A Symbolic Representation of Motion Capture Data for Behavioral Segmentation

Ruxiang Wei, Weibin Liu
Institute of Information Science
Beijing Jiaotong University
Beijing 100044, China
e-mail: wbliu@bjtu.edu.cn

Weiwei Xing
School of Software Engineering
Beijing Jiaotong University
Beijing 100044, China

***Abstract*—For building and understanding computational models of human motion, behavioral segmentation of human motion into actions is a crucial step, which plays an important part in many domains such as motion compression, motion classification and motion analysis. In this paper, we present a novel symbolic representation of human motion capture data, called the Behavior String (BS). Based on the BS, a further motion segmentation algorithm for human motion capture data is proposed. The human motion capture data is treated as a high-dimensional discrete data points, which are clustered by an alternative algorithm based on density, and each cluster is divided into a character. Then, the BS is produced for the motion data by temporal reverting. By analyzing the BS, the human motion capture data is segmented into distinct behavior segments and the cycles of motion are found. Experiments show that our method not only has a good performance in behavioral segmentation for motion capture data, but also finds cycles of motion and the motion clips of the same behaviors from long original motion sequence.**

Keywords-motion analysis; behavioral segmentation; clustering; motion capture data; cycle

I. INTRODUCTION

Computer animation technology has been extensively employed in film and television production, entertainment and games, virtual reality, and many other fields [1]. Human animation is the most challenging topics in computer animation. Motion capture technique has become a major means of human body animation production which has been applied extensively in recent years. For the benefit of motion compression, motion classification and motion analysis in computer graphics, behavioral segmentation is part of the hot research topics, the motivation of which is to get the independently semantic subsequences from long original motion sequence [2], as shown in Fig. 1. It is very tedious to segment long sequences by hand. Therefore, the research of motion capture data segmentation algorithm is of major importance. This paper focuses on efficient and quite robust technique which is able to automatically create a segmentation even when the behaviors have not been seen before.

A new symbolic representation of human motion capture data for behavioral segmentation is proposed from this paper, which provides two key abilities: the ability to gets a higher rate of recall and precision, the ability to finds the cycles of motion and extracts the motion clips which represent the same behaviors from long original motion sequence. Our method can be simply described as follows: Each frame in the motion capture data is considered as a point. Thus, human motion capture data is treated as a high-dimensional discrete data points. These points are clustered by an alternative algorithm based on density [3]. Then, the points reordered according to the order of the original frames, and different clusters are indicated by different letters. Thus, a symbolic representation of motion capture data is produced which is called the BS. The human motion capture data is segmented into distinct behavior segments and the cycles of motion are discovered by analyzing the BS. From experimental results, we have found that this algorithm has a very good performance in behavioral segmentation for motion capture data.

The paper is organized as follows. In Section II, we discuss related work. In Section III, we detail how to create the BS including computing the distance, clustering and the parameters setting. In Section IV, we describe the analyzing of BS, so that the segmentation points and the cycles will be discovered. In Section V, we analyze experimental results and compare with other state-of-the-art motion segmentation methods. Finally, in Section VI we provide the conclusions and provide directions for future work.

II. RELATED WORK

Finding the segmentation points of different behaviors for virtual humans with motion capture data is a challenging task that has seen much work over the last decade. Numerous researches have made efforts and a series of achievements, but still have some problems.

A few years ago, for the method based on classifier, Arikan et al. [4] constructed Support Vector Machine (SVM) to realize human motion segmentation with the manually annotation training database. But this method depended on large training

This research is partially supported by National Natural Science Foundation of China (No. 61370127, No.61100143, No.61473031, No. 61472030), Program for New Century Excellent Talents in University (NCET-13-0659), Fundamental Research Funds for the Central Universities(2014JBZ004), Beijing Higher Education Young Elite Teacher Project (YETP0583). The opinions expressed are solely those of the authors and not the sponsors.
{ Corresponding author: Weibin Liu, wbliu@bjtu.edu.cn }

DOI reference number: 10.18293/DMS2015-004

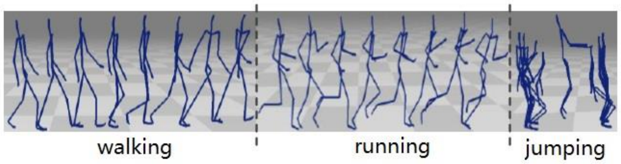


Figure 1. Segmenting motion capture data into different behaviors.

datasets and empirical value. For the method based on dimension reduction. Barbic et al. [5] assumed that different human motion behaviors can be represented with different intrinsic dimensionality, and realized the behavioral segmentation by subspace analysis of PCA theory. For the longer human motion data, however, the recall and precision through this method were lower [5]. For the method based on dimension model. Lu et al. [6] presented a two-threshold, multidimensional segmentation algorithm to automatically decompose a complex motion into a sequence of simple linear dynamic models. However, the scope of application was restricted. This method could only process human motion data with repetition period [6]. Barbic et al. [5] proposed PPCA segmentation method, which assumed that different motion classes belonged to different Gaussian distributions and could assign behavioral cuts with Mahalanobis distance. In addition, Gaussian Mixture Model (GMM) is utilized to cluster motion sequences with distinct motion classes. Nevertheless, users needed to set the number of motion classes beforehand.

