Discovery of Time-Series Motif from Multi-Dimensional Data Based on MDL Principle

YOSHIKI TANAKA   yoshiki@ai.cs.scitec.kobe-u.ac.jp
KAZUHISA IWAMOTO   kazuhisa@ai.cs.scitec.kobe-u.ac.jp
KUNIAKI UEHARA   uehara@kobe-u.ac.jp
Graduate School of Science and Technology, Kobe University 1-1 Rokko-dai, Nada, Kobe 657-8501, Japan

Editor: Eamonn Keogh

Abstract.   Recently, the research on efficient extraction of previously unknown, frequently appearing patterns in a time-series data has received much attention. These patterns are called '*motifs*'. Motifs are useful for various time-series data mining tasks. In this paper, we propose a motif discovery algorithm to extract a motif that represents a characteristic pattern of the given data based on Minimum Description Length (MDL) principle. In addition, the algorithm can extract motifs from multi-dimensional time-series data by using Principal Component Analysis (PCA). In experimental evaluation, we show the efficiency of the motif discovery algorithm, and the usefulness of extracted motifs to various data mining tasks.

Keywords: time-series motifs, multi-dimensional time-series data, PCA, MDL principle

1. Introduction

Many researchers have been studying the extraction of various characteristics from time-series data. One of these challenges, efficient extraction of '*motifs*' has received much attention (Lin et al., 2002). The motif is known as a previously unknown pattern that appears frequently in a time-series data. Figure 1 depicts the concept of motifs that can be perceived from intuition. Formal definition of motifs will be discussed in the next section.

Motif extraction is useful for various time-series data mining tasks described as follows:

– We can apply the motif to mining association rules in time-series data. For example, Mori and Uehara (2001) extract association rules from the 3-dimensional time-series data that represents the human motion while doing some actions. This extraction requires time-series subsequences called '*primitive motion*' that represent a basic movement of a human body part. A '*Time-Series Subsequence (TSS)*' is a part of a time-series data. By using the motif as the primitive motion, we can extract more useful and efficient association rules to obtain novel knowledge about the human motions.
– In the field of physiotherapy, Caraca-Valente and Lopez-Chavarrias (2000) discover '*sequential patterns*' from an isokinetics machine for muscular diagnosis and rehabilitation, etc. We can use motif discovery algorithm for efficient extraction of the sequential patterns.
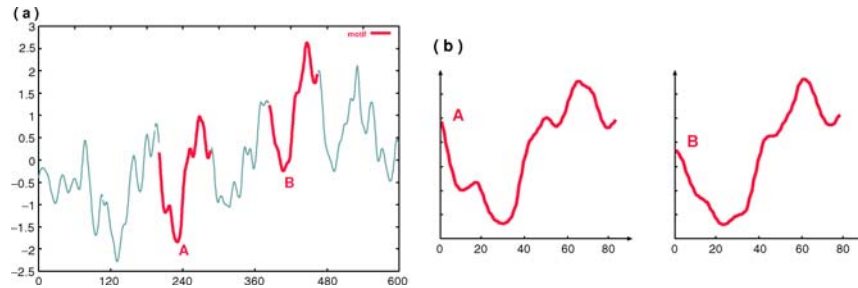
Figure 1. (a) An example of a motif that appears two times (bold lines) in an 1-dimensional time-series data. (b) A zoom-in of two time-series subsequences of motif. These figures show that two time-series subsequences are similar to each other.

– In financial domain, Koopman and Ooms (2003) model the characteristics of the tax revenues series by finding the '*recurring patterns*' from daily tax revenues. We can regard these recurring patterns as motifs.

Many researchers have proposed algorithms for discovering a motif (Lin et al., 2002; Berberidis et al., 2002; Chiu, Keogh, & Lonardi, 2003; Tanaka & Uehara, 2003). Among them, our previous algorithm (Tanaka & Uehara, 2003) can extract motifs from multi-dimensional time-series data. First, the motif discovery algorithm transforms multi-dimensional time-series data into 1-dimensional time-series data by using PCA (Principal Component Analysis) (Heras et al., 1996) for reducing dimensions of the data. Then, to extract patterns that have appeared in the 1-dimensional time-series data, we transform the data into a sequence of symbols. Finally, we discover a motif by calculating a 'description length' of a pattern based on the Minimum Description Length (MDL) principle (Rissanen, 1989).

Our motif discovery algorithm reported in Tanaka and Uehara (2003) is useful based on the assumption that the lengths of all TSS's in the same pattern are identically same. Hereafter, we call such pattern '*Same Length pattern (SL pattern)*'. However, in the real world, the lengths of each TSS in a pattern are a little bit different from each other. Hereafter, we call such pattern '*Different Lengths pattern (DL pattern)*'.

For instance, a pattern that appeared in an electrocardiogram is described in figure 2.
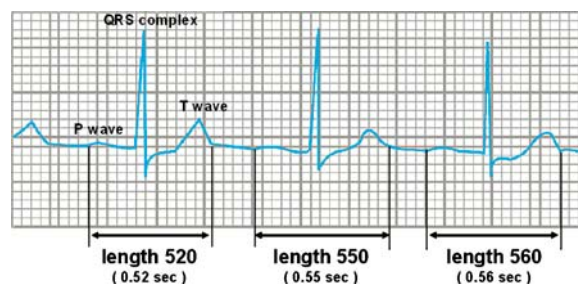

Figure 2. An example of a pattern in an electrocardiogram.

Every TSS represents a series of heartbeat that includes a P wave, a QRS complex and a S wave. However, the lengths of the time series are a little bit different from each other. In this example, the first TSS has a length of about 520, the second has a length of about 550 and the third has about 560 (A sampling frequency of the time-series data is 1 kHz). Here, we mention about our interesting level of DL pattern length. We want to extract TSS's whose length is little bit different. As in this instance, we can say that most of the patterns in real time-series data are DL patterns. Therefore, we must extract motifs that are not only SL patterns but also DL patterns.

This paper extends the work previously introduced in Tanaka and Uehara (2003). We extend this motif discovery algorithm by modifying the following two points. First, we transform the symbol sequence that represents a behavior of a given time-series data to extract DL patterns. Second, we augment the definition of a description length of a time-series data to process not only SL patterns but DL patterns as well.

The rest of this paper is organized as follows. In Section 2, we introduce some approaches related to our motif discovery algorithm. In Section 3, we show the efficiency of PCA to reduce dimensions of a multi-dimensional time-series data. In addition, we note the efficiency of Independent Component Analysis (ICA) (Hyvrinen & Oja, 2000) for the same purpose. Section 4 describes in detail the motif discovery algorithm introduced in Tanaka and Uehara (2003), upon which our algorithm is based. In this section, by showing experimental evaluation, we note the efficiency of MDL principle for detecting motif that is the characteristic pattern of a given time-series data. In Section 5, we extend our previous algorithm to extract a motif that is the DL pattern. We call the algorithm '*Extended Motif Discovery (EMD) algorithm*'. We show the efficiency of EMD algorithm by comparing it with motifs extracted by our previous algorithm. Furthermore, we also show the usefulness of motifs to various data mining tasks. Finally, we note the directions for future work in Section 6, and conclude our work in Section 7.

2. Background

2.1. Definitions

In this section, we present some definitions used in this paper. *1-dimensional time-series data T* of *length n* is represented as follows:

$$T = x_1, \ldots, x_n \tag{1}$$

Every x_i is a real-valued variable. In particular, *m-dimensional time-series data T_m* of length *n* is a sequence of *m* set of real-value variables. It is represented as follows:

$$T_m = (x_{11}, \ldots, x_{m1}), \ldots, (x_{1n}, \ldots, x_{mn}) \tag{2}$$

Time-series subsequence (TSS) C of length *q* is a part of *T*. It is represented as follows:

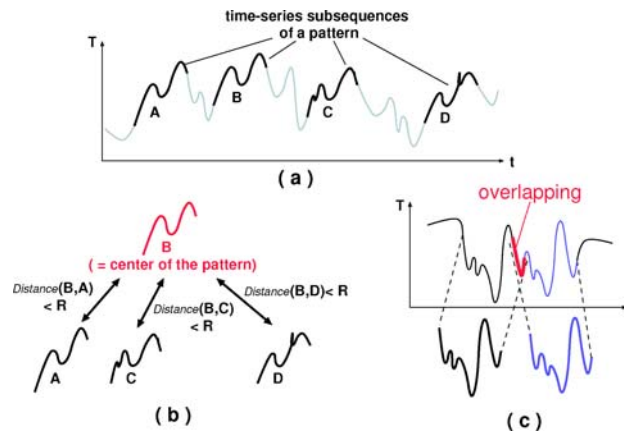$$C_{p,q} = x_p, \ldots, x_{p+q-1}(q \leq n, 1 \leq p \leq n - q + 1) \tag{3}$$

Figure 3. Constraints of a pattern: (a) 'behavior constraint', (b) 'distance constraint' and (c) 'non-overlapping constraint'. Figure 3(c) represents an example of non-overlapping constraint violation.

Here, we define the *pattern* of a time-series data. All TSS's that belong to the same pattern need to conform to the following constraints:

Behavior constraint: Every TSS has the same *behavior* (temporal variation) as described in figure 3(a).

Distance constraint: Distances between all possible pairs of TSS's are lower than the threshold R. R is a user defined threshold distance. For example, figure 3(b) represents that all distances between **B** and other TSS's are lower than R. Therefore, these TSS's conform to the 'distance constraint'. We call **B** the *center of the pattern*.

Non-overlapping constraint: TSS's should not overlap to each other. For example, the two TSS's described in figure 3(c) have the same behavior. However, these TSS's are overlapping (represented by the bold line). Therefore, they do not belong to the same pattern. By introducing this non-overlapping constraint, our motif algorithm can be categorized as a tool for a segmentation of time-series. For example, if overlapping is accepted, "CABB-BABBBAA", ABBBA can be extracted as a motif. But, in this case, two subsequences share the same A, and we can get two possible segmentations: The one is C, ABBBA and BBBAA. The other is CABBB, ABBBA and A. Whereas, if we introduce the non-overlapping constraint, ABBB can be extracted as a motif. In this case, the sequence can be segmented as C, ABBB, ABBB and AA.

2.2. *Related works*

Data mining tasks for time-series data, such as indexing (Vlachos et al., 2003), clustering (Cyril et al., 1999), discovery of association rules (Mori & Uehara, 2001; Das et al., 1998), have some common purposes. One of them is to reduce calculation time. For this purpose,

many approaches are proposed to obtain a discrete representation from a time-series data. Our approach transforms a time-series data into a symbol sequence by SAX algorithm (Lin et al., 2002, 2003). We will note the details of SAX algorithm in Section 4.1.

Similar to our approach, '*time-series active search*' (Kashino, Smith, & Murase, 1999) and pattern recognition based on '*episode*' (Colomer, Melendez, & Gamero, 2002) use a discrete representation. The time-series active search transforms a time-series data into a sequence of symbols based on histograms that obtained by the short-time spectrum analysis. The time-series active search is used for the discrete representation at a frequency domain.

The work in Colomer, Melendez, and Gamero (2002) divides a time-series data into some '*episodes*.' Each episode represents a basic behavior of the data. Then, this method obtains a symbol sequence by assigning a unique symbol to each type of episode. However, this discrete representation is easily influenced by the noise in the data. In addition, ambiguity in the definition of the distance between episodes is also a problem.

Similar to our approach, there are some approaches for extracting unknown patterns (motifs) in a time-series data, such as '*Enumeration of Motifs through Matrix Approximation (EMMA) algorithm* (Lin et al., 2002)' and the motif discovery algorithm using random projection (Chiu, Keogh, & Lonardi, 2003). The EMMA algorithm discovers a motif by obtaining all possible TSS's with a user defined length. On the other hand, the approach (Chiu, Keogh, & Lonardi, 2003) is extended from EMMA algorithm by adapting the random projection (Buhler & Tompa, 2001) to discover motif more efficiently. Note that they can deal with only SL patterns.

