孤立点检测算法及其在数据流挖掘中的可用性*)

孙 云[1] 李舟军[2] 陈火旺[1]

(国防科技大学计算机学院 长沙 410073)[1] (北京航空航天大学计算机学院 北京 100083)[2]

摘 要 孤立点(也称为噪声、异常点等)是那些不符合数据一般模型的数据,它们与数据集的其他部分不同或不一致。检测孤立点的主要目的是为了从数据集中找出那些不正常的观察结果。随着现实世界和工程实践中不断产生大量的数据流,在数据流上有效检测孤立点越来越引起国内外研究者的广泛关注。在系统地分析了目前国内外孤立点检测相关文献的基础上,本文对孤立点检测算法进行了较为全面的阐述,并就这些算法是否可以用于数据流上孤立点检测进行了深入探讨和研究,同时指出了这些算法存在的主要问题以及未来的研究方向。

关键词 孤立点,孤立点检测,数据流

Outlier Detection Algorithms and their Availability to Data Streams Mining

SUN Yun[1] LI Zhou-Jun[2] CHEN Huo-Wang[1]

(School of Computer, National University of Defence Technology, Changsha 410073)[1]

(School of Computer Science & Engineering, Beihang University, Beijing 100083)[2]

Abstract An outlier (also referred to as noise, novelty, anomaly, deviation, exception) is one that appears to deviate markedly from other members of the dataset in which it occurs. The purpose of outlier detection is to find the anomalous observation from datasets. With more and more massive data streams were generated in real-world application, the problem of efficient outlier detection in data streams has received considerable attention in research of information science. In this paper, we summarized the literature of outlier detection, dissertated the algorithms of outlier detection comprehensively, and discussed if these algorithms could be applied to outlier detection of data streams. Finally, we presented the limitation of these algorithm and their future research topics.

Keywords Outliers, Outliers detection, Data streams

1 引言

在统计学和数据库理论中对孤立点的研究由来已久。近年来,随着数据挖掘技术的飞速发展,孤立点检测又成为数据挖掘技术中一个重要的组成部分。虽然孤立点不符合数据的一般规律,与数据的其他部分不一致,是数据集中远远偏离其他对象的那些小比例对象,但在一些应用中,对孤立点的检测却能为我们提供重要的信息。John 在文[17]中说明了这样一个事实:孤立点可能是一个令人惊奇的数据的真实写照。也就是说,一个数据点除了确实属于类 A 外,也可能属于类 B,而这种真实性对一个观察者来说可能会有意想不到的收获。Aggarwal 和 Yu[6]还注意到,孤立点可能的确是某个聚类集合的"噪声",但也有可能是这些聚类集合以外的非噪声的数据点。这些孤立点与其他正常点的行为有着明显的不同。本文以 Hawkins[10]给出的定义为标准考虑孤立点,并不分别考虑有双重身份的点,或区别于噪声的孤立点。

及时准确地发现孤立点有着广泛的应用前景。在那些对安全性有相当高要求的系统环境中,孤立点检测是一项重要任务。孤立点在这些环境中暗示不正常状况的出现,这种不正常往往预示着环境或性能的重大退化或危险的出现。如飞机性能统计数据中的一个孤立点,可能表示飞机发动机的一个设计缺陷;地理图像上的一个孤立点可能标志着一个危险对象(如一颗地雷);另外,系统中的一个孤立点还可能是对某个恶意入侵的精确定位。因此,迅速准确地检测孤立点是非常必要的。孤立点检测可以是某工厂通过对其产品的细节特征进行连续监测,并将得到的实时数据与产品的正常或异常数据进行比较,发现生产线中存在的缺陷的过程;孤立点检测也可以是通过对信用卡或移动电话使用方式的监视,来发现使用模式的突然变化的过程,这种变化往往暗示着盗用信用卡或盗用通话时间的欺诈行为。除了上述用途之外,孤立点检测还可以用于各种审批处理,如在处理贷款申请或社会保险支付时,孤立点检测可用于监视基金申请人的行为,确保资金支付不会被骗。股票或商品的投资者也可以通过孤立点检测的方式来监视个体配额或市场变化,以便发现新的买或卖的投资方向。一个新闻投递系统可以通过检测新闻事件情节的改变来确信新闻提供者身份的正确性[32]。

孤立点产生的原因是多方面的,概括起来,大致可以归为两类:第一,由错误产生的孤立点,如测量仪器的缺陷造成的度量错误、网络黑客的入侵和机器故障的出现可造成系统的行为异常。第二,由数据变异产生的孤立点,这是数据分布真实性的反映。从数据库的一致性和完整性的角度来看任何方面的问题都是必须考虑的。