In recent years, for the method based on clustering. Zhou et al. [7] used Aligned Cluster Analysis (ACA) method to segment human motion data, measured the similarities among diverse human motion sequences through dynamic time warping kernels, and accomplished motion segmentation with kernel k-means classification. However, users required to determine the cluster number with respect to temporal constraint [7]. In the past two years, Zhou et al. [8] utilized HACA to combine kernel k-means with the generalized dynamic time alignment kernel to cluster time series data. Moreover, HACA provides a natural framework to locate a low-dimensional embedding for the time series. HACA is efficiently optimized with a coordinate descent strategy and dynamic programming. While this method depended on the choice of the kernel parameters and the functional form of the kernel [8]. Yang et al. [9] presented a symbolic representation of motion capture data which called the Motion String (MS). MS was used to segment human motion data. But users needed to set the number of motion classes beforehand. On the other hand, this algorithm could only segment of the human motion data which have small changes between each behaviors, and could not segment the data which have complex behaviors [9]. Beaudoin et al. [10] developed a string-based motif-finding algorithm which allows for user-controlled compromise between motif length and the number of motions in a motif. Motion motifs represent clusters of similar motions and together with their encompassing motion graph they lend understandable structure to the contents and connectivity of substantial motion datasets.

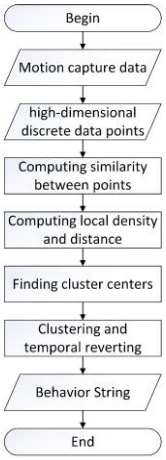

Begin
Motion capture data
high-dimensional discrete data points
Computing similarity between points
Computing local density and distance
Finding cluster centers
Clustering and temporal reverting
Behavior String
End

Figure 2. The procedure of creating the BS.

To solve the shortcomings and limitations of existing motion segmentation methods, inspired by the Motion String [9] and Motion-Motif Graph [10]. This paper proposes a novel symbolic representation of human motion capture data, which called the Behavior String (BS). Based on the BS, a further motion segmentation algorithm for human motion capture data is proposed. This algorithm is able to automatically find the number of cluster centers, while the Motion String algorithm needs the user to set the number of motion classes beforehand which is impractical. Our method can also find the cycles of motion and the motion clips of the same behaviors from long original motion sequence which the Motion String cannot do. Distinct from the Motion String which treats the string to periodic substrings, static substrings and transitional substrings [9]. In this paper, a string matching method is presented to find cut points and the cycles of motion, which regards each letter as a part of a meaningful behavior rather than a transition between different behaviors.

III. CREATING THE SYMBOLIC REPRESENTATION

In this section, the procedure of creating the BS which is a symbolic representation of motion capture data is presented, as shown in Fig.2. First, compute the similarity between every two points which are regarded as the frames in the motion capture data sequences. Second, calculate the local density and the distance for the clustering method. Third, find cluster centers which have a large local density and distance at the same time. Finally, cluster each remaining point which is not the cluster centers and recorded all points with the time sequence.

A. *Computing similarity between points*

To build the behavior strings, we need to cluster each frame from one of the human motion capture data sequences which also called the long original motion sequence. We regard each frame as a high-dimensional point, and compute the distance between every two points. Lee et al. [1] presented an algorithm to calculate the distance between two frames from the motion sequence and Wang et al. [2] improved this algorithm. But these algorithms only compute the principal joints and ignore other joints. So that these algorithm cannot distinguish the relatively complex behaviors. Therefore, we use the algorithm [11] which compute not only the difference of pose but also the velocity between the two frames.

