

时间序列相似性度量方法

王 燕, 安云杰⁺

(兰州理工大学 计算机与通信学院, 甘肃 兰州 730050)

摘 要: 在时间序列相似性度量中, 符号聚合近似 (symbolic aggregate approximation, SAX) 方法没有将符号化后的模式序列进一步处理, 导致存在一定误差, 为此提出将算术编码技术引用到 SAX 中, 即将符号化序列转换为编码序列, 实现时间序列在概率区间上的分析与度量; 在计算序列间的相似度时采用分层欧式距离算法, 综合考虑序列的统计距离和形态距离, 由粗到细地进行筛选, 达到序列整体趋势匹配以及细节拟合的目标。实验结果表明, 该方法在不同的数据集上都有一定的可行性, 具有较高的准确度和较好的鲁棒性。

关键词: 时间序列; 相似性度量; 关键点等; 算术编码技术; 符号化; 分层欧式距离

中图法分类号: TP311 **文献标识号:** A **文章编号:** 1000-7024 (2016) 09-2520-06

doi: 10.16208/j.issn1000-7024.2016.09.043

Method of time series similarity measure

WANG Yan, AN Yun-jie⁺

(College of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China)

Abstract: In time series similarity measure, symbolic aggregate approximation (SAX) does not further process the patterns sequence, which leads to certain error. To solve the problem, the arithmetic encoding technology was introduced into SAX. The symbol sequence was changed to encoding sequence, which was implemented to measure and analyze time series in probability interval. The hierarchical Euclidean distance algorithm was used in similarity measure of time series. Statistical and morphological distances of time series were considered, sequences were filtered from rough to subtle, and the overall trend of sequence was matched, and the goal of fitting was reached in detail, thus, the ultimate realization of time series similarity measure was achieved. Experimental results verify that using this method in different data sets has certain feasibility, and higher accuracy and better robustness.

Key words: time series; similarity measure; equivalence of key point; arithmetic encoding; SAX; hierarchical Euclidean distance

0 引 言

时间序列是对某一物理过程中的某一变量 $A(t)$ 分别在时刻 t_1, t_2, \dots, t_n ($t_1 < t_2 < \dots < t_n$) 进行观察测量而得到的离散有序的数据集合, 但由于时间序列数据的复杂, 多种类、高维度等特性, 为处理这些数据的分析带来了很大的困难, 因此时间序列数据挖掘工作变得尤为重要^[1]。

在整个时间序列数据挖掘过程中, 相似性度量技术是许多其它工作 (比如聚类、分类、关联规则等) 的基础, 吸引了大量学者的深入研究^[2-6]。其中, 基于特征的符号聚合近似 (SAX)^[7] 方法成为了最流行的相似性度量方法。例如, Antonio Canelas 等用 SAX 方法处理时间序列^[8], 具有

简单易用、不依赖具体实验数据、并能准确表示时间序列统计特征的优点, 但该方法弱化了序列的形态变化信息; 张海涛等提出基于趋势的时间序列相似性度量^[9], 能够客观的描述序列形态变化, 但由于选择的符号数太多, 丧失了处理意义, 使度量算法变的繁琐; 肖瑞等提出了编码匹配算法在不确定时间序列相似性度量上的应用^[10]; Yan Wang 将关键点提取和序列对等技术应用到了 SAX 算法中^[11], 为时间序列相似性度量提供了可以借鉴和参考的方向。虽然学者们对基于特征的相似性度量方法提出了一些改进, 但是缺少对符号化后的模式序列进行处理, 使得相似性度量的准确率较低。

为此本文提出对符号化序列用算术编码技术做进一步

收稿日期: 2015-09-28; 修订日期: 2015-11-29

基金项目: 国家自然科学基金项目 (61263019)

作者简介: 王燕 (1971-), 女, 甘肃兰州人, 硕士, 教授, 研究方向为模式识别、数据挖掘; ⁺通讯作者: 安云杰 (1990-), 女, 黑龙江绥化人, 硕士研究生, 研究方向为数据挖掘。E-mail: anyunjie@163.com

处理, 使其既便于理解又提高了相似性度量的准确率。在计算序列间相似度时采用分层欧式距离算法, 综合考虑均值距离和斜率距离, 使度量结果更加准确、高效。

1 数据预处理

时间序列存在噪声干扰、短期起伏性、非稳态及长期性等特点, 如果直接对原始序列进行研究, 难以获得令人满意的结果^[4]。所以需要原始时间序列进行预处理, 为了减少分段误差, 预处理中采用先规范化再分段的方法。