On the other hand, 'Approximates Periodicity Detection' (APPD) algorithm (Berberidis et al., 2002) discovers unknown patterns by using Fast Fourier Transform (FFT) and auto-correlation function. Although APPD algorithm can deal with DL patterns, it can extract only patterns that occur periodically in a given time-series data. Most of the patterns in real time-series data appear non-periodically, so APPD algorithm is not adequate to be considered as the motif discovery algorithm.

3. Transforming multi-dimensional time-series data into 1-dimensional time-series data

This section presents our approach relative to multi-dimensional time-series data. Generally, there are two approaches. One is to treat each time-series data separately and another is dimensionality reduction. Navarro and Baeza-Yates (1999) and Vlachos et al. (2003) proposed a method which treats multi-dimensional time-series data separately, matching the approximate patterns that appear in the multi-dimensional time-series data. This method can be used as a tool for data mining. However, treating multi-dimensional time-series separately needs much more calculation time. For example, our approach without dimensionality reduction takes about 468 seconds for discovering a motif from 63-dimensional time-series of length 3600. But, our approach with dimensionality reduction takes almost 7.43 second for discovering a motif from the same data.

Furthermore, dimensionality reduction, that is transforming multi-dimensional time series data into a 1-dimensional time-series data seems to be an easier method. Because analyzing 1-dimensional time-series data is easier then analyzing multi-dimensional time-series

sources. Many research efforts have been focussed on pattern discovery from 1-dimensional time-series data (Das et al., 1998).

Due to these reasons, we transform multi-dimensional time-series data into 1-dimensional time-series data. However, for the purpose of motif discovery, we need to maintain at least the characteristic patterns that appear frequently in the original multi-dimensional time-series data.

3.1. *Different methods of transformation*

To transform a multi-dimensional time-series data into a 1-dimensional time-series data, some dimensionality reduction methods have already been proposed. For example, RP (random projection) (Buhler & Tompa, 2001), PCA (Principal Component Analysis) (Heras et al., 1996), ICA (Independent Component Analysis) (Hyvrinen & Oja, 2000) are popular methods. First, we focus on RP and PCA. Recently RP has appeared as a tool for dimensionality reduction and performs better than PCA in most of the cases. However, in Fradkin and Madigan (2003), RP predictively underperforms PCA in some conditions. For example, RP leads to good results for high-dimensional data such as 2000 and 3571 dimensions. But, PCA is able to do better for relative low-dimensional data such as 34 and 57 dimensions. In this paper, we mainly experiment time-series data obtained from the motion capture system. So, the dimension of our experimental data is at most 63-dimensions. Therefore, we consider that PCA is more adequate for our experiments than RP method. Second, we focus on PCA and ICA. PCA and ICA are effective methods to find the characteristics of the data expressed by some observed variables. For example, PCA is used to extract an index for two or more stock prices (Levin, Leen, & Moody, 1993). ICA is used to separate brain activity from artifacts in Magnetoencephalograrhy (MEG) (Vigario et al., 1998).

Both methods extract 'components' (called 'principal components' in PCA, and 'independent components' in ICA) that are linear combinations of the original variables. Assume that we observe n linear mixtures $x_1 \ldots x_n$ of n components $s_1 \ldots s_n$, each component is represented by the equation:

$$s_i = a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \tag{4}$$

Both methods estimate s_i without any background knowledge of the original data. However, an assumption for estimating principal components by PCA is different from that of ICA. PCA estimates s_i based on the assumption that every principal component is statistically non-correlate to each other. So, PCA extracts a projective representation that maximizes the variance of the data as principal components. On the other hand, ICA estimates s_i based on the assumption that every independent component is statistically independent to each other. So, ICA extracts a projective representation that maximizes the nongaussianity of the data as independent components.

In our approach, we use principal components instead of independent components. Because, we notice several problems in using ICA to reduce dimensions of the time-series data. First, ICA cannot determine the order of the independent components. But, PCA determines the order of principal components by eigenvalues calculated from eigenvectors of

original data. The first principal component maintains the largest amount of information of the original data in all principal components (Heras et al., 1996). On the other hand, it is known that the order of independent components is random (Hyvrinen & Oja, 2000). Therefore, we cannot uniquely determine the independent component for reduction of dimensions. So, we use the first principal component to effectively reduce the dimensions of multi-dimensional time-series data.

The second problem is that it is not quite comprehensible what is represented by independent components. From a view point of 'Projection Pursuit (Friedman & Tukey, 1974)', we can say that independent components are the most interesting projective representations. However, it is not clear what kind of statistical information of the original data remains. Therefore, a motif extracted from an independent component may not be a characteristic pattern in the original time-series data.

The third problem is a contradiction between the assumption for ICA and our motif discovery algorithm. ICA needs the assumption that every observed variable x_i and independent component s_i do not have Gaussian distribution. It is well known that we cannot estimate independent components if every s_i has Gaussian distribution (Hyvrinen & Oja, 2000). In addition, if every x_i has Gaussian distribution, then statistically non-correlated components are always independent to each other. Therefore, the input time-series data needs to have non-Gaussian distribution for ICA. On the other hand, in our algorithm, we transform the 1-dimensional time-series data into a sequence of symbol based on PAA representation. SAX representation is obtained based on the assumption that the given data has a Gaussian distribution. Due to the fact, we cannot use ICA to transform the multi-dimensional time-series data into the 1-dimensional time-series data for motif detection. In contrast, PCA assumes that the original time-series data has a Gaussian distribution. Therefore, we can use PCA for discovering motifs.

3.2. Principal component analysis

In this section, we illustrate the specific method to apply PCA to the m-dimensional time-series data T_m of length n. First, we need to calculate a covariance matrix A_{T_m} by using the following equation:

$$A_{T_m} = \begin{bmatrix} \sum_t x_{1t}x_{1t} & \sum_t x_{1t}x_{2t} & \cdots & \sum_t x_{1t}x_{mt} \\ \sum_t x_{2t}x_{1t} & \sum_t x_{2t}x_{2t} & \cdots & \sum_t x_{2t}x_{mt} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_t x_{mt}x_{1t} & \sum_t x_{mt}x_{2t} & \cdots & \sum_t x_{mt}x_{mt} \end{bmatrix} \tag{5}$$

Each eigenvalue λ_i is ordered as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$. The eigenvector is represented as $[e_{1\lambda_i} e_{2\lambda_i} \ldots e_{m\lambda_i}]$. Then, the i-th principal component pc_{t,λ_i} is calculated by using means of $x_1, x_2, \ldots, x_m$.

$$pc_{t,\lambda_i} = e_{1\lambda_i}(x_{1t} - \bar{x}_1) + e_{2\lambda_i}(x_{2t} - \bar{x}_2) + \cdots + e_{m\lambda_i}(x_{mt} - \bar{x}_m) \tag{6}$$

In our approach, we use the first principal component to effectively transform the multi-dimensional time-series data into 1-dimensional time-series data. Finally, we obtain 1-dimensional time-series data T as follows:

$$T = x_1, \ldots, x_t, \ldots, x_n \tag{7}$$

$$x_t = e_{1\lambda_1}(x_{1t} - \bar{x}_1) + e_{2\lambda_1}(x_{2t} - \bar{x}_2) + \cdots + e_{m\lambda_1}(x_{mt} - \bar{x}_m) \tag{8}$$

PCA dynamically detects the significant coordinates that include characteristic patterns of the original data T_m, because the significance of each coordinate is represented in each coefficient $e_{i\lambda_1}$. In addition, the first principal component maintains the largest amount of information of the original data (Heras et al., 1996). So, we can say that the first principal component is a linear combination of the original variables weighted according to the contribution in the original data. Therefore, we can assume that the discovered motif from T is same as that of the original multi-dimensional time-series data T_m.

4. The motif discovery algorithm

In this section, we present an overview of our motif discovery algorithm. The motif discovery algorithm dynamically detects a motif from the 1-dimensional time-series data based on information theoretic criterion. We mention about the reason why we exploit MDL principle for discovering a motif. As for information theoretic criterion, we know AIC (Akaike's Information Criterion) (Akaike, 1969), BIC (Bayesian Information Criterion) (Schwarz, 1981) and MDL (Minimum Description Length) principle (Rissanen, 1989). First of all, AIC estimates the best model based on "prediction capability". But, our approach is finding frequent patterns rather than prediction of time-series. Secondly, BIC is the way to estimate the best model based on bayesian theorem. Finally, MDL principle states that the best model to describe a set of data is the model which minimizes the description length of the entire data set. Here, BIC equals to MDL from the viewpoint of selecting a simpler model. We focus on the concept of MDL. The concept of MDL is to select the best model which compresses data well. Here, our motif discovery algorithm can be used as a method for summarizing time-series data. So, we consider that the model which has minimum description length is the best model for summarization of time-series data. Therefore, we consider that the MDL principle is adequate for our approach.

4.1. Transforming time-series into a sequence of symbols

We use MDL principle for extracting a pattern that is expected to be a motif. However, there lies the problem that the same patterns hardly appear in time-series. In addition, we want to extract a pattern without the influence of the "noise" of the time-series. For these reasons, we transform the time-series data into a sequence of symbols that represents the behavior excluding the noise. For the purpose, we use dimensionality reduction algorithm based on PAA representation. Here, we show the visualization of this transformation algorithm in figure 4.
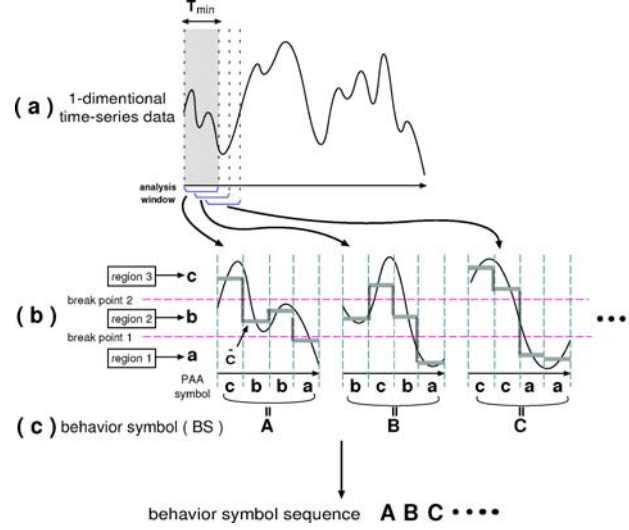
Figure 4. Visualization of the algorithm to transform a time-series into a symbol sequence. (a) we obtain TSS's by shifting the analysis window. (b) Each TSS is transformed into a SAX symbol sequence. (c) 'behavior symbol' is assigned for every SAX symbol sequence.

First, we prepare the analysis window of T_{min} (figure 4(a)). T_{min} is the minimum length of motif for the data. By shifting the analysis window, we obtain all TSS's with length of T_{min} in the data. Second, each TSS is represented by PAA representation, and transformed into a sequence of '*PAA symbols*' (figure 4(b)). PAA representation is a vector expression obtained by dividing a time-series data into some segments and calculating the average value in each segment. In figure 4(b), each TSS is divided into 4 segments. By using PAA representation, a time series $T = x_1, \ldots, x_n$ of length n can be represented as w-dimensional space by a vector $\bar{C} = \bar{c}_1, \ldots, \bar{c}_w$ as follows:

$$\bar{c}_i = \frac{w}{n} \sum_{j=\frac{n}{w}(i-1)+1}^{\frac{n}{w}i} x_i \tag{9}$$

In figure 4(b), PAA representation of each TSS is represented by the vector $\bar{C}$. Then, '*break points*' are determined to transform the vector of w-dimension into a sequence of '*SAX symbols*'. Break points provide some equiprobable regions of PAA representation under a Gaussian distribution (Lin et al., 2002). For example, in figure 4(b), two breakpoints are determined and three regions are provided. Then, every region is assigned a unique SAX symbol, and every PAA coefficient is transformed into the SAX symbol in the area where PAA coefficient belongs to. For example, $\bar{C}$ of the first TSS is transformed into SAX symbol sequence '*cbba*'.