This paper employs the following human skeleton model which has 31 joints shown in Fig.3. There are 62-dimensional in one frame, including root position vector, root orientation vector

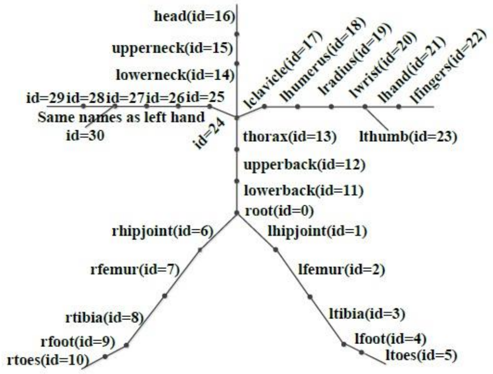


Figure 3. The human skeleton model.

and other joints' direction vector. The ith frame's pose consists of all joints rotation angle in the ith frame except the root position vector and the root orientation vector which including 6-dimensional. Each pose $p_i = \{a_{i,1}, a_{i,2}, a_{i,3}, \dots, a_{i,56}\}$ is represented as a point in 56-dimensional space, where $a_{i,m}$ is one of an Euler angle. The ith frame's velocity v_i is computed by the Euclidean distance between p_{i+1} and p_i. Especially the last frame's velocity v_n is equal to the last but one frame's velocity v_{n-1}.

$$v_i = \begin{cases} \sqrt{\left(a_{i+1,1} - a_{i,1}\right)^2 + \left(a_{i+1,2} - a_{i,2}\right)^2 + \cdots + \left(a_{i+1,56} - a_{i,56}\right)^2}, & i \neq n \\ v_{i-1}, & i = n \end{cases} \tag{1}$$

We calculate the distance by:

$$d_{ij} = \alpha p_{ij} + \beta v_{ij} \tag{2}$$

Where the p_{ij} is the Euclidean distance of pose between the ith frame and the jth frame. The v_{ij} is the difference of velocity between the ith frame and the jth frame. The α and the β are the weights. In our experiments, the α and the β is equal to 1. With this method, we compute the all distance between any two frames. Then we get the distance matrix $\mathrm{D}_{n\times n}$, where n is the length of the long original motion sequence. Obviously, $d_{ij} = d_{ji}$ $(i \neq j)$, and $d_{ij} = 0$ $(i = j)$.

B. *Local density and distance for clustering*

This clustering approach based on the idea that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities [3].

For each point i, we compute two quantities: its local density ρ_i and its distance δ_i from points of higher density. Both of the quantities only depend on the distances d_{ij} between every two points. We use the Gaussian kernel function [12] [13] to compute the local density ρ_i:

$$\rho_i = \sum_j e^{-\left(\frac{d_{ij}}{d_c}\right)^2} \tag{3}$$

Where d_c is a distance which is the parameter we need to set in this clustering algorithm. The ρ_i is larger when there are more points closer than d_c to point i.

We reorder the local density $\{\rho_i\}_{i=1}^{n}$ in descending order, and use $\{q_i\}_{i=1}^{n}$ to indicate the local density's subscript. Thus $\rho_{q_1} \geq \rho_{q_2} \geq \cdots \geq \rho_{q_n}$. The distance δ_i can be defined as:

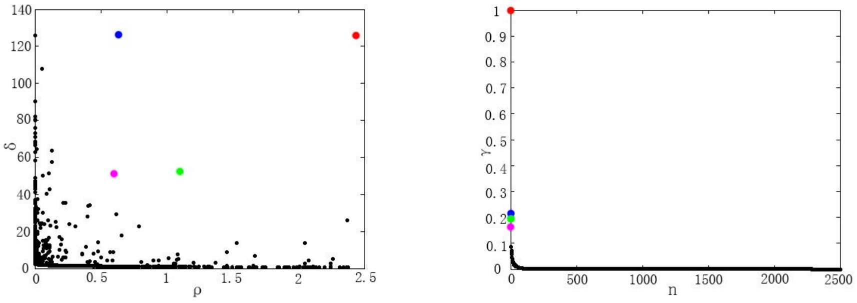
$$\delta_{q_i} = \begin{cases} \min\limits_{\substack{q_j \\ j<i}} \left\{d_{q_i q_j}\right\}, & i \geq 2 \\ \max\limits_{j\geq 2} \left\{\delta_{q_j}\right\}, & i = 1 \end{cases} \tag{4}$$

Obviously if the point i has the biggest local density, the distance δ is the maximum distance of other points. Otherwise, the distance δ is the minimum distance from points with higher densities.

C. *Finding cluster centers*

According to our assumption in the beginning of 3.2. Cluster centers have a large ρ and δ at the same time. Considering the ρ and δ may have a different order of magnitude. We multiply ρ by δ after normalization, and let the results denoted by $\gamma_i = \rho_i \times \delta_i$. Then we reorder the $\{\gamma_i\}_{i=1}^{n}$ in descending order, and let $\{\gamma_j\}_{j=1}^{n}$ indicate this sequence. Thus, cluster centers are recognized as points for which the value of γ is anomalously large. The first j points which can be found by the maximum j satisfying $\gamma_\mathrm{j} - \gamma_{\mathrm{j}+1} > \theta$. The θ is a threshold which is set to 0.05 in our Experiments.

