Temporal Clustering of Motion Capture Data with Optimal Partitioning

Yang Yang* Hubert P. H. Shum† Nauman Aslam† Lanling Zeng*
*Department of Computer Science, Jiangsu University
†Department of Computer Science, Northumbria University

Abstract

Motion capture data can be characterized as a series of multi-dimensional spatio-temporal data, which is recorded by tracking the number of key points in space over time with a 3-dimensional representation. Such complex characteristics make the processing of motion capture data a non-trivial task. Hence, techniques that can provide an approximated, less complicated representation of such data are highly desirable. In this paper, we propose a novel technique that uses temporal clustering to generate an approximate representation of motion capture data. First, we segment the motion in the time domain with an optimal partition algorithm so that the within-segment sum of squared error (WSSSE) is minimized. Then, we represent the motion capture data as the averages taken over all the segments, resulting in a representation of much lower complexity. Experimental results suggest that comparing with the compared methods, our proposed representation technique can better approximate the motion capture data.

Keywords: Motion capture, temporal clustering, optimal partition

Concepts: • **Computing methodologies ~ Computer Graphics;** *Methodology and Techniques;*

1 Introduction

Motion capture is a well-established technology for tracking and recording motions. Nowadays, it has been used in many fields such as entertainment [Huang 2015], sports [Alderson 2015], and medical [Marshall 2015] applications. One common problem for these applications is motion analysis, for example, [Huang 2015] analyzes the motion in order to build the motion graphs that can be used to produce 3D animations, [Alderson 2015] analyzes the motions to evaluate the athletes' performances, [Marshall 2015] analyzes the movements in order to evaluate the rehabilitation status. However, due the high complexity of motion capture data, the processing of this high dimensional time series data is quite difficult, for example, motif discovery and anomaly detection from the time series data mining field are both NP-hard. In this paper, we propose a novel technique to reduce the complexity of motion capture data to generate an approximate representation. The proposed technique allows much easier interpretations of motions, thus enhances the efficiency of data mining operations such as gesture recognition, posture detection etc.

We adapt temporal clustering to derive the representation so as to minimize the within-segment sum of squared error (WSSSE), thus closely approximate the original motion capture data. It is an algorithm similar to K-means clustering in the data mining field. The difference lies in the fact that the former requires considering the temporal order, while the latter does not. Temporal clustering groups continuous frames into segments. Giving a motion of n frames, the goal is to partition it into k segments in which each frame belongs to the segment with the nearest mean. The approximate representation is then derived as the mean of these segments so that a minimum WSSSE is achieved.

The search space of the segmentation problem in temporal clustering is exponential to the number of frames n, therefore a brute force algorithm is not practical. In this paper, the segmentation problem is perfectly solved with Fisher's optimal partition algorithm [Fisher 1958], which is based on the idea of dynamic programming. The algorithm can solve the segmentation problem in $O(n^2)$ time with space complexity $O(n^2)$.

Through rigorous experimentation, the proposed approximate representation was evaluated. The dataset involved in the experiment contains 8 kinds of motions that are taken from CMU motion capture database. We compare the WSSSE of our system with that of the compared methods, and conclude that our proposed representation could accurately approximate the original motion capture data.

The rest of this paper is organized as follows. Section 2 contains the review of relevant work, Section 3 covers the details of our proposed method. In Section 4, the evaluation of the proposed representation is given. Section 5 concludes the paper, and highlights the direction for our future work.

2 Related Work

In the past, many researchers have applied approximation techniques used for time series data to the motion capture data. Among these, three techniques are the most notable. They are discrete Fourier transform (DFT), discrete wavelet transform (DWT) [Kin and Fu 1999] and piecewise aggregate approximation (PAA) [Keogh et al. 2001]. DFT attempts to represent the temporal series data as a linear combination of the complex sinusoids. It can capture the general shape of the time series data well and it is relatively fast to compute ($O(nlog(n))$), so it has broad applications such as compression, smoothing. However, it is not good at approximating flat time series with sudden bursts. In order to overcome this problem, DWT was proposed. DWT describes the time series data as a combination of shifted and scaled version of wavelets, since the contribution of wavelets is localized, it can be used to model a variety of time series data. It is also very fast ($O(n)$). PAA tries to model time series data as segmented means, i.e., it first divides the time series data into segments, then each segment is represented as the average value. It is very intuitive and simple to calculate ($O(n)$), and it retains the similar representation power just like the other two sophisticated representations. However, PAA divides data into segments of equal length, which is inappropriate

*e-mail:yyoung@ujs.edu.cn.

