Knowledge-Based Systems xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Piecewise statistic approximation based similarity measure for time series

Qinglin Cai, Ling Chen *, Jianling Sun

College of Computer Science and Technology, Zhejiang University, Hangzhou 310000, China

ARTICLE INFO

Article history:
Received 18 December 2014
Received in revised form 28 April 2015
Accepted 1 May 2015
Available online xxxx

Keywords:
Time series
Piecewise representation
Dynamic time warping
Similarity measure
Pattern matching

ABSTRACT

In the research field of time series analysis, dynamic time warping distance (DTW) is a prevalent similarity measure with high precision. However, the computational complexity of DTW is high, which makes it difficult to be applied to the high dimensional time series. An effective solution is to compute DTW on the piecewise representation (PR-DTW), which employs the features of the subsequences of time series for similarity measure. However, the features that most existing piecewise representations focus on are too simple, which capture only one aspect of the fluctuation information of time series, and thus influence the precision of PR-DTW. In order to solve this problem, we propose a novel piecewise representation model, named *piecewise statistic approximation* (PSA), for supporting the PR-DTW measure. Rather than focusing on a single type of features, PSA extracts multiple statistical features to capture the synthetic fluctuation information for similarity measure. Besides, by taking the weighted Euclidean distance for the subsequence matching in the subroutine, PSA based DTW (PSADTW) can discriminate the expressivities of the multiple features. Comprehensive experiments over 45 real-world datasets empirically demonstrate that, PSA is well suited to support both precise and efficient PR-DTW measure.

1. Introduction

In the past two decades, a plethora of topics were proposed in the research field of time series analysis [1–3], including similarity search [4–6], classification [7–9], clustering [10,11], prediction [12,13], anomaly detection [14], and motif discovery [15]. Many methods on these topics, e.g., *k*NN classification or *k*-means clustering algorithm, need to assess the similarity between time series. Researchers usually employ the similarity measure (e.g., L_p-norms distance [16], dynamic time warping distance (DTW) [4,17], and the variations of edit distance (LCSS, EDR, ERP) [18], etc.) to evaluate the similarity between time series. With high precision, DTW is one of the most prevalent similarity measures for time series [1,4,7,17]. However, the computational complexity of DTW is $O(n^2)$, which greatly limits its application to the high dimensional time series and the dynamic data stream.

Many methods have been proposed to cut down the computational complexity of DTW, which can be roughly classified into two types: (1) Methods that take advantage of the speedup techniques to improve the basic algorithm of DTW, e.g., dynamic programming constraint [17], lower boundary heuristics [4]; (2) Methods that compute DTW in the low-dimensional feature space. Recent results suggest that we are at the asymptotic limit of speeding up DTW [17]. For the second type, a lot of data representation models can be used to construct the feature space, e.g., discrete fourier transformation (DFT) [19], discrete wavelet transformation (DWT) [3], piecewise aggregation approximation (PAA) [11,20], adaptive piecewise constant approximation (APCA) [21], piecewise linear approximation (PLA) [22,23], bag-of-pattern representation [24,25], shapelet transformation [8], etc. Among these models, the piecewise representations are prominent for capturing the local characteristics of time series. However, the features that most existing piecewise representations focus on are too simple, which reflect only one aspect of the fluctuation information of time series. For instance, PAA [20] and APCA [21] extract only the mean values, PLA [23] extracts only the linear fitting slopes, the derivative time series segment approximation (DSA) [6] extracts only the mean values of the derivative subsequences, and the state-of-the-art piecewise cloud approximation (PWCA) [26] extracts only the cloud models of subsequences, etc. This shortcoming would lead to the weak expressivities of the representation methods, and influence the precision of the piecewise representation based DTW (PR-DTW).

* Corresponding author. Tel.: +86 13606527774.
E-mail addresses: qlcai@zju.edu.cn (Q. Cai), lingchen@zju.edu.cn (L. Chen), sunjl@zju.edu.cn (J. Sun).

http://dx.doi.org/10.1016/j.knosys.2015.05.005
0950-7051/

In order to improve the precision of existing PR-DTWs, we propose a novel piecewise representation model, named piecewise statistic approximation (PSA), and the corresponding similarity measure, i.e., PSA based DTW (PSADTW). Rather than focusing on a single type of features, multiple statistical features (e.g., mean value, standard deviation, and skewness, etc.) are extracted from the subsequences of time series in PSA, which can capture the synthetic fluctuation information of time series for similarity measure. Besides, in the PSADTW, we take the weighted Euclidean distance for matching the extracted features, which can discriminate the expressivities of the features by the weights, and make PSADTW sensitive to the effective information. Since the subroutine of PSADTW is running on the subsequence matching, it can overcome the problem of phase-shift between subsequences. The contributions of this paper are listed as follows:

(1) Propose the PSA representation for time series, which is capable of capturing the synthetic fluctuation information of time series for similarity measure.
(2) Propose the PSADTW for time series similarity measure, which can discriminate the expressivities of multiple features and overcome the phase-shift between subsequences.
(3) Conduct comprehensive experiments to evaluate the performance of PSADTW.

The structure of this paper is as follows: The related work regarding data representation and similarity measure is given in Section 2; Section 3 gives the details of PSA representation; the details of PSADTW measure are presented in Section 4; Section 5 provides the comprehensive experiment results and analysis; Section 6 concludes this paper.

2. Related work

2.1. Data representation

In many application fields, the high dimensionality of time series has limited the performance of a myriad of algorithms. With this problem, a great number of data representation methods have been proposed to reduce the dimensionality of time series [1,2]. They can be roughly classified into two categories: One sort is based on the global mapping, which projects the whole time series into a low-dimensional feature space; the other sort is based on the local mapping, which projects the segmented subsequences of time series into the feature space.

The first attempt based on the global mapping is the DFT method [19], which projects the time series into the frequency domain, and extracts the coefficients of the main frequencies as features to reduce the dimension of the raw data. Likewise, DWT [3] projects the raw data into the wavelet domain, and extracts the wavelet coefficients as features. Compared with DFT, DWT can capture the multi-resolution characteristics of time series in both time and frequency domains. Recently, a novel global mapping based representation named shapelet transformation [8] was proposed, which maps the time series into the shapelet space and takes the distances between time series and the shapelets as features. This approach is comprehensible and can offer insight into the problem domain. Besides, another recent research direction on the global mapping is to adopt the bag-of-word model from the text mining and information retrieval communities. The first attempt is the bag-of-pattern representation (BOP) [25], which projects time series into the vector space of local patterns. Then, Senin et al. [27] improved BOP by using the tf-idf score to select the pattern words. Different from BOP, Baydogan et al. [24] proposed the bag-of-features representation (TSBF) method, which employs the typical subsequences to compose a compact codebook as the vector space.

Since the global mapping based representation methods approximate time series with the global features, they would lose the local characteristics of time series, which are crucial to the similarity measure. In the contrary, the local mapping based representation methods can overcome this drawback, e.g., the time series forest method [29], which can perform the accurate classification by extracting multiple local features from time series.

In the local mapping based methods, the piecewise representation methods are most prevalent. The first attempt is the PAA [20], which segments time series into the equal-length subsequences, and extracts the mean values of the subsequences as features to approximate the raw data. However, the extracted single sort of features only indicates the height of the subsequences, which may cause the local information loss. Consecutively, an adaptive version of PAA, named piecewise constant approximation (APCA) [21], was proposed, which can segment time series into the subsequences with adaptive lengths and thus can approximate time series with less error. As well, a multi-resolution version of PAA, named MPAA [11], was proposed, which can iteratively segment time series into 2^i subsequences. However, both of the variations inherit the poor expressivity of PAA. Another pioneer piecewise representation is the PLA [22,23], which extracts the linear fitting slopes of the subsequences as features to approximate the raw data. However, the fitting slopes only reflect the movement trends of the subsequences. For the time series fluctuating sharply with high frequency, the effect of PLA on dimension reduction is not prominent.

In addition, two novel piecewise representations were proposed recently. One is the DSA [6] representation, which takes the mean values of the derivative subsequences of time series as features. However, it is sensitive to the small fluctuation caused by the noise. The other is the PWCA [26] representation, which employs the cloud models to fit the data distribution of the subsequences. However, the extracted features only reflect the data distribution characteristics and cannot capture the fluctuation information of time series.

2.2. Similarity measure

The similarity measure methods for time series have been elaborated in many occasions [1,2,7,17,18,30,31], and a complete review on all methods is out of the scope of this paper. We would only focus on two prevalent sorts of methods: the lock-step measure and the elastic measure.

The lock-step measure [1,2,16] is computed by strictly aligning the indices of the time series. L_p-norms distance [16,30] is the most prevalent lock-step measure, which is the Manhattan distance, the Euclidean distance, and the maximum distance respectively as $p = 1$, $p = 2$, and $p \to \infty$. L_p-norms is well known for its simple implementation, low computational complexity, i.e., $O(n)$, free parameters, and satisfying the triangular inequality. However, its measure precision is sensitive to the noise, the outliers, and the basic shape variations. To improve the measure precision of L_p-norms, the weighted Euclidean distance [22] was proposed, which introduces the weights to indicate the importance of the points in the measure. However, it retains the sensitivity of L_p-norms to the basic shape variations.

To overcome the shortcomings of the lock-step measure, the elastic measure was proposed, which is computed by realigning the indices of time series and is robust to the basic shape variations. DTW [4,7,17] is the most prevalent elastic measure, which is computed by the dynamic programming. Although it is robust to the time warping and the phase-shift, and has very high measure precision, the $O(n^2)$ computational complexity largely limits

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

the application of DTW to the high dimensional time series and the high-speed data stream [4,32]. Besides, the "singularities" problem [33] may also exist in DTW, where a single point on one time series may map onto a large subsection of another series. To overcome this problem, Keogh et al. [33] proposed the derivative dynamic time warping distance (DDTW), which is computed on the derivatives of time series and could find the natural alignments between time series.

Recently, a novel penalty-based DTW named WDTW [34] was proposed, which can penalize the points of time series with their phase difference and thus prevent the minimum distance distortion caused by the outliers. Another novel DTW named CIDDTW [30] was also proposed, which introduces the overall variation regularization into DTW, and can identify the time series with the similar complexity. However, the two novel variations retain the high computational complexity of DTW.

To improve the high computational complexity of DTW, many methods have been proposed, which can be roughly classified into two categories: the methods of improving the basic algorithm of DTW, and the methods of computing DTW in the low-dimensional feature space.

For the first type, the global constraints are mostly used, such as the Sakoe-Chiba Band and the Itakura Parallelogram [22] Band, which can restrict the DTW computation into a small range in the dynamic programming table. Besides, four novel lower boundary heuristics [4] were proposed for the task of similarity search with DTW, which can raise the search efficiency under DTW as fast as that under the Euclidean distance over large-scale datasets. Although this type of methods is effective, they are at the asymptotic limit of speeding up DTW.

