为该次算法执行结果的噪声数据集, 直到遍历完所有的 Eps . 具体见如下算法.

算法 3 FinalClustering(多步聚类算法)

输入: 代表 Eps 集合 Rep , 数据集 D , 邻居数 k , 限定参数 Den ;

输出: 多密度数据集上的簇的集合 $Cluster$.

- 1) 聚类数据集 $D_s = D$, $Cluster = \langle \rangle$
- 2) 对 Rep 集合进行排序; $length = Rep$ 的大小;
- 3) For $i = 0$ to $length$
- 4) 将 $EXTDBSCAN(D_s, Rep[i], k, Den)$ 产生的簇添加到 $Cluster$ 集合;
- 5) $UPDATE D_s = EXTDBSCAN(D_s, Rep[i], k, Den)$ 产生的噪声数据;
- 6) End For;

注 3 函数 $EXTDBSCAN(D_s, Rep[i], k, Den)$ 中参数 k 是传统 DBSCAN 算法中的邻域最少对象数参数 $MinPts$, 它与 2.2 节和 2.3 节中 k 最近邻距离的数值 k 是相同的, $Rep[i]$ 为邻域半径参数 Eps . $EXTDBSCAN$ 算法与传统 DBSCAN 算法的区别在于采用定义 4 中的核对象代替原来的核心对象的定义来获得多密度数据集上相对完整的簇, 其中主要用参数 Den 限制所产生的簇中元素的数量. 在每次执行 DBSCAN 算法时, 检测产生的簇的大小, 保留簇元素数量超过 Den 的簇, 将小于 Den 的那些簇看成是噪声数据并重新参与下一次聚类. 在大的密度区域中可能存在少量十分密集的部分数据, 参数 Den 主要是为了防止用小的邻域半径 Eps 进行聚类时在大的数据密度分布中产生微小的簇的情况, 以得到各个密度分布上较完整的簇.

2.5 SCCD 算法性能分析

本文提出的形态复杂的多密度数据的半监督聚类算法 SCCD 在 2.2 节和 2.3 节中选取各密度级别

上的 Eps 的时间复杂性为 $(2PN)$, 在 2.4 节聚类阶段的时间复杂性为 $O(N^2)$, SCCD 算法总的时间复杂性 $O(2PN + N^2) = O(N^2)$. 其中 P 为成对限制的数目, N 为多密度数据集的个数, $P \leq N$. SCCD 算法的时间复杂性与 DBSCAN 算法的时间复杂性相同, 但 DBSCAN 算法在计算时间复杂性时, 并没有将确定参数 Eps 的第 k 最近邻距离图 ($k2dist$ 图) 的绘制时间算进去. 而绘制 $k2dist$ 图需花费大量的时间, 其中计算数据对象间的距离 $O(N^2)$ 和计算数据对象的 k 最近距离 $O(KN^2)$, 而且 DBSCAN 必须通过人工确定最终的阈值 Eps . 本文提出的算法除了不必人工选择 Eps 以外, 还大大降低了 Eps 选取的时间复杂性. 如果采用 R^*2tree 或者 $kd2tree$ 作为数据集的存储结构, 时间复杂性为 $O(N \log N)$.

3 实验结果及分析

实验环境为: CPU AMD sempron 2200 + 1.5Hz, 256 内存, windows xp 操作系统. 算法编程语言为 java, 平台环境 JDK1.5+ Eclipse.

由于空间的限制, 为检验 SCCD 算法的有效性, 实验选取 4 个不同的多密度数据和均匀数据的二维仿真数据集进行实验, 其中包括相关文献的类似的数据集^[1,4]. 成对限制由随机程序抽取产生.

数据集 1 是人工产生的数据集, 包含 3 个密度区域和 7 个簇. SCCD 在给定的少量密度区域限制时, 能够发现包含在其他簇数据中的两个圆形的簇, 以及距离接近的另外 3 个复杂形状的簇, 表明 SCCD 可以很好地区分不同密度区域的簇和同一密度区域的簇, 如图 2 所示. 实验数据: 数据集数目为 1182; Eps 为 4.08, 8.88, 20.45; bound 为 4.15; Den 为 12; $MinPts$ (即 k) 为 6; 成对限制的数目为 20 (其中 must2link 数 12).

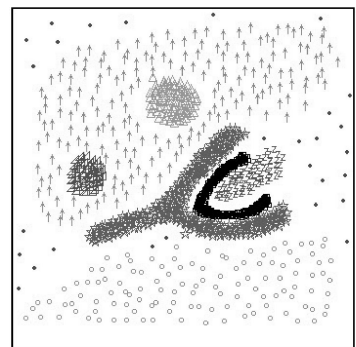


图 2 数据集 1 的聚类效果

数据 2 是模拟文献[4]提供的数据集. 文献[16]中的实验显示, SNN 算法对类似的数据集的聚类不能有效地处理噪声数据和离群点数据. 而 SCCD 采用 DBSCAN 算法作为聚类的一部分, 能有效地处

理噪声数据和离群点数据,并且能发现形态复杂的簇,如图3所示.实验数据:数据集数目为3135, Eps为8.09, bound为0.74, Den为10, MinPts(即k)为7,成对限制的数目为16(其中 must2link 数10).