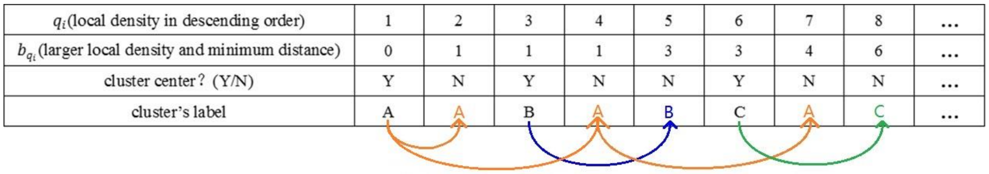
We select a long original motion sequence from the CMU database [14], which has 2500 frames including 'running' and 'walking'. Fig. 4(a) shows the ρ and δ computed from these



(a)

(b)

Figure 4. Cluster analysis of the human motion capture data from CMU Database. (a) The ρ and δ computed from 2500 frames that selected from the CMU database. (b) The value of $\gamma_i = \rho_i \times \delta_i$ in decreasing order for the data in (a). Colored points correspond to cluster centers.



q_i(local density in descending order)	1	2	3	4	5	6	7	8	…
b_{q_i}(larger local density and minimum distance)	0	1	1	1	3	3	4	6	…
cluster center？(Y/N)	Y	N	Y	N	N	Y	N	N	…
cluster's label	A	A	B	A	B	C	A	C	…

Figure 5. Clustering other points except the cluster centers. Each remaining point is assigned to the same cluster as its nearest neighbor of higher density.

frames. Fig. 4(b) shows the $\{\gamma_j\}_{j=1}^{2500}$ in descending order. Colored points correspond to cluster centers. Notice that there are 4 cluster centers, each cluster center is not indicate each behavior, but represent the feature of behaviors.

D. Clustering and temporal reverting

After the cluster centers have been found, each remaining point is assigned to the same cluster as its nearest neighbor of higher density, shown in Fig. 5. A detailed description of this procedure is as follows. We define the b_i as:

$$b_{q_i} = \begin{cases} \arg\min\limits_{\substack{q_j \\ j<i}} \left\{d_{q_i q_j}\right\}, & i \geq 2 \\ 0, & i = 1 \end{cases} \qquad (5)$$

The $\{b_i\}_{i=1}^{n}$ indicate a sequence number of the point which has a larger local density than i$^{\text{th}}$ point and a minimum distance. The $\{q_i\}_{i=1}^{n}$ is defined in 3.2 which indicate the subscript of local density. First, we classify the cluster centers. Then, each remaining point will be classified in the cluster that belongs to b_{q_i}. For example, if the $q_m$$^{\text{th}}$ point is not the cluster center, the cluster of $q_m$$^{\text{th}}$ point is assigned to the cluster of $b_{q_m}$$^{\text{th}}$ point. Finding the cluster according to $\{q_i\}_{i=1}^{n}$ which indicate the subscript of local density assure the $b_{q_i}$$^{\text{th}}$ point has a definite cluster, because the point which has a largest global density is certainly a cluster center.

After the clustering, each frame which is regarded as a high-dimensional point has a cluster's label, like A, B or C. These letters are recorded according to the order of the original frames, and this step is called the temporal reverting. The motions can now be converted into strings using the letters associated with each frame. This string can be further simplified by removing consecutive repetitions of the same letter, and the length of sequential repetitions saved by the subscript of each letter. For instance, the string {AAAABBCCC} can be simplified to $\{\, \mathrm{A}_4\mathrm{B}_2\mathrm{C}_3 \,\}$. This symbolic representation which called the Behavior String (BS) represents human motion capture data. For example, the sequence which have 1200 frames including 'running' and 'walking' can be shown as:

$\{\mathrm{A}_{100}\mathrm{B}_{100}\mathrm{A}_{100}\mathrm{B}_{100}\mathrm{A}_{100}\mathrm{B}_{100}\mathrm{C}_{150}\mathrm{D}_{150}\mathrm{C}_{150}\mathrm{D}_{150}\}$. By analyzing this Behavior String, $\{\mathrm{A}_{100}\mathrm{B}_{100}\}$ is a circular sequence meaning a motion 'x', while $\{\mathrm{C}_{150}\mathrm{D}_{150}\}$ is a circular sequence meaning other motion 'y'. There are three 'x' and two 'y' in this sequence. The cycle of 'x' is 200 frames, while the cycle of 'y' is 300 frames.