For the second type, the PAA representation based PDTW [35] and the PLA representation based SDTW [36] are the early pioneers, and the DSA representation based DSADTW [6] is the state-of-the-art method. Rather than in the raw data space, they compute DTW in the PAA, PLA, and DSA spaces respectively. Since the segment numbers are much less than the original time series length, this type of methods can greatly decrease the computational complexity of DTW. Nonetheless, their measure precision depends on the used representation methods, where the extracted features are crucial to the measure precision.

3. Piecewise statistic approximation

Without loss of generality, we first give three basic definitions relevant to time series as follows.

Definition 1. Time series. The sample sequence of a variable over the contiguous time moments is called time series, denoted as $T = \{t_1, t_2, \ldots, t_i, \ldots, t_n\}$, where t_i denotes the sample value of the variable on the i-th moment.

Definition 2. Subsequence. Given time series $T = \{t_1, t_2, \ldots, t_i, \ldots, t_n\}$, the subset S of T that consists of the continuous elements $\{t_{i+1}, t_{i+2}, \ldots, t_{i+l}\}$, where $0 \leqslant i \leqslant n-l$ and $0 \leqslant l \leqslant n$, is called the subsequence of T.

Definition 3. Time series data representation. Given time series set $\boldsymbol{T} = \{T_1, T_2, \ldots, T_n\}$ and $\boldsymbol{V} \in \boldsymbol{R}^n$, if $\exists f : \boldsymbol{T} \rightarrow \boldsymbol{V}$, then $\boldsymbol{V}$ is called time series data representation.

At the beginning, it is necessary to z-normalize the raw time series to have zero mean and standard deviation 1, as a pre-processing step [4].

Since the piecewise representation methods employ the disjoint window to reduce the dimensionality of time series [6,20,26], the local characteristics of time series can be investigated independently, and the global characteristics of time series can be explored from a high-level perspective of subsequences.

Definition 4. Disjoint window. Given the time series $T = \{t_1, t_2, \ldots, t_i, \ldots, t_n\}$ and the window W with length $l, l < n$, if W segments T into the adjacent subsequence set $\boldsymbol{S} = \{S_1, S_2, S_3, \ldots, S_i, S_{i+1}, \ldots\}$, where $S_i = \{t_i, t_{i+1}, t_{i+2}, \ldots, t_{i+l-1}\}$ and $S_{i+1} = \{t_{i+l}, t_{i+l+1}, \ldots, t_{i+2l-1}\}$, then W is called disjoint window.

Rather than focusing on one type of features, we would extract multiple statistical features in PSA, e.g., mean value μ, standard deviation σ, and skewness SK, etc., to describe the characteristics of time series, which can capture the synthetic fluctuation information from the different aspects. As follows, we show the significances of six general statistical features [37], i.e., mean value μ, variance D, standard deviation σ, dispersion factor CV, skewness SK, and kurtosis K, and compute them as Formulas (1)–(6).

- **Mean value**($\boldsymbol{\mu}$) is the balance point of a subsequence. It indicates the height and the central tendency of the subsequence.
- **Variance** ($\boldsymbol{D}$) or **standard deviation** ($\boldsymbol{\sigma}$) indicates the intensity and the dispersion of the subsequence.
- **Dispersion factor** ($\boldsymbol{CV}$) indicates the relative dispersion of the subsequence.
- **Skewness**($\boldsymbol{SK}$) or **kurtosis**($\boldsymbol{K}$) indicates the data distribution of the subsequence.

$$\mu = \left(\sum_{i}^{i+l-1} s_i\right) \Big/ l \tag{1}$$

$$D = \sum_{i}^{i+l-1} (s_i - \mu)^2 \Big/ (l-1) \tag{2}$$

$$\sigma = \sqrt{\textstyle\sum_{i}^{i+l-1} (s_i - \mu)^2 \big/ (l-1)} \tag{3}$$

$$CV = \frac{\sigma}{\mu} \tag{4}$$

$$SK = \frac{l \sum_{i=1}^{i+l-1} (s_i - \mu)^3}{(l-1)(l-2)\sigma^3} \tag{5}$$

$$K = \frac{l(l+1)\sum_{i=1}^{i+l-1} (s_i - \mu)^4 - 3(l-1)\left[\sum_{i=1}^{i+l-1} (s_i - \mu)^2\right]^2}{(l-1)(l-2)(l-3)\sigma^4} \tag{6}$$

With the extracted features, we construct the local feature vector (LFV) to approximate the subsequence.

Definition 5. Local feature vector. Given the subsequence set $\boldsymbol{S} = \{S_1, S_2, \ldots, S_i, \ldots, S_m\}$ of time series T and $\boldsymbol{V} = [v_1, v_2, \ldots, v_n] \in \boldsymbol{R}^n$, if $\exists f : S_i \rightarrow \boldsymbol{V}$, then $\boldsymbol{V}$ is called local feature vector, denoted as LFV.

Theoretically, the dimension of LFV is not limited. In fact, with the time series from the different application fields, the practical features extracted in PSA are data-reliable. For instance, as the vital signs samples in Fig. 1, the time series are periodic with low frequency and has the bald movement pattern with less noise. Thereby, the linear fitting slope of the subsequence is more effective than the other features for describing the fluctuation characteristics. While in the stock market, the investors are more interested in the open price, the high price, the low price, and the close price within a period, thus we only need to extract the first value, the last value, the maximum, and the minimum of a subsequence to construct LFVs.

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

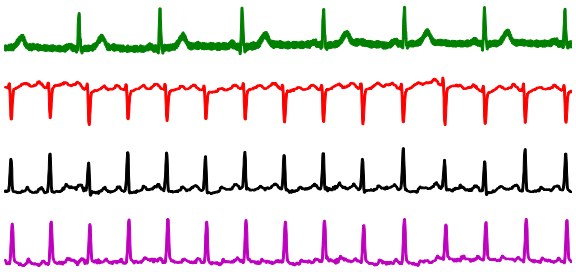


Fig. 1. Vital signs samples.

Fig. 2 shows the examples of (a) PAA, (b) PLA, (c) DSA, and (d) PSA representations for the stock price time series of Google Inc. (symbol: GOOG) from The NASDAQ Stock Market, which consists of the close prices at 800 consecutive trading days (2010/10/4–2013/12/5). As shown in Fig. 2(a), PAA extracts the mean values of the subsequences with equal-length as features. In Fig. 2(b), PLA takes the linear fitting slopes and the spans of the subsequences with adaptive-length as features, e.g., [0.5, 96] for the first subsequence. In Fig. 2(c), DSA takes the mean values of the derivative subsequences with adaptive-length as features. The above subplot shows the derivative series of the raw data, and the below one shows the DSA approximation. In Fig. 2(d), PSA extracts all the features in the former three methods to construct *LFV*s to approximate the time series. For the clear exhibition, the above subplot only shows the approximation with the mean values and the linear fitting slopes of the subsequences, e.g., [596.5, 0.5] for the first subsequence, and the below subplot shows the approximation with the DSA features. Different from PLA and DSA, PSA extracts the features from the equal-length subsequences. As shown in Fig. 2(c), the data-adaptive DSA approximation shows the poor performance on dimension reduction for the time series with high frequency, while PSA could overcome this drawback.

Algorithm 1. Piecewise statistic approximation

Input: $T = \{t_1, t_2, \ldots, t_i, \ldots, t_n\}$, $F = \{f_1, f_2, \ldots, f_m\}$, $l \in N^*$;
Output: *PSA*;
1: $T' \leftarrow z\text{-}norm(T)$;
2: $w \leftarrow n/l$;
3: $PSA \leftarrow \varnothing$;
4: **for** $i = 1{:}w$
5: $S \leftarrow T'[(i-1)\cdot l + 1 : i \cdot l]$;
6: $LFV \leftarrow \emptyset$;
7: **for** $j = 1{:}m$
8: $LFV \leftarrow [LFV; f_j(S)]$;
9: **end for**
10: $PSA \leftarrow \{PSA; LFV\}$;
11: **end for**
12: **Return** *PSA*;

Algorithm 1 shows the procedure of representing time series with PSA. The inputs are the time series T and the feature set F, whose elements are the designated features to be extracted from T, as well as the disjoint window length l used for segmentation. The output is the PSA representation of T. The inner loop (lines 7–9) extracts m statistical features from each subsequence to construct the *LFV*. In fact, if the extracted features are limited to Formulas (1)–(6), only the mean value μ and the difference $s_i - \mu$ are necessary for computing all features. Thus, in the inner loop,

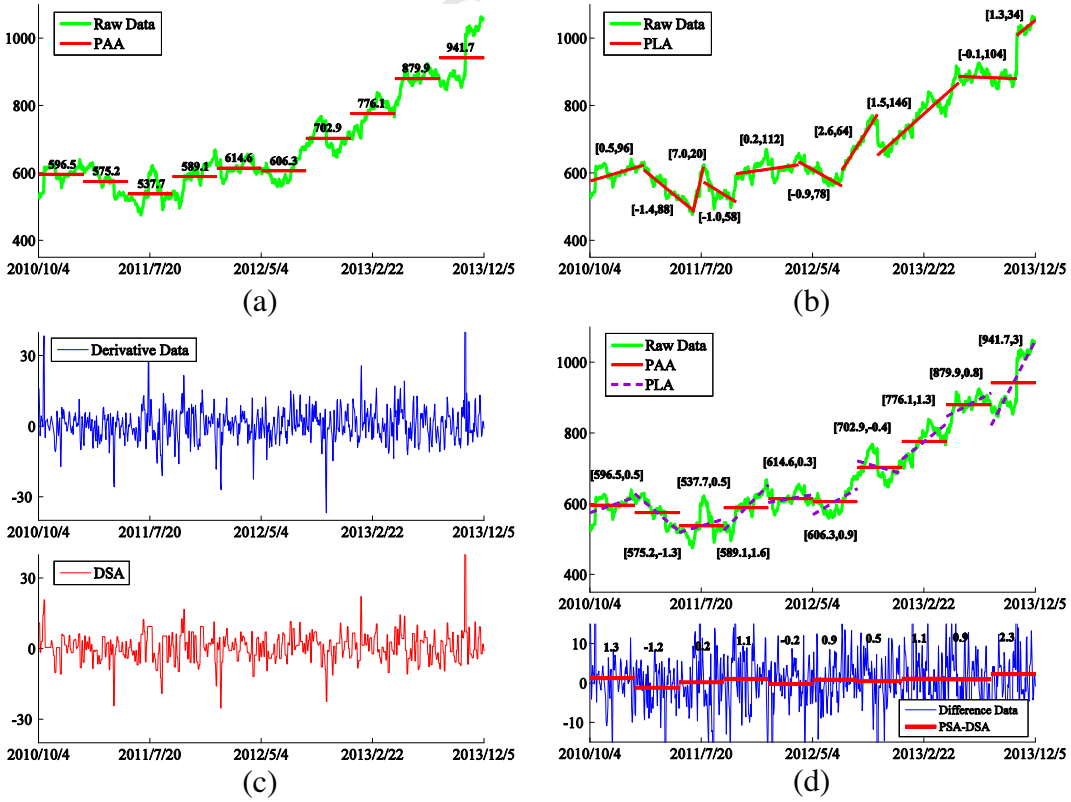


Fig. 2. PAA/PLA/DSA/PSA representation examples.

each subsequence only needs to be scanned twice: computing the mean value μ of the subsequence for the first time and the difference $s_i - \mu$ for the second time. If the *LFV* is constructed with the features of Formulas (1)–(6), the computational complexity is $O(n)$.