VRCAI '16, December 03-04, 2016, Zhuhai, China
© 2016 ACM. ISBN 978-1-4503-4692-4/16/12$15.00
DOI: http://dx.doi.org/10.1145/3013971.3014019

for motion data because it's doesn't adapt to different motion data.

Another closely related topic is motion segmentation, which is very useful to applications like motion analysis, motion recognition, and motion graph. The task of motion segmentation is to divide the motion into segments. In order to do that, many previous works tried to empirically estimate the threshold for the local minimum of various features, such as velocity [Gibet and Marteau 2007], entropy [So et al. 2005] and curvature [Zhao 2002]. [Yang et al. 2015] divide the motion into segments by applying hierarchical clustering with cosine distance. However, these methods all suffer from the sensitivity to noise. In addition, they are not flexible, because they do not allow the user to specify the number of segments. Similarly, keyframe extraction extracts keyframes that can be seen as segmentation boundaries. The related work in this area can be generally divided into three categories, namely curve simplification [Takeshi et al 2014], clustering [Halit and Capin 2011] [Qiang et al. 2013] and matrix decomposition [Xin et al. 2007]. Curve simplification method is good for keyframe extraction, however, it is not applicable to represent a motion. Current clustering based methods try to divide the motion into clusters, while not segments, they do not consider the temporal order of the motion capture data. [Zhou et al. 2013] consider the temporal order in clustering, while they focus on segmenting motions into meaningful behaviors.

In order to address the limitations mentioned above, we intend to improve the work of PAA by allowing varying length segments. Optimal partition algorithm is applied to divide the motions into varying length segments, so as to minimize the WSSSE.

3 Proposed method

Figure 1 shows the overview of our proposed method, as we can see, it mainly consists of 2 steps, and we will go through each of the steps in this section.

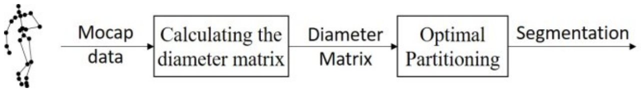


Figure 1: *Overview of the proposed method*

The motion capture data used in this work contains sample points for 21 joints all over the human body (we have reduced the redundant joints in the original CMU motion capture data, so that there are only 21 joints left). The 3D joint angles of all the joints are recorded. In total, there are 63 dimensional time series data. In this way, the motion capture data can be represented as a matrix, in which each row represents a frame and each column represents the data corresponding to one degree of freedom (DOF). With the explanation above, the problem can be formulated as the following:

Given a motion capture data *M* which consists of *n* frames,

$$M = (f_1,\ f_2, \dots f_n)^T, \text{ where } T \text{ represents transpose} \qquad (1)$$

We want to divide the motion into *k* segments

$$S(n,k) = (s_1,\ s_2 \dots s_k,\ s_{k+1}) \qquad (2)$$

where s_i represents the i^{th} segment boundary, and we have $1 = s_1 < s_2 < \cdots < s_k < s_{k+1} = n+1$. In this way, the i^{th} segment is from frame s_i to frame $s_{i+1}-1$, such that the centers of these segments $C = (c(s_1, s_2 - 1), c(s_2, s_3 - 1), \dots, c(s_k, s_{k+1} - 1))$, can be used to best approximate the motion, i.e., the within segment sum of squared error (WSSSE) $L(S(n,\ k))$ reaches minimum:

$$argmin\ L(S(n,\ k)) = \sum_{i=1}^{k} \sum_{j=s_i}^{s_{i+1}-1} (f_j - c(s_i, s_{i+1} - 1))(f_j - c(s_i, s_{i+1} - 1))' \qquad (3)$$

where the center function c is defined as:

$$c(s_i, s_{i+1} - 1) = (1/(s_{i+1} - s_i)) \sum_{j=s_i}^{s_{i+1}-1} f_j \qquad (4)$$

Fisher's optimal partition algorithm [Fisher 1958] is applied to solve the above-mentioned problem. It is a dynamic solution which runs in *O(n²)* time. The mathematical foundation in Fisher's optimal partition algorithm can be found below.

The diameter of a segment from frame u to frame v is defined as:

$$d(u, v) = \sum_{j=u}^{v} \left(f_j - c(u, v)\right) (f_j - c(u, v))' \qquad (5)$$

Thus, the objective function (3) can be rewritten as:

$$argmin\ L(S(n,\ k)) = \sum_{i=1}^{k} d(s_i, s_{i+1} - 1) \qquad (6)$$

Assume that $S^*(n,\ k)$ is the segmentation of a motion capture data that minimizes the objective function L. If $k = 2$, i.e., divide the data into 2 segments, then $S^*(n,\ k) = (1, t, n+1)$, where t is a value in [2,n] that minimizes $d(1, t-1) + d(t, n)$. If $k > 2$, i.e., divide the data into more than 2 segments, the problem can then be seen as divide the first $t - 1$ frames into $k - 1$ segments, the remaining $n - t + 1$ frames will be the last segment, i.e., $S^*(n, k) = S^*(t-1, k-1) \cup (n+1)$, where the value $t \in [2, n]$ which minimizes $L(S^*(t-1, k-1)) + d(t, n)$. According to the above discussion, we will have the following formula.