Here, we obtain SAX symbol sequences for every TSS. To obtain a sequence of symbols that represents the behavior of T, every SAX symbol sequence is transformed into a single

unique symbol. We call the symbol '*Behavior Symbol (BS)*', since every SAX symbol sequence represents the behavior of each TSS. For example, in figure 4(c), the behavior symbol '*A*' is assigned to SAX symbol sequence '*cbba*', and '*B*' is assigned to SAX symbol sequence '*bcba*' and so on. Finally, we obtain a '*Behavior Symbol sequence (BS sequence)*' of T. Here, we present in detail the meaning of behavior symbols. A behavior symbol represents a unique behavior in the data. For example, in figure 4(b), SAX symbols obtained from the data in initial window are '*cbba*' where '*c*' is high, '*b*' is middle and '*a*' is low. So, the string '*cbba*' can be assigned to the letter '*A*' which conceptually means "the time-series has two peaks, the second one being lower than the first one". The string '*bcba*' can be assigned to the letter '*B*' which conceptually means "the time-series starts in middle range, reaches one high peak and then decrease". In this way, by introducing behavior symbols, we can detect TSS's with the same behavior. Another reason for introducing behavior symbol is to reduce memory space for symbol sequences and search space for discovering a motif rather than SAX symbol sequences.

4.2.   *Estimating extracted patterns based on MDL principle*

To estimate the optimality of BS subsequence patterns by MDL principle, we need to define a description length of BS sequence. The '*BS subsequence*' is a part of the BS sequence, and we call BS subsequence '*BSS*' hereafter. In addition, the BS subsequence pattern is called a *BSS pattern*. We assume that n_p is the length of the BSS pattern SC and s_p is the number of unique BSs in SC. First, we need $\log_2 n_p$ bits to express the number of BSs of SC. Then, we need $n_p \log_2 s_p$ bits to express the number of unique BSs of SC. Hence, the description length of SC is defined as follows:

$$DL(SC) = \log_2 n_p + n_p \log_2 s_p \tag{10}$$

In addition, we need to define the description length $DL(\tilde{C} \mid SC)$ where $\tilde{C}$ represents *BS sequence*. This is the description length of $\tilde{C}$ where SC is replaced with one symbol. We assume that the length of such a sequence is n_a, the number of unique BSs in $\tilde{C}$ is s_a and the frequency of appearance SC in $\tilde{C}$ is q. The description length $DL(\tilde{C} \mid SC)$ is calculated as follows:

$$DL(\tilde{C} \mid SC) = \log_2 n_a + n_a \log_2(s_a + q) \tag{11}$$

Here, $\log_2 n_a$ is the number of bits required to express the number of BSs in $\tilde{C}$. In addition, $n_a \log_2(s_a + q)$ is the number of bits required to encode the number of unique BSs of $\tilde{C}$. Finally, MDL estimation function $MDL(\tilde{C} \mid SC)$ is defined as follows:

$$\begin{aligned} MDL(\tilde{C} \mid SC) &= DL(\tilde{C} \mid SC) + DL(SC) \\ &= \log_2 n_a + n_a \log_2(s_a + q) + \log_2 n_p + n_p \log_2 s_p \end{aligned} \tag{12}$$

Now, this $MDL(\tilde{C} \mid SC)$ should be small. In order to obtain small values for $MDL(\tilde{C} \mid SC)$, the length of the pattern and its the number of unique BSs in SC should be small, that is the number of types of patterns should be small. In the other words, the frequency of the patterns should be high, and the complexity of the model should be large. That is, we want to extract a high-frequency and complex patterns. We consider the BSS pattern SC which has the minimum value of MDL estimation function is the 'MDL pattern' of $\tilde{C}$. It corresponds to a motif of the time-series data.

4.3. Discovering motif from a BS sequence

In this section, we illustrate the MDL pattern detection algorithm. Figure 5 shows the visualization of the algorithm.

Figure 5(a) is a BS sequence $\tilde{C}$. Here, we prepare an analysis window with certain length. By shifting the window, we obtain BSS's whose lengths are equal to the length of the window. For example, we shift the analysis window with the length of 3, and obtain BSS's, such as '*ABC*', '*BCB*' and '*CBC*'. Then, we extract the most frequently appearing BSS in $\tilde{C}$. In figure 5(b), '*BCB*' appears 5 times, so '*BCB*' is extracted as the most frequently appearing BSS pattern in $\tilde{C}$.

Here, we should recall that all TSS's that belong to the same pattern have to conform to the '*distance constraint*' defined in Section 2.1. However, all TSS's that correspond to the extracted BSS pattern do not always conform to the '*distance constraint*'. Because, this pattern conforms to only '*behavior constraint*'. Therefore, to obtain a pattern that conforms to '*distance constraint*', we need to calculate distances between every two TSS's.

Figure 5(c) represents TSS's that correspond to the BSS pattern '*BCB*'. Then, we calculate Euclidean distances between every two TSS's and make the '*distance matrix*' described in figure 5(d). For example, the distance matrix shows that the Euclidean distances between the first TSS and the others are 10, 50, 20 and 45 respectively. Next, we count the number of TSS's whose distance are lower than threshold R. In this example, we assume $R = 30$ and the count of the first TSS is 2.

Furthermore, to determine members of the pattern, we need to determine the '*center of the pattern*'. In the approach, TSS that has the maximum count is considered to be the candidate of '*center of the pattern*'. In figure 5(d), the maximum count is 2, and three TSS's (first, second and fourth one) are considered to be the candidates of the center of the pattern. Next, we calculate the sum of distances that are lower than R for each of them. For example, the sum value of the first TSS is $10 + 20 = 30$, the second TSS is $10 + 25 = 35$, and the fourth TSS is $20 + 25 = 45$. Then, the minimum sum value is 30, so the first. TSS is regarded as the center of the pattern. Finally, we extract TSS's whose distance from the center are lower than R. These TSS's are considered as members of the pattern. In figure 5(e), the first, the second and the fourth are extracted as members of the pattern. Then, in figure 5(f), BSS's that correspond to these TSS's are considered as members of the BSS pattern '*BCB*'.

Next, we calculate MDL estimation function M_1 of the BSS pattern '*BCB*'. At the same time, we calculate the length of the pattern L_1 and the location of the pointer P_1 which shows the beginning of every BSS. For instance, in figure 5(f), the length L_1 is 3 and the obtained pointers are located at 2, 9 and 21. Then, in figure 5(g), we extract another pattern
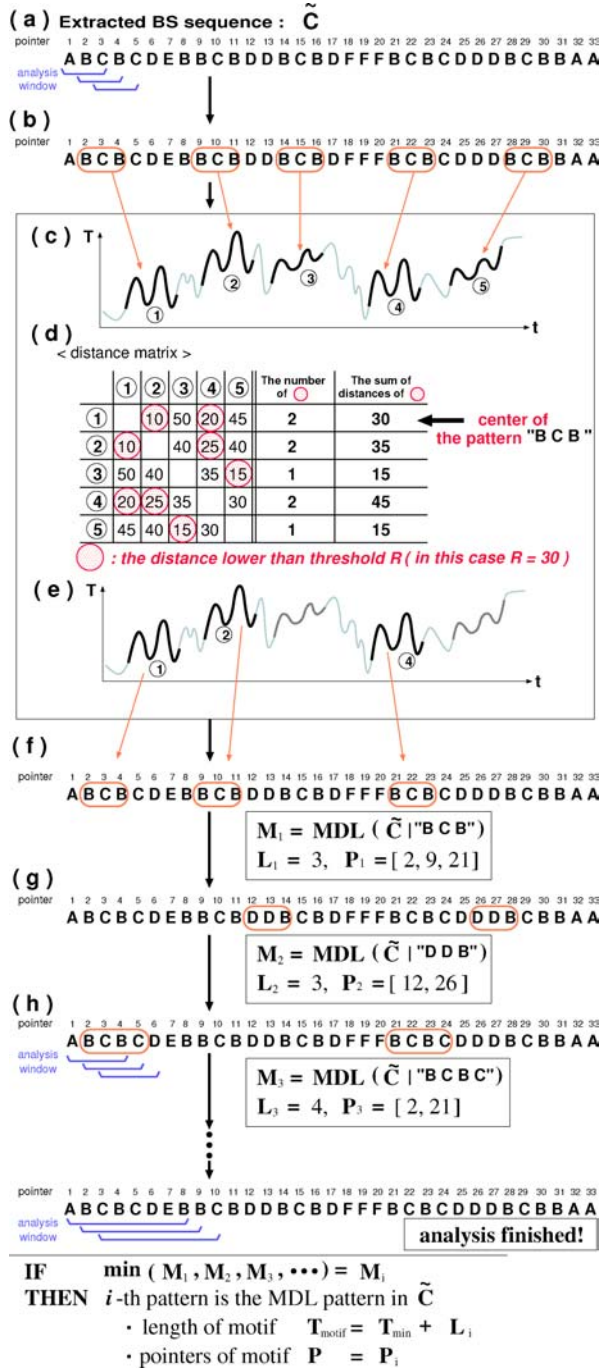
Figure 5.   Visualization of the MDL pattern detection algorithm.

'*DDB*' with the length of 3, and calculate MDL estimation function M_2 and the pointer P_2 of the pattern. If we find all patterns whose lengths are 3, In figure 5(h), by using the analysis window with length of 4, we extract a pattern '*BCBC*'. In figure 5, a BS sequence pattern length is determined by MDL principle mentioned above. And, the range of varying sequence pattern length is from 3 to at most the length of $\tilde{C}$. So, our analysis stops looking for other patterns if no patterns have been discovered for that specific length.

The above analysis is repeated until we find all patterns with arbitrary length in $\tilde{C}$. When it is finished, the pattern with the smallest value of MDL estimation function is considered as the MDL pattern in $\tilde{C}$. Using the length of the MDL pattern L_i, the length of the motif T_{motif} is calculated as follows:

$$T_{\text{motif}} = T_{\text{min}} + L_i \tag{13}$$

Then, we regard the MDL pattern as a motif of T.

4.4. *Experimental evaluation*

In this section, we show the efficiency of our motif discovery algorithm. In the experiment, we use 3-dimensional time-series data set of human motion obtained from the motion capture system. In this system, an actor puts on 18 markers which reflect infra-red light, and performs some actions being surrounded with 6 cameras (figure 6(a)). These cameras record the actor's action as video images and calculate 3-dimensional locations of the markers. Finally, the 3-dimensional time-series data is obtained as shown in figure 6(b). The figure represents the movement of '*the right hand while the actor is pitching a ball*'.