4. Piecewise statistic approximation based similarity measure

As mentioned above, DTW is a prevalent elastic measure that exploits the one-to-many aligning scheme on time series. Overall, it tries to find the optimal alignment between time series. Given a sample space F, time series $T = \{t_1, t_2, \ldots, t_i, \ldots, t_m\}$ of length m, and time series $Q = \{q_1, q_2, \ldots, q_i, \ldots, q_n\}$ of length $n, t_i, q_j \in F$, a local distance measure d should be first set in DTW for measuring two samples t_i, q_j, defined as d: $F \times F \to \boldsymbol{R} \geqslant 0$. Then, a distance matrix $C \in \boldsymbol{R}^{m \times n}$ is computed by evaluating the distance between each pair of points (t_i, q_j), i.e., $C(i,j) = d(t_i, q_j)$. There is an optimal warping path ξ in C, which has the minimal total distance D_ξ.

Definition 6. Warping path. Given the distance matrix C, if the sequence $p = \{p_1, \ldots, p_l, \ldots, p_L\}$, where $p_l = (a_l, b_l) \in [1{:}\ n] \times [1{:}\ m]$ for $l \in [1{:}\ L]$, satisfies the following conditions:

(1) Boundary: $p_1 = (1, 1)$ and $p_L = (m, n)$;
(2) Continuity: $p_{l+1} - p_l \in \{(1, 0), (0, 1), (1, 1)\}$ for $l \in [1{:}\ L\text{-}1]$;
(3) Monotonicity: $a_1 \leqslant a_2 \leqslant \ldots \leqslant a_L$ and $b_1 \leqslant b_2 \leqslant \cdots \leqslant b_L$, then p is called warping path. The total distance D_p of the warping path p is defined as Formula (7).

$$D_p = \sum_{l=1}^{L} C(a_l, b_l) \tag{7}$$

The minimal total distance D_ξ is eventually defined as the DTW distance between T and Q. The entire computation is implemented by the dynamic programming algorithm, which can lead to the quadratic computational complexity to the time series length. Fig. 3 respectively shows the dynamic programming table with the optimal warping path of computing DTW, and the one-to-many aligning scheme of DTW.

In order to reduce the high computational complexity of DTW, PR-DTW has been proposed for measuring the similarity between time series, which takes the subsequence matching as the subroutine in the dynamic programming computation. Rather than on the raw data, PR-DTW is computed on the extracted features in the piecewise representation. Since the features can reflect the characteristics of the subsequences, and capture the fluctuation information from a higher level than the raw data, PR-DTW has the strong interpretability. The computational complexity of PR-DTW is $O(MN)$, where M and N are the segment numbers of two time series and much less than the original time series length. Fig. 4 shows the dynamic programming table with the optimal warping path of computing PR-DTW, and the one-to-many aligning scheme based on the subsequence matching.

For the piecewise representation based similarity measure, the phase-shift between the subsequences caused by the time scaling is a global factor of influencing the measure precision, which cannot be easily captured by the features of the subsequences. As shown in Fig. 5, the two real time series can be both segmented into four subsequences with the similar movements. The subsequence pair S_4 can be aligned well, but from right to left, the phase-shift between the subsequence pair is more and more prominent. Under the dynamic programming computation, PR-DTW can well address this problem.

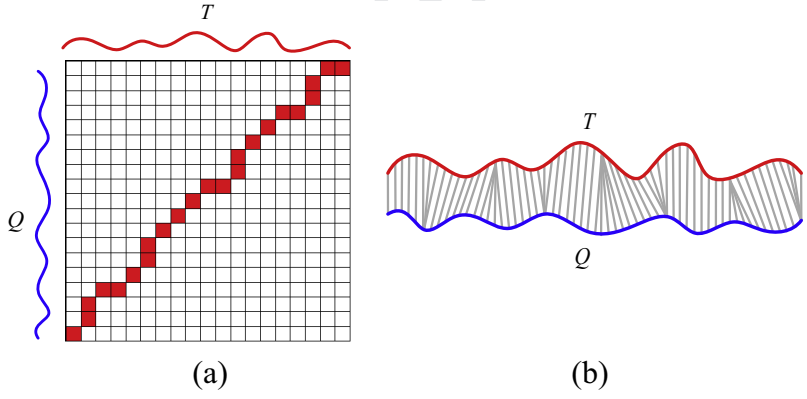


Fig. 3. (a) The dynamic programming table with the optimal warping path of computing DTW (b) The one-to-many aligning scheme of DTW.

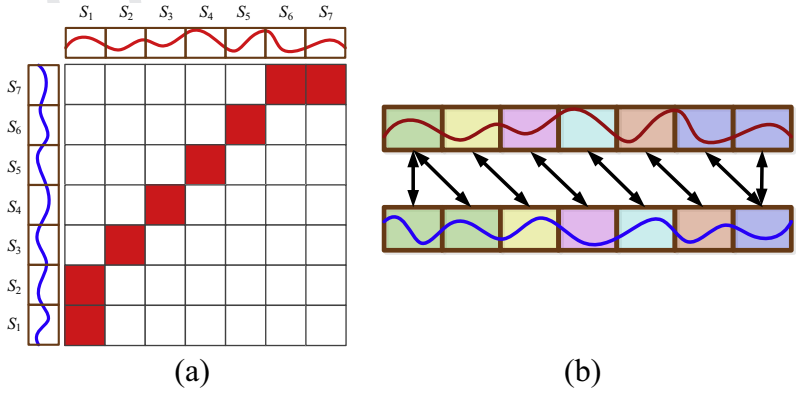


Fig. 4. (a) The dynamic programming table with the optimal warping path for computing PR-DTW (b) The one-to-many aligning scheme of PR-DTW.

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

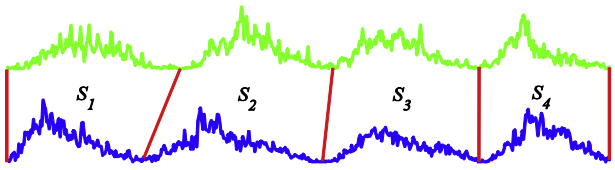


Fig. 5. The phase-shift between the subsequences.

However, as a prerequisite, the subsequences should be matched accurately based on the extracted features in the subroutine of PR-DTW. Thus, (1) one crucial factor of determining the precision of PR-DTW is the extracted features, which is also one of the main differences between PSADTW and the other PR-DTWs. However, in practice, some feature values may dominate the others, and the expressivities of the features are different from each other. In view of this, we highlight some features and restrain the others in PSADTW. Concretely, we adopt the weighted Euclidean distance for the subsequence matching, where the weights [22] can be learned to reflect the expressivities of the features. (2) This not only exhibits the interpretability of PSADTW, but also is another different point between PSADTW and the other PR-DTWs.

Overall, the framework of our methodology is shown in Fig. 6, which is composed of three parts:

(1) PSA Representation, where the original time series is segmented and approximated with multiple statistic features.
(2) Parameter Training, where the weights are learned from the labelled dataset for the subsequence matching.
(3) PSADTW Computing, where the dynamic programming computation is performed based on the subsequence matching to get the measure result.

4.1. Local distance measure

Based on the PSA representation, the subsequences of time series are projected into the statistical feature space and represented in the form of *LFV*. The subsequence matching based on PSA is in essence to measure the distance between *LFV*s. In order to distinguish the expressivities of the features, we adopt the weighted Euclidean distance for the subsequence matching. Besides, the weighted Euclidean distance can make PSADTW applicable in the feedback learning to boost the measure precision. The subsequence matching is performed as Formula (8).

$$dist\{LFV[T(S_i)], LFV[Q(S_j)]\} = \sqrt{\sum_{k=1}^{n} a_k \cdot (v_k - v'_k)^2} \quad (8)$$

where $T(S_i)$ denotes the subsequence S_i of time series T and $LFV[T(S_i)]$ denotes the *LFV* of $T(S_i)$, the same meaning is taken for $Q(S_j)$ and $LFV[Q(S_j)]$; v_k and v'_k are the k-th elements of $LFV[T(S_i)]$ and $LFV[Q(S_j)]$ respectively; a_k is the weight of the k-th element of *LFV*.

Proposition 1. *Formula (8) satisfies the triangle inequality.*

Proof. Given that $LFV[T(S_i)] = [v_1, v_2, \ldots, v_n]$, $LFV[Q(S_j)] = [v'_1, v'_2, \ldots, v'_n]$, $LFV[R(S_t)] = [v''_1, v''_2, \ldots, v''_n]$, then

$$\begin{aligned} dist\{LPV[T(S_i)], LPV[Q(S_j)]\} &= \sqrt{\sum_{k=1}^{n} a_k \cdot (v_k - v'_k)^2} \\ &= \sqrt{\sum_{k=1}^{n} (\sqrt{a_k} \cdot v_k - \sqrt{a_k} \cdot v'_k)^2} \end{aligned}$$

Set $LPV' = \{\sqrt{a_1}v_1, \sqrt{a_2}v_2, \ldots, \sqrt{a_n}v_n\} = \{\mu_1, \mu_2, \ldots, \mu_n\}$, then

$$\begin{aligned} dist\{LPV[T(S_i)], LPV[Q(S_j)]\} &= dist\{LPV'[T(S_i)], LPV'[Q(S_j)]\} \\ &= \sqrt{\sum_{k=1}^{n} (\mu_k - \mu'_k)^2} \\ &= \|LPV'[T(S_i)] - LPV'[Q(S_j)]\| \end{aligned}$$

According to the Cauchy–Schwarz inequality, we have

$$|LPV'[T(S_i)] \cdot LPV'[R(S_t)]|^2 \leqslant \|LPV'[T(S_i)]\|^2 \cdot \|LPV'[R(S_t)]\|^2$$