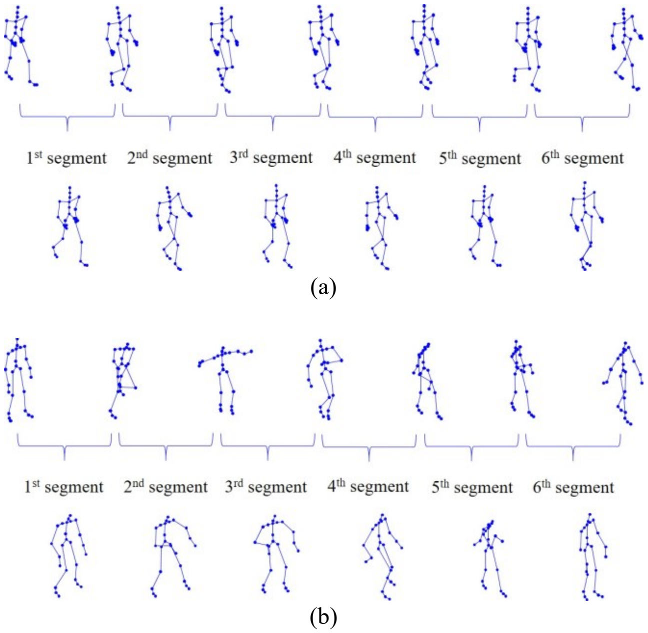
$$\begin{cases} L(S^*(n, 2)) = \min\limits_{2 \le t \le n} \{d(1, t-1) + d(t, n)\} \\ L(S^*(n, k)) = \min\limits_{2 \le t \le n} \{L(S^*(t-1, k-1)) + d(t, n)\} \end{cases} \qquad (7)$$

The algorithm to calculate the optimum partition can be described as follows:

```
Algorithm Optimal Partition
Input: motion capture data m, number of frames n, number of
        segments k
Output: WSSSE matrix L, partition matrix P
begin
    Calculate the diameter matrix d;
    for i=2 to k
        for j=3 to n
            if i==2
                find the value t* which minimizes d(1, t-1)+d(t, n);
                L(j, i)=d(1, t*-1)+d(t*, n);
            else
                find the value t* which minimizes L(t-1, i-1)+d(t, n);
                L(j, i)= L(t*-1, i-1)+d(t*, n);
            end
            P(j, i)=t*;
        end
    end
end
```

The optimal partition algorithm can be implemented in three steps: 1) Calculate the diameter matrix d, i.e., for every pair i, j, where $i < j$, we calculate $d(i, j)$, this can be simply accomplished in $O(n^2)$ time; 2) Based on the diameter matrix d, compute the optimal partitions $S^*(n, 2)$ which divide the first j $(1 < j \leq n)$ frames into 2 segments; 3) Iteratively deduce $S^*(j, k)$ from $S^*(j, k-1)$, $(1 < j \leq n)$.

Figure 2 illustrates the temporal clustering of a “walking” and a “dance” motion, we can see that the two motions are both divided into 6 segments, the postures on the top of the figure show the segment boundaries, while the postures on the bottom demonstrate the mean poses of each segment. We can draw the conclusion from this figure that the segmentation is quite intuitive, and from the segment boundaries and mean poses, we can actually understand and replicate the corresponding motion.



(a)

(b)

Figure 2: *An illustration to the temporal clustering of a “walking” motion (a) and a “dance” motion (b).*

4 Evaluation

In order to evaluate the proposed method, an experiment is conducted. The dataset is taken from CMU motion capture database (http://mocap.cs.cmu.edu/). 8 different kinds of motions are involved in our experiment, which are “Walk, Run, Pickup, Jump, Backflip, Basketball, Boxing and Dance”. The $\sqrt{WSSSE}$ of these motions derived by various methods are compared when the number of segments k takes different values, as shown in Figure 3, Figure 4, Figure 5, and Figure 6. All the methods are implemented with MATLAB 7.11, which runs on an i3-4000M platform. It can be observed from these figures that comparing with DWT (Haar) and PAA, our proposed method produces better approximate representations with a lower $\sqrt{WSSSE}$. While comparing with DFT, for most of the cases, our proposed method outperform DFT, except for 2. As shown in Figure 3, for “Walk” and “Run”, our proposed method results in a higher $\sqrt{WSSSE}$ than DFT method when dividing the data into 6 segments. It is mainly because that 1) “Walk” and “Run” contains periodical movements that are quite similar to sine and cosine waves; 2) DFT has a very good approximation ability when the number of segments k is small. We can see our method gains its advantage over DFT as k increases to 12, 24, and 48.