Algorithm 1. The 'Key Word' discovery algorithm.

```
Input:
      N is the length of the BS,
      Vector BS.
Output:
      Vector KeyWord.
Begin:
      Window size = 2 or 3;
      for ( i = first frame to last frame ) do
        Compare the Word in the neighbour Window;
        Count the Word with 2 or 3 character;
        Save the Word and the number of Word;
      end if
      for ( i = 1 to the number of Word ) do
        Delete the smaller number of the similar Word;
      end if
      for ( i = 1 to the number of the simplified Word ) do
        if ( the number of Word[i] >= 3 )
          KeyWord[i] = Word[i];
          end if
      end for
End
```

IV. ANALYZING BS FOR BEHAVIORAL SEGMENTATION

How to get the cut points and cycles of motion from BS are presented in this section. Extract the 'Key Word' which represent the behavior and use the string matching method to find the cut points and the cycles of motion.

A. Extracting 'Key Word' representing the behavior

We use the Moving Window to get the statistics of 'words'. In our experiments, a behavior can be converted to the 'word' which has at most three letters, like 'AB', 'BC' or 'ADC'. As shown in Algorithm 1. Setting the step size to one, from the first frame to the last frame, the 'word' will be counted with the window size of 2. Then we do the same work with the window size of 3. Specially, we ignore the 'word' which has a same letter in it, like 'ABA' or 'CDC'. Note that the 'word' which is constituted by the letters in the window does not consider the order of the letters. For example, 'AB' is equal to 'BA', and 'ABC' is equal to 'BCA'. If the number of the 'word' is less than 3, which as a rule of thumb in our experiments, the 'word' will be also removed, because the 'word' is a transition between two different behaviors. Moreover, if letters from one 'word' are

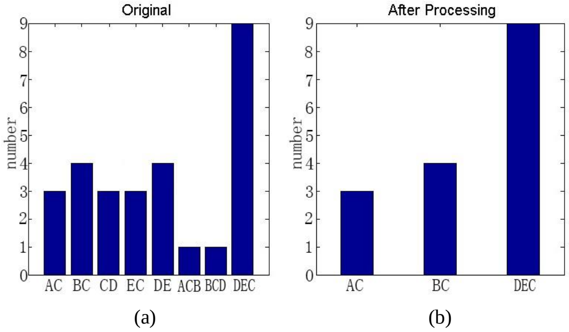


Figure 6. Finding 'Key Word'. (a) The statistics of 'word'. (b) The 'word' after processing is called the 'Key Word'.

included in another ‘word’, save the ‘word’ which has largest counts and abandon others. For example, to facilitate the expression, let the $\mathrm{N_{AB}}$ to indicate the number of ‘AB’. If the Behavior String is {ABCABCABCDEDEDE} which hid the subscripts, we only keep the ‘ABC’ and ‘DE’, because the $\mathrm{N_{AB}} = 3$, $\mathrm{N_{BC}} = 3$ and $\mathrm{N_{AC}} = 2$ are less than $\mathrm{N_{ABC}} = 7$, while $\mathrm{N_{BCD}} = \mathrm{N_{CDE}} = \mathrm{N_{CD}} = 1$ are less than 3 meaning the transition between two different behaviors. The remaining ‘words’ after the aforementioned processing are called the ‘Key Words’.

Fig. 6 shows the ‘Key Word’ found from the statistics of the ‘words’ in Behavior String:

$$\{\mathrm{A_{500}C_{99}A_{398}C_{195}B_{127}C_{217}B_{118}C_{286}D_{34}E_{60}C_{51}D_{41}E_{68}C_{42}D_{37}E_{70}C_{49}D_{52}E_{56}}\}$$

Which have 2500 frames from the CMU database [14] including ‘walking’, ‘stretching’ and ‘swinging arms’, represented by ‘AC’, ‘BC’ and ‘DEC’. In Fig. 6(a), there are 8 ‘words’ computed by the statistic of the ‘words’ in Behavior String. As we can see, the number of ‘ACB’ and ‘BDC’ is less than 3, so these ‘words’ are regarded as the transition of two behaviors, in other words, these two ‘words’ are removed. The ‘CD’, ‘EC’ and ‘DE’ are included in ‘DEC’ which not consider the order of the letters. According to our algorithm, the largest number of ‘word’ is reserved, while the smaller number of ‘words’ are removed. The number of ‘DEC’ is 9 which is larger than ‘CD’, ‘EC’, and ‘DE’. So ‘DEC’ is the only one ‘word’ which is reserved. In general, there are 3 ‘words’ which is regarded as the ‘Key Words’ as shown in Fig. 6(b).