This dataset includes various data such as stable data, fluctuating data, periodic data, non-periodic data etc. By using these datasets, we can estimate the efficiency of our motif discovery algorithm to various type of time-series data. In addition, by comparing extracted
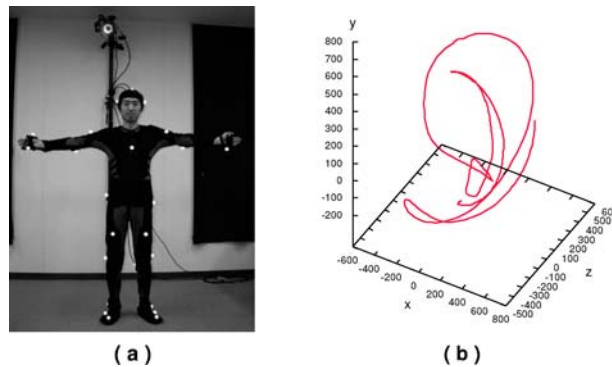

Figure 6.   (a) The motion capture system and the actor who puts on 18 markers (white spots). (b) An example of the 3-dimensional time-series data obtained from the motion capture system.
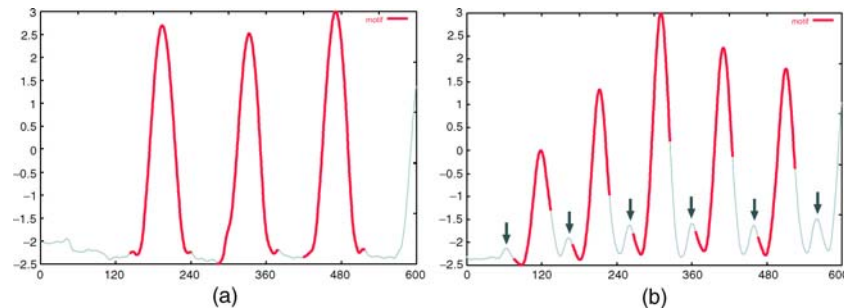
Figure 7. Motifs discovered from 1-dimensional time-series data: (a) 'Feet movement while walking' and (b) 'Feet movement while running'.

motif with a real human motion, we can confirm whether the extracted motif is a characteristic pattern in the given data or not.

4.4.1. Efficiency of MDL principle for discovering motif. Figure 7 shows the motifs extracted from some time-series data. Both time-series data have a common characteristic pattern, i.e. single peak with high amplitude (bold lines). However, the length and the interval of occurrence of the peaks are different from each other. From the result, our algorithm can discover motif whose length are approximately equal to the peak's length.

Here, we can find that the motif of figure 7(b) is not the most frequently occurring pattern. Because, the number of appearance of a pattern with small peak indicated by arrows is larger than the number of motif's appearance. However, based on MDL principle, this motif can describe the given time-series data with the shortest description length. It is due to the fact that MDL principle takes account of not only the number of pattern's appearance but also the description length of the given time-series data. Therefore, we can say that MDL principle is useful for discovering motif from time-series data.

4.4.2. Efficiency of our algorithm for multi-dimensional time-series data. Next, we direct our attention toward the validity of multi-dimensionality. Figure 8 shows the example of the motif extracted from a motion data '*Feet movement while walking*'. As seen from these
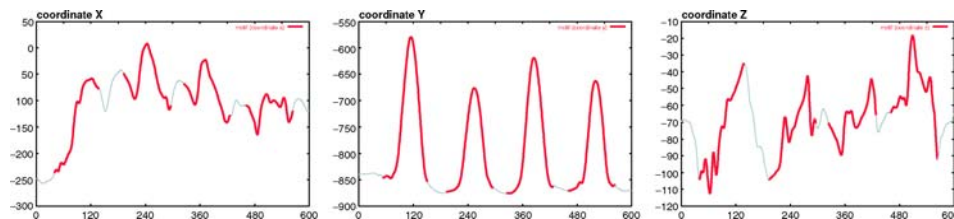

Figure 8. An example of motif in the original 3-dimensional time-series data, 'Feet movement while walking.' From left to right, the figures represent coordinate '*x*', '*y*', '*z*', respectively.

results, the motif of coordinate 'y' satisfies the definition of motif, because we can intuitively find that these TSS's have the same behavior. On the other hand, the motif of coordinate 'x' and 'z' are far from the characteristic of a motif. Because, in both of coordinate 'x' and 'z', the behavior of first and fourth TSS's are different from that of second and third one. It occurs due to the transformation of the original multi-dimensional time-series data. In this process, the PCA regards coordinate 'y' as significant coordinate, whereas coordinate 'x' and 'z' are insignificant. So, the algorithm mainly extracts information based on the coordinate 'y'.

However, it has validity from the viewpoint of human motion. We can recognize intuitively the feature of motion while walking. That is, the coordinate 'y' (expressing the movement towards upper and lower sides), the coordinate 'x' (expressing the movement towards left and right), and the coordinate 'z' (expressing movement towards backward and forward).

In addition, by checking the human motion, we find the motif represents the movement of '*raising and lowering the actor's leg*'. We can intuitively understand that this movement is a characteristic pattern appeared frequently in the motion '*walking*'. So, we can say that the motif discovery algorithm can detect motif that represents the feature of the human motion. Therefore, our motif discovery algorithm is useful for analyzing various multi-dimensional time-series data.

5. Extended motif discovery (EMD) algorithm

As mentioned in Section 1, to make the motif discovery algorithm useful in real world problems, we must extend the motif discovery algorithm to discover a motif that is the DL pattern. We call this extended motif discovery algorithm '*EMD algorithm*'. In this section, we introduce two modifications for extending the motif discovery algorithm.

5.1. Modifying the BS sequence

The behavior symbol sequence $\tilde{C}$ represents a behavior of the given time-series data. So, it is preferred that we should detect patterns from the behavior symbol sequence. However, all patterns extracted from $\tilde{C}$ are always SL patterns. For example, we assume that a BSS pattern '*ABC*' is extracted from a BS sequence described in figure 9(a). Here, we recall
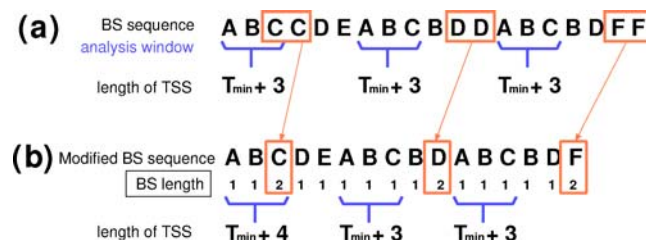

Figure 9.   (a) BS sequence obtained from the time-series data. (b) Modified BS sequence.

that each BS represents a behavior of each TSS of length T_{min}. Therefore, all TSS's that correspond to the BSS pattern '*ABC*' have the length of $T_{min} + 3$ and it is a SL pattern.

To solve this problem, we attempt to modify the original BS sequence. First, we detect BSS's where the same BS appears repeatedly, and count the number of it (this number is called '*BS length*'). For example, in figure 9(a), the third BS and the fourth BS are '*C*', so we find BSS '*CC*' with BS length of 2. Then, we transform such BBS into one unique BS. For example, '*CC*' is transformed into '*C*' with BS length 2. This is called '*modified BS sequence*'. By extracting BSS pattern from the modified BS sequence, we can detect the DL pattern.

Here, by using figure 10, we note the reason that DL patterns extracted from the modified BS sequence conform to the '*behavior constraint*'. In figure 10(a), the first BS '*A*' represents a behavior in area from 1 to 3. The TSS in area from 2 to 4 has the same behavior. So, the second BS is assigned also '*A*'. Here, it is inferred that the TSS in area from 1 to 4 has almost same behavior. Therefore, we can transform the first two '*AA*' into a single BS '*A*'. Similar to this example, we can transform the rest of such BSS into a single BS.

As mentioned above, we can extract DL patterns. But, we cannot calculate Euclidean distances between every two TSS's of DL patterns. Because two TSS's of DL patterns might have different lengths. Therefore, we must introduce DTW (Dynamic Time Warping) (Myers & Rabiner, 1981) distance function rather than Euclidean distance function. DTW calculates the distance between two TSS's whose lengths are different. We calculate DTW distances between every TSS's and make a 'distance matrix', such as described in figure 5(d).
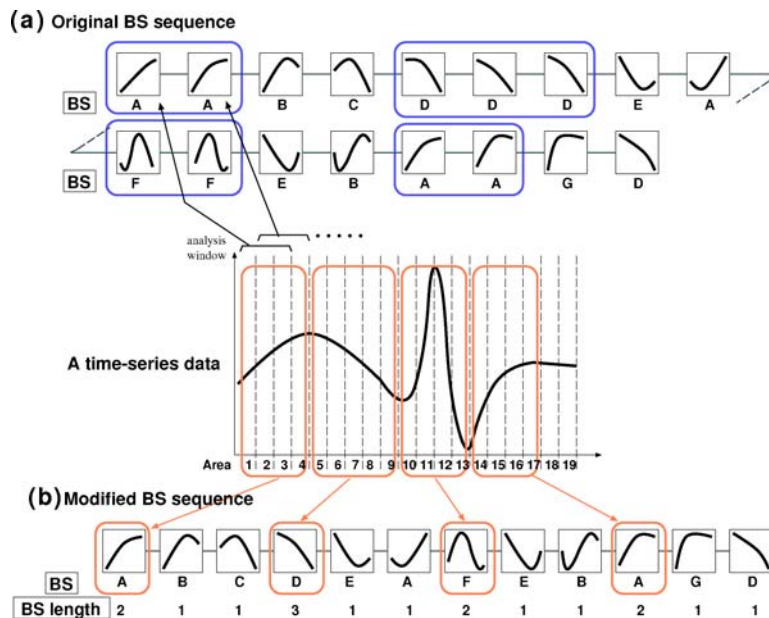

Figure 10. (a) An original BS sequence and (b) a modified BS sequence.

5.2. *New definition of description length of a time-series data*

We must augment the definition of description length according to the modified BS sequence. That is, for the purpose of extracting a motif of DL pattern, we must redefine the description length of a time-series data. Here, we assume that if we segment a time-series data using a motif, the cost for this segmentation is minimized. Therefore, we consider a new assumption for motif discovery: 'the best model of a time-series data is the best segmentation using the motif.' Based on the new assumption, we redefine the description length of time-series data using cost of the segmentation.

We are concerned about the significant behavior. This means that we are interested in the variation of BS sequences along time. This kind of variation can be measured by the number of bits in which BS sequence can be encoded. The motif using very few bits is simple, and the more bits mean more complex and significant. Secondly, according to MDL principle, the length of the motif should be short. Finally, the number of segments should be small. In order to, express our idea, we define new description length based on Chakrabarti, Sarawagi, and Dom (1998).

It is used for detecting "surprising patterns" in a binary time-series data '*market basket sequence*'. We apply the definition of the description length to the modified BS sequence. The new description length of a time-series data consists of three costs, '*data encoding cost*', '*parameter encoding cost*' and '*segmentation cost*'. *Data encoding cost* is the lower bound of description length that is required to encode each segment. This description length is calculated based on the Shannon's theorem. *Parameter encoding cost* is the description length that is required to describe the order of BS in each segment. *Segmentation cost* is required to describe the location of all segments.

We illustrate these costs in figure 11. We assume that a modified BS sequence $\tilde{C}$ is extracted from a time-series data in figure 11(a). In addition, we assume that we find the BSS pattern '*BCD*'. Then, we divide $\tilde{C}$ into m segments by '*BCD*'. For example, in figure 11(b), we obtain 7 segments. Next, we calculate the data encoding cost and the parameter encoding cost for each segment.