2 基本概念

2.1 孤立点

可以用于孤立点检测的技术非常多,但当前许多已经用

*)本文工作受到国家自然科学基金项目(60573057,60473057,90604007)的资助。孙 云 博士研究生,研究方向为数据挖掘技术;李舟军 博士,教授,博士生导师,研究方向为进程代数理论、数据挖掘技术、安全协议形式化验证;陈火旺 院士,主要研究方向为软件理论与软件工程。

于检测孤立点的技术，除了其定义的形式与命名方式不同之外，其基本的思想却都是一致的。另外，对于什么是“孤立点”，虽然目前已有许多不同形式的定义，但至今还没有一个能被人们普遍接受的统一方式。迄今为止，最具有代表性的是 Hawkins 给出的一个比较直观的定义：孤立点的表现是如此与众不同，不禁让人怀疑它们是由另外一种完全不同的机制所产生的[10]。如图 1[32] 中标记为 V,W,X,Y,Z 的五个点相对于正常数据点有明显的不一致。类似这样的点就是孤立点。

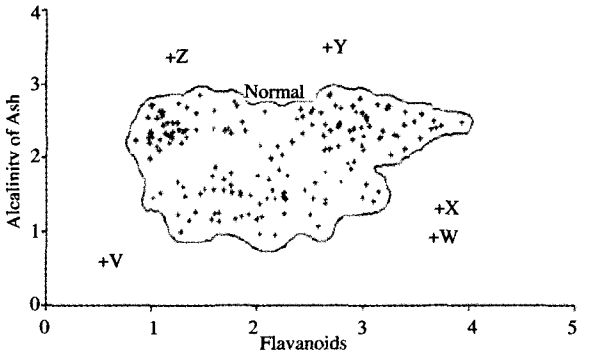

图 1　带有孤立点的数据分布图

除 Hawkins 给出的定义外，许多研究者根据特定的研究背景，给出了孤立点的不同的定义[1,10,11,15,27,35]。尽管它们不尽相同，但都反映了孤立点的特点：孤立点看起来令人惊奇；孤立点是一个相对定义，如果初始分布模型的假设不同，会产生不同的结论；孤立点有较强的主观性，几乎所有研究者在进行孤立点检测研究时都定义特有的检测背景。

所谓孤立点检测，就是从大量数据中提取隐含在其中的、人们事先不知道的但又是潜在有用的信息和知识的过程[20]。孤立点检测可以形式化地描述为：

定义 1[20]　给定一个有 n 个数据点或对象的集合及预期的孤立点数目 k，发现与剩余的数据相比是显著异常的、孤立的或不一致的前 k 个对象的过程。因此，孤立点检测实际上可以被看作两个子问题：

(1) 在给定的数据集合中定义什么样的数据是不一致的；

(2) 找到一个有效的方法来检测这样的不一致数据。

2.2　数据流

数据流模型是伴随大规模数据集应用而产生的，例如，用户点击、电话记录、大的网页集合、多媒体数据、财务往来、观测的科学数据等，这些数据基本上都是通过数据流来建模的。还有某些数据集，例如路由器数据包统计、气象数据、传感器网络数据等，这些都是时变数据，不适合在磁盘上进行处理，必须在其生成时就进行处理，也适合于通过数据流进行处理。为了处理这些情况，人们开始了对数据流的研究。

定义 2[33]　设 $A=(A_1,A_2,\cdots,A_k)$ 为属性集合，相应的 k 维数据空间 $S=(A_2\times A_2\times\cdots\times A_k)$，数据流(Data Stream)可以看成一个 k 元关系 $R=\{\vec{r}_t|\vec{r}_t\in S,t=1,2,\cdots\}$，其中元组 $\vec{r}_t$ $(t=1,2,\cdots)$ 连续到达，t 是数据 $\vec{r}_t$ 的时间戳。