Figure 7 shows the sum of $\sqrt{WSSSE}$ of all the 8 kinds of motions when k takes different values, and it can be concluded that our method produces the approximate representations with minimum $\sqrt{WSSSE}$. On the other hand, it is a fact that the lower the $\sqrt{WSSSE}$ value, the better the representation in approximating the original data. Thus, it can be concluded from the above result that our proposed method could produce better approximate representations than the 3 compared methods.

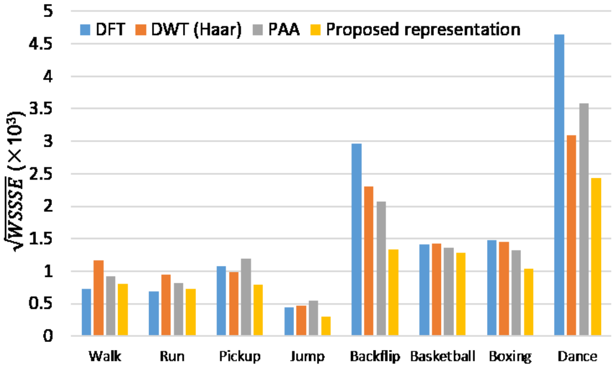


Figure 3: $\sqrt{WSSSE}$ *for various kinds of motions (k=6).*

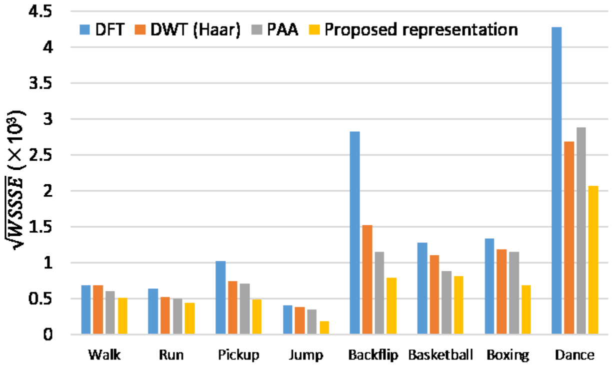


Figure 4: $\sqrt{WSSSE}$ *for various kinds of motions (k=12).*

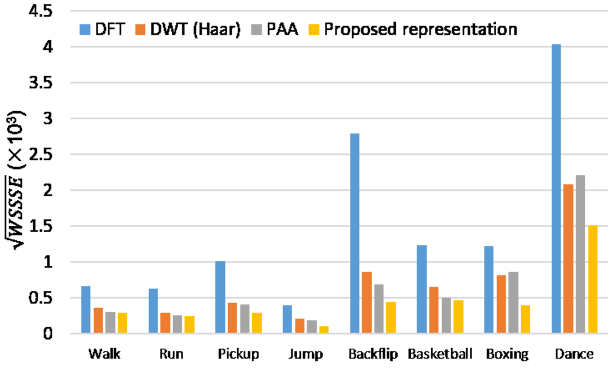


Figure 5: $\sqrt{WSSSE}$ *for various kinds of motions (k=24).*

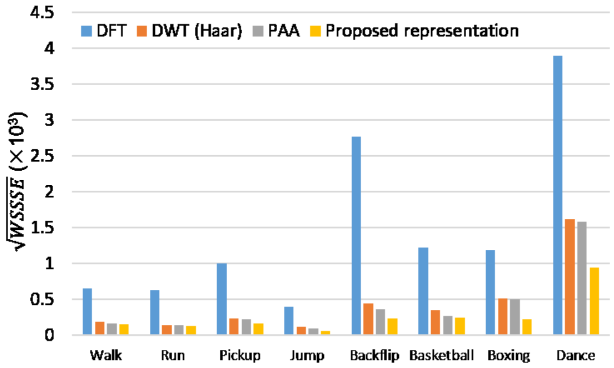


Figure 6: *$\sqrt{WSSSE}$ for various kinds of motions (k=48).*

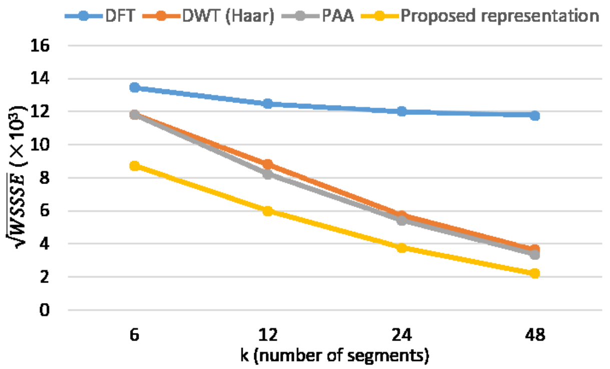


Figure 7: *The sum of $\sqrt{WSSSE}$ of all the 8 kinds of motions when k takes different values*

5 Conclusion

In this paper, we proposed a method for temporally clustering of motion capture data based on optimal partition algorithm. Our proposed method provides an approximate representation of motion capture data, which will ease the processing of such high-dimensional time series data. In order to evaluate the proposed method, an experiment is conducted, and the results suggest that our proposed method produces a better approximate representation of motion capture data than the compared methods.

Acknowledgements

The work described in this paper was supported by National Natural Science Foundation of China (Grant No. 61402205), China Postdoctoral Science Foundation (Grant No. 2015M571688), a grant from City University of Hong Kong (Grant No. 7004548), Engineering and Physical Sciences Research Council (EPSRC) (Ref: EP/M002632/1), and a Research Fund granted by Jiangsu University (Grant No. 13JDG085).

References

Alderson, J. 2015. A markerless motion capture technique for sport performance analysis and injury prevention: Toward a big data, machine learning future. *Journal of Science and Medicine in Sport* 19, e79.

Fisher, W. D. 1958. On grouping for maximum homogeneity. *Journal of the American Statistical Association*, 53, 284, 789-798.