B. *Finding cut points and cycles*

We use the string matching method to find the cut points and the cycles of motion. Use every ‘Key Word’ to match with the original Behavior String, and save the last letter’s frame number to the Cut Array if the next ‘word’ is not this ‘Key Word’. Considering the Behavior String may have a letter which is not included in all of the ‘Key Words’. If the subscript of the letter is more than 600, as a rule of thumb in our experiments, adding the last letter’s frame number to the Cut Array. This letter can be regarded as an independent behavior, which means each frame in this behavior is clustered in one cluster, but we cannot find its cycle. Else, we will do nothing because the letter is regarded as a noise. Each element in Cut Array adds one. Then reorder the number after the repeated numbers are removed. Cut points between two behaviors are given as a frame number in Cut Array. Next, for each ‘Key Word’, find the average length of the matched ‘word’ in Behavior String. The length is the cycle of motion represented by the ‘Key Word’.

V. ANALYZING BS FOR BEHAVIORAL SEGMENTATION

We test our techniques on the largest freely available motion capture databases, the CMU motion databases [14], proving that our approach is not only theoretically applicable but also solves the problem of segmenting motion capture data into distinct behaviors in practice. d_c Is a parameter deciding the results of the clustering which has been presented in Section III.B. In our experiments, we set the d_c so that the average number of neighbors is around 1% of the total number of points in the data set [3]. We choose 34 motion sequences with different behaviors such as walking, running, jumping, punching, stretching, and swinging arms. We use our method to segment this 34 long original motion sequences into distinct behaviors. As comparison, we use manual labeling method, PCA method, PPCA method [5], ACA method [7] and HACA method [8] to accomplish motion segmentation. There are 8 representative sequences as shown in Table I and the results of the behavioral segmentation using these sequences are shown in Fig. 7. The exact frame of the transition is particularly difficult to determine when the motion capture subject makes a smooth transition from one behavior to another. Consequently, we allow a range of frames to be specified as the ground truth by the human subjects. In Fig. 7, the black stripes in the sequences indicate the cut points assigned by the algorithms. Specially, for the human observer, instead of a single frame, the range in which the transition occurred is given, as all the frames in the range are acceptable positions for a motion cut. Notice that the ‘ACA’, ‘HACA’, ‘BS’ and ‘Human’ sequences in Fig. 7 have many colors, while the ‘PCA’ and ‘PPCA’ sequences only have a single color. Because the former methods can find not only the cut points but also the motion clips which represent the same behaviors in the original motion sequence, while the latter methods could only find the cut points. For one original motion sequence, the different colors correspond to different behaviors while the same colors correspond to the motion clips which represent the same behaviors in this sequence.

TABLE I. SOME OF THE HUMAN MOTION CAPTURE DATA FOR BEHAVIORAL SEGMENTATION IN OUR EXPERIMENTS.