数据流一般具有以下特性[38]：

(1) 数据实时到达；

(2) 数据到达次序独立，不受应用系统所控制；

(3) 数据规模宏大且不能预知其最大值；

(4) 数据一经处理，除非特意保存，否则不能被再次取出处理，或者再次提取数据代价昂贵。

数据流的这些特征，使其成为近几年数据挖掘技术的研究热点之一。当然，作为数据挖掘重要组成部分的孤立点检测在数据流挖掘中同样是必不可少的。

3　面向数据流分析孤立点检测算法

现有的孤立点检测方法绝大多数都是针对静态数据集给出的，真正的数据流上检测孤立点的方法非常少。造成这一现象不是偶然，这是由于数据流的特征对数据流挖掘提出了相当强的限制：(1)必须对每个数据快速处理以适用数据流的流速，(2)在流速限制下，只能通过有限的存储空间来对数据进行处理，(3)挖掘算法必须能够适应流的变化趋势，(4)某些时候，挖掘必须能够(几乎)实时支持决策系统，这是因为大量源于监测系统的软件要求能够提供实时报警、运算和管理。

本文从静态数据集上的孤立点检测算法入手，通过对比数据流的特征，分析这些算法在数据流上进行孤立点检测的可用性。根据对孤立点的不同定义，现有的孤立点检测方法大致可以分为 6 种类型：基于统计的方法[31]、基于深度的方法[30]、基于距离的方法[11~14]、基于聚类的方法[3]、基于密度的方法[27,34]和基于关联的方法[37,1,22]。

3.1　基于统计的算法

基于统计模型的方法是最早应用于孤立点检测的方法。人们在使用各种早期的统计方法进行孤立点检测之前，常常要事先假设给定的数据集合服从某种确定的分布或概率模型[2,4,5,8,24,25](如正态分布、泊松分布等)，并通过一致性检验将偏离这些模型的对象看作是孤立点[31]。一致性检验要求已知数据集的模型(如假设的数据分布)、分布参数(如均值、方差或协方差)以及预期的孤立点数目。在早期的这类方法中，公认的最健壮、最高效的孤立点检测工具是 Hampel identifier[21,23]。

为了摆脱需要预知分布模型的限制，研究人员从均匀样本的数据之间存在自相关性入手，针对自相关性来研究孤立点检测技术，并取得了可喜的成果。1982 年，Martin 和 Thomson[26] 通过对基于参数估计的自回归(AR)模型 Kalman filter[9] 进行改进，给出了一种不需要事先假设数据分布模型的数据清理工具 MT filter-cleaner。该工具首先利用从现在开始到过去的某段时间内已有的数据，进行回归分析并确定其模型参数，然后利用该模型对下一个数据点进行检测，判断是否为孤立点。

2004 年，Liu Hancong 等人[18] 对 MT filter- cleaner 的健壮性进行了进一步的加强，给出了一种改进的 MT filter-cleaner。这个改进的 MT filter- cleaner 算法不仅不需要准确的模型知识，而且能很好地用于时间序列数据流上，捕捉在线数据的动态信息，并以此建模。该算法很好地降低了噪声对估计模型的影响，使得模型的健壮性得到进一步提高，能够很好地适应时间序列数据流并准确地捕捉流上的孤立点。

虽然该算法在不事先假设分布模型的统计方法研究方面取得了可喜的成果，但这些都只是集中在对均匀的单维数据(如时间序列)的研究之上。而事实上，绝大多数的数据集都是高维的或不均匀的。对于这些情况，在没有事先假设的情况下，想要找出准确的处理模型并准确地检测孤立点，是一件非常困难的事。

由于数据流对算法的速度要求非常严格，为了避免参数

估计带来的时间复杂性，1999 年，Jagadis[19] 等人通过对直方图进行动态重组，给出了一种无参数、应用于时间序列的孤立点检测新方法。该方法在保持直方图中 bucket 的数量 b 与孤立点数量 k 之和不变(有限存储)的条件下，利用统计量(方差)进行动态规划，在 bucket 数量与孤立点数量之间找到一个最好的折中，从而得到全局“最优”的孤立点。也就是说，当从初始的 $b+k$ 个 bucket 中选出 k 个孤立点单独存储，并将原来的 bucket 重新组合成 b 个 bucket 后，使得新得到的所有 bucket 方差之和取得最小值。该算法在检测时间序列孤立点方面非常有效，但却不能应用于大规模的数据流。

Muthukrishnan 等人[28] 对 Jagadis 的算法进行了研究，发现该算法找到的并不是最优孤立点，而且由它所产生的孤立点的数量非常庞大，且只能用于时间序列数据流。针对这些问题，Muthukrishnan 等对文[19]中的算法进行了修改，给出了第一个能够找到最优孤立点的算法，且证明了这些孤立点的最优性。同时，他们还给出了第一个能够用于大规模数据流(包括单维和多维)的近似系数为 $1+\varepsilon$ 的孤立点检测近似算法，也证明了算法的近似最优性(其中的 ε 是用户根据自己的不同需要而设置的最大容忍程度)。该算法使用 $O((k^3/\varepsilon)\log n)$ 的空间来存储数据流的一个概要数据结构(每个 bucket 的位置、其中所有数据之和、平方和，k 个升序排列的最大值，k 个降序排列的最小值)，使用 $O((k^4/\varepsilon)\log n)$ 时间来处理每一项，当用户提出查询请求时，算法可以在 $O((k^3/\varepsilon)\log n)$ 时间内输出孤立点。