1.1 数据规范化

规范化是将时间序列数据转化为均值为 0, 标准差为 1 的标准正态分布序列。即原始时间序列 $A = \{A_1, A_2, \dots, A_n\}$ 转化为由 $S = \{S_1, S_2, \dots, S_n\}$ 表示, 其数据规范化如式 (1) 所示

$$S_i = \frac{A_i - u(A)}{\delta} \quad (i = 1, 2, \dots, n) \quad (1)$$

式中: A_i ($i = 1, 2, \dots, n$) 为原始时间序列的每一个取值, u 为整条时间序列的均值, δ 为整条时间序列的方差。

1.2 序列分段与对等

表 1 为时间序列分段与对等过程中所涉及到的符号的定义。

表 1 分段过程中涉及的符号的定义

符号	代表意义	符号	代表意义
A	原始时间序列	ω	子段 S' 的段数
S	标准时间序列	S'_i	子段 S' 的第 i 段
n	序列 S 的长度	S'_{ij}	子段 S' 第 j 个数据
S'	子段	η	时间段阈值
k	子段 S'_i 的长度	θ	筛选角度阈值

对于时间序列的分段, 本文采用关键点分段技术。现阶段关键点的提取方法有多种, 例如: 基于极值点、弧度、曲率、夹角等。本文为了达到对时间序列进行有效压缩和保留序列重要特征的目的, 采用基于极值点的关键点选取方法。具体的选取原则请参考文献 [11]。将标准时间序列 S 分段得到子段 S' ($S = \{S'_1, S'_2, \dots, S'_i, \dots, S'_\omega\}$) 其中每个子段 S'_i 中包含 k (k 值不等) 个数据点, 即 $S'_i = \{S'_{i1}, S'_{i2}, \dots, S'_{ij}, \dots, S'_{ik}\}$ ($i \leq \omega$)。如图 1 所示: 两条时间序列 SX 和 SY 利用关键点分段技术可得到 $SX = \{(SX'_1, t_{x1}), (SX'_5, t_{x5}), (SX'_6, t_{x6}), (SX'_7, t_{x7}), (SX'_9, t_{x9}), (SX'_{10}, t_{x10})\}$, $SY = \{(SY'_1, t_{y1}), (SY'_4, t_{y4}), (SY'_5, t_{y5}), (SY'_6, t_{y6}), (SY'_8, t_{y8}), (SY'_9, t_{y9}), (SY'_{10}, t_{y10})\}$ 。

关键点是时间序列中的重要节点, 利用关键点分段技术得到的子段可以保持原序列的基本形态, 为计算序列间的相似度提供了方便。但通常情况下, 任意两条时间序列 SX 和 SY 在采用关键点分段后各端点不会完全对齐。为了

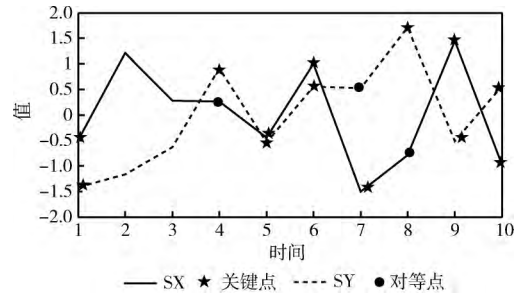


图 1 关键点对等

计算序列间相似性的方便, 需对分段后的序列进行关键点对等。对于上述 SX 和 SY 经过关键点对等后表示为: $SX = \{(SX'_1, t_{x1}), (SX'_4, t_{x4}), (SX'_5, t_{x5}), (SX'_6, t_{x6}), (SX'_7, t_{x7}), (SX'_8, t_{x8}), (SX'_9, t_{x9}), (SX'_{10}, t_{x10})\}$, $SY = \{(SY'_1, t_{y1}), (SY'_4, t_{y4}), (SY'_5, t_{y5}), (SY'_6, t_{y6}), (SY'_7, t_{y7}), (SY'_8, t_{y8}), (SY'_9, t_{y9}), (SY'_{10}, t_{y10})\}$ 。