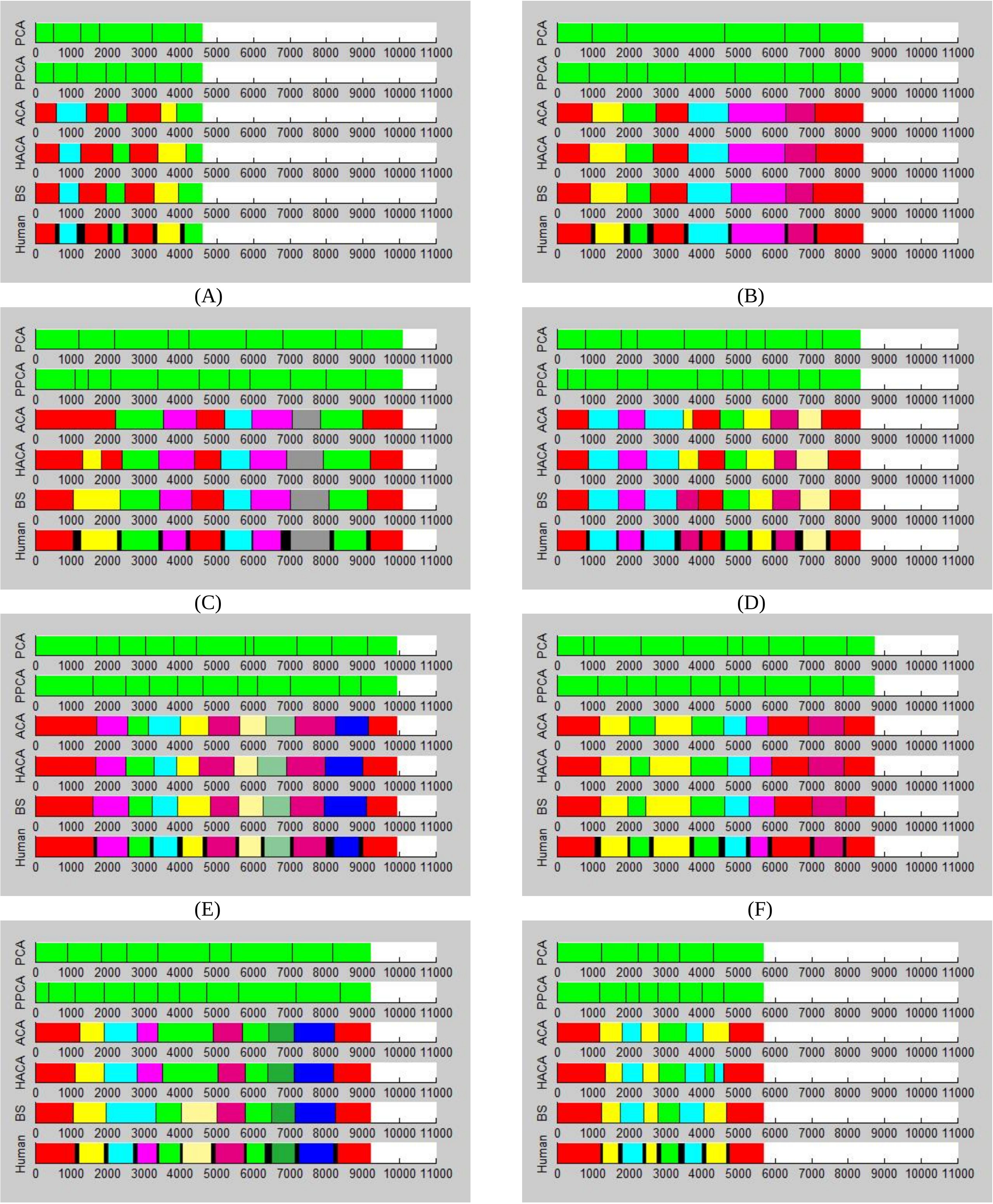
Serial number	Name	Number of frames	Number of behaviors
A	CMU_86_01	4579	4
B	CMU_86_03	8401	6
C	CMU_86_04	10078	6
D	CMU_86_05	8340	8
E	CMU_86_06	9939	9
F	CMU_86_07	8702	6
G	CMU_86_08	9206	9
H	CMU_86_11	5674	4

We compared five algorithms using the standard precision/recall framework [5]. Precision is defined as the ratio of reported correct cuts versus the total number of reported cuts. Recall is defined as the ratio of reported correct cuts versus the total number of correct cuts. The closer precision and recall are to 1, the more accurate the algorithm is. Table II gives precision and recall scores for the five algorithms.

From the experimental results in Table II, we can see that the precision and recall of our method are obviously higher than the PCA method, a little higher than the PPCA method, and similar as the ACA and HACA methods. Because the clustering algorithm in this paper is able to cluster the motion capture data which have a high-dimensional and nonspherical clusters with

TABLE II. PRECISION AND RECALL SCORES FOR THE PCA, PPCA, ACA, HACA AND OUR ALGORITHMS.

Algorithm	Precision	Recall
PCA	74.11%	80.72%
PPCA	88.30%	90.36%
ACA	90.89%	92.37%
HACA	91.12%	92.53%
Our algorithm	91.25%	92.77%



(A) (B)

(C) (D)

(E) (F)

(G) (H)

Figure 7. Motion segmentation points. Each chart corresponds to one motion from the Table I, the x-axis corresponds to the frame number, and the vertical bars specify the cut points assigned by the algorithms. For the human observer, instead of a single frame, the range in which the transition occurred is given, as all the frames in the range are acceptable positions for a motion cut. Different colors correspond to different behaviors while the same colors correspond to the motion clips which represent the same behaviors in the original motion sequence.

TABLE III. The cycle of motion in the 'A' sequence.

Behaviors	Cycle by BS	Cycle by human observer
walking	186 frames	170 to 190 frames
jumping	271 frames	250 to 270 frames
punching	164 frames	150 to 160 frames
kicking	287 frames	290 to 320 frames

the non-visual number of clusters. Thus, this clustering algorithm can accurately cluster the each frame into each cluster so that the cut points can be found exactly than PCA and PPCA methods compared with the human observer, shown in Fig. 7. Moreover, our segmentation method can find the motion clips representing the same behaviors which the PCA and PPCA methods cannot, because the clustering algorithm is able to cluster similar frames into one cluster. Notice that in Fig. 7(D), although ACA and HACA methods find the right cut points, they could cluster the clips which represent different behaviors into one cluster that leads to the wrong recognition of behaviors. However, in the experiments with our method, this situation not happened due to the clustering algorithm has a better performance for motion capture data.