In case of calculating the data encoding cost of the i-th segment, we calculate the length of the i-th segment t_i. For example, in figure 11(c), the length of the first segment is $t_1 = 2$, the length of the second segment is $t_2 = 8$ and so on. In addition, we assume that the j-th BS has a length l_{ij}. A data encoding cost for the j-th BS in the ith segment is calculated as follows:

$$-l_{ij} \log_2 \frac{l_{ij}}{t_i} \tag{14}$$

For example, in figure 11(c), the first BS of the second segment is '*B*', and the data encoding cost of '*B*' is $-2 \log_2 \frac{2}{8}$. By calculating the data encoding cost of all unique BSs in the i-th segment, we obtain the data encoding cost of the segment as follows:

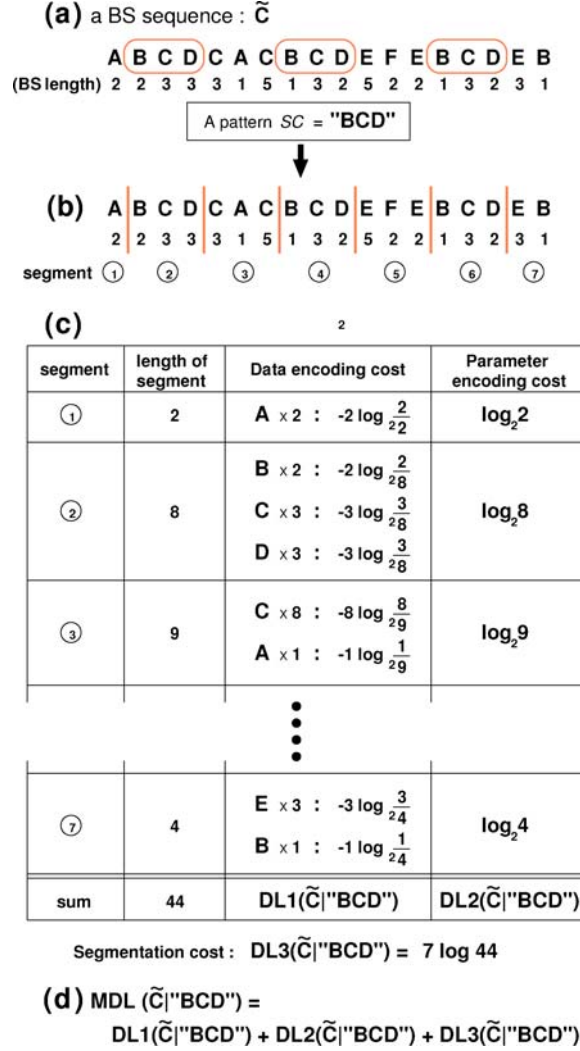$$\sum_j -l_{ij} \log_2 \frac{l_{ij}}{t_i} \tag{15}$$

Figure 11.   Calculation by using the new definition of the description length.

By using the following equation, we calculate the data encoding cost $DL1(\tilde{C} \mid SC)$ of $\tilde{C}$ that is segmented by the pattern SC:

$$DL1(\tilde{C} \mid SC) = \sum_{i}^{m} \sum_{j} -l_{ij} \log_2 \frac{l_{ij}}{t_i} \tag{16}$$

On the other hand, we calculate the parameter encoding cost of each segment as follows:

$$\log_2 t_i \tag{17}$$

For example, in figure 11(c), the parameter encoding cost of the first segment is $\log_2 2$, second segment is $\log_2 8$ and so on. Thus, we calculate it $DL2(\tilde{C} \mid SC)$ of $\tilde{C}$ as follows:

$$DL2(\tilde{C} \mid SC) = \sum_{i}^{m} \log_2 t_i \tag{18}$$

Next, we calculate the segmentation cost $DL3(\tilde{C}|SC)$ of $\tilde{C}$ as follows:

$$DL3(\tilde{C} \mid SC) = m \log_2 \left(\sum_{i}^{m} t_i \right) \tag{19}$$

For example, in figure 11(c), the length of $\tilde{C}$ is 44, so the segmentation cost is $7 \log_2 44$.

Finally, in figure 11(d), we obtain the description length of $\tilde{C}$ that is segmented by the pattern SC as follows:

$$MDL(\tilde{C} \mid SC) = DL1(\tilde{C} \mid SC) + DL2(\tilde{C} \mid SC) + DL3(\tilde{C} \mid SC) \tag{20}$$

We use Eq. (20) as the MDL estimation function for the MDL pattern detection algorithm. Thus, the EMD algorithm can extract a motif of DL pattern.

5.3. Experimental evaluation

In the section, we show the efficiency of EMD algorithm. In the experiment, we also use 3-dimensional time-series data sets of human motion.

5.3.1. Efficiency of EMD algorithm.   Figure 12(a) shows an example of motif extracted by EMD algorithm. Extracted motifs have the same behavior that includes a valley with high amplitude. In figure 12(b), we also find that the lengths of TSS are different from each other. In addition, all of them represent the movement of '*lowering the actor's head*'. So, we can say that the motif satisfies the definition of motif.

To compare EMD algorithm with our previous algorithm, we show a motif extracted by the previous algorithm in figure 12(c). Two TSS's of the motif also include the valley. However, the third valley indicated by an arrow is not extracted as a motif. The phenomenon occurs due to the fact that our previous algorithm can detect only SL patterns. On the other hand, EMD algorithm regards the TSS with the third valley as the motif because it can detect DL patterns. Hence, EMD algorithm is more efficient than our previous algorithm.

Furthermore, figure 13 shows a time-series data that represents an actor's right hand movement while performing '*kendo*'. '*Kendo*' is a series of different motions (attack, defense, strike etc.) like '*fencing*.' In figure 13(b), it is inferred that the lengths of three
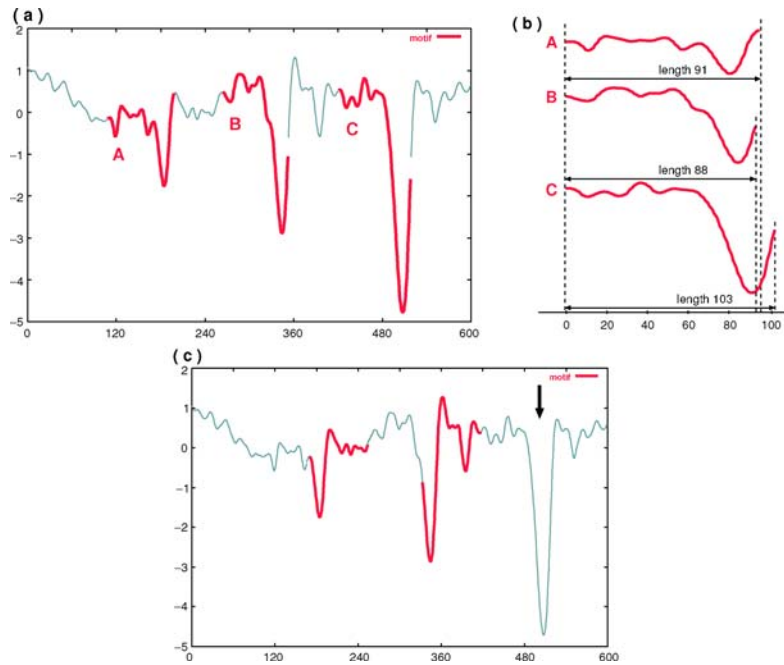
Figure 12. The motif discovered from the motion data '*Head movement while walking*'. (a) The motif extracted by using EMD algorithm. (b) A zoom-in of three TSS's of the motif. (c) A motif extracted by using our previous algorithm.
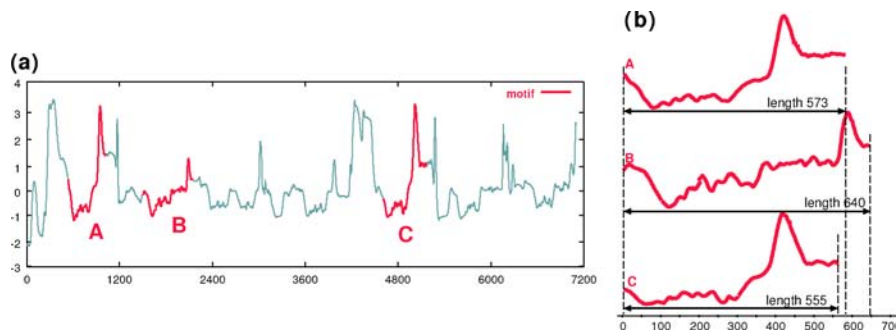

Figure 13. This data represents the right hand movement while performing '*kendo*'. (a) Three TSS's are extracted as a motif of '*kendo*' data. (b) A zoom-in shows the difference of their lengths.

TSS's are different from each other. However, we find intuitively that they have the same behavior. Therefore, the EMD algorithm is useful to extract motifs from the time-series data.

Next, to evaluate a motif obtained from higher dimensional data, we experimented on even higher dimensional data. Figure 14 shows an example of motif extracted from
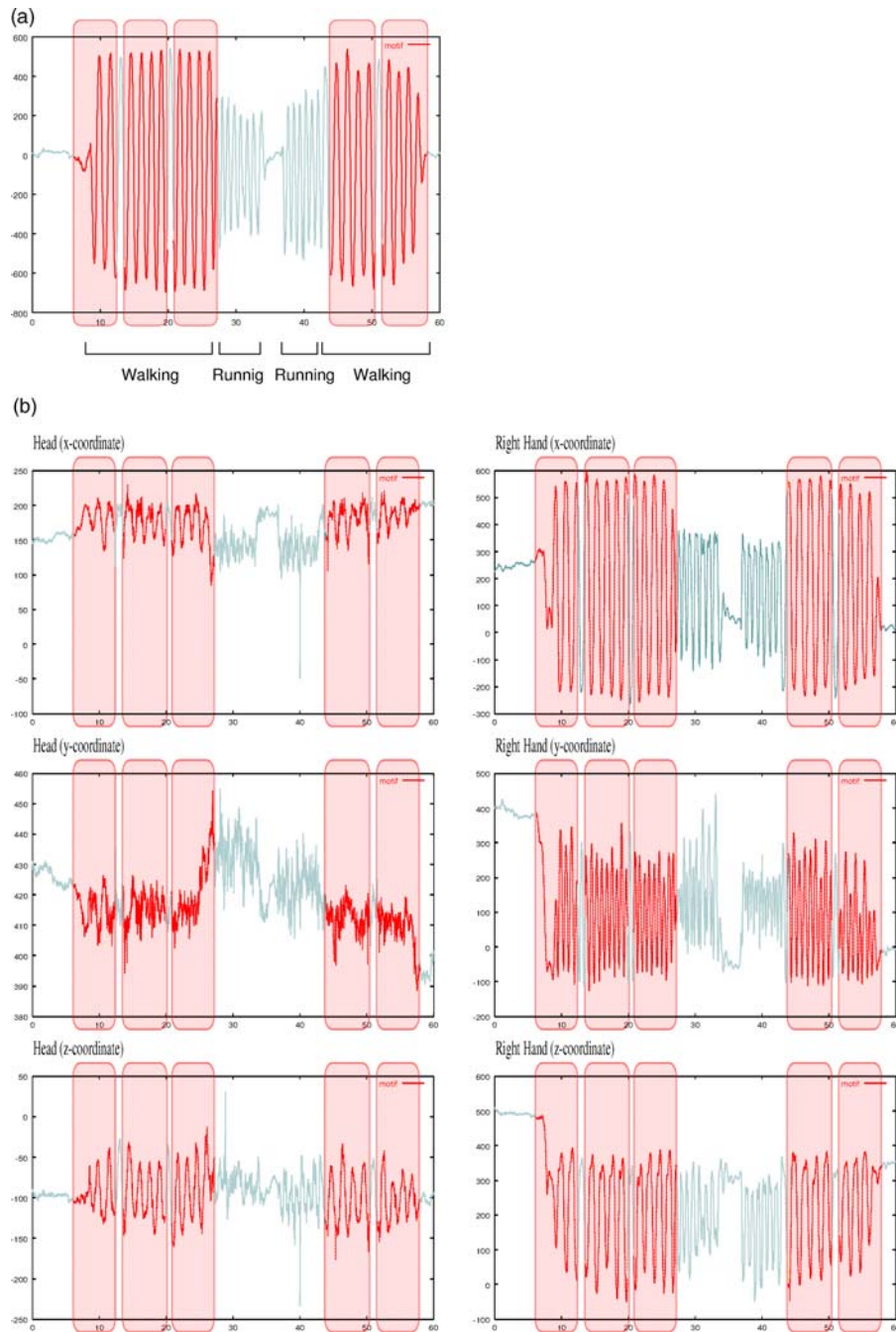
Figure 14.   This data represents 45-dimensional data while 'running' and 'walking'. (a) Five TSSs are extracted as a motif. This figure shows first principal component. (b) 6-dimensional time-series data.