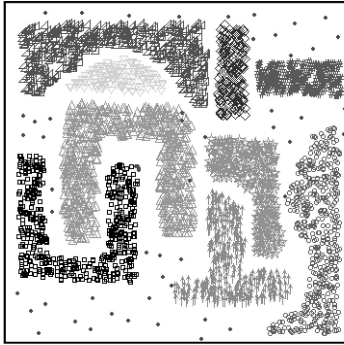


图3 数据集2的聚类效果

数据集3是模拟文献[1]的数据集合,包含2个密度区域和5个簇. SCCD在给定的少量密度区域限制时,在存在噪声的情况下,可以发现任意形状的多密度的簇,如图4所示. SCCD可有效地发现/S0形的簇和它所包围的两个紧密的簇,以及与它相连和相邻的两个复杂形状的簇.实验数据:数据集数目为945; Eps为6.43, 13.45; bound为1.96; Den为10; MinPts(即k)为6;成对限制的数目为7(其中 must2link 数4).

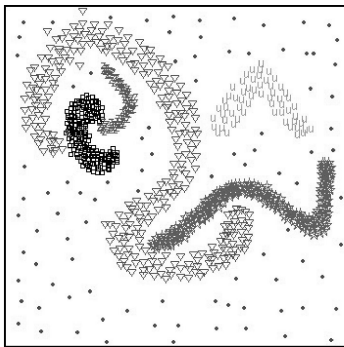


图4 数据集3的聚类效果

数据集4是人工产生的数据集合,包含2个密度区域和6个簇. SCCD在给定的少量密度区域限

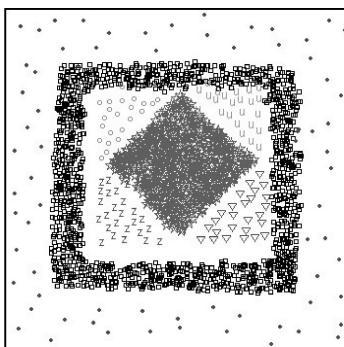


图5 数据集4的聚类效果

制,并存在噪声的情况下,可以发现正方形簇所包围的5个簇,如图5所示.实验数据:数据集数目为2155; Eps为7.08, 25.74; bound为3.99; Den为6; MinPts(即k)为6;成对限制的数目为8(其中 must2link 数5).

4 结论

为了有效地处理多密度且任意形状的数据,本文把半监督学习的方法融入经典的聚类算法DBSCAN,提出一种半监督聚类算法SCCD.解决了聚类算法DBSCAN参数难以确定的问题,并且可在存有多种不同密度的数据集中发现任意形状的簇.实验结果表明了算法的有效性.

参考文献(References)

- [1] Ester M, Kriegel H P, Sander J, et al. A density-based algorithm for discovering clusters in large spatial database with noise [C]. Proc of 2nd Int Conf on Knowledge Discovery in Databases and Data Mining. Portland: AAAI Press, 1996: 226-231.
- [2] 周水庚, 周傲英, 曹晶. 基于数据分区的DBSCAN算法[J]. 计算机研究与发展, 2000, 37(10): 1153-1159. (Zhou S G, Zhou A Y, Cao J. A data-partitioning-based DBSCAN algorithm[J]. Computer Research and Development, 2000, 37(10): 1153-1159.)
- [3] Karayipis G, Han E H, Kumar V. Chameleon: A hierarchical clustering algorithm using dynamic modeling [J]. IEEE Computer, 1999, 32(8): 682-75.
- [4] Ertoz L, Steinbach M, Kumar V. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data [C]. Proc of the 3th SIAM Int Conf on Data Mining. San Francisco, 2003: 472-58.
- [5] 赵艳厂, 宋梅, 宋俊德, 等. 用于不同密度聚类的多阶段等密度线算法[J]. 北京邮电大学学报, 2003, 26(2): 42-47. (Zhao Y C, Song M, Song J D, et al. Clustering datasets containing clusters of various densities[J]. J of Beijing University of Posts and Telecommunications, 2003, 26(2): 42-47.)
- [6] Wagstaff K, Cardie C. Clustering with instance-level constraints [C]. Proc of the 17th Int Conf on Machine Learning. Palo Alto: Morgan Kaufmann Publishers, 2000: 1103-1110.
- [7] Klein D, Kamvar S, Manning C. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering [C]. Proc of the 19th Int Conf on Machine Learning. Sydney: Morgan Kaufmann Publishers, 2002: 307-314.
- [8] Law M, Tophy A, Jan A. Clustering with soft and group constraints [C]. Proc of the Joint IAPR Int Workshop on Structural, Syntactic, and Statistical