Our method can find not only the cut points but also the cycles of motion which the other four methods cannot. Table III gives the cycle of motion computed by our method and human observer from the sequence 'A' in Table I. Considering the difference of each motion, like the manual labeling method to find the cut points, we also allow a range of frames to be specified as the ground truth by the human subjects.

From the experimental results in Table III, we can see that the cycle of motion computed by our method accord with the human observer. While the cycle of motion may not be found by our method when all the frames in the behavior are clustered in one cluster. For example, if an original motion sequence has one behavior which has an enormous difference to other behaviors, and each frame in this behavior has a little different to other frames, the behavior may be clustered in one independent letter in Behavior String. Essentially, the parameter d_c deciding the result of the clustering which has been set to a constant value in our experiments rather than automatically changed with each motion sequence.

VI. Conclusions and Future Work

This paper introduces a new symbolic representation of motion capture data called the Behavior String. The BS can be created by three steps: First, each frame in the motion capture data is regarded as a point. Thus, human motion capture data is treated as a high-dimensional discrete data points. The distances between every two points are computed by considering the pose and the velocity. Second, these points are clustered by an alternative algorithm based on the idea that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities, which is able to cluster the motion capture data having a high-dimensional and nonspherical clusters with the non-visual number of clusters. Third, the points reordered according to the order of the original frames, and different clusters are indicated by different letters. Then the BS is produced by the further simplified.

Based on the BS, in the behavioral segmentation area, the behavioral cut points and the motion clips which represent the same behaviors are found with the string analysis method. This method is able to automatically create a segmentation even when the behaviors have not been seen before, and the user need not to know the number of behaviors in the original motion sequence beforehand, which make it closer to the real. Experimental results show that our method has a good performance than other state-of-the-art methods in behavioral segmentation for motion capture data. Furthermore, the cycles of motion are also found with this method which plays an important role in many domains such as motion compression, motion classification and motion analysis.

Although experimental results have shown our method provided a good performance. Our method could only segment the original motion sequence which has simple behaviors like walking, running, jumping and so on. For complicated behaviors, such as dances, this method could not distinguish them.

References

[1] J. Lee, J. Chai, and P. Reitsma, "Interactive control of avatars animated with human motion data," ACM Transactions on Graphics (TOG), Vol. 21, No. 3, pp. 491-500, 2002.

[2] J. Wang and B. Bodenheimer, "An evaluation of a cost metric for selecting transitions between motion segments," Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. Eurographics Association, pp. 232-238, 2003.

[3] R. Alex and A. Laio, "Clustering by fast search and find of density peaks," Science,Vol. 344, No. 6196, pp. 1492-1496, 2014.

[4] O. Arikan, D. A. Forsyth, and J. F. O'Brien, "Motion synthesis from annotations," ACM Transactions on Graphics (TOG), Vol. 22, No. 3, pp. 402-408, 2003.

[5] J. Barbič, et al, "Segmenting motion capture data into distinct behaviors," Proceedings of the 2004 Graphics Interface Conference, London, May, 2004.

[6] C. Lu and N. J. Ferrier, "Repetitive motion analysis: segmentation and event classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26, No.2, pp. 258-263, 2004.

[7] F. Zhou, F. Torre, and J. K. Hodgins, "Aligned cluster analysis for temporal segmentation of human motion," IEEE International Conference on Automatic Face and Gesture Recognition, pp. 1-7, 2008.

[8] F. Zhou, F. Torre, and J. K. Hodgins, "Hierarchical aligned cluster analysis for temporal clustering of human motion," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 35, No. 3, pp. 582-596, 2013.

[9] Y. Yang, L. Wang, and A. H. Aimin, "Motion String: A Motion Capture Data Representation for Behavior Segmentation," Journal of Computer Research and Development, Vol. 45, No.3, pp. 527-534, 2008.

[10] P. Beaudoin, S. Coros, M. Panne, and P. Pierre, "Motion-motif graphs," Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Eurographics Association, pp. 117-126, 2008.

[11] Y. Lee, K. Wampler, G. Bernstein, J. Popovic, and Z. Popovic, "Motion fields for interactive character locomotion," ACM Transactions on Graphics (TOG), Vol. 29, No. 6, pp. 138-146, 2010.

[12] A. Shrivastava, V. Patel, and R. Chellappa, "Multiple Kernel Learning for Sparse Representation-based Classification," IEEE Transactions on Image Processing, Vol. 23, No. 7, pp. 3013-3024, 2014.

[13] S. Gao, I. Tsang, and L. Chia, "Sparse representation with kernels," IEEE Transactions on Image Processing, Vol. 22, No. 2, pp. 423-434, 2013.

[14] CMU: http://mocap.cs.cmu.edu, 2003.