虽然利用直方图进行孤立点检测十分有效，但它们检测到的孤立点都是全局孤立点。而在数据流上，由于数据随时间的变化而产生概念漂移现象，使得全局孤立点对实际应用的价值并不大。因此，如何使直方图这种有效的孤立点检测方法能够检测局部孤立点，并能适应概念漂移的影响，以提高其应用价值，是一个值得深入探讨和研究的重要课题。

3.2 基于距离的算法

为了解决基于统计的方法必须假设数据集符合特定分布模型的局限性，Knorr 和 Ng[11~14] 提出了一种只依赖于数据之间距离的孤立点定义：在数据库 T 中，若有超过 $p\%$ 的对象与某对象 O 的距离大于 D，则称该对象 O 为一个基于距离的孤立点。这个定义可形式化地表示如下：

定义 3[12] 如果 $|\{O_i|d(O_i,O)\geqslant D\}/|T|\geqslant p\%$，则称对象 O 是一个孤立点，记为 $DB(p,D)$- Outlier。其中 $|T|$ 表示数据库 T 中对象的数量，$d(O_i,O)$ 表示对象 O_i 与对象 O 之间的距离。

根据上述定义，主要有三种基于距离的孤立点检测算法[12]：复杂度为 $O(c^k+N)$ 的 cell-based 算法复杂度为 $O(k*N^2)$ 的 Nested-Loop(NL)算法和复杂度为 $O(k*N^2)$ 的 index-based 算法。

Ramaswamy 等人[29] 使用 k-最近邻的孤立点定义对基于距离的算法进行了进一步的扩展，并将 top-k 数据对象作为孤立点进行输出。他们还提供了一种 partition-based 算法，该算法使用一个聚类方法来划分数据库，然后分别在每一个划分中检测孤立点，此算法可以给出每个点的孤立度，但是这依赖于算法中的聚类过程，因此该算法的效率受到了影响。Tao Yufei[34] 给出了一种新的基于距离的孤立点检测方法，该方法能够以最小的 I/O 消耗代价，找出满足三角不等式的任何度量空间下大型数据集中的所有孤立点，而且最多只需扫描两次数据库，但它不能对找到的孤立点给出这个孤立点近似的偏离程度。

然而，对每个数据对象来说，所有的这些基于距离的算法都需要搜索整个数据集来确定它的邻域，因此需要一遍以上的数据集扫描次数，不能适应数据流的单遍扫描要求。同时，这些算法找到的只是全局孤立点，且只能发现球形区域的孤立点，不适用于数据分布形状不均匀的数据集。由于这些自身的原因，这类算法虽然在静态数据集上非常有效，但它们并不适用于数据流。因此就笔者所能了解的知识范围，还没有发现基于距离的数据流孤立点检测算法。

3.3 基于密度的算法

为了弥补基于距离方法的不足，Breuning 等人受基于密度聚类思想的启发，于 2000 年提出了一种基于局部密度的检测孤立点新方法[35]。在该方法中，对于每个数据对象，只以其周围小范围内的数据对象作为参考，来计算它的局部孤立点因子 LOF(Local Outlier Factor)。通过该对象周围区域的局部密度，与它邻居的局部密度之比来确定该对象的 LOF，LOF 的值越大说明该对象越可能是孤立点。这一方法的提出打破了孤立点二值(是或否)定义的传统，开创了研究局部孤立点的先河。

Papadimitriou 等[27] 通过对文[35]中算法的研究，发现在使用 LOF 进行孤立点检测时，对标志着局部范围的 MinPts 值存在选择上的困难。为了克服这一缺陷，他们定义了一个新的评价孤立点偏离程度的标准——多粒度偏差因子 $MDEF$(Multi- Granularity Deviation Factor)：

$$MDEF(p,r,\alpha)=1-\frac{n(p,\alpha r)}{\hat{n}(p,\alpha,r)}$$

其中 $n(p,\alpha r)$ 表示对象 p 的 αr 邻域内的邻居数量，$\hat{n}(p,\alpha,r)$ 表示对象 p 的 r 邻域内所有邻居的 αr 邻域内邻居的平均数量，α 是为了能检测到图 2 左边所示的局部孤立点而加入的粒度因素。根据这个定义作者给出了一个准确检测孤立点的算法 LOCI(LOcal Correlation Integral)，同时给出了一个近似算法 aLOCI，以牺牲适当精度为代价，大大提高了算法速度。LOCI 算法使用派生自数据本身的统计值，解决了 LOF 算法中对 MinPts 的选择难题。