45-dimensional time-series data. Figure 14(a) shows an example of motif extracted from first principal component of 45-dimensional time-series data. Figure 14(b) shows only 6-dimensional time-series data, although the original is 45-dimensional time-series data of actor's body parts while performing 'running' and 'walking'. Here, compared with the time-series data in figure 14(b), the first principal component data in figure 14(a) preserves characteristic of 'walking' and 'running' movements, and the discovered motif represents 'walking' movement. It can be said that the 'walking' movement pattern is correctly extracted as a motif. So, we can state that our approach is also useful for high dimensional data.

Up to here, we mainly experimented on a time-series data which is obtained from motion capture system. Next, we experiment on more general data. The purpose is to evaluate a motif extracted from general and public data. So, we experimented on the fetal ECG data (Keogh, 2004). The fetal ECG data in figure 15 is 8-dimensional time-series data, and each TSS of shaded region represents a motif. Figure 15(a) shows the first principal component. Figure 15(b) presents only 4-dimensional time-series data, although the original is 8-dimensional fetal ECG data. Here, in figure 15(a), the first principal component data preserves the large variance of each time-series in figure 15(b). So, the motif extracted from the data in figure 15(a) represents the characteristic pattern of time-series of each dimension.

Up to here, we mainly experimented on time-series data which are periodical and correlated with each other. Next, we experiment on the shuttle data (Keogh, 2004). The purpose is to evaluate the extraction of a motif from non-periodical data. The shuttle data in figure 16 is 6-dimensional time-series data, and each TSS in the shaded region represents a motif. Figure 16(b) presents the original time-series data. Figure 16(a) shows the first principal component. Here, the first two time-series data are correlated with each other. The others are also correlated with each other but almost independent relative to the first two series. But, in figure 16(a), the first principal component data preserves the behavior of original time-series data. So, although the shuttle data is not periodical data, our algorithm discovered two TSS's as a motif.

The readers may be concerned that a motif which is extracted from the first principal component is not representative for the motifs which are extracted from each original multi-dimensional time-series data separately. We extracted motifs from original 8-dimensional fetal ECG data separately in figure 17(a). Figure 17(a) shows only 4-dimensional time-series data, although the original is 8-dimensional time-series data. Figure 17(b) shows a motif extracted from the first principal component. In figure 17, the frequency and occurrence of TSS's in A, B and D are almost same and correlated with each other. The first principle component indicates that there is a considerable inter-correlation between A, B and D. But, the frequency of TSS's occurrence in C differs from that expected. Therefore C is not correlated with A, B, and D, the contribution ratio of C for first principal component is low. So, we consider that a motif extracted from the first principle component may not be representative, if the contribution ratio of the time-series data is low.

5.3.2. Usefulness of motifs to data mining tasks.   Each time-series data described in figure 18 represents the movements of left shoulder, left biceps, left arm and left hand while
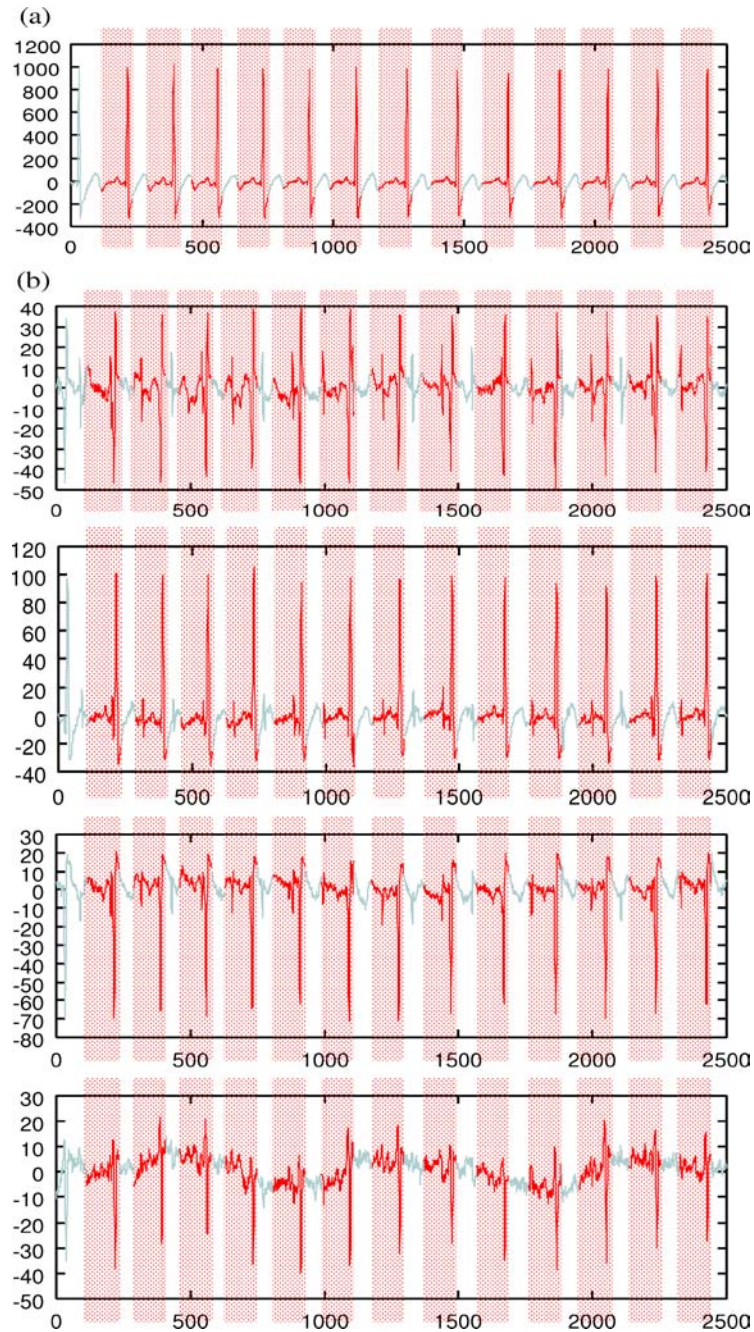
Figure 15. This data represents the 8-dimensional fetal ECG data. (a) Thirteen TSSs are extracted as a motif of the fetal ECG data. (b) 4-dimensional fetal ECG data.
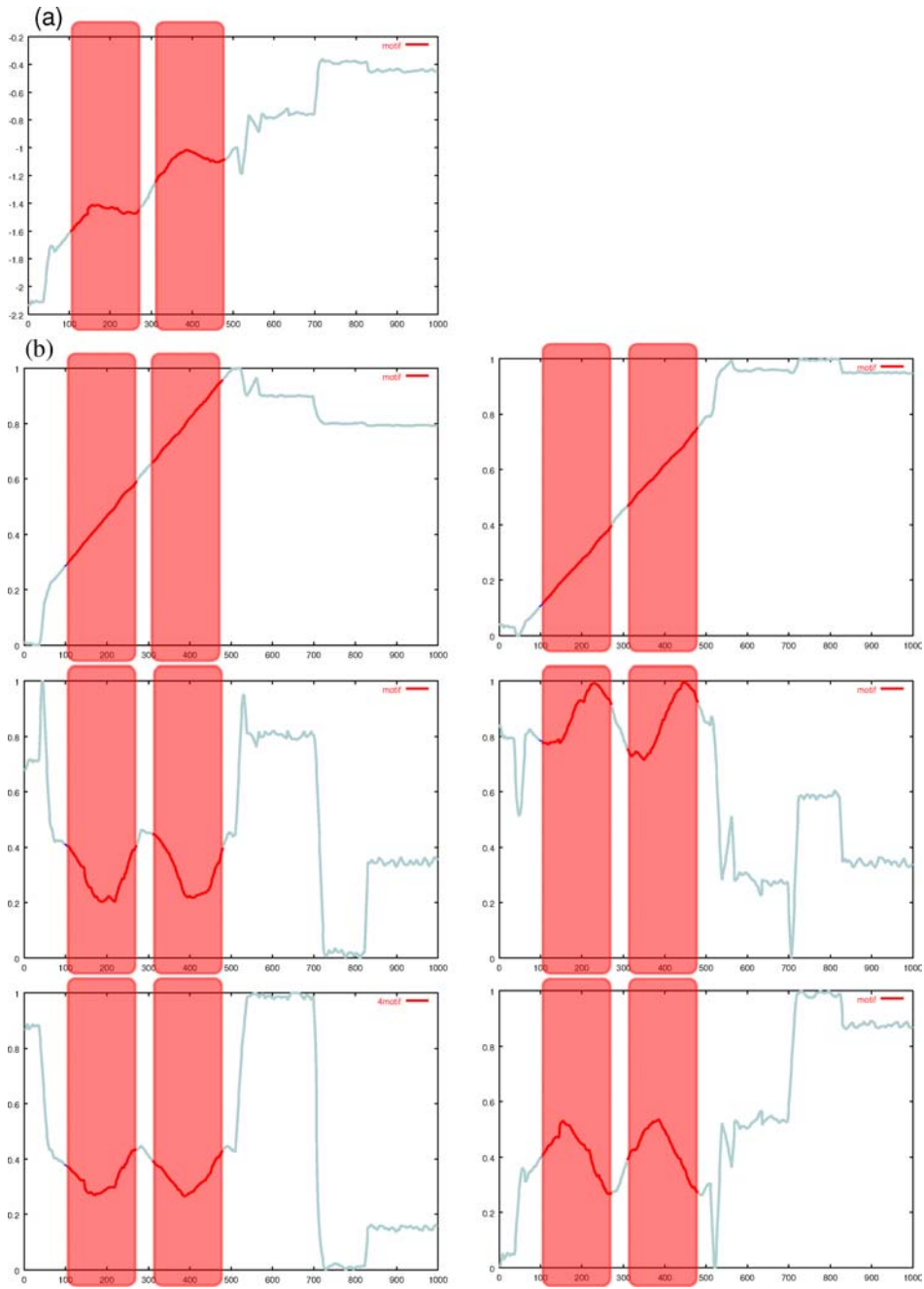
Figure 16. This data represents the 6-dimensional shuttle data. (a) Two TSSs are extracted as a motif of the shuttle data. This figure shows the first principal component. (b) Original 6-dimensional shuttle data.
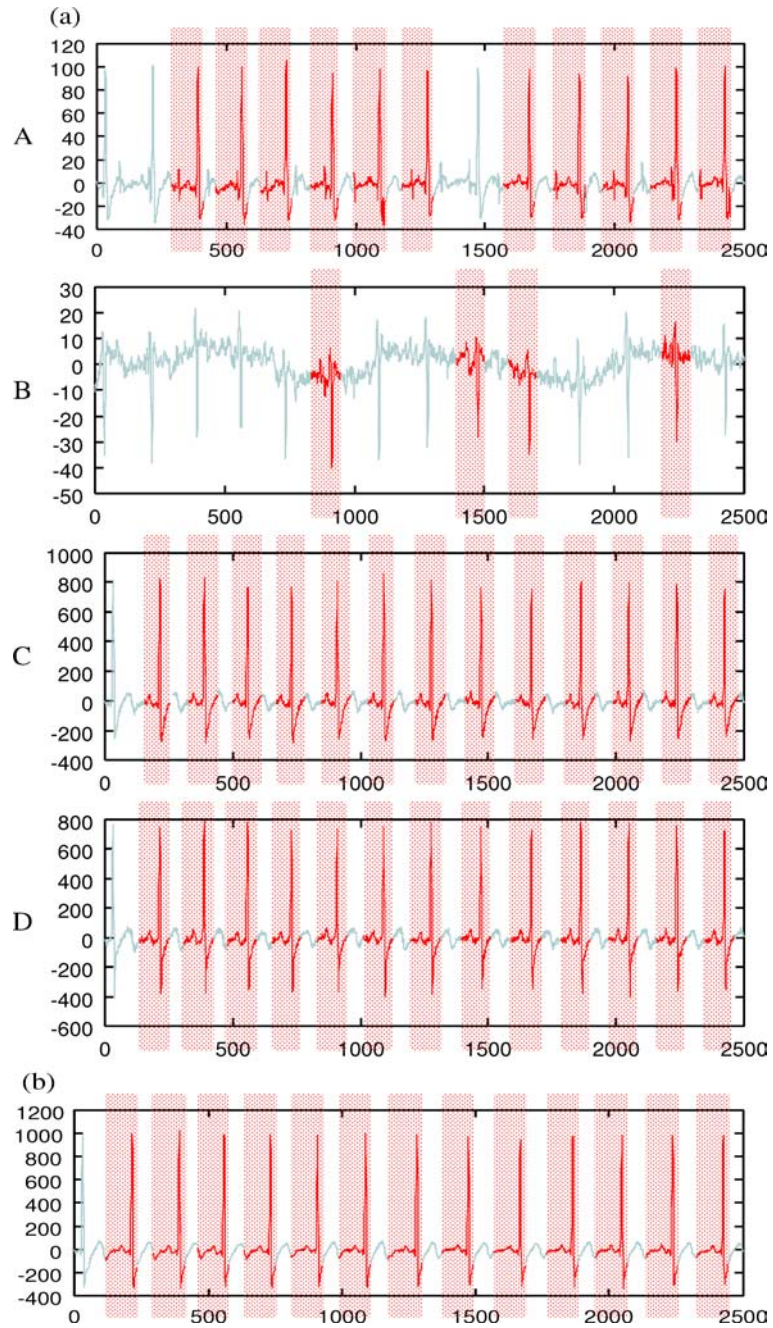
Figure 17.   This data represents the 8-dimensional fetal ECG data. (a) 4-dimensional fetal ECG data. Each TSS is extracted from a time-series separately. (b) Thirteen TSSs are extracted from the first principle component of the fetal ECG data.
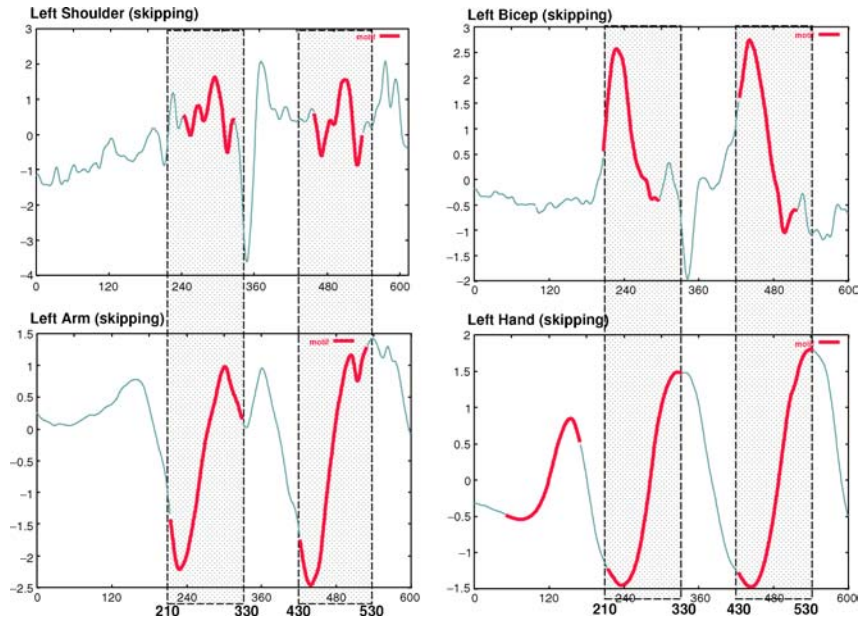
Figure 18. Each time-series data represents the movement of left shoulder, left bicep, left arm and left hand while an actor is skipping.