Namely,

$$\begin{aligned} (\mu_1\mu''_1 + \mu_2\mu''_2 + \cdots + \mu_n\mu''_n)^2 \leqslant\ & (\mu_1^2 + \mu_2^2 + \cdots + \mu_n^2) \\ & \cdot (\mu_1'^2 + \mu_2'^2 + \cdots + \mu_n'^2) \end{aligned}$$

by which we have

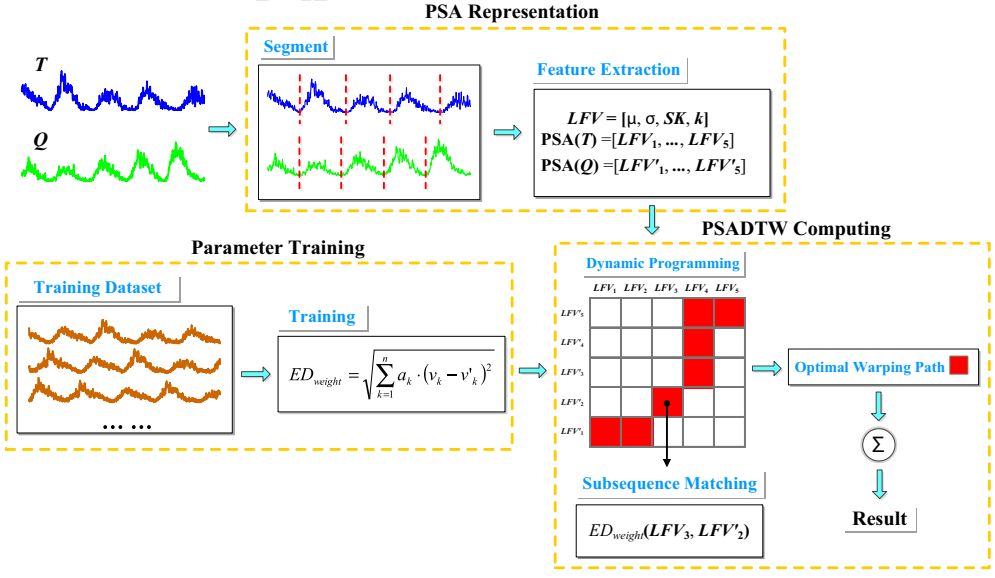


Fig. 6. The methodology framework of PSADTW.

$$\begin{aligned}\|LPV'[T(S_i)] + LPV'[R(S_t)]\| &\leqslant \|LPV'[T(S_i)]\| + \|LPV'[R(S_t)]\| \\ &\Rightarrow \|LPV'[T(S_i)] - LPV'[R(S_t)]\| \\ &= \|(LPV'[T(S_i)] - LPV'[Q(S_j)]) \\ &\quad + (LPV'[Q(S_j)] - LPV'[R(S_t)])\| \\ &\leqslant \|LPV'[T(S_i)] - LPV'[Q(S_j)]\| \\ &\quad + \|LPV'[Q(S_j)] - LPV'[R(S_t)]\|\end{aligned}$$

Namely,

$$\begin{aligned}dist\{LPV[T(S_i)], LPV[R(S_t)]\} &\leqslant dist\{LPV[T(S_i)], LPV[Q(S_j)]\} \\ &\quad + dist\{LPV[Q(S_j)], LPV[R(S_t)]\}\end{aligned}$$

∴ Formula (8) satisfies the triangle inequality.

4.2. Dynamic programming computation

Given that under the disjoint window with length l, time series T with length m is segmented into M subsequences, and time series Q with length n is segmented into N subsequences, PSADTW can be computed as Formula (9).

$$PSADTW(T,Q) = \begin{cases} 0, & \text{if } m = n = 0 \\ \infty, & \text{if } m = 0 \text{ or } n = 0 \\ dist\{LFV[T(S_1)], LFV[Q(S_1)]\} & \\ + \min\begin{cases} PSADTW\{rest[PSA(T)], PSA(Q)\}, \\ PSADTW\{PSA(T), rest[PSA(Q)]\}, \\ PSADTW\{rest[PSA(T)], rest[PSA(Q)]\} \end{cases}, & \\ otherwise & \end{cases} \quad (9)$$

where $dist\{LFV[T(S_1), LFV[Q(S_1)]]\}$ is computed as Formula (8), $PSA(T) = \{LFV_1, LFV_2, \ldots, LFV_i, \ldots, LFV_M\}$, $LFV_i = LFV[T(S_i)]$, $rest[PSA(T)] = \{LFV_2, LFV_3, \ldots, LFV_M\}$, and the same meaning is taken for $PSA(Q)$ and $rest[PSA(Q)]$.

Algorithm 2 shows the procedure of PSADTW computation. The inputs are the PSA representations of T and Q, and the output is their PSADTW distance. Actually, the computation is a dynamic programming procedure. The subsequence matching is performed as Formula (8) (line 6). If the dimensionality of LFV is λ, the entire computational complexity is $O(\lambda MN)$. Compared with the original DTW, the computational complexity speedup gained by PSADTW is $O(mn)/O(\lambda MN)$, namely $O(l^2/\lambda)$, where λ is much less than l.

Algorithm 2. PSADTW distance

Input: $PSA(T) = \{LFV_1, LFV_2, \ldots, LFV_M\}$,
$PSA(Q) = \{LFV'_1, LFV'_2, \ldots, LFV'_N\}$,
$W = \{a_1, a_2, \ldots, a_n\}$;
Output: $PSADTW(T,Q)$;
1: $table \leftarrow zeros(M,N)$;
2: $table(1,:) \leftarrow dist\{LFV_1, LFV'_{1\sim N}\}$;
3: $table(:,1) \leftarrow dist\{LFV_{1\sim M}, LFV'_1\}$;
4: **for** $i = 2{:}M$
5: **for** $j = 2{:}N$
6: $table(i,j) = dist\{LFV_i, LFV'_j\}+$
$\min\{table(i\text{-}1, j), table(i, j-1), table(i\text{-}1, j\text{-}1)\}$;
7: **end**
8: **end**
9: $PSADTW(T,Q) = table(M,N)$;
10: **Return** $PSADTW(T,Q)$;

However, the "singularities" [33] may still exist in PSADTW, as shown in Fig. 7(a). This problem can be addressed by the constraint window (e.g., Sakoe-Chiba Band) that limits the subsequence matching to a small range, as shown in Figs. 7(b) and 8.

5. Experiments

In the experiments, we will firstly evaluate the effect of the extracted multiple features in PSA for similarity measure. Then, the effect of the weighted Euclidean distance introduced for PSADTW is examined. Thirdly, by the comprehensive experiments, we evaluate the precision improvement of PSADTW over the existing PR-DTW measures, as well as the computational speedup gained by PSADTW over the original DTW.

We use 45 real-world datasets provided by the UCR time series archive [38] in the experiments. All the datasets have been z-normalized and partitioned into training and testing sets by the provider. Appendix A shows the sample instances of all datasets.

In order to evaluate the precision of the similarity measure, the renowned 1NN classifier is used, where the nearest neighbor o' is searched under the similarity measure for the unlabeled object o, and the label of o' is taken as the predicted label of o. The final classification accuracy can significantly indicate the precision of the similarity measure [24,29,31].

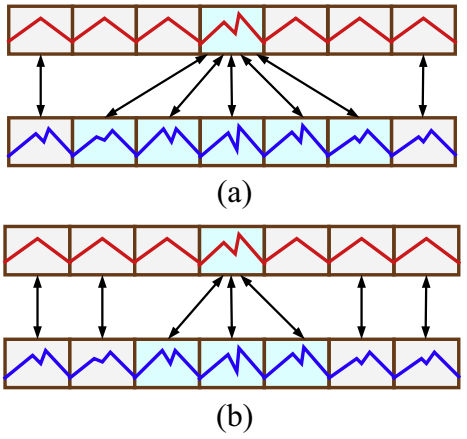


Fig. 7. (a) The "singularities" in PSADTW (b) The constraint effect of Sakoe-Chiba Band with width = 1.

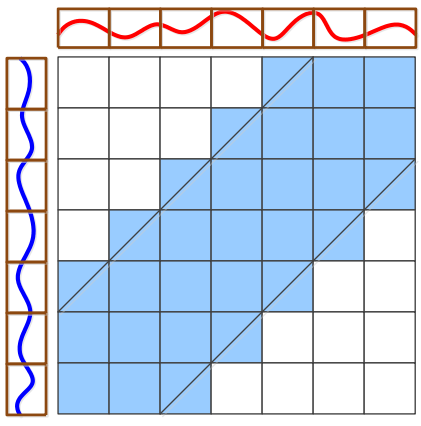


Fig. 8. The computation of PSADTW with Sakoe-Chiba Band.

Besides, all parameters in the measure methods will be learned by the DIRECT global optimization algorithm [27,39], which is used to seek for the global minimum of the multivariate function with a constraint domain. For PSADTW, the weights $[a_1, a_2, \ldots, a_n]$ for the multiple features must satisfy the constraint condition $a_1 + a_2 + \ldots + a_n = 1$.

The experiment environment is Intel(R) Core(TM) i5-2400 CPU @ 3.10 GHz; 8G Memory; Windows 7 64-bit OS; MATLAB 8.0_R2012b.

5.1. Feature selection

With the extensive datasets from the various application fields, we will construct the *LFVs* for PSA with the general statistical features listed in Section 3. Since some of the features have the same expressivity for the data fluctuation, e.g., σ and D, SK and K, it may be needless to extract all features. In order to examine this assumption, we compare the precision of the PSADTWs respectively based on $\{\mu, \sigma, SK\}$ and $\{\mu, \sigma, D, CV, SK, K\}$, which are called $PSADTW_3$ and $PSADTW_6$. The Euclidean distance is used for matching the *LFVs* in both measures.

The experiment results are visually summarized in Fig. 9. In the scatter plot, the x-axis and y-axis represent $PSADTW_6$ and $PSADTW_3$ respectively. Each dot represents a particular dataset, whose coordinate is determined by the accuracy of both measures on this dataset. The dots on the diagonal line indicate that both measures have the equal accuracy, while the dots above the diagonal mean that the measure of y-axis is more accurate than that of x-axis, and vice versa. It can be seen that the precision of $PSADTW_3$ is considerably higher than $PSADTW_6$ over a large number of datasets. The results indicate that it is needless to extract the features with the same expressivities. The intrinsic reason is that, since some features have stronger expressivity than the others, the contributions of the former will be diluted by the latter, so that the precision of PSADTW decreases. Hereafter, we take use of the features $\{\mu, \sigma, SK\}$ in PSADTW.

5.2. Evaluation for subsequence matching

In PSADTW, we adopt the weighted Euclidean distance for the local measure, where the weights are learned from the labelled dataset. We assume that the introduction of the weight mechanism can further improve the precision of PSADTW. In this experiment, we compare the precision of PSADTWs respectively based on the Euclidean and the weighted Euclidean distances. Fig. 10 shows the comparison between the results.

In Fig. 10, it can be seen that PSADTW with the weights has higher precision than that without weights over the most datasets.

