Original Paper DOI: 10.5604/20831862.1044460 Biol. Sport 2013;30:145-151

3D HUMAN MOTION RETRIEVAL BASED ON HUMAN HIERARCHICAL INDEX STRUCTURE

Accepted for publication 07.11.2012

AUTHORS: Zhang Q., Guo X.

Key Laboratory of Advanced Design and Intelligent Computing, Dalian University, Ministry of Education, Dalian, China

Reprint request to:
Qiang Zhang
Key Laboratory of Advanced Design and Intelligent Computing (Dalian University) Ministry of Education, Dalian, 116622, China
E-mail: zhangq@dlu.edu.cn

ABSTRACT: With the development and wide application of motion capture technology, the captured motion data sets are becoming larger and larger. For this reason, an efficient retrieval method for the motion database is very important. The retrieval method needs an appropriate indexing scheme and an effective similarity measure that can organize the existing motion data well. In this paper, we represent a human motion hierarchical index structure and adopt a nonlinear method to segment motion sequences. Based on this, we extract motion patterns and then we employ a fast similarity measure algorithm for motion pattern similarity computation to efficiently retrieve motion sequences. The experiment results show that the approach proposed in our paper is effective and efficient.

KEY WORDS: motion capture, human hierarchical structure, motion pattern, KMP algorithm, motion retrieval

INTRODUCTION

Human motion capture has been a new rising technology for motion data collection in recent years. With the rapid development and increasing maturity of motion capture techniques, more and more human motion data have become available and have been widely used in many research areas such as television animation, 3D game, intelligent control, athletic training, visual surveillance and so on. This makes efficient and effective retrieval of motion data an important issue. However, effective retrieval of motion data is not an easy task due to the complexity of the motion data set. Thus an efficient motion data retrieval method is needed.

Human motion can be represented in the form of 3D positional and rotational information of joints in space over time. Motion data comprise a frame sequence and each frame records a posture at a given time, so a motion sequence can be regarded as a multivariate time series. Such information can be used to better analyse and quantify the complex human body motions for gait analysis and several orthopaedic applications such as joint mechanics, prosthetic designs, and sports medicines. Directly computing the similarity between two human motion sequences is very difficult because motion sequences have multiple attributes; for example, a motion sequence usually consists of a large number of frames and different motion sequences have different lengths, which implies that similar frames may correspond to very different positions in the sequence [1]. Therefore an efficient motion data retrieval method is needed to deal with it.

In this paper, in the data pre-processing stage, we first divide the human body into a number of meaningful parts and build a hierarchical structure based on the correlations among these body parts. Then, a motion segmentation scheme is employed for each part to partition the original motion sequences into part-based motion representations. After these steps, an adaptive threshold clustering algorithm is then performed upon these motion segments to extract motion patterns by detecting and grouping the similar motion segments. In the retrieval stage, given a query motion sequence, our approach first chops it into motion segments and further creates its corresponding motion pattern lists by matching the existing patterns in the pre-constructed motion pattern library. Then, we adopt a fast string match algorithm for motion pattern similarity computation to efficiently retrieve logically similar motion sequences.

The paper is organized as follows. In Section 2 the related work on indexing and retrieval of motion capture data is discussed. In Section 3 the human motion data pre-processing is given. The indexing and retrieval method is introduced in Section 4. Experimental results are given in Section 5. Finally, in Section 6 we conclude this paper by discussing the advantages and limitations of our methods and providing directions for future work.

Related work

In the field of computer animation, data-driven motion synthesis is an important technique in order to create new, realistic motions from recorded motion capture data. So the efficient re-use of such data becomes a hot topic with the rapid increase of motion capture data. In recent years, researchers have obtained many meaningful results on various aspects of human motion capture data such as features representation, feature extraction, indexing and retrieval.