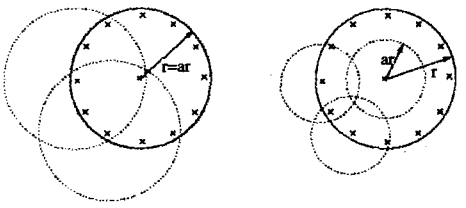

图 2 左边是 $\alpha=1$ 的情况，右边是 $\alpha<1$ 的情况

虽然这些算法都有相当高的精确度，但其时间复杂度却接近 $O(n^2)$，不适合直接用于数据流。通过综合它们源自密度聚类的方法特征，很多研究者将聚类思想与孤立点检测巧妙地结合在一起，在数据流上通过类似聚类的方式，快速过滤掉稠密区域的数据(这样做的理由是基于局部密度的孤立点定义只需考虑其某个邻域范围内的数据，而不用对整个数据集中的数据进行考察；而且数据具有天生的聚类性)，在稀疏区域找出可能成为孤立点的候选集合，然后利用聚类特有的统计信息对孤立点的偏离程度进行估计，以便用户进行查询。如文[7]和[33]，就是分别在静态数据库和数据流上利用网格聚类的思想，通过对网格的划分有效地过滤处于稠密区域的大量数据主体，只对稀疏区域的数据采用近似方法计算出它

们的偏离程度，这样既大大减少了算法所需考察数据的规模，又有效地捕捉到了数据流上的概念漂移。可以肯定地说，这种方法对维度较低的数据流模型非常有效，但对高维数据与聚类一样存在指数爆炸的存储问题。文[36]中，首先对数据流上的微聚类按照其密度进行排序，然后利用给定的阈值将序列末尾的几个稀疏微聚类作为孤立点候选集，并估计出候选集中数据的偏离程度。文[16]的主要目的在于进化数据流上的聚类挖掘，它对噪声的处理只是找到并清除，而并没有给出孤立度，但该文对噪声的查找方式对于数据流上的孤立点检测来说也是值得一提的。

3.4 基于关联的算法

前面叙述的各种各样的孤立点检测算法都只能适用于连续属性数据集，而不适用于离散属性数据集，这是因为很难对离散属性数据进行求和、求距离等数字运算。

2004 年，He Zengyou[37] 等人提出了一种通过发现频繁项集来检测孤立点的新方法。其思想非常简单：由关联规则算法发现的频繁模式反映了数据集中的“普遍模式”，这也就使人们相信，不包含频繁模式或只包含极少频繁模式的数据其实就是孤立点。也就是说，频繁模式不会包含在作为孤立点的数据中。He 定义了一种利用频繁模式度量孤立点偏差程度的频繁模式孤立点因子 *FPOF*(Frequent Pattern Outlier Factor)：

$$FPOF(t)=\frac{\sum_{X\subseteq t}\text{support}(X)}{|FPS(D,\text{mimisupport})|}$$

其中 t 表示数据集 D 中的一个对象(事务)，$FPS(D,\text{mimisupport})$表示 D 中满足最小支持度的频繁模式集，并通过对频繁项集的挖掘和比较给出了检测孤立点的新算法 FindFPOF，该算适用于离散属性数据集。