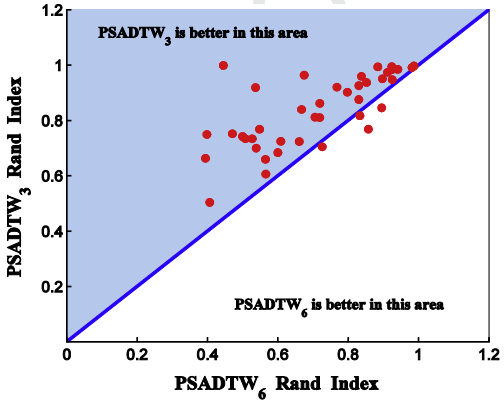


Fig. 9. The precision comparison between $PSADTW_3$ and $PSADTW_6$.

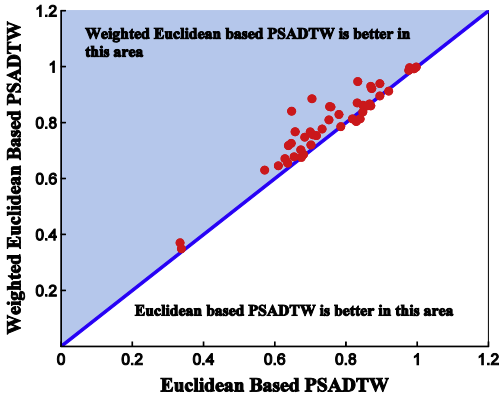


Fig. 10. Precision comparison between PSADTWs based on the Euclidean and the weighted Euclidean distances.

The reason is apparent that, the weights can highlight the features with the strong expressivity but restrain the others. Such discrimination makes PSADTW sensitive to the effective information but robust to the useless information.

5.3. Comprehensive evaluation

In this section, we conduct the comprehensive evaluation for PSADTW on both the precision and efficiency. Since PSA extracts more features for similarity measure than existing piecewise representation methods, it is assumed that PSADTW would have higher precision than the other PR-DTWs. Meanwhile, the significant efficiency superiority of PR-DTW over the original DTW would be retained in PSADTW.

In order to validate the performance of PSADTW, besides the original DTW and the state-of-the-art CIDDTW, we take four PR-DTWs as baselines, i.e., PDTW, SDTW, APCADTW, and DSADTW, which are based on PAA, PLA, APCA, and DSA, respectively. All the seven measures are shown in Table 1.

5.3.1. Precision

We first conduct the 1NN classification to evaluate the precision of PSADTW. Table 2 shows the accuracy of the 1NN classifiers respectively based on the seven measures over all datasets. The best accuracy of five PR-DTWs on each dataset is highlighted in bold, and two statistical test results are shown in the last two rows.

Firstly, the established sign test [28,40] is used to evaluate the absolute total wins for each PR-DTW. This test assigns each dataset an absolute wins score 1 and averages the score to the measure(s) with the best accuracy on the dataset. By accumulating the scores of each measure over all datasets, we can get the absolute total wins for each measure, as shown in the "absolute total wins" row. In terms of this figure, PSADTW overwhelmingly beats the four PR-DTW baselines.

Secondly, we compare PSADTW with each PR-DTW baseline by the Wilcoxon signed-ranks test [7,8,40], which is regarded as a

Table 1
Seven DTW measures compared in the experiments.

	Method	Description
1	DTW	The original DTW on the raw data
2	CIDDTW	The DTW complied with the complexity factor
3	PDTW	The DTW distance on the PAA representation
4	SDTW	The DTW distance on the PLA representation
5	APCADTW	The DTW distance on the APCA representation
6	DSADTW	The DTW distance on the DSA representation
7	PSADTW	The DTW distance on the PSA representation

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

Table 2
The 1NN classification accuracy and the statistical test results.

Dataset	Class	Length	Size	PSAD.	PDTW	SDTW	APCAD.	DSAD.	DTW	CIDD.
50Words	50	270	905	0.69	**0.71**	0.65	0.62	0.63	0.69	0.68
Adiac	37	176	781	**0.68**	0.61	0.34	0.28	0.38	0.60	0.61
Beef	5	470	60	**0.77**	0.50	0.57	0.57	0.47	0.50	0.50
CBF	3	128	930	**1.00**	0.98	0.95	0.91	0.50	1.00	1.00
Chlorine	3	166	4307	**0.68**	0.60	0.55	0.56	0.62	0.65	0.64
CinC	4	1639	1420	**0.72**	0.65	0.63	0.61	0.63	0.65	0.70
Coffee	2	286	56	0.79	0.79	0.75	**0.82**	0.61	0.82	0.82
Cricket_X	12	300	780	0.72	0.75	**0.76**	**0.76**	0.36	0.78	0.74
Cricket_Y	12	300	780	0.78	**0.79**	0.74	0.78	0.33	0.79	0.74
Cricket_Z	12	300	780	0.76	0.78	0.78	**0.79**	0.31	0.79	0.75
Diatom	4	345	322	0.92	0.92	**0.96**	**0.96**	0.88	0.97	0.96
ECG200	2	96	200	**0.86**	0.80	0.83	0.77	0.81	0.77	0.81
ECGFiveDays	2	136	884	**0.81**	0.79	0.68	0.68	0.57	0.77	0.76
FaceALL	14	131	2250	**0.75**	0.63	0.50	0.63	0.71	0.81	0.85
FaceFour	4	350	112	**0.84**	0.77	0.75	0.76	0.32	0.83	0.73
FacesUCR	14	131	2250	**0.86**	0.60	0.57	0.72	0.70	0.90	0.83
Fish	7	463	350	0.86	0.75	0.73	0.60	**0.89**	0.83	0.79
Gun_Point	2	150	200	**0.95**	0.90	0.87	0.89	**0.95**	0.91	0.88
Haptics	5	1092	463	**0.37**	**0.37**	0.35	**0.37**	0.30	0.38	0.38
InlineSkate	7	1882	650	0.35	0.35	0.38	0.35	**0.40**	0.38	0.40
Italy	2	24	1096	0.86	**0.93**	0.80	0.90	0.87	0.95	0.92
Lightning2	2	637	121	0.89	0.87	**0.90**	**0.90**	0.67	0.87	0.85
Lightning7	7	319	143	**0.77**	0.68	0.70	0.70	0.51	0.73	0.60
MALLAT	8	1024	2400	0.91	0.86	0.89	**0.95**	0.85	0.93	0.94
MedicalImages	10	99	1141	0.67	0.67	0.70	**0.73**	0.59	0.74	0.72
MoteStrain	2	84	1272	**0.84**	0.79	0.78	0.83	0.65	0.84	0.77
NonInvasive.1	42	750	3765	**0.83**	0.79	0.58	0.72	0.60	0.79	0.78
NonInvasive.2	42	750	3765	**0.86**	**0.86**	0.74	0.80	0.77	0.87	0.86
OliveOil	4	570	60	**0.87**	**0.87**	0.57	0.77	0.37	0.87	0.83
OSULeaf	6	427	442	0.70	0.55	0.61	0.56	**0.77**	0.59	0.63
SonyAI	2	70	621	0.75	**0.76**	0.73	**0.76**	0.70	0.73	0.81
SonyAI.II	2	65	980	0.81	**0.85**	0.75	0.71	0.78	0.83	0.90
StarLight	3	1024	9236	0.94	0.89	0.88	0.89	**0.96**	0.91	0.93
SwedishLeaf	15	128	1125	**0.87**	0.74	0.62	0.56	0.80	0.79	0.82
Symbols	6	398	1020	0.90	0.90	0.91	0.92	**0.94**	0.95	0.94
Synthetic	6	60	600	**0.99**	0.97	0.92	0.95	0.49	0.99	0.96
Trace	4	275	200	0.93	0.99	**1.00**	**1.00**	**1.00**	1.00	1.00
TwoLeadECG	2	82	1162	0.81	0.72	0.80	0.66	**0.91**	0.90	0.90
TwoPatterns	4	128	5000	**1.00**	**1.00**	0.99	**1.00**	0.97	1.00	1.00
UWave.X	8	315	4478	0.73	**0.74**	**0.74**	**0.74**	0.58	0.73	0.73
UWave.Y	8	315	4478	0.63	**0.67**	0.64	0.64	0.43	0.63	0.63
UWave.Z	8	315	4478	0.65	**0.66**	**0.66**	0.65	0.46	0.66	0.66
Wafer	2	152	7164	**0.99**	**0.99**	0.98	0.97	0.94	0.98	0.98
Words	25	270	905	0.66	**0.68**	0.63	0.60	0.64	0.65	0.65
Yoga	2	426	3300	0.80	**0.85**	**0.85**	0.81	0.70	0.84	0.83
Absolute total wins				17.67	10	3.17	7.33	6.83		
Wilcoxon signed rank (z)					−2.49	−4.16	−3.47	−4.75		

safe, robust, and powerful test for the statistical comparison between two classifiers. In this test, when the statistic $z < -1.96$, the null-hypothesis is rejected with 95% confidence ($p \leqslant 0.05$). As the test results shown in Table 2, PSADTW is superior to the four PR-DTW baselines with the statistical significance.

Fig. 11 shows the one-to-one comparison between PSADTW and all baselines. We can clearly find that, the precision of PSADTW is significantly superior to the four PR-DTW baselines in a large range of application fields, and even achieves the comparative precision with the original DTW and the state-of-the-art CIDDTW.

The reason to the above results is that, the features $\{\mu, \sigma, SK\}$ used in PSADTW contain the feature u used in PDTW and APCADTW, thus PSADTW can synthesize more information for similarity measure than the latter two measures. Besides, we find that the datasets, where PSADTW wins over SDTW and DSADTW, are surprisingly similar, e.g., Beef, CricketX, CricketY, CricketZ, FaceFour, Lightning2, Medical., Synthetic., and uWaveY. The reason may be that the features used in SDTW and DSADTW can capture the same information for similarity measure, which is sensitive to the noise and thus influences the precision of SDTW and DSADTW.

5.3.2. Efficiency

The entire consumption of PR-DTW consists of two parts: data representation and distance computation. The computational complexities of five PR-DTW measures are shown in Table 3.

Although the multiple features extracted for similarity measure can increase the precision of PR-DTW, the efficiency of data representation may be influenced. However, compared with the single type of features, the strong expressivity of the multiple features could make time series be segmented into longer subsequences (less number of subsequences), which can raise the efficiency of distance computation in PR-DTW. In view of this, we compare PSADTW with the other PR-DTWs, in terms of the data representation efficiency and the distance computation efficiency.

As shown in Table 4, we present the average runtime of data representation for the different datasets in the 1NN classification. In order for clear comparison, all datasets are partitioned into three adaptive ranges in terms of time series length, i.e., 24–180, 270–470, 570–1882, and the minimum, the median, and the maximum within each range are picked out for comparison, as shown in Table 5.