2 符号化与算术编码

SAX 是时间序列相似性度量中应用最为广泛的方法。但是 SAX 方法最大的缺陷是只采用均值信息直接衡量序列间的相似性。本文提出在计算时间序列相似性时, 应用算术编码技术对符号化序列进一步处理, 使得到的编码序列能够准确的反映出序列的变化特征, 并为相似性度量奠定了可靠的基础。

2.1 符号化

时间序列符号化是通过离散化方法将连续实数值的时间序列或者时间序列的某段时间内的波形映射到有限的符号集上, 从而实现时间序列模式转换^[12]。将时间序列映射到不同区域上, 同一区域映射为一个字符。对于字符集大小的选择问题, 若太大会丧失符号化的意义, 太小又起不到一定的效果, 一般情况下, 符号集大小选为 2。因此, 本文采用两个字符 $\{0, 1\}$ 和一个划分点 O 对时间序列进行符号化表示。子段 S'_i 的均值 a_i 按照式 (2) 来计算, 斜率 k_i 按照式 (3) 计算

$$\begin{cases} \text{if} \left[\sum_i = \frac{\sum_{j=1}^k S'_{ij}}{k} \geq 0 \right] & a_i = 1 \\ & i = 1, 2, \dots, \omega \quad (2) \\ \text{if} \left[\sum_i = \frac{\sum_{j=1}^k S'_{ij}}{k} < 0 \right] & a_i = 0 \\ \text{if} (90^\circ \leq \theta_i \leq 0^\circ) & k_i = 1 \\ \text{if} (180^\circ \leq \theta_i \leq 90^\circ) & k_i = 0 \end{cases} \quad i = 1, 2, \dots, \omega \quad (3)$$

其中, \sum_i 为正规化子段 S'_i 的均值; θ_i 为每个子段第一个点与最后一个点的连线与 x 轴正方向的夹角。由此, 子段 S'_i 的均值和斜率均可用一个字符 0 或 1 表示, 时间序列 S 的均值和斜率可由 ω 个 0, 1 字符串表示。对于图 1 中的时间序

列 SX 采用上述方法得到的均值符号化序列为：1, 0, 1, 0, 0, 1, 0；斜率符号化序列为：1, 0, 1, 0, 1, 1, 0。

2.2 算术编码

算术编码是一种对数据进行的无损压缩的方式，是一种在给定符号集合和符号概率集合的情况下进行的编码方式。该编码方式的效果接近最优，能够精确表示出数据的关键信息^[10]。本文的符号集为符号化的取值 {0, 1}，符号概率集为 {0.5, 0.5}。具体编码过程如下：

(1) 输入时间序列 S 的均值和斜率符号化序列，符号集 {0, 1}，符号概率集 {0.5, 0.5}。首次编码区间定为 [0, 1]，则对于首次编码符号为 0 时，编码区间为 [0, 0.5]；首次编码符号为 1 时，编码区间为 [0.5, 1]。

(2) 根据 (1) 的首次编码区间对时间序列 S 的均值符号序列和斜率符号序列进行算术编码。现以均值序列编码为例：设子段 S'_i(1 < i < w) 的编码区间为 [a, b] (a ≥ 0, b ≤ 1, a < b)；对于下一个子段 S'_{i+1}(i+1 ≤ w) 时，若为

字符 0，则对应的编码为 [a, (a+b)/2]；若为字符 1，则对应的编码为 [(a+b)/2, b]；若 i=w，编码结束，输出所有的均值编码区间值。

(3) 每一个均值编码区间的均值编码值 Sa'_i 的计算公式如下

$$Sa'_i = \frac{(a+b)}{2} \tag{4}$$

式中：a ∈ [0, 1], b ∈ [0, 1]。

(4) 最后输出时间序列 S 的均值编码序列 Sa 和斜率编码序列 Sk。

对于图 1 中的时间序列 SX 按照上述算术编码规则得到均值编码区间序列为 [0.5, 1]；[0.5, 0.75]；[0.625, 0.75]；[0.625, 0.6875]；[0.625, 0.65625]；[0.640625, 0.65625]；[0.640625, 0.648437]。与此同时，均值编码序列为：0.75；0.625；0.6875；0.65625；0.640625；0.648437；0.644531。对于 SX 和 SY 的均值编码和斜率编码如图 2 所示。