A variety of qualitative features describing geometric relations between specific body points of a posture are constructed in [9], and these features are used to induce a time segmentation of motion capture data streams for motion indexing. For each query, users have to select suitable features in order to obtain high-quality retrieval results. Yi et al. [17] proposed an indexing method based on geometric features, in which the bone structure is used to tectonic indexed tree and the geometric features of bone are regarded as branches. They segment the motion sequence according to the geometric feature and make similar motion segments as leaf nodes of tree structure, and then define characteristic coding functions to realize the feature extraction. Gu et al. [5] use hierarchical motion description for a posture, and then clustering-based key frame extraction is performed for retrieving and compressing the motions respectively. For the extraction of key frames, similarity needs to be found between each consecutive frame, which is time-consuming. Yan et al. [16] proposed a normalized algorithm to ensure that motion data have the same skeleton length, and they improved Muller's retrieval method. Their method makes the retrieval process have the function of sport reorganization and is capable of automatically splicing to retrieve motion that does not exist in a motion database by introducing an automatic conversion strategy. Sonoda et al. [10] investigated digital archiving of Japanese dance movements by using a motion capture system. By employing Laban motion analysis and Kansei information processing, their results related to motion segment, motion and player identification, extraction of characteristic poses, similarity retrieval of dance motions, and qualitative analysis of dance motion. Yamasaki et al. [15] proposed a scheme based on the content of a cross search for 3D human motion data retrieval, including time-varying mesh (TVM) and shape geometric feature extraction for motion capture data.

Annotation of motion capture data can be found by using clustering techniques such as self-organizing mapping (SOM). Xing et al. [14] use the SOM clustering algorithm to partition the frames into different classes to obtain the associated class reference vectors. Then probabilistic principal component analysis (PPCA) is applied to estimate the distribution of its data. Finally the similarity between the query example and the motion sequence in a database is measured by using the Mahalanobis distance. Wu et al. [12] present an efficient motion data indexing and retrieval method based on SOM and the Smith-Waterman string similarity metric. But when the SOM converts high-dimensional data into low-dimensional space (typically 2D), some key information of motion posture are missed and thus result in primitive posture distortion and the reduction of retrieval accuracy. Gaurav et al. [3] developed an efficient indexing approach for 3D motion capture data, which supports queries involving both sub-body motions as well as whole-body motions. The proposed indexing structure is based on the hierarchical structure of the human body segments consisting of independent index trees corresponding to each subpart of the body. Gaurav et al. [4] improved the above index methods and improved the index function to solve the information loss problem that was brought on by the feature space division. Dynamic time warping (DTW) is a popular tool in the processing of time series data such as speech recognition, which can also be used in the matching and searching of human motion data [6]. Because human motion data exhibit a strong continuum in the time axis with no obvious segmentation information, it is difficult to find the start and end points of motion data series, so comparing and matching the data is hard by DTW directly. Worawat et al. [11] proposed a quick filtering method for similarity queries in motion capture databases and introduced a new technique for dimensionality reduction based on the average and variance of joint angles. The new dimensional reduction named Constant Approximation with Average and Variance (CAAV) exploits simple comprehensible average and variance of the joint angles of a human body.

Although many results on the index and the retrieval methods for the human motion data have been obtained, both the efficiency and the retrieval accuracy need to be improved

Human body motion data pre-processing

Hierarchy structure

In this section, we use joint angles rather than 3D marker positions for representing human motion data due to the fact that the joint angle representation is convenient for unifying the motions of human bodies with different bone sizes. For example, right/left foot in front, right/left foot raised, right/left knee bent, right/left leg side stretch, and legs crossed are distinguishable in the lower body part. We choose a human hierarchy representation because it provides a logical control granularity. Furthermore, a multilayer hierarchy naturally makes the correlations among several human parts, and this human hierarchy representation can also be exploited for subsequent motion similarity computation.

A hierarchical human structure [2] is illustrated in Fig. 1, which is constructed based on the spatial connectivity of the human body. The whole human body is first divided into ten meaningful basic

parts, for example, head, torso, left arm, left hand, etc., and then a hierarchy with four layers is built accordingly. The hierarchy includes a total of eighteen nodes: ten leaf nodes stand for basic body parts, the parent nodes in the middle layers correspond to meaningful combinations of child nodes, and the root node represents the entire human body.