但当前真正有实用价值的数据集大多具有混合属性，因此只能对单一属性数据集进行检测的算法显然无法满足这一现状。于是 Amol Ghoting 等人[1,22] 为检测这类数据集中的孤立点提供了第一个检测算法 LOADED，该算法分为两部分，第一部分是对整个数据集的离散属性在给定最小支持度的条件下进行频繁模式的挖掘；第二部分是将每个对象 P 中的所有可能模式分别与前面中得到的频繁模式集合 D 进行比较，如果项 $d\subseteq P$ 且 $d\notin D$，则为对象 P 打一个分值 $1/|d|$；如果 $d\in D$，则考虑对象 P 的连续属性之间的关联程度 $V(P)=\sum_i\sum_j v(P_{ij})$，其中 $v(P_{ij})$表示对象 P 的第 i 个连续属性与第 j 个连续属性值之间的关联系数是否偏离了这两个属性之间的平均关联系数，如果是则将其值设为 1，如果不是其值为 0，这样在 $V(P)$的值超过某个容忍阈值 δ 时，就为对象 P 打一个分值 $1/|d|$，对 P 中所有可能模式得到的分值的累加就是该对象可能成为孤立点的分值。同时，文[22]中还通过每隔一段时间就将保存在 Hash 表中的频繁模式的频率降低一定数量的时间偏斜方式，将算法应用于数据流模型。但该算法存在两个明显的缺陷：第一，当离散属性的数量较小(例如是 1)时，算法的准确度明显降低；第二，因为频繁模式的挖掘是非常耗时的工作，频繁模式的保存又需要消耗大量的存储空间，因此，该算法很难适应快速的数据流。

3.5 其它算法

除了我们上面提到的孤立点检测算法之外，还有许多其它的检测算法。如文[15]给出了一种新的孤立点定义和检测方法，该方法使用 Pawlak 的粗糙集理论，首先通过对使用任意数据集 X 使用一个等价关系族 $R=\{r_1,r_2,\cdots,r_m\}$，来定义 X 的一个内边界(inner boundary)族 $B=\{B_1,B_2,\cdots,B_m\}$，然后使用这个内边界族定义了 X 的异常集合 e，$e\in X$ 且 $e\cap B_i\neq\Phi$，e 可以是 $\bigcap_{i=1}^{m}B_i$ 的子集，也可以不是，并将所有这类集合中没有冗余对象的集合作为极小异常集(Minimal Exceptional Set) f；然后将包含对象 $x\in X$ 的 B_i 的个数作为 x 的边界度 $Degree\_B(x)$，并用边界度定义了对象的异常度 $ED\_Object(x)=Degree\_B(x)/m$ 和集合的异常度 $ED\_Set(Y)=\sum_{y\in Y}ED\_Object(y)/|Y|$；最后通过这两个定义对极小异常集中的对象进行计算，当结果大于某个给定的阈值时说明对象是一个孤立点，否则不是。当然也可以计算整个极小异常集的孤立程度。

该文只是从理论的角度提出了使用粗糙集定义孤立点和孤立点检测方法，还未对其进行过充分的实验检验，也没有给出其算法复杂性的分析结果。但由于粗糙集理论本身处理模糊、不精确或不完全信息的能力，因此为其将来应用于数据流提供了良好的理论基础，在实际应用时需要充分考虑最小异常集合的概念漂移问题。笔者相信，该方向具有良好的研究价值。但还是由于粗糙集理论本身的限制，该算法仅能适用于离散属性数据集，对于连续属性数据集需要进行离散化处理。

结束语 对孤立点检测的研究经过近一个世纪的起伏，随着现代数据挖掘技术的崛起，以及其自身应用价值的显露，又一次迎来了它的新生命热点，即在数据流模型上检测孤立点。本文在系统地分析和研究了目前国内外孤立点检测相关文献的基础上，对孤立点检测算法进行了较为全面的阐述，并从数据流自身特征对数据流上挖掘技术的要求出发，就这些算法是否可以用于数据流上孤立点检测进行了深入探讨和研究。

我们下一步的工作就可以针对已有算法相对数据流存在的弊端进行改造，给出真正适用于数据流的快速孤立点检测算法。

当然，孤立点检测除了数据流这一发展方向外，根据数据库发展现状和实际应用的需要，还有以下几个研究热点：(1)检测高维空间数据集中的孤立点；(2)开发非参数的孤立点检测算法；(3)检测混合类型属性数据集中的孤立点；(4)检测 Web 数据的孤立点。

参 考 文 献

1 Ghoting A, Otey M E, Parthasarathy S. Loaded: Link-based outlier and anomaly detection in evolving data sets. In: Proc. of the IEEE International Conference on Data Mining, 2004

2 Fox A J. Outliers in Time Series. Journal of Royal Statistics Society, 1972, B 34: 350～363

3 Jain A K, Murty M N, Flynn P J. Data clustering: A review. ACM Computing Surveys, 1999, 31(3): 264～323

4 Bianco A M, Garcia B M, Martinez E J, et al. Robust procedures for regression models with ARIMA errors. In: COMPSTAT96, Proceeding in Computational Statistics Part A, 1996. 27～38