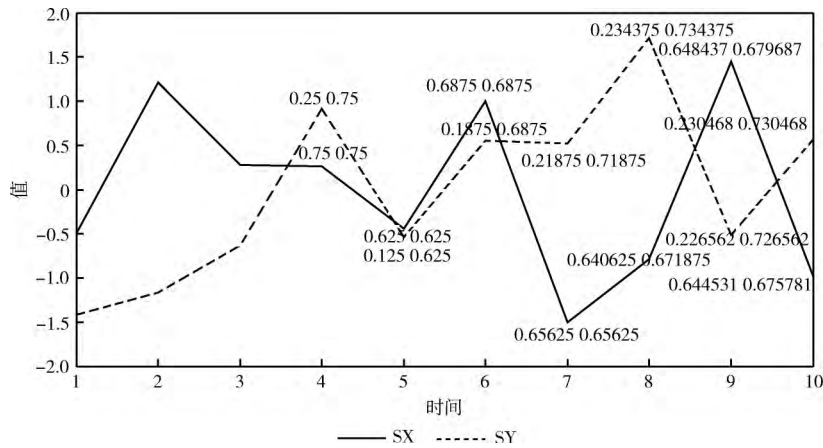


图 2 两条时间序列的均值编码和斜率编码

在图 2 中，标注出的关键点的前一个数字代表本段的均值编码值，后一个数字代表斜率编码值。对于编码的两条时间序列计算相似性距离时，需要计算出两条时间序列每段均值编码差和斜率编码差。以时间序列段之间的均值编码差为例，按式 (5) 计算得出第 i 个子段 S'_i 的均值编码差 Da'_i。斜率编码差的计算公式与均值编码差的相似，此处不再列出

$$Da'_i = |SXa'_i - SYa'_i| \tag{5}$$

3 相似性度量

本文采用分层欧氏距离 (HE) 算法对编码序列计算相似度。主要思想是：利用均值编码对待度量序列进行粗度量，对于满足要求的序列形成候选集；接下来在候选集中利用斜率编码进行细度量。这样有层次地进行相似性搜索，逐步缩小搜索范围，使转换后的新序列与原始序列的拟合度更高，相似性度量效率也更高。具体的过程如下：

(1) 均值编码粗度量：在待度量序列中，对经过符号化和算术编码后的均值序列用欧氏距离计算两条时间序列在统计特征上的相似度。对满足式 (6) 的序列形成候选集序列

$$Da(SX, SY) = \sqrt{\sum_{i=1}^w (v_i^2 (Da'_i)^2)} \leq \epsilon \tag{6}$$

式中：Da(SX, SY) 为两条时间序列 SX 和 SY 之间的均值编码距离，ε 为均值编码阈值。v_i 表示子段的权值，其值等于该子段的时间长度，即 v_i = t_i - t_{i-1} (i = 1, 2, ..., w)。v_i 的值可以反映出该子段在所处的分段中的重要性，t_i 表示第 i 个子段的结束时刻，t_{i-1} 表示第 i-1 个子段的结束时刻，t_n 表示序列结束时刻。对于 t_i, t_{i-1} 和 t_n 之间的关系如式 (7)

$$\frac{(\sum_{i=1}^w (t_i - t_{i-1}))}{t_n} = 1 (i = 1, 2, \dots, w) \tag{7}$$

(2) 斜率编码细度量: 对候选集中的时间序列比较形态距离。给定的斜率编码阈值 λ , 斜率编码距离的计算如式 (8)

$$Dk(SX, SY) = \sqrt{\sum_{i=1}^w (v_i^2 (Dk'_i)^2)} \leq \lambda \quad (8)$$

本文将分层的思想运用到计算序列间相似性问题中, 可以达到在统计距离和形态距离上的相似, 从而提高相似性度量的准确率。本文的时间序列相似性度量流程如图 3 所示。

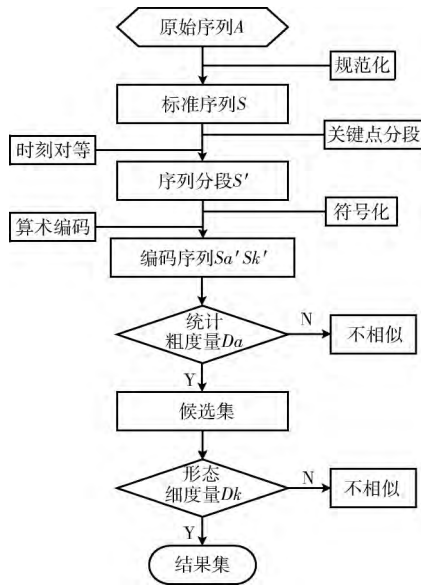


图 3 算法流程

其主要算法为:

输入: 待度量时间序列, 角度阈值 θ , 时间阈值 η , 均值编码阈值 ϵ , 斜率编码阈值 λ ;

输出: 相似的结果集;

步骤 1 将时间序列规范化, 得到标准时间序列 S ;

步骤 2 利用关键点和对等技术分段, 得到 w 个子段 S' ;

步骤 3 对每个子段 S' 的均值与斜率进行符号化;

步骤 4 对符号化后的均值和斜率序列进行算术编码, 得到每个子段的均值编码序列 Sa' 和斜率编码序列 Sk' ;

步骤 5 计算两条时间序列对应的每段的均值编码差 Da'_i 和斜率编码差 Dk'_i 。

步骤 6 对编码后的序列采用基于分层欧式距离算法进行度量, 首先进行均值编码粗度量, 形成候选集, 再进行斜率编码细度量, 最终得到相似序列。

4 实验与分析

4.1 实验环境与数据

实验的环境为: Windows 7, Inter (R) Core (TM) 2 Duo CPU T5870 2.00 GHz, Memory 2.00 GB, MatlabR 2010b。为了突出本文方法比 Euclid、分层动态时间弯曲 (HD)、传统 SAX 方法在相似性度量方面的优势, 采用相

同的且已知分类结果的一元时间序列数据集 UCI 进行实验并对比。表 2 对 UCI 中的 3 组数据集进行了介绍。

表 2 时间序列数据集

序号	名称	类别个数	训练集个数	测试集个数	序列长度
1	Synthetic control	6	300	300	60
2	beef	5	30	30	470
3	coffee	2	28	28	286

4.2 实验方案与结果分析

实验 1: 本文参考文献 [13] 提出的把实验数据分为测试集和训练集两个已知的集合的方法, 利用所选的度量算法衡量测试集中的样本与训练集中的样本的相似度。即给定相似度距离阈值, 若测试集样本和训练集样本之间的相似度距离低于给定阈值, 则判定这两条时间序列相似, 高于则为非相似序列, 完成某一算法在测试集中所有序列的相似性度量后, 把这些结果的平均值作为此算法在这个数据集上的实验结果^[13]。实验结果用正确率和误检率来衡量。正确率指的是原本就相似的时间序列, 经相似性度量后依旧判定为相似序列的条数与原相似序列总数的比值。误检率是指原本不相似的序列, 经相似性度量后错误的判断为相似序列的条数与原不相似序列总数的比值。例如在 Synthetic control 数据集中有 6 类数据, 计算正确率时是在每一类中随机选取一条数据与训练集中同类的 50 条数据进行度量, 计算经过实验判定为相似的序列条数与实验总数之比; 在计算误检率时是在 6 类测试集中选一类中任意一条和训练集中不同类的 250 条数据进行实验, 计算误认为相似的条数与进行实验总数之比。为了体现本文方法的优越性, 表 3 显示了传统 SAX 方法与本文方法在相同的数据集上得到结果的对比。

表 3 实验结果对比

序号	名称	误检率/%		正确率/%	
		传统 SAX	本文	传统 SAX	本文
1	Synthetic control	34.8	24.4	85	88.67
2	beef	33.33	16.67	86.67	93.33
3	coffee	35.71	14.29	89.29	92.85

表 3 显示出了本文方法和传统 SAX 方法在正确率和误检率方面的对比。可以看出本文方法的正确率明显高于传统 SAX 方法, 误检率可以达到传统 SAX 误检率的一半。其主要原因有如下两点: ①是传统的 SAX 方法采用 PAA 分段, 用平均值表示每个分段, 而平均值并不能完全的表示出时间序列的重要信息, 忽略了原始序列的形态特征。本文用关键点技术分段, 既能提取序列的重要特征点, 又能有效的压缩序列并降低维数, 可以更加准确代表原始序列。本文综合考虑了形态特征和统计特征, 不但拥有传统 SAX 方法的简单方便的特点, 而且有效地描述了时间序列的形态

变化。②是在计算相似性距离之前没有对 SAX 序列充分处理，导致传统 SAX 方法在度量上有一定的误差。而本文采用基于均值和斜率特征的基础上应用算术编码对字符串进一步处理，将序列转换到概率区间上，使度量结果更加准确。