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

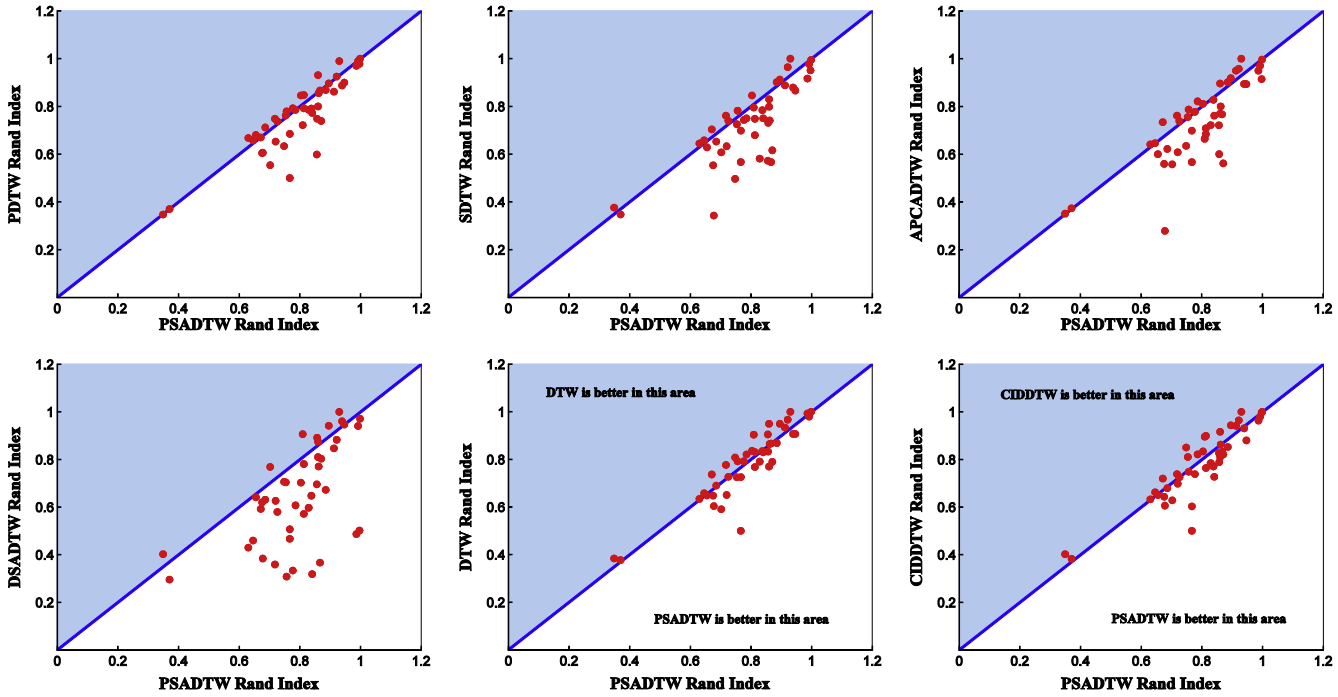


Fig. 11. The one-to-one comparison between PSADTW and six baselines.

Table 3
The computational complexities of five PR-DTW measures.

Method	Data representation	Distance computation
PDTW	$O(n)$	$O(w^2)$
SDTW	$O(n \log n)$	$O(w^2)$
APCADTW	$O(n \log n)$	$O(w^2)$
DSADTW	$O(n)$	$O(w^2)$
PSADTW	$O(n)$	$O(w^2)$

In Table 5, we observe that within all ranges, (1) PSADTW has the comparative data representation efficiency with PDTW. This result conforms to the theoretical analysis on Algorithm 1 that, in PSA representation, only the mean value μ and the difference $s_i - \mu$ are necessary for computing all features, and thus the computational complexity is $O(n)$. (2) The data representation efficiency of PSADTW is higher (up to two orders of magnitude) than the state-of-the-art DSADTW. This is because DSA is a

Table 4
The average runtime of data representation in the 1NN classification (ms).

Dataset	Length	PSADTW	PDTW	SDTW	APCAD.	DSADTW
50Words	270	0.09	0.07	165.22	133.04	5.84
Adiac	176	0.04	0.04	118.27	83.51	4.07
Beef	470	0.27	0.07	337.71	279.82	10.75
CBF	128	0.03	0.03	77.68	74.84	3.06
Chlorine	166	0.04	0.04	104.62	84.39	3.83
CinC	1639	0.34	0.11	1257.60	1175.81	37.46
Coffee	286	0.09	0.06	194.19	162.86	6.45
Cricket_X	300	0.08	0.08	189.36	150.10	7.10
Cricket_Y	300	0.08	0.06	185.16	145.25	7.00
Cricket_Z	300	0.08	0.08	186.76	146.28	7.08
Diatom	345	0.09	0.05	229.25	162.86	7.88
ECG200	96	0.02	0.02	59.12	45.19	2.23
ECGFiveDays	136	0.03	0.01	91.70	86.26	3.23
FaceALL	131	0.03	0.03	78.46	60.58	2.94
FaceFour	350	0.15	0.01	217.99	171.92	7.95
FacesUCR	131	0.03	0.03	78.43	61.50	2.93
Fish	463	0.10	0.02	291.34	233.39	10.92
Gun_Point	150	0.04	0.03	101.43	69.87	3.36
Haptics	1092	0.26	0.11	782.80	723.32	26.97
InlineSkate	1882	0.37	0.08	1455.38	1414.59	43.70
Italy	24	0.01	0.01	13.36	10.34	0.51
Lightning2	637	0.19	0.10	424.04	349.08	15.92
Lightning7	319	0.09	0.05	196.27	151.88	8.07
MALLAT	1024	0.21	0.09	754.80	658.78	21.64
MedicalImages	99	0.02	0.03	59.89	45.51	2.18
MoteStrain	84	0.02	0.02	56.15	46.81	2.03
NonInvasive.1	750	0.16	0.14	515.46	433.85	16.91
NonInvasive.2	750	0.16	0.13	507.46	454.11	16.91
OliveOil	570	0.32	0.13	363.31	313.20	12.48

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

Table 4 (*continued*)

Dataset	Length	PSADTW	PDTW	SDTW	APCAD.	DSADTW
OSULeaf	427	0.09	0.07	266.75	210.30	9.96
SonyAI	70	0.02	0.01	44.99	39.66	1.58
SonyAI.II	65	0.02	0.01	38.20	38.16	1.46
StarLight	1024	0.24	0.14	765.62	651.61	21.59
SwedishLeaf	128	0.03	0.01	86.30	60.51	2.91
Symbols	398	0.08	0.02	271.14	204.95	9.41
Synthetic	60	0.02	0.01	35.83	31.13	1.35
Trace	275	0.10	0.06	173.94	169.57	6.38
TwoLeadECG	82	0.02	0.02	52.54	48.18	1.84
TwoPatterns	128	0.03	0.03	82.60	75.05	3.13
UWave.X	315	0.07	0.08	214.75	148.18	7.67
UWave.Y	315	0.07	0.08	195.25	148.15	7.80
UWave.Z	315	0.07	0.08	210.16	152.02	7.77
Wafer	152	0.03	0.01	93.77	70.51	4.17
Words	270	0.07	0.07	167.56	130.33	5.68
Yoga	426	0.09	0.11	264.60	228.29	9.82

Table 5
Comparison for the data representation efficiency of five measures (ms).

PR-DTW	24–180			270–470			570–1882		
	Min	Med	Max	Min	Med	Max	Min	Med	Max
PSADTW	0.01	0.03	0.04	0.07	0.09	0.27	0.16	0.24	0.37
PDTW	0.01	0.02	0.04	0.01	0.07	0.11	0.08	0.11	0.14
DSADTW	0.51	2.92	4.17	5.68	7.78	10.92	12.48	21.59	43.70
SDTW	13.36	78.05	118.27	165.22	203.21	337.71	363.31	754.80	1455.38
APCADTW	10.34	60.54	86.26	130.33	157.44	279.82	313.20	651.61	1414.59

Table 6
The segment numbers and the DCRs of five PR-DTWs over all datasets.

Dataset	Length	PSADTW		PDTW		SDTW		APCADTW		DSADTW	
		Seg.	DCR	Seg.	DCR	Seg.	DCR	Seg.	DCR	Seg.	DCR
50Words	270	49	5.51	66	4.09	57	4.74	63	4.29	125.37	2.15
Adiac	176	11	16.00	36	4.89	13	13.54	43	4.09	70.00	2.51
Beef	470	98	4.80	61	7.70	10	47.00	61	7.70	192.32	2.44
CBF	128	18	7.11	30	4.27	27	4.74	15	8.53	46.19	2.77
Chlorine	166	39	4.26	36	4.61	29	5.72	34	4.88	64.77	2.56
CinC	1639	66	24.83	103	15.91	94	17.44	84	19.51	655.49	2.50
Coffee	286	15	19.07	60	4.77	33	8.67	40	7.15	117.34	2.44
Cricket_X	300	63	4.76	72	4.17	65	4.62	66	4.55	109.31	2.74
Cricket_Y	300	74	4.05	57	5.26	73	4.11	71	4.23	112.96	2.66
Cricket_Z	300	70	4.29	72	4.17	70	4.29	70	4.29	110.03	2.73
Diatom	345	45	7.67	45	7.67	45	7.67	86	4.01	136.54	2.53
ECG200	96	5	19.20	14	6.86	19	5.05	23	4.17	35.84	2.68
ECGFiveDays	136	6	22.67	9	15.11	9	15.11	5	27.20	48.24	2.82
FaceALL	131	26	5.04	32	4.09	32	4.09	32	4.09	53.96	2.43
FaceFour	350	73	4.79	6	58.33	83	4.22	83	4.22	138.22	2.53
FacesUCR	131	9	14.56	24	5.46	32	4.09	31	4.23	54.43	2.41
Fish	463	27	17.15	17	27.24	106	4.37	103	4.50	169.86	2.73
Gun_Point	150	21	7.14	25	6.00	12	12.50	36	4.17	61.97	2.42
Haptics	1092	107	10.21	107	10.21	123	8.88	95	11.49	373.23	2.93
InlineSkate	1882	66	28.52	68	27.68	124	15.18	49	38.41	721.45	2.61
Italy	24	6	4.00	5	4.80	6	4.00	6	4.00	10.61	2.26
Lightning2	637	66	9.65	89	7.16	103	6.18	112	5.69	196.45	3.24
Lightning7	319	33	9.67	44	7.25	73	4.37	76	4.20	92.65	3.44
MALLAT	1024	38	26.95	84	12.19	66	15.52	103	9.94	487.19	2.10
MedicalImages	99	18	5.50	24	4.13	22	4.50	24	4.13	42.88	2.31
MoteStrain	84	13	6.46	17	4.94	6	14.00	11	7.64	28.01	3.00
NonInvasive.1	750	26	28.85	127	5.91	106	7.08	120	6.25	308.60	2.43
NonInvasive.2	750	25	30.00	127	5.91	121	6.20	99	7.58	308.93	2.43
OliveOil	570	107	5.33	120	4.75	123	4.63	101	5.64	251.50	2.27
OSULeaf	427	21	20.33	67	6.37	99	4.31	100	4.27	161.16	2.65
SonyAI	70	6	11.67	13	5.38	9	7.78	8	8.75	27.41	2.55
SonyAIII	65	13	5.00	12	5.42	15	4.33	4	16.25	25.84	2.52
StarLight	1024	121	8.46	127	8.06	48	21.33	112	9.14	488.73	2.10
SwedishLeaf	128	9	14.22	12	10.67	12	10.67	31	4.13	48.05	2.66
Symbols	398	20	19.90	19	20.95	52	7.65	83	4.80	134.62	2.96