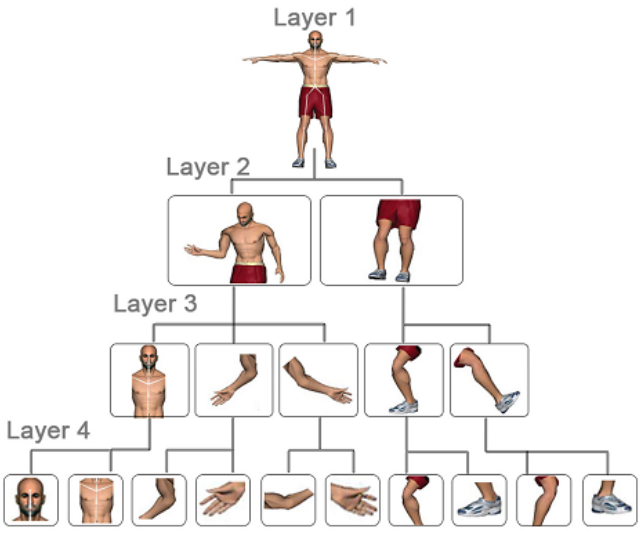

FIG. 1. ILLUSTRATION OF THE CONSTRUCTED HUMAN BODY HIERARCHY [2]

Motion segment

For a common motion capture database, its movement data are a hybrid of many common motion types. For example, in a period of time a man first warms up by walking and then starts to run, he resumes walking after finishing the run motion and stops finally. This human movement process contains two common movement types, i.e., run and walk, and in this movement process the run and walk motion alternates twice. Obviously, natural human movements are a mixture of such common motion types. For this kind of motion sequence, it is not easy to calculate the similarity and index the motion database directly. So it is necessary to segment the movement sequence appropriately before subsequent work.

The PCA method has been applied in motion segmentation and good results have been achieved. But human motion is inherently nonlinear; just by using linear methods it is difficult to find out the inherent characteristics of some complicated motions, and thus it is probable that some important motion information will be lost. This method cannot segment some long motion sequences correctly, for example, sitting on a chair and suddenly standing up; continuous jumping, etc. It either segments by mistake or the place that should be segmented is not segmented. Furthermore, the PCA method cannot correctly distinguish the change of speed. So in this paper we employ the nonlinear method to segment the long motion sequence into primitive motion clips.

In this work, we adopt the ISOMAP method to segment motion sequences. It is capable of motion sequence segmentation as well as dimension reduction. ISOMAP is a manifold learning method, in which the concept of topology is manifold. Its principle can be described as follows: firstly, it uses the shortest path in the nearest diagram to determine the approximate geodesic distance to replace the Euclidean distance that cannot express inner manifold structure, then this is inputted into the multidimensional dimension analysis (MDS) and processed, and then it finds the low-dimensional coordinates that are embedded in the high-dimension space.

The specific steps of the ISOMAP algorithm are described as follows.

(1) Calculate the neighbouring points of each point (with K neighbour or ε neighbourhood).

(2) Define empowerment without a directed graph of the sample sets; if x_i and x_j are mutually neighbouring points, the edge of weights will be $d_x(i, j)$.

(3) Employ Dijkstra's or Floyd's algorithm to calculate the shortest distance between two points on the chart, which defines the distance matrix as $D_G = \{d_G(i, j)\}$.

(4) Use MDS for low-dimensional embedded manifold. The low-dimensional embedding is the eigenvectors of $\tau(D)$ corresponding to the second smallest to the (d + 1)th smallest eigenvalues

$$S = (S_{ij}) = (D_j^{\ 2})\ H = (H_j) = (\delta_j - 1/N)\ \tau(d) = -HSH/2$$

Methods of motion indexing and retrieval

Because human motion in the database has multiple attributes, obviously it is very difficult to directly retrieve the motion data, and the retrieval efficiency is very low. The purpose of establishing the index is to exclude most irrelevant motion to the retrieval samples from the motion database, which can avoid unnecessary traverse of a large-scale database and thus the retrieval efficiency is improved. Numerical similarity between two full-body motion sequences is not necessarily equivalent to their part-body similarity mainly due to high dimensionality of human motion data. For this reason, in this paper we break the full-body motion into a part-based hierarchy, in which the motions of each part have a much lower dimensionality than the full-body motions.

Motion pattern extraction

A motion pattern is a representative motion segment for a node in the constructed human hierarchy, where a node is a part of the human body. To extract motion patterns from the motion segments, we can cluster the motion segments into different groups. In our paper, we use quality threshold (QT) clustering [12] to partition the motion segments. QT clustering requires more computing power than K-means but does not require specification of the number of clusters in advance and always returns the same results when run several times. QT clustering is a recursive algorithm which can be briefly described as follows:

1. Set a threshold value for maximum diameter of clusters.
2. For each point in the current dataset, build a candidate cluster firstly by itself, and then add the closest point, the next closest, and so on until the diameter, which is the maximum distance of a point to all points in the current candidate cluster.
3. Take the candidate cluster with the largest number of points as the result of current recursion and remove all points of it from the current dataset.
4. Recursion with the reduced set of points.