实验 2：分层欧式距离（HE）和分层动态时间弯曲（HD）算法在数据集 synthetic control 上随机抽取 5 组时间序列数据，每组数据抽取 60 条序列，分别属于 6 个不同的类别，每个类别 10 条。图 4 显示出 HE 和 HD 在每组数据的相似性距离运算时间的对比。

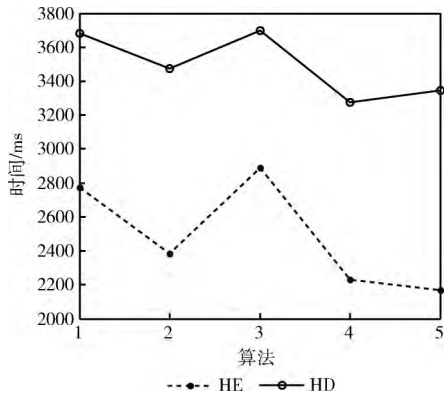


图 4 HE 与 HD 运算时间的对比

由图 4 可以很容易得到：分层欧式距离（HE）比分层动态时间弯曲（HD）算法的时间消耗小的多。由于两种分层思想是基于传统算法的，传统的动态时间弯曲（DTW）的时间代价很大^[14]，限制了 HD 使用范围。而传统的欧式距离算法运算速度快，复杂度比较低，所以计算相似性距离时运用 HE 算法可以达到较高的效率。

实验 3：对比传统欧氏距离（Euclid）与分层欧式距离（HE）在数据集 coffee 中随机抽取 10 组数据进行实验的结果，表 4 显示了两种算法在计算序列间相似性度量精度方面的对比。

表 4 显示出了 Euclid 和 HE 在每组数据集上得到的时序相似性度量在精度方面的对比，可以得出 Euclid 在不同数据集上的度量精度波动很大，而 HE 除了在第 10 组数据之外，其它 9 组的相似性度量精度都高于传统欧式距离并且十分稳定。从两种算法在 10 组数据的相似性度量平均值上可以看出 HE 比 Euclid 要高出将近 10%。本文 HE 算法综合考虑了序列的统计差异和形态差异，即便两条时间序列的斜率距离为 0 均值距离不同，其差异也会体现出来，因此更加符合时间序列的相似性度量的要求。由此可得出 HE 比 Euclid 在计算时间序列相似性度量方面具有较高的精度和较好的稳定性。

表 4 Euclid 和 HE 在相似性度量精度上的对比

	1	2	3	4	5	6	7	8	9	10	平均值
Euclid	0.66	0.8	0.75	0.79	0.81	0.71	0.74	0.68	0.76	0.85	0.755
HE	0.79	0.85	0.84	0.87	0.9	0.87	0.76	0.8	0.9	0.82	0.84

5 结束语

本文针对时间序列符号化相似性度量结果存在误差的问题，提出一种基于符号化与算术编码相结合的相似性度量方法，它通过分层欧式距离算法结合均值编码距离和斜率编码距离计算时序间的相似性，从而实现了时间序列在概率区间上的准确分析。本文方法可以对时间序列提供更多的描述信息，提高了相似性度量的准确性和可靠性，因而它在时间序列的相似性度量中获得更加出色的表现。接下来我们着力研究的方向是实现构造数据索引。

参考文献：

[1] XIE Fuding, LI Ying, SUN Yan, et al. Improved symbolization time series method [J]. Computer Engineering and Design, 2012, 33 (10): 3950-3953 (in Chinese). [谢福鼎, 李迎, 孙岩, 等. 改进的符号化时间序列处理方法 [J]. 计算机工程与设计, 2012, 33 (10): 3950-3953.]

[2] Fu Tak-chung. A review on time series data mining [J]. Engineering Application of Artificial Intelligence, 2012, 24 (1): 164-181.

[3] GONG Jinli, PENG Xianwu. Pattern similarity measurement of traffic flow time series [J]. Computer Engineering and Applications, 2015, 51 (6): 103-107 (in Chinese). [弓晋丽, 彭贤武. 交通流时间序列模式相似性度量法 [J]. 计算机工程与应用, 2015, 51 (6): 103-107.]