- Pattern Recognition. Berlin: Springer-Verlag, 2004: 662-670.
- [9] Wagstaff K, Cardie C, Rogers S, et al. Constrained K2 means clustering with background knowledge [C]. Proc of 18th Int Conf on Machine Learning. Williamstown: Morgan Kaufmann Publishers, 2001: 572-584.
- [10] Basu S, Banerjee A, Mooney R. Active semi-supervised learning for pairwise constrained clustering [C]. Proc of the SIAM Int Conf on Data Mining. Cambridge: MIT Press, 2004: 332-344.
- [11] Grira N, Crucianu, Boujemaa N. Semi-supervised fuzzy clustering with pairwise constrained competitive agglomeration [C]. IEEE Int Conf on Fuzzy Systems. Reno: IEEE Press, 2005: 222-225.
- [12] Davidson I, Ravi S S. Agglomerative hierarchical clustering with constraints: Theoretical and empirical results [C]. Proc of the 9th European Conf on Principles and Practice of Knowledge Discovery in Databases. Porto: Springer, 2005: 52-70.
- [13] Davidson I, Ravi S S. Clustering with constraints: Feasibility issues and the K2means algorithm [C]. Proc of the 5th SIAM Int Conf on Data Mining. Newport Beach, 2005: 132-149.
- [14] Davidson I, Ravi S S. Intractability and clustering with constraints [C]. Proc of 24th Int Conf on Machine Learning. New York: ACM Press, 2007: 201-208.
- [15] Wei T, Hui X, Shi Z, et al. Enhancing semi-supervised clustering: A feature projection perspective [C]. Proc of the 13th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM Press, 2007: 702-716.
- [16] 张西芝, 姬波, 邱保志. 基于网格的多密度聚类算法 [J]. 微计算机信息(管控一体化), 2005, 21(36): 102-103.
(Zhang X Z, Ji B, Qiu B Z. Grid-based clustering algorithm for multi-density [J]. Control & Automation, 2005, 21(36): 102-103.)

(上接第 7 页)

- [30] Xu L, Oja E. Randomized hough transform (RHT): Basic mechanisms, algorithms and computational complexities [J]. Computer Vision Graphics and Image Processing, 1993, 57(2): 131-154.
- [31] Li Q, Zheng N, Cheng H. Lane boundary detection using an adaptive randomized hough transform [C]. Proc of the 5th World Congress on Intelligent Control and Automation. Hangzhou, 2004: 4084-4088.
- [32] Li Q, Zheng N, Cheng H. Springrobot: A prototype autonomous vehicle and its algorithms for lane detection [J]. IEEE Trans on Intelligent Transportation Systems, 2004, 5(4): 302-308.
- [33] Yu B, Jain A. Lane boundary detection using a multi-resolution hough transform [C]. Int Conf on Image Proc. Washington D C, 1997: 742-751.
- [34] Wang Y, Teoh E K, Shen D. Lane detection using B-snake [C]. Proc of Information Intelligent Systems. Bethesda, 1999: 432-443.
- [35] Wang Y, Teoh E K, Shen D. Lane detection and tracking using B-snake [J]. Image and Vision Computing, 2004, 22(4): 262-280.
- [36] Arulampalam S, Maskell S, Gordon N, et al. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking [J]. IEEE Trans on Signal Processing, 2001, 50(2): 174-188.
- [37] Doucet A, Godsill S, Andrien C. On sequential monte carlo sampling methods for Bayesian filtering [J]. Statistic and Computing, 2000, 10(3): 197-208.
- [38] Gordon N J, Salmond D J, Smith A F M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation [C]. IEE Proc F, 1993, 140(2): 107-113.
- [39] Kim Z W. Robust lane detection and tracking in challenging scenarios [J]. IEEE Trans on Intelligent Transportation Systems, 2008, 9(1): 12-26.
- [40] Crisman J D, Thorpe C E. SCARF: A color vision system that tracks roads and intersections [J]. IEEE Trans on Robotics and Automation, 1993, 9(1): 49-58.
- [41] Bahlmann C, Zhu Y, Visvanathan Ramesh Pellkofer M, et al. A system for traffic sign detection tracking and recognition using color shape and motion information [C]. Proc of the IEEE Intelligent Vehicles Symposium. Las Vegas, 2005: 252-260.
- [42] Tipping M E. Sparse Bayesian learning and the relevance vector machine [J]. J of Machine Learning Research, 2001, 1(1): 21-244.
- [43] Lawrence N, Seeger M, Herbrich R. Fast sparse Gaussian process methods: The informative vector machine [C]. Advances in Neural Information Processing Systems. Cambridge, 2003: 609-616.