With QT clustering, most of the motion segments are clustered into different groups, which are mainly differentiated by the types of the motion segments. From each of the groups, we can extract its motion pattern, which is defined as follows:

$$MotionPattern(k)=\left\{node(i)\left|\sum_{j=1}^{J}S_k^j(i)>J\alpha \ for \quad all \quad i\right.\right\} \quad (1)$$

$$S_k^j(i)=\begin{cases}1 \ if \quad string \quad j \quad in \quad cluster \quad k \quad possesses \quad node \quad i\\ 0 \qquad\qquad\qquad otherwise\end{cases} \quad (2)$$

Where

J is the number of segments in cluster k, and α is a ratio threshold value.

The representative motion segments, obtained from the above motion pattern extraction process, are stored as motion patterns in a library. Each human motion sequence is transformed to a total of 18 lists of pattern indexes, in which each node in the hierarchy has a pattern index list. The mappings between these pattern index lists and the original motion sub-sequences are retained for search purposes.

However, motions in some groups belong to the same motion type with only minor differences between the groups, such as jumps of different heights and behaviours with personal characteristics. So we implement the QT clustering algorithm again on the groups to remove the influence of minor differences and gather similar clusters together.

Computation of motion similarity

The typical matching method is dynamic time warping (DTW), which is time-consuming in computation, requiring O (m*n) time, where m and n are the lengths of two motion sequences, respectively, and $m \le n$. DTW is used in most of the current motion retrieval systems. In our paper, we adopt the classical Knuth-Morris-Pratt (KMP) string match algorithm [2] for motion pattern similarity computation. It is a kind of improved pattern matching algorithm whose advantage is to look for the biggest "jump", and the time complexity is O (n), where n is the number of motion sequences in the data repository. We compute the motion similarity scores of each body part between the pattern index lists of the query motion and that of any existing human motion in the database.

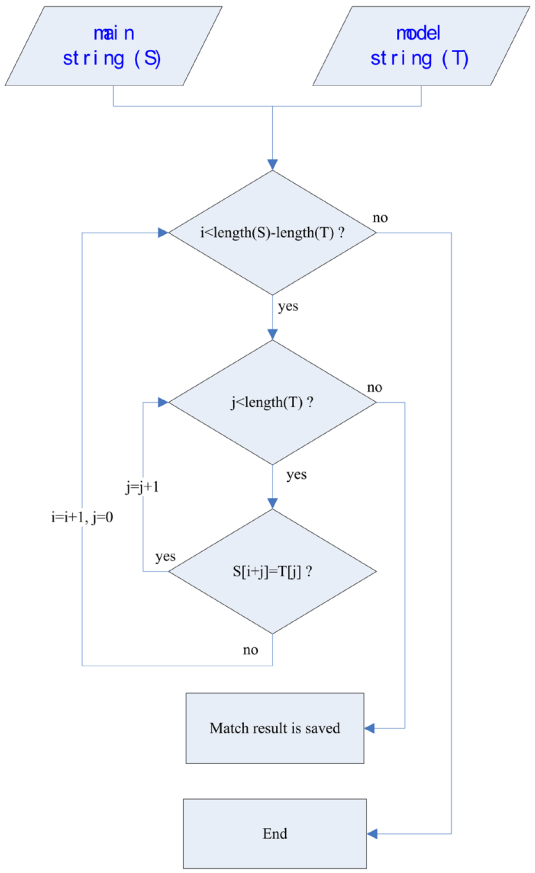

FIG. 2. THE FLOW CHAT OF THE KMP ALOGORITHM

The KMP algorithm is an improved pattern matching method in strings. The key of this algorithm is the definition of a so-called *next* function according to the given pattern strings. The *next* function includes the information of local matching of the pattern strings. The steps of this algorithm can be described as follows:

Begin: input main string S and model string T;

Step1: If the compare position i does not exceed the main string S go to Step2, else go to End;

Step2: If the compare position j does not exceed the main string T go to Step3, else go to Step4;

Step3: If the string S[i+j] is equal to the string T[j], j=j+1 and go to Step2, else i=i+1, j=0 and go to Step1;

Step4: Save the match result;

End: Matching process is ended, output the match results;

The flow chart of the algorithm is illustrated in Fig. 2.