an actor is skipping. Each motif has different behavior from the others. However, we find an interesting phenomenon that every motif appears at almost the same time. In the example, every motif appears during 210 to 330 and 430 to 530. Here, by comparing these motifs with a real human motion, we find that these motifs represent the movement of '*raising arm*'. So, we can assume that '*raising arm*' has a big influence on the human motion '*skipping*'. Therefore, we consider that we can model the human motion '*skipping*' by using these motifs.

Furthermore, figure 19 shows motifs that represent the movement of head and right foot while an actor performs '*skipping*', '*running*' and '*walking*' respectively. For each kind of human motion, both head movement and right foot movement appear at almost same time. From the result, we assume that there is a relationship between the head and right foot when the actor performs various actions. As described above, we consider that the motif detection is useful for discovering association rules of the multi-dimensional time-series data.

6.   Open problems

There are several directions to extend this work. One of them is to dynamically determine several parameters in our motif discovery algorithm. These are a number of segments, unique SAX symbols and the threshold of distance R. In particular, the number of
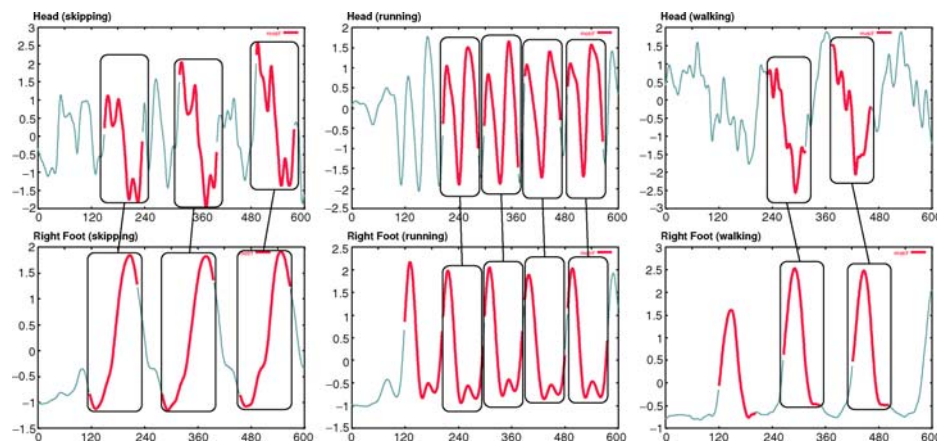
Figure 19. Comparing the motif the head movement with that of the right foot movement.

segments of PAA representation has a big influence on the result of our motif discovery algorithm. Figures 20 and 21 show PAA representations using 5, 10 segments respectively.

Here, we will compare figure 20(b) with figure 21(b). While being segmented differenty, we can obtain the same behavior, symbol sequence ABC. On the other hand, we can get the behavior symbol sequence ABC from figure 20(c) whereas DEF from figure 21(c).

However, from the view point of Shannon's sampling theorem (Shannon, 1949), it is not surprising that the phenomenon has occurred in the above example, because the phenomenon is similar to '*aliasing*' in the theorem. The theorem states that, when sampling a signal at discrete intervals, the '*sampling frequency*' must be greater than twice of the '*highest frequency*' of the input signal to reconstruct the original signal perfectly from the sampled version. If the '*sampling frequency*' is less than twice of the '*highest frequency*', then the frequencies in the original signal that are above half of the sampling rate will be aliased and will appear in the reconstructed signal as lower frequencies. By the phenomenon '*aliasing*', PAA representation of figure 20(B) cannot reconstruct the original data.

Here, we consider that the number of segments of PAA representation is equal to the number of the sampling frequency described in the Shannon's sampling theorem. However, note that our motivation is to represent only significant behavior of a given time-series data. So, based on the Shannon's sampling theorem, if we find the '*highest frequency*' of a time-series data for representing at least significant behavior, the number of segments should be larger than the number of the '*highest frequency*'. However, there is no criterion for determining which behavior is significant for the time-series data, so we cannot find such '*highest frequency*'.

Another problem is about discovering DL pattern. We can not extract TSS's whose lengths are greatly different from each other though they have the same behavior. We
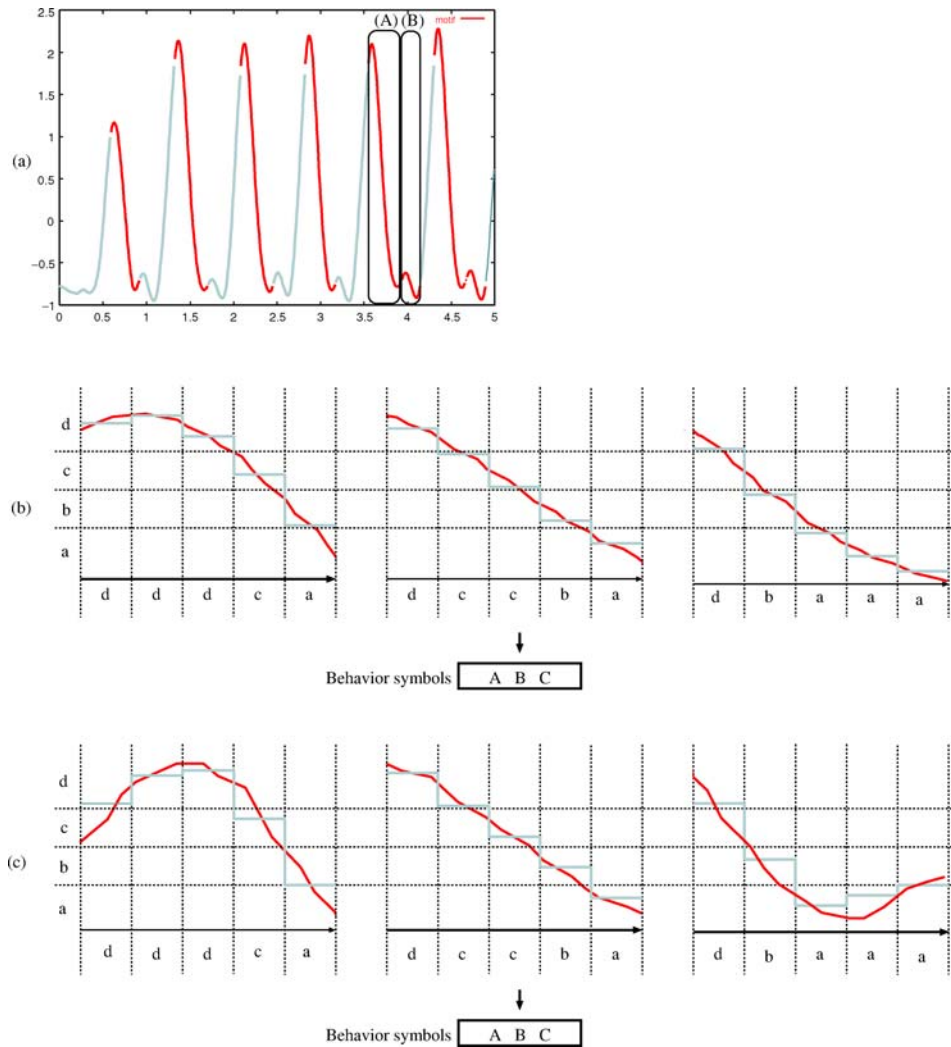
Figure 20. A result of a motif using 5 PAA segments. (a): (A) and (B) is analysis window. (b) By shifting analysis window, we obtain 3 behavior symbols of TSS in (A). (c) By shifting analyasis window, we obtain 3 behavior symbols of TSS in (B).