5 Bianco A M, Garcia B M, Martinez E J, et al. Outlier detection in regression models with ARIMA errors using robust estimations. Journal of Forecasting, 2001, 20: 565～579

6 Aggarwal C C, Yu P S. *Outlier Detection for High Dimensional* Data. In: Proceedings of the ACM SIGMOD Conference, 2001

7 Li C H, Sun Z H. GridOF: An efficient outlier outlier detection algroithm for very large datasets. Journal of Computer Research and Development, 2003, 40(11): 1586～1592

8 Chen C, Liu L M. Forecasting time series with outliers. Journal of Forecast, 1993, 12: 13～35

(下转第 225 页)

$$e_t=\sum_{i}^{k_t}\| q_i-\bar{p}_u \|^2+(n-k_t)(T\sigma)^2 \quad (19)$$

通过退火温度的控制可使 e_t 以较快速度收敛。初始阈值 σ 可选择为点集中点之间最小距离，T 为 10～20 之间，退火速率为 0.8～0.95 之间。

4)比较分析

实验统计结果表明，本文算法与基于 GA 的算法的抗仿射形变能力不相上下，但抗噪性能和抗出格点性能则更胜一筹。真实图像实验也能说明，本文算法更具实用性。就时间复杂度来说，基于 GA 的算法每一次迭代复杂度为 $O(mn)\times N$，其中 N 为遗传种群的规模，取值为 100～1000。本文算法每次迭代复杂度至多为 $O(mn)$，因而从复杂度上小于基于 GA 的算法。此外，基于 GA 的算法需要较多的迭代步数(60～100)才能达到收敛，本文算法只需较少的迭代(5～20)则完全收敛。

结论 本文提出一种新的基于仿射参数估计的迭代点匹配算法，很好地解决了由仿射带来的非刚性形变点集匹配问题。该算法把点匹配基本问题转化为函数优化问题，通过对点集间变换关系和匹配关系的反复估计迭代得到问题的最优解。文中给出了在匹配未知和已知两种情况下的仿射参数估计方法。在点集间匹配关系未知情况下，我们通过构造虚拟点对估计仿射参数，该方法简单有效，为算法提供了良好的初值。在匹配关系已知时，利用最小方差法进行参数估计，它确保了算法的收敛性。利用改进的最近点原则求解点集的匹配关系，减小了算法匹配误差，加快了收敛速度。文章还对虚拟点对初值估计方法和阈值 σ 的选取问题进行了讨论，并给出了改进办法。通过实验，并与基于 GA 的匹配方法进行比较，说明算法具有较强的鲁棒性和实用性。

参 考 文 献

1 Belongie S, Malik J, Puzicha J. Shape Matching and Object Recognition Using Shape Contexts. IEEE Trans on Pattern Analysis and Machine Intelligence, 2002,24(4): 509～522

2 Shapiro L S, Brady J M. Feature-Based correspondence: an eigenvector approach. Image and Vision Computing, 1992, 10(5): 283～288

3 Besl P J, Mckay N D. A method for registration of 3-D shapes. IEEE Trans on Pattern Analysis and Machine Intelligence, 1992, 14(2): 8～14

4 Talbi H, Batouche M C. Particle Swarm Optimization for Image Registration. In: 2004 International Conference on Information and Communication Technologies: From Theory to Applications, 2004,4: 397～398

5 Chui H, Rangarajan A. A New Algorithm for Non-Rigid Point Matching. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition, 2000,2: 44～51

6 Chang S, Cheng F, Hsu Wen-hsing, et al. Fast algorithm for point pattern matching: invariant to translations, rotations and scale changes. Pattern Recognition, 1997, 29(1): 11～16

7 Huttenlocher D P, Ullman S. Object recongnition using alignment. In: Proceeding of the First International Conference on Computer Vision, London, 1987. 102～111

8 Gold S, Rangarajan A, Lu C, et al. New algorithms for 2D and 3D point matching pose estimation and correspondence. Pattern Recognition, 1998, 31(8): 1019～1031

9 Zhang L, Xu W, Chang C. Genetic algorithm for affine point pattern matching. Pattern Recognition Letters, 2003, 24(1-3): 9～19

10 孙森，王秀坤，邵刚，冯林，贺明峰. 二维点模式图像的仿射变换配准. 计算机辅助设计与图形学学报，2005,7(17): 1497～1503

(上接第 203 页)

9 Masreliez C J, Martin R D. Robust Bayesian estimation for the linear model and robustifying the Kalman filter. IEEE Transactions on Automatic Control, 1977, AC 22(3): 361～371