If let $next[j]=k$, then $next[j]$ shows that when the jth character in the patterns mismatches the corresponding character in the main string, re-comparisons with the location of this character are needed. This leads to the definition of the *next* function for the pattern string as follows:

$$next[j]=\begin{cases}-1 & j=0\\ Max\left\{k\left|0<k<j\text{且}"T_0T_1\cdots T_{k-1}"="T_{j-k}T_{j-k+1}\cdots T_{j-1}"\right.\right\} & \text{when this set is not empty}\\ 0 & \text{el se}\end{cases}$$

Where $T_0T_1\cdots T_{k-1}$ is the prefix substring of $T_0T_1T_2\cdots T_{k-2}T_{k-1}$, $T_{j-k}T_{j-k+1}\cdots T_{j-1}$ is the postfix substring of $T_0T_1T_2\cdots T_{k-2}T_{k-1}$.

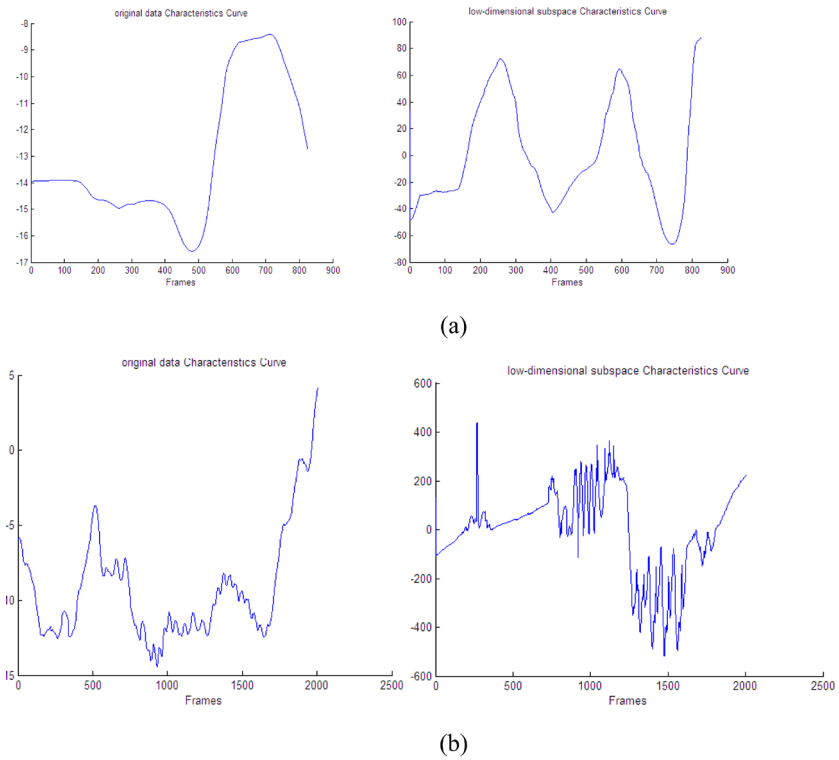

FIG. 3. THE CHARACTERISTIC CURVE OF DIFFERENT MOTION SEQUENCE

Sometimes two motions may exhibit high local similarities while they have notable differences from the global perspective. Therefore, we can use the following formula to fuse similarity scores at different layers for the purpose of result ranking.

$$S_{uchild} = \alpha \times S_{child} + (1-\alpha) \times S_{parent} \quad (3)$$

The basic idea is to transmit the similarity scores of the nodes at the lower layers to their parent nodes in the hierarchical tree. After every time of transmitting, we update the similarity score lists of the corresponding parent nodes. The propagation continues until the root node is reached.

Experimental results

Our experiment dataset comes from the CMU motion capture database [1], which is made up of 700 independent motion sequences. The number of frames is from hundreds to thousands with various lengths and velocities. Additionally, the database includes many motion types such as walking, running, kicking, boxing, jumping, etc. Users can retrieve similar movements through different types of motion examples. All experiments were implemented in Matlab 7.1 on a PC with 2 GB memory and a 2.70 GHz dual-core processor.