illustrate the situation in figure 22. All three TSS's have the same behavior with different lengths. For example, in TSS (a) is transformed into '*AABCD*', and TSS (b) is transformed into '*ABCD*'. However, by transforming the BSS (a) into '*ABCD*', they are quite same. On the other hand, the TSS (c) has the length that is greatly different from TSS (b). In figure 22, BSS (c) is quite different from BSS (b) although their behaviors are almost same.
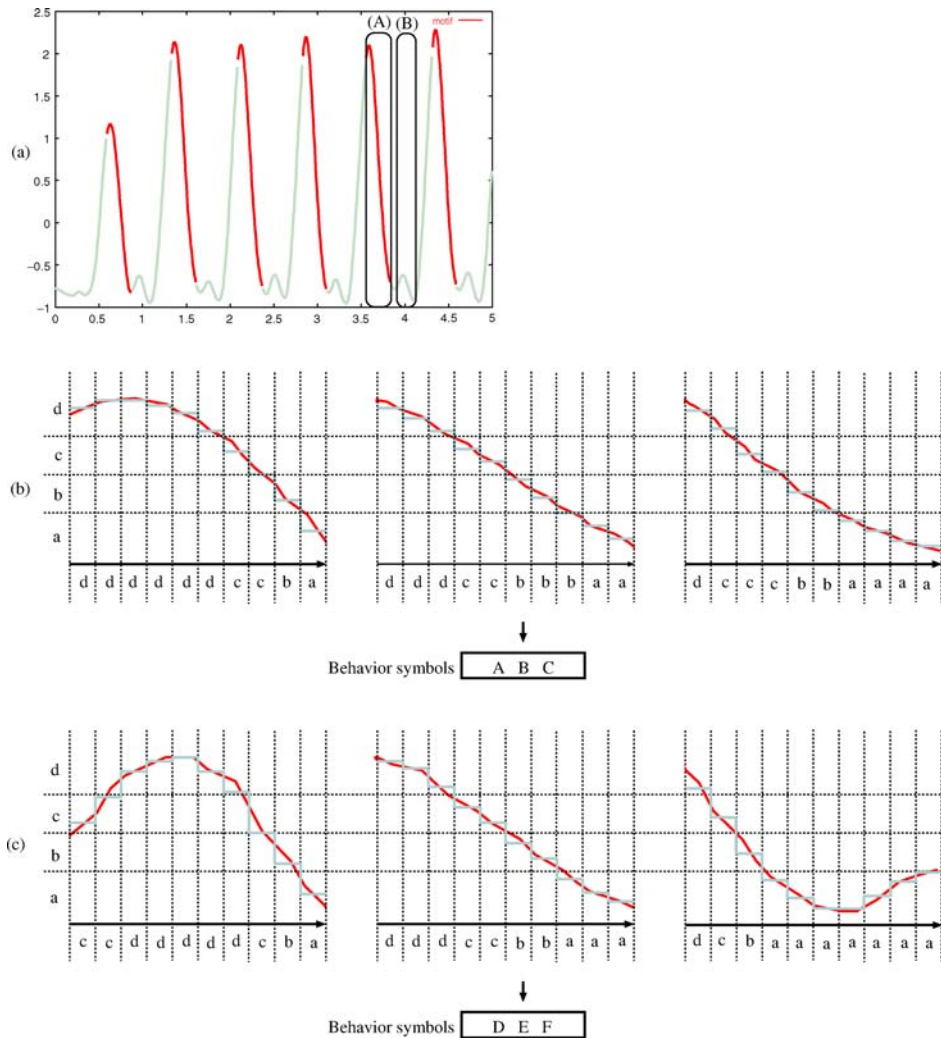
Figure 21. A result of a motif using 10 PAA segments. (a) (A) and (B) is analysis window. (b) By shifting analysis window, we obtain 3 behavior symbols of TSS in (A). (c) By shifting analyasis window, we obtain 3 behavior symbols of TSS in (B).

For instance, in experiment, figure 23 shows the movement of the right foot while 'walking and running' in coordinate 'y' (expressing the movement towards the upper and lower sides). All TSS's indicated by 'A' to 'D' represent '*the movement of feet while walking*.' Here, we can find that 'E' has the same behavior. However, from figure 23(b), the length of 'E' is different from them. Therefore, 'E' and TSS's from 'A' to 'D' doesn't belong to the same pattern.
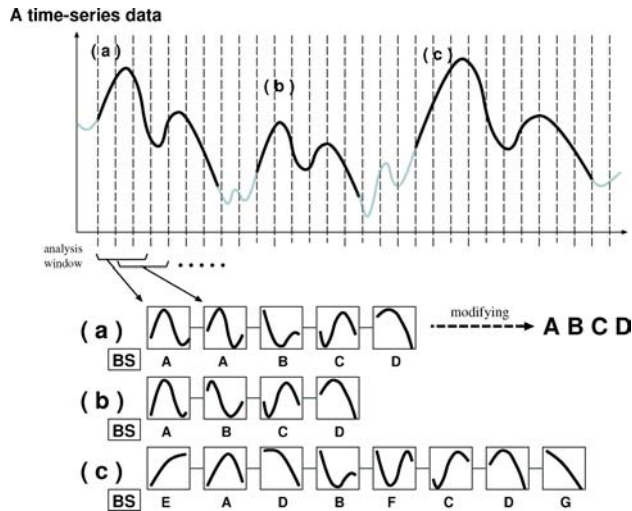
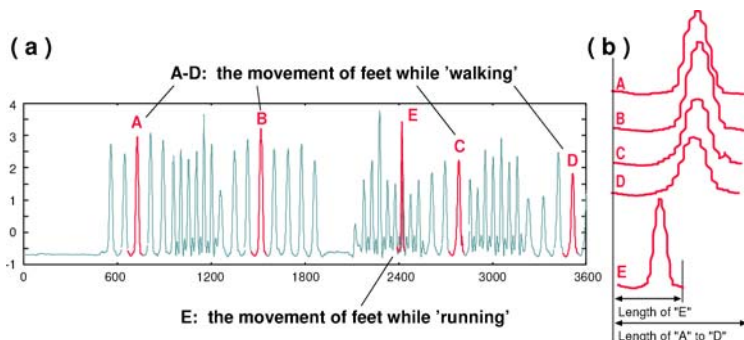*Figure 22.*   BS sequences with different lengths.

Figure 23.   (a) A time-series data 'feet movement while running and walking' obtained from motion capture system. (b) A zoom-in reveals the difference between the length of 'E' and the lengths of 'A' to 'D'.

7.   Conclusions

In this paper, we presented an algorithm for efficiently discovering a motif of DL pattern from multi-dimensional time-series data. We proved our algorithm's advantage that, it can extract a motif that can be recognized intuitively by human. From the result of our experiments, it is evident that our algorithm is effective to mine the various unexpected periodicities and extract association rules from time-series, etc. However, to make the algorithm more useful in the real world data, we need further development of the motif discovery algorithm by solving the problem mentioned above.

References

Akaike, H. (1969). Fitting autoregressive model for prediction. *The Annals of the Institute of Statistical Mathematics, 21*, 243–247.

Berberidis, C., Vlahavas, I., Aref, W. G., Atallah, M., & Elmagarmid, A. K. (2002). On the discovery of weak periodicities in large time series. In *Proc. of PAKDD 2002* (pp. 51–61).

Buhler, J., & Tompa, M. (2001). Finding motifs using random projections. In *Proc. of 5th International Conference on Computer Molecular Biology* (pp. 67–74).

Caraca-Valente, J. P., & Lopez-Chavarrias, I. (2000). Discovering similar patterns in time series. In *Proc. of 6th ACM SIGMOD International Confference on Knowledge Discovery and Data Mining* (pp. 126–133).

Chakrabarti, S., Sarawagi, S., & Dom, B. (1998). Mining surprising patterns using temporal description length. In *Proc. of 24th International Conference on Very Large databases* (pp. 606–617).

Chiu, B., Keogh, E., & Lonardi, S. (2003). Probabilistic discovery of time series motif. In *Proc. of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 493–498).

Colomer, J., Melendez, J., & Gamero, F. I. (2002). Pattern recognition based on episodes and DTW, application to diagnosis of a level control system. In *Proc. of 16th International Workshop on Qualitative Reasoning* (pp. 37–43).

Cyril, G., Peter, T., Egill, R., Arup, N. F., & Kai, H. L. (1999). On clustering fMRI time series. *NeuroImage, 9:3*, 298–310.

Das, G., Lin, K., Mannila, H., Renganathan, G., & Smyth, P. (1998). Rule discovery from time series. In *Proc. of the 4th ACM SIGMOD International Conference on Knowledge Discovery and Data Mining* (pp. 16–22).

Fradkin, D., & Madigan, D. (2003). Experiments with random projections for machine learning. In *Proc. of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 517–522).

Friedman, J. H., & Tukey, J. W. (1974). A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on Computers, 23:9*, 881–889.

Heras, D. B., Cabaleiro, J. C., Perez, V. B., Costas, P., & Rivera, F. F. (1996). Principal component analysis on vector computers. In *Proc. of Vector and Parallel Processing 1996* (pp. 416–428).

Hyvrinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks, 13:4*, 411–430.

Kashino, K., Smith, G., & Murase, H. (1999). Time-series active search for quick retrieval of audio and video. In *Proc. of 1999 International Conference on Acoustics, Speech and Signal Processing* (pp. 2993–2996).

Keogh, E. (2004). The UCR Time Series Data Mining Archive [http://www.cs.ucr.edu/~eamonn/TSDMA/datasets.html]. Riverside CA, University of California: Computer Science & Engineering Department.

Koopman, S. J., & Ooms, M. (2003). Time-series modeling of daily tax revenues. *Statistica Neerlandica, 57:4*, 439–469.

Levin, A. U., Leen, T. K., and Moody, J. E. (1993), Fast pruning using principal components. In *Proc. of NIPS '93* (pp. 35–42).

Lin, J., Keogh, E., Lonardi, S., & Chiu, B. (2003). A symbolic representation of time series, with implications for streaming algorithms. In *Proc. of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*.

Lin, J., Keogh, E., Lonardi, S., & Patel, P. (2002). Finding motifs in time series. In *Proc. of the 2nd Workshop on Temporal Data Mining* (pp. 53–68).

Mori, T., & Uehara, K. (2001). Extraction of primitive motion and discovery of association rules from human motion. In *Proc. of the 10th IEEE International Workshop on Robot and Human Communication* (pp. 200–206).

Myers, C. S., & Rabiner, L. R. (1981). A comparative study of several dynamic time-warping algorithms for connected word recognition. *The Bell System Technical Journal, 7:60*, 1389–1409.

Navarro, G., & Baeza-Yates, R., (1999). Fast multi-dimensional approximate pattern matching. In *Proc. of the 10th Annual Symposium on Combinatorial Pattern Matching* (pp. 243–257).

Rissanen, J. (1989). *Stochastic Complexity in Statistical Inquiry*, Vol. 15. World Scientific Publishing.

Schwarz, G. (1981). Estimating the dimension of a model. *The Annals of Statistical Mathematics, 6:2*, 461–464.

Shannon, C. E. (1949). Communication in the presence of noise. In *Proc. of 2nd Institute of Radio Engineers* (pp. 10–21).

Tanaka, Y., & Uehara, K. (2003). Discover motifs in multi dimensional time-series using the principal component analysis and the MDL principle. In *Proc. of 3rd International Conference on Machine Learning and Data Mining in Pattern Recognition* (pp. 252–265).

Vigario, R., Jousmaki, V., Hamalainen, M., Hari, R., and Oja, E. (1998), Independent component analysis for identification of artifacts in magnetoencephalographic recording. In *Proc. of NIPS '97* (pp. 229–235).

Vlachos, M., Hadjieleftheriou, M., Gunopulos, D., & Keogh, E. (2003). Indexing multi-dimensional time-series with support for multiple distance measure. In *Proc. of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 216–225).