Faloutsos, C. 1994. Fast subsequence matching in time-series databases. *Acm Sigmod Record*, 23, 2, 419-429.

Gibet, S., & Marteau, P. F. 2007. Approximation of Curvature and Velocity for Gesture Segmentation and Synthesis. *Gesture-Based Human-Computer Interaction and Simulation*. 13-23.

Halit, C., & Capin, T. (2011). Multiscale motion saliency for keyframe extraction from motion capture sequences. *Computer Animation & Virtual Worlds*, 22, 1, 3-14.

Huang, P., Tejera, M., Collomosse, J., & Hilton, A. 2015. Hybrid skeletal-surface motion graphs for character animation from 4d performance capture. *ACM Transactions on Graphics*, 34, 2, 1-14.

Keogh, E., Chakrabarti, K., Pazzani, M., & Mehrotra, S. 2001. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge & Information Systems*, 3, 3, 263-286.

Kin-Pong Chan, & Fu, A.W.-C. 1999. Efficient time series matching by wavelets. *IEEE TENCON-Digital Signal Processing Applications*, 514-519.

Marshall, B. 2015. The use of 3d in motion capture in ACLR and athletic groin pain rehabilitation. *Biomechanics*.

Qiang, Z., Yu, S. P., Zhou, D. S., & Wei, X. P. 2013. An efficient method of key-frame extraction based on a cluster algorithm. *Journal of Human Kinetics*, 39, 1, 5-13.

So, C. K. F., & Baciu, G. 2005. Entropy-based motion extraction for motion capture animation. *Computer Animation & Virtual Worlds*, 16, 3-4, 225-235.

Takeshi M., Takaaki K., Takeshi S., Hiroaki K., Katsubumi T., Hideo T. 2014. A hybrid approach to keyframe extraction from motion capture data using curve simplification and principal component analysis. *Ieej Transactions on Electrical & Electronic Engineering*, 9, 6, 697–699.

Xin, W. K., Tateno, K. K., Konma, T., & Shimamura, T. 2007. Discrete wavelet based keyframe extraction method from motion capture data. *International Journal of Asia Digital Art & Design*, 6.

Yang, Y., Chen, J., Liu, Z., Zhan, Y., & Wang, X. 2015. Low level segmentation of motion capture data based on hierarchical clustering with cosine distance. *International Journal of Database Theory & Application*, 8, 4, 231-240.

Zhao, Liwei. 2002. *Synthesis and acquisition of laban movement analysis qualitative parameters for communicative gestures*. PhD thesis, University of Pennsylvania.

Zhou, F., De, l. T. F., & Hodgins, J. K. 2013. Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Transactions on Pattern Analysis & Machine Intelligence,* 35, 3, 582-96.