A *Jump* motion sequence with 826 frames is shown in Fig. 3 (a), and a *Stretches and jumps* motion sequence with 2028 frames is shown in Fig. 3 (b). The left figures in Figure 3(a) and Figure 3(b) are the characteristic curve of the original data and the right figures in Figure 3(a) and Figure 3(b) are the characteristic curve of the low dimensional subspace data. From Fig. 3, we can see that the characteristic of the motion in a low-dimensional subspace is more obvious than that in the original data. So the motion segmentation has a higher efficiency and better effect in the low-dimensional subspace than in the original motion data.

It follows from the characteristic curve of the low dimensional subspace data in Fig. 3(a) that this *Jump* motion is a periodic motion and a manual segment can be made in some key points such as the 405th frame and 745th frame. Thus, this motion sequence can be segmented as several motion clips and only one clip can be reserved when the motion synthesis is done later so that memory sizes and computational complexity could be reduced.

From the characteristic curve of the low dimensional subspace data in Fig. 3(b) we can see that this motion involves two types of motions. At the initial stage it is a stretch motion and then it becomes a periodic jump motion. This shows that we can segment this motion at the boundary of the two types of motions, i.e., at the 750th frame.

The clustering result by using the QT method is shown in Table 1 with comparisons to other methods. As an unsupervised learning approach, the QT method used here can cluster the motions automatically with the threshold values set in advance, while the kWAS [8] methods need to manually select a training dataset of each motion type and then classify new motion clips. Although

TABLE 1. CLUSTERING ACCURACY

Methods	kWAS[14]	K-means	QT
Clustering accuracy	90.3%	94.7%	97.4%
Learning method	supervised	unsupervised	unsupervised

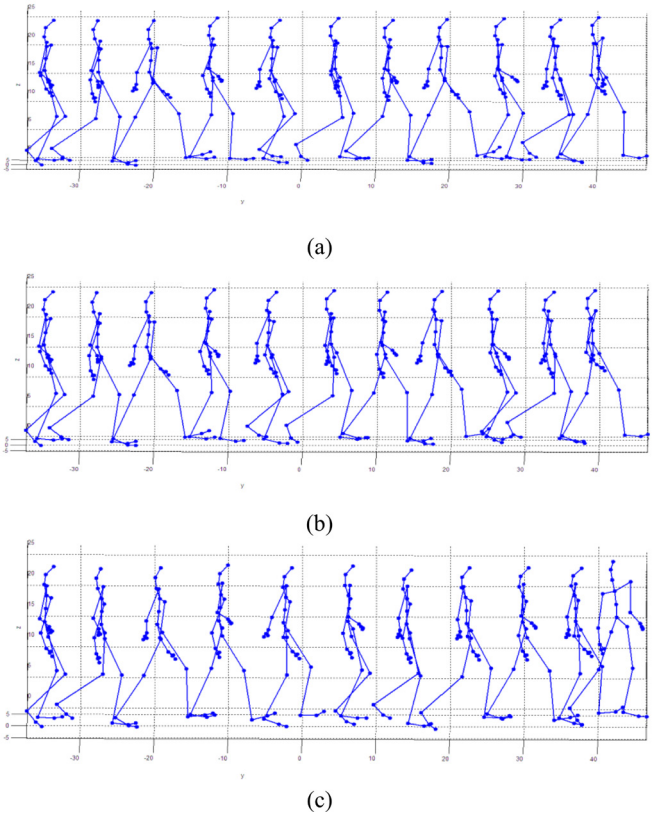

FIG. 4. (A) SAMPLES OF A WALK QUERY MOTION (B) THE BEST MATCH (C) THE FOURTH MATCH

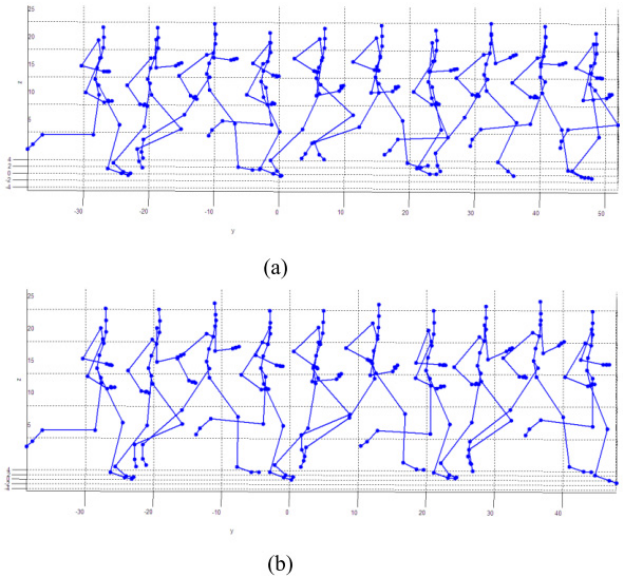

FIG. 5. (A) SAMPLES OF A RUN QUERY MOTION (B) THE BEST MATCH

the K-means method is also an unsupervised learning approach, it needs to specify the number of clusters in advance and the result is obtained always when the random points are chosen very skillfully.