(*continued on next page*)

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

Table 6 (*continued*)

Dataset	Length	PSADTW		PDTW		SDTW		APCADTW		DSADTW	
		Seg.	DCR	Seg.	DCR	Seg.	DCR	Seg.	DCR	Seg.	DCR
Synthetic	60	10	6.00	12	5.00	14	4.29	10	6.00	22.50	2.67
Trace	275	64	4.30	57	4.82	57	4.82	24	11.46	98.12	2.80
TwoLeadECG	82	19	4.32	19	4.32	12	6.83	7	11.71	31.39	2.61
TwoPatterns	128	21	6.10	31	4.13	21	6.10	14	9.14	37.69	3.40
UWaveX	315	19	16.58	72	4.38	34	9.26	77	4.09	95.91	3.28
UWaveY	315	77	4.09	77	4.09	72	4.38	77	4.09	91.04	3.46
UWaveZ	315	7	45.00	74	4.26	43	7.33	73	4.32	96.53	3.26
Wafer	152	29	5.24	5	30.40	32	4.75	37	4.11	27.55	5.52
Words	270	66	4.09	65	4.15	64	4.22	64	4.22	125.33	2.15
Yoga	426	21	20.29	102	4.18	105	4.06	80	5.33	161.30	2.64
DCR average rank			1.96		2.86		2.62		2.63		4.93

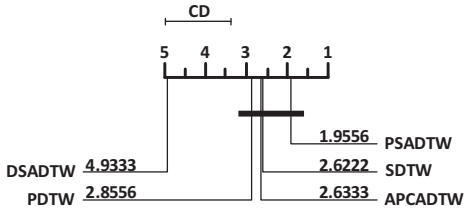


Fig. 12. The critical difference diagram for the DCRs of five PR-DTW measures.

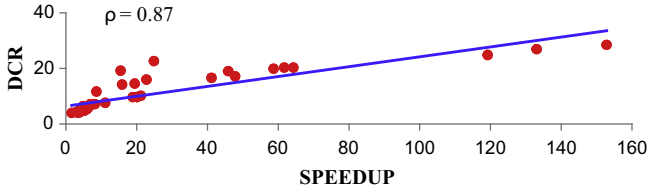


Fig. 13. The relationship between the speedup of PSADTW over the original DTW and the DCR of PSA over the raw data

data-adaptive representation method, which needs to compute the threshold (σ) to judge the segment points once a new sample comes. While PSA segments time series based on the trained parameter and the equal-length windows, thus it is more efficient than DSA. (3) The data representation efficiency of PSADTW is higher (even up to four orders of magnitude) than SDTW and APCADTW. This is also apparent that, PSA only scans time series once (or twice) to segment time series, while PLA and APCA need multiple-times scan for the global optimal segmentation.

By the above evaluation, we find the previous assumption that the multiple features extracted for similarity measure may influence the efficiency of data representation is not supported. PSA has significant superiority on the data representation efficiency over the other piecewise representation baselines. In the following, we will evaluate the distance computation efficiency of PSADTW.

The theoretical speedup (Ω) gained by PR-DTW over the original DTW is $O(n^2/\ w^2)$. It is determined by the data compression rate ($DCR = n/w$) of the piecewise representation over the raw data. Thus, the distance computation efficiency of PR-DTW can be examined in terms of the DCR performance of five PR-DTWs on all datasets. Specifically, the optimal segment numbers learned in the 1NN classifier are used to compute the DCRs for PSADTW, PDTW, SDTW, and APCADTW; while for the nonparametric DSADTW, we take the average segment numbers on each dataset to compute the DCR. The segment numbers and the DCRs of five measures over all datasets are shown in Table 6.

In order to compare the five measures comprehensively, we make a critical difference diagram [40] as Fig. 12, which is based on the average ranks shown in Table 6. For the comparison of multiple classifiers (>2), the Friedman test is recommended [40], which follows F distribution with the $(k-1)$ and $(k-1)\cdot(n-1)$ degrees of freedom. For the significance level $\alpha = 0.05$, the critical difference is 1.6129. Under the null hypothesis, as the statistical difference between the methods is not significant, i.e., the difference between the average ranks is smaller than 1.6129, they are grouped into a *clique*.

In Fig. 12, PSADTW, SDTW, APCADTW, and PDTW locate in the same *clique*. That means the DCRs of the four measures have no significant difference. However, PSADTW is superior to SDTW, APCADTW, and PDTW with a large gap. Obviously, this result derives from the stronger expressivity of the multiple features extracted in PSA, which could make time series be segmented into less number of subsequences than the single type of features. Thus, the above assumption on the distance computation efficiency is supported. Besides, the above four measures are significantly superior to DSADTW. In Table 6, we can observe that the DCRs of DSADTW locate in the range of 2–4 over all datasets. That means, for whatever type of time series, the subsequences segmented in DSA only contain 2–4 samples, which is too short to reflect the significant improvement of PR-DTW over the original DTW on the efficiency.

Furthermore, in Fig. 13, we present the relationship between the speedup of PSADTW over the original DTW and the DCR of PSA over the raw data on all datasets. The correlation ρ between the DCR and the SPEEDUP is 0.87, which indicates that the practical speedup has high positive linear correlation with DCR.

By the above experiments, we find that PSADTW is efficient in the procedure of distance computation. Therefore, although PSA extracts multiple features for similarity measure, PSADTW has high computational efficiency on both data representation and distance computation.

6. Conclusions and future work

In this paper, we proposed a novel piecewise representation model, named piecewise statistic approximation (PSA), for supporting the PR-DTW measure, which extracts multiple statistical features to capture the synthesized characteristics of time series for similarity measure. Besides, in PSADTW, we take the weighted Euclidean distance for matching the extracted features, which can discriminate the expressivities of the features by the weights, and make PSADTW sensitive to the effective information. The comprehensive experiments empirically prove that, PSADTW not only increases the precision of existing PR-DTW measures significantly,

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005

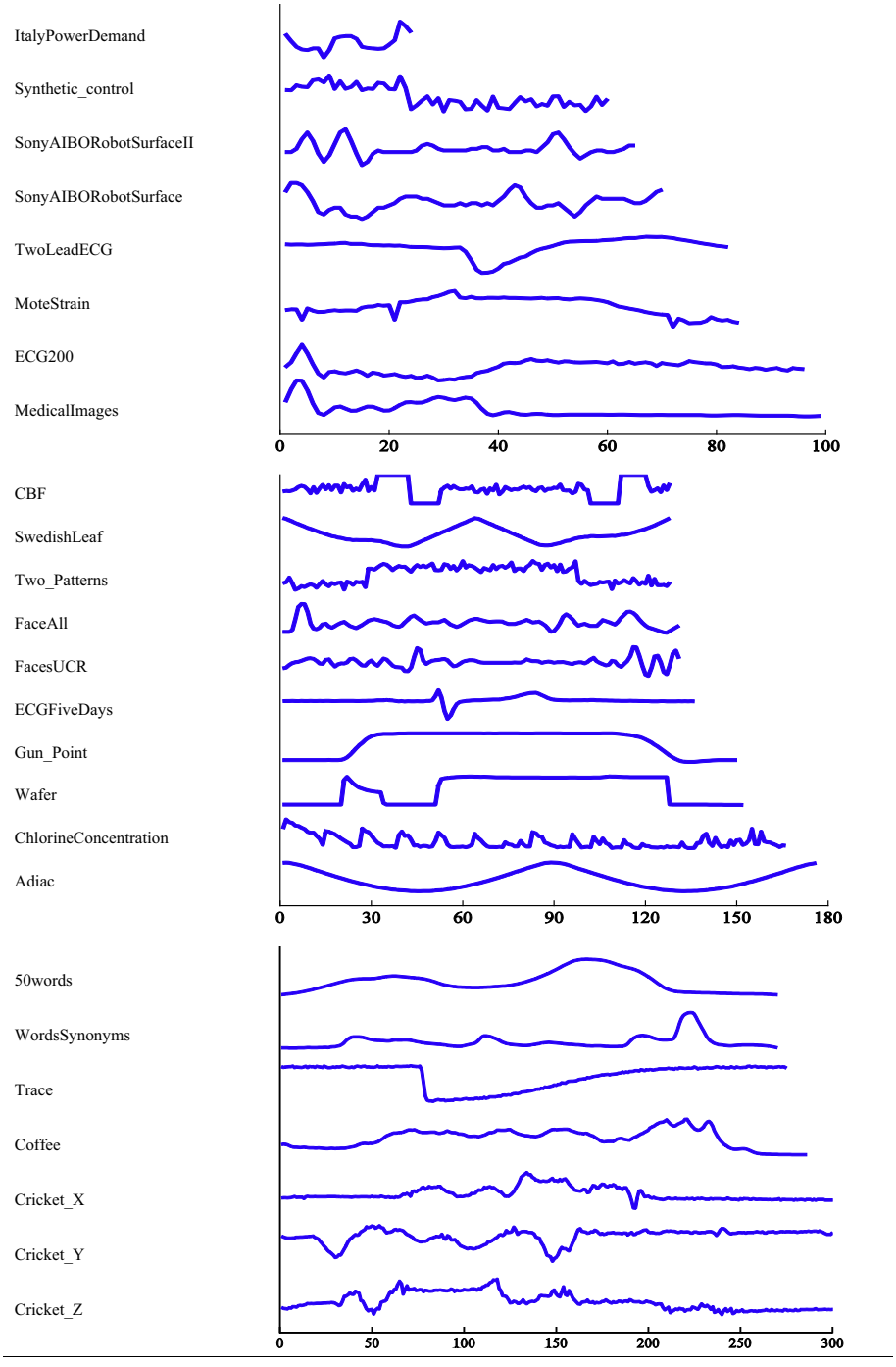
but also preserves the efficiency superiority of PR-DTW over the original DTW.

However, in PSA representation, time series is segmented into the subsequences with equal lengths. For the time series with irregular fluctuation, such a global segmentation scheme may be unable to get the subsequences with adaptive lengths or complete trends. Thus, in the next work, we will focus on the data-adaptive piecewise representation, which can identify the segmentation points automatically and divide the original time series into the subsequences with adaptive lengths.