10 Hawkins D. Identification of Outliers. London: Chapman and Hall, 1980

11 Knorr E M, Ng R T. A unified notion of outliers: Properties and computation. In: Proc. KDD 1997, 1997. 219～222

12 Knorr E M, Ng R T. Algorithms for mining distance-based outliers in large datasets. In: Proc. VLDB 1998, 1998. 392～403

13 Knorr E M, Ng R T. Finding intentional knowledge of distance-based outliers. In: Proc. VLDB 1999, 1999. 211～222

14 Knorr E M, Ng R T, Tucakov V. Distance-based outliers: Algorithms and applications. VLDB Journal, 2000, 8: 237～253

15 Jiang F, Sui Y F, Cao C G. Outlier Detection Using Rough Set Theory. In: RSFDGrC 2005, LNAI, 2005. 79～87

16 Cao Feng, Ester M, Qian Weining, et al. Density-Based Clustering over an Evolving Data Stream with Noise. In: SDM'2006

17 John G H. Robust Decision Trees: Removing Outliers from Databases. In: Proceedings of the First International Conference on Knowledge Discovery and Data Mining, 1995. 174～179

18 Liu H C, Shah S, Jiang W. On-line outlier detection and data cleaning. Computers and Chemical Engineering, 2004, 28: 1635～1647

19 Jagadis H V, Koudas N, Muthukrishnan S. Mining Deviants in a Time Series Database. VLDB, Edinburgh: Morgan Kaufmann Publishers, 1999. 102～113

20 Han J W, Kamber M. Data mining: concepts and techniques. NewYork: Morgan Kaufmann Publishers, 2001

21 Davies L, Gather U. The identification of multiple outliers. Journal of the American Statistical Association, 1993, 88: 782～792

22 Otey M E, Ghoting A, Parthasarathy S. Fast Distributed Outlier Detection in Mixed-Attribute Data Sets. Data Mining and Knowledge Discovery, 2005

23 Perarson R K. Outliers in Process Modeling and Identification. IEEE Transactions on Control Systems Technology, 2002, 10: 55～63

24 Tsay R S. Outliers, level shifts, and variance changes in time series. Journal of Forecasting, 1998, 7: 1～20

25 Tsay R S. Time series model specification in the presence of outliers. Journal of the American Statistical Association, 1996, 81: 132～141

26 Martin R D, Thomson D J. Robust-resistant spectrum estimation. In: Proceeding of the IEEE 1982, 70: 1097～1115

27 Papadimitriou S, Kitawaga H, Gibbons P, et al. LOCI: Fast Outlier Detection Using the Local Correlation Integral. In: Proc of the International Conference on Data Engineering, 2003. 315～326

28 Muthukrishnan S, Shah R, Vitter J S. Mining Deviants in Time Series Data Streams. In: Proc. of the 16th Int'l Conf. on Scientific and Statistical Database Management. Santorini Island: IEEE Computer Society, 2004. 41～50

29 Ramaswamy S, Rastogi R. Kyuseok S. Efficient Algorithms for Mining Outliers from LargeData Sets. In: SIGMOD' 00 2000. 427～438

30 Johnson T, Kwok I, Ng R T. Fast computation of 2-dimensional depth contours. In: Proc. KDD 1998, 1998. 224～228

31 Barnett V, Lewis T. Outliers in Statistical Data. John Wiley, 1994

32 Hodge V J, Austin J. A Survey of Outlier Detection Methodologies. Artificial Intelligence Review, 2004, 22: 85～126

33 Yang Y D, Sun Z H, Zhu Y Q, et al. A Fast Outlier Detection Algorithm for Data Streams Based on Dynamic Grids. Journal of Software, 2006, 17(8): 1796～1803

34 Tao Y F, Xiao X K, Zhou S G. Mining Distance-based Outliers from Large Databases in Any Metric Space. KDD'06, 2006. 394～403

35 Breuning M M, Kriegel H P, Ng R T, et al. Lof: Identifying density-based local outliers. In: Proc. ACM SIGMOD Conf 2000, 2000. 93～104

36 Cui Hongyin. Online Outlier Detection Over Data Streams. Paper for Master Degree. Smon Fraser University, 2005

37 He Z Y, Xu X F, Huang J Z, et al. A Frequent Pattern Discovery Method for Outlier Detection. WAIM 2004, LNCS 3219, 2004. 726～732

38 金澈清，钱卫宁，周傲英. 数据流分析与管理综述. 软件学报，2004, 15(8): 1172～1181