[4] LI Hailin, GUO Chonghui. Similarity measure based on the multidimensional shape feature representation for time series [J]. System Engineering Theory and Practice, 2013, 33 (4): 1024-1034 (in Chinese). [李海林, 郭崇慧. 基于多维形态特征表示的时间序列相似性度量 [J]. 系统工程理论与实践, 2013, 33 (4): 1024-1034.]

[5] LI Hailin, GUO Chonghui. Piecewise aggregate approximation method based on cloud model for time series [J]. Control and Decision, 2011, 26 (10): 1525-1529 (in Chinese). [李海林, 郭崇慧. 基于云模型的时间序列分段聚合近似方法 [J]. 控制与决策, 2011, 26 (10): 1525-1529.]

[6] SUN Ya, LI Zhihua. Clustering algorithm for time series based on locally extreme point [J]. Computer Engineering, 2015, 41 (5): 33-37 (in Chinese). [孙雅, 李志华. 基于区域极值点的时间序列聚类算法 [J]. 计算机工程, 2015, 41 (5): 33-37.]

- [7] Keogh E, Chakrabarti K, Pazzani M, et al. Dimensionality reduction for fast similarity search in large time series databases [J]. Knowledge and Information Systems, 2001, 3 (3): 263-286.
- [8] Antonio Canelas, Rui Neves, Nuno Horta. A SAX-GA approach to evolve investment strategies on financial markets based on pattern discovery techniques [J]. Expert Systems with Application, 2013, 40 (5): 1579-1590.
- [9] ZHANG Haitao, LI Zhihua, SUN Ya. New similarity measure method on time series [J]. Computer Engineering and Design, 2014, 35 (4): 1279-1284 (in Chinese). [张海涛, 李志华, 孙雅. 新的时间序列相似性度量方法 [J]. 计算机工程与设计, 2014, 35 (4): 1279-1284.]
- [10] XIAO Rui, LIU Guohua, SONG Zhuan. Encoding similarity matching for uncertain time series [J]. Computer and Digital Engineering, 2013, 11 (41): 1797-1800 (in Chinese). [肖瑞, 刘国华, 宋转. 不确定时间序列的编码匹配 [J]. 计算机与数字工程, 2013, 11 (41): 1797-1800.]
- [11] Wang Yan, Su Yu. The method for time series based on symbolic form and area difference [J]. Journal of Information and Computational Science, 2015, 12 (4): 1657-1666.
- [12] BAO Weiwei, XIA Shixiong, Niu Qiang. Similarity search for time series based on triangle inequality [J]. Computer Engineering and Design, 2014, 35 (12): 4191-4201 (in Chinese). [包伟伟, 夏士雄, 牛强. 基于三角不等式的时间序列相似性搜索算法 [J]. 计算机工程与设计, 2014, 35 (12): 4191-4201.]
- [13] LI Hailin, GUO Chonghui. Symbolic aggregate approximation based on shape features [J]. Pattern Recognition and Artificial Intelligence, 2011, 24 (5): 665-672 (in Chinese). [李海林, 郭崇慧. 基于形态特征的时间序列符号聚合近似方法 [J]. 模式识别与人工智能, 2011, 24 (5): 665-672.]
- [14] Banko Z, Abonyi J. Correlation based dynamic time warping of multivariate time series [J]. Expert Systems with Applications, 2012, 39 (17): 12814-12823.

(上接第 2463 页)

- [17] Lin Yongbing, Zheng Jianhua, Zheng Xiaozhen, et al. Removal of dependency between multiple PUs in a CU for disparity vector derivation [C] //JCT3V-B0096, Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 2nd Meeting, 2012: 13-19.
- [18] Kang Jewon, Chen Ying, Zhang Li. CU-based disparity vector derivation in 3D-HEVC [C] //JCT3V-D0181, Joint Collaborative Team on 3D Video Coding Extensions of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 4th Meeting, 2013: 20-26.
- [19] Zhou Zuocheng, Jia Kebi. Group searching and proportion combination based disparity vector derivation for 3D-HEVC [C] //IEEE International Conference on Image Processing, 2015: 3828-3831.
- [20] Dmytro Rusanovskyy, Karsten Müller, Anthony Vetro. Test conditions of 3DV core experiments [C] //JCT3V-E1100, Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 5th Meeting, 2013: 1-8.
- [21] Na Taeyoung, Kim Munchurl. A novel no-reference PSNR estimation method with regard to deblocking filtering effect in H.264/AVC bitstreams [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2014, 24 (2): 320-330.