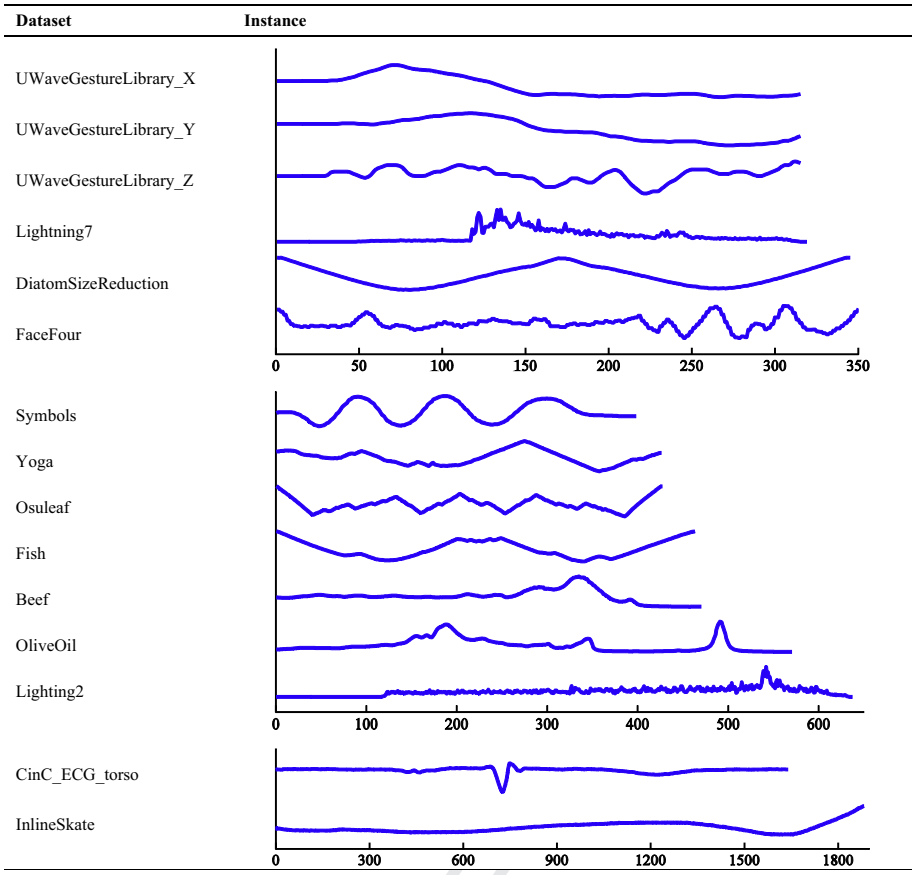
Acknowledgements

This work was funded by the Ministry of Industry and Information Technology of China (No. 2010ZX01042-002-003-001), China Knowledge Centre for Engineering Sciences and Technology (No. CKCEST-2014-1-5).

Appendix A. Sample instances of 40 datasets from the UCR time series archive and the optimal parameters in PSADTW



Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005



References

[1] P. Esling, C. Agon, Time-series data mining, ACM Comput. Surv. (CSUR) 45 (1) (2012) 12.
[2] T. Fu, A review on time series data mining, Eng. Appl. Artif. Intell. 24 (1) (2011) 164–181.
[3] P. Chaovalit, A. Gangopadhyay, G. Karabatis, Z. Chen, Discrete wavelet transform-based time series analysis and mining, ACM Comput. Surv. (CSUR) 43 (2) (2011) 6.
[4] T. Rakthanmanon, B. Campana, A. Mueen, et al., Searching and mining trillions of time series subsequences under dynamic time warping, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2012, pp. 262–270.
[5] A. Camerra, J. Shieh, T. Palpanas, T. Rakthanmanon, E. Keogh, Beyond one billion time series: indexing and mining very large time series collections with iSAX2+, Knowl. Inf. Syst. 39 (1) (2014) 123–151.
[6] F. Gullo, G. Ponti, A. Tagarelli, S. Greco, A time series representation model for accurate and fast similarity detection, Pattern Recogn. 42 (11) (2009) 2998–3014.
[7] J. Lines, A. Bagnall, Time series classification with ensembles of elastic distance measures, Data Min. Knowl. Disc. (2014) 1–28.
[8] J. Hills, J. Lines, E. Baranauskas, J. Mapp, A. Bagnall, Classification of time series by shapelet transformation, Data Min. Knowl. Disc. 28 (4) (2014) 851–881.
[9] J. Serra, J.L. Arcos, An empirical evaluation of similarity measures for time series classification, Knowl.-Based Syst. 67 (2014) 305–314.
[10] T. Rakthanmanon, E. Keogh, S. Lonardi, S. Lonardi, S. Evans, Time series epenthesis: clustering time series streams requires ignoring some data, in: Proceedings of the 11th International Conference on Data Mining (ICDM), 2011, pp. 547–556.
[11] J. Lin, M. Vlachos, E. Keogh, et al., A MPAA-based iterative clustering algorithm augmented by nearest neighbors search for time-series data streams, Advances in Knowledge Discovery and Data Mining, Springer, Berlin Heidelberg, 2005, pp. 333–342.
[12] T. Schlüter, S. Conrad, Hidden markov model-based time series prediction using motifs for detecting inter-time-serial correlations, in: Proceedings of the 27th Annual ACM Symposium on Applied Computing, 2012, pp. 158–164.
[13] E. Chalmers, M. Mizianty, E. Parent, Y. Yuan, E. Lou, Toward maximum-predictive-value classification, Pattern Recogn. 47 (2014) 3949–3958.
[14] G. Nychis, V. Sekar, D.G. Andersen, H. Kim, H. Zhang, An empirical evaluation of entropy-based traffic anomaly detection, in: Proceedings of the 8th ACM SIGCOMM Conference on Internet Measurement (IMC), 2008, pp. 151–156.
[15] S. Yingchareonthawornchai, H. Sivaraks, T. Rakthanmanon, C.A. Ratanamahatana, Efficient proper length time series motif discovery, in: Proceedings of the 13th International Conference on Data Mining (ICDM), 2013, pp. 1265–1270.
[16] B.K. Yi, C. Faloutsos, Fast time sequence indexing for arbitrary Lp norms, in: Proceedings of the 26th International Conference on Very Large Data Bases (VLDB), 2000, pp. 385–394.
[17] C.A. Ratanamahatana, E. Keogh, Three myths about dynamic time warping data mining, in: Proceedings of SIAM International Conference on Data Mining (SDM), 2005, pp. 506–510.
[18] L. Chen, R. Ng, On the marriage of lp-norms and edit distance, in: Proceedings of the 30th International Conference on Very Large Data Bases (VLDB), 2004, pp. 792–803.
[19] R. Agrawal, C. Faloutsos, A. Swami, Efficient similarity search in sequence databases, in: Proceedings of the International Conference on Foundations of Data Organization and Algorithms (FODO), 1993, pp. 69–84.
[20] E. Keogh, K. Chakrabarti, M. Pazzani, S. Mehrotra, Dimensionality reduction for fast similarity search in large time series databases, Knowl. Inf. Syst. 3 (3) (2001) 263–286.
[21] K. Chakrabarti, E. Keogh, S. Mehrotra, M. Pazzani, Locally adaptive dimensionality reduction for indexing large time series databases, ACM Trans. Database Syst. 27 (2) (2002) 188–228.
[22] E. Keogh, M.J. Pazzani, Relevance feedback retrieval of time series data, in: Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 1999, pp. 183–190.
[23] E. Keogh, S. Chu, D. Hart, M. Pazzani, Segmenting time series: a survey and novel approach, Data Min. Time Ser. Databases 57 (2004) 1–22.
[24] M.G. Baydogan, G. Runger, E. Tuv, A bag-of-features framework to classify time series, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2796–2802.
[25] J. Lin, R. Khade, Y. Li, Rotation-invariant similarity in time series using bag-of-patterns representation, J. Intell. Inf. Syst. 39 (2) (2012) 287–315.
[26] H. Li, C. Guo, Piecewise cloud approximation for time series mining, Knowl.-Based Syst. 24 (4) (2011) 492–500.
[27] P. Senin, S. Malinchik, SAX-VSM: interpretable time series classification using SAX and vector space model. in: Proceedings of the 13th International Conference on Data Mining (ICDM), 2013, pp. 1175–1180.

[28] J. Grabocka, L. Schmidt-Thieme, Invariant time-series factorization, Data Min. Knowl. Disc. 28 (5–6) (2014) 1455–1479.
[29] H. Deng, G. Runger, E. Tuv, M. Vladimir, A time series forest for classification and feature extraction, Inf. Sci. 239 (2013) 142–153.
[30] G.E. Batista, E.J. Keogh, O.M. Tataw, V.M. Souza, CID: an efficient complexity-invariant distance for time series, Data Min. Knowl. Disc. 28 (3) (2014) 634–669.
[31] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, E. Keogh, Querying and mining of time series data: experimental comparison of representations and distance measures, in: Proceedings of the International Conference on Very Large Data Bases (VLDB), 2008, pp. 1542–1552.
[32] E. Keogh, C.A. Ratanamahatana, Exact indexing of dynamic time warping, Knowl. Inf. Syst. 7 (3) (2005) 358–386.
[33] E.J. Keogh, M.J. Pazzani, Derivative dynamic time warping, in: Proceedings of SIAM International Conference on Data Mining (SDM), 2001, pp. 5–7.
[34] Y.S. Jeong, M.K. Jeong, O.A. Omitaomu, Weighted dynamic time warping for time series classification, Pattern Recogn. 44 (9) (2011) 2231–2240.
[35] E.J. Keogh, M.J. Pazzani, Scaling up dynamic time warping for data mining applications, in: Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2000, pp. 285–289.
[36] E.J. Keogh, M.J. Pazzani, Scaling up dynamic time warping to massive datasets, Principles of Data Mining and Knowledge Discovery, Springer, Berlin Heidelberg, 1999, pp. 1–11.
[37] M. Lovric, International Encyclopedia of Statistical Science, Springer, London, 2011.
[38] E. Keogh, Q. Zhu, B. Hu, Hao. Y., X. Xi, L. Wei, C.A. Ratanamahatana, UCR time series classification/clustering homepage: <www.cs.ucr.edu/~eamonn/time_series_data/>, 2011.
[39] M. Björkman, K. Holmström, Global optimization using the DIRECT algorithm in matlab, Adv. Model. Optim. 1 (2) (1999) 17–37.
[40] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.

Please cite this article in press as: Q. Cai et al., Piecewise statistic approximation based similarity measure for time series, Knowl. Based Syst. (2015), http://dx.doi.org/10.1016/j.knosys.2015.05.005