Some of the retrieval results for a *Walk* motion example and a *Run* motion example are shown in Fig. 4 and Fig. 5, respectively. We can see that both the motions of best matches are very close to the corresponding query examples. While the difference from the fourth match and the query in Fig. 4 is quite obvious, they are still similar walking motions from the viewpoint of visual sense. From Table 2 below we can see that both the precisions and recall rates obtained by retrieving samples of different motion types are different. When the motion type changes are complicated, the precisions and recall rates are also reduced correspondingly. By comparing with the retrieval method based on the double-reference index (DRI) [13], we can find that the precisions, recall rates and retrieval time have significant improvements. The results confirm that our algorithm has a more ideal effect and reflect the effectiveness of the proposed algorithm.

Conclusion and future work

With the rapid development and increasing maturity of motion capture techniques, the motion datasets are becoming larger and larger. Thus an efficient motion data retrieval method is needed. In this paper, a novel efficient indexing and retrieval method of motion capture data is presented. We first hierarchically represent human motion. Due to the fact that the motion sequence contains complex movement types, the ISOMAP algorithm is adopted to segment the motion sequence. And then, motion patterns are extracted and the classical KMP string matching algorithm for motion pattern similarity computation is employed to efficiently retrieve motion sequences. Experimental results show that our method is effective.

The limitations of our method are as follows. First, the hypothesis of the ISOMAP segment method is that the original motion data are in the high-dimensional manifold. An uncorrected segment may occur when the motion data in the mapped low-dimensional subspace is segmented. For some motions in which their similar parts

TABLE 2. THE RETRIEVAL PERFORMANCE STATISTICS

Examples	Precision/%		Recall/%		Retrieval time/s	
	DRI[15]	Our method	DRI[15]	Our method	DRI[15]	Our method
Walk	91.1	93.4	94.5	94.7	4.2	0.4
Run	90.2	92.7	93.1	93.2	4.4	1.1
Jump	89.3	91.5	92.3	92.5	5.2	1.5
Dance	85.5	87.2	89.7	90.8	5.3	2.1

are too simple, these similar parts are mapped into the same area and cannot be distinguished and segmented correctly when they are mapped into the low-dimensional space. For example, for the right hand boxes and the left hand boxes, the starting postures of these two motions are standing and ready to punch, but the postures in the mapping low-dimensional space cannot be distinguished. For this reason, the corrected results cannot be obtained by using the ISOMAP method. In the next work, we will seek more appropriate segment methods to reduce the erroneous segment. Second, the clustering time of the QT clustering algorithm we used here is a bit long. In the future work, how to get a more efficient clustering algorithm remains to be investigated.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 60875046), by the Program for Changjiang Scholars and Innovative Research Team in University (No. IRT1109), the Key Project of the Chinese Ministry of Education (No. 209029), the Program for Liaoning Excellent Talents in University (No. LR201003), the Program for Liaoning Science and Technology Research in University (No. LS2010008, 2009S008, 2009S009, LS2010179), the Program for Liaoning Innovative Research Team in University (Nos. 2009T005, LT2010005, LT2011018), the Natural Science Foundation of Liaoning Province (201102008) and by „Liaoning BaiQianWan Talents Program (2010921010, 2011921009)".

REFERENCES

1. CMU Mocap Database. http://mocap.cs.cmu.edu, 2008.
2. Deng Z., Gu Q., Li Q. Perceptually consistent example-based human motion retrieval. Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games, I3D'2009. Boston, Massachusetts 2009;pp.191-198.
3. Gaurav N.P., Chuanjun L., Balakrishnan P. Hierarchical indexing structure for 3D human motions. In: T.-J. Cham et al. (eds.) MMM 2007, LNCS, 2007;4351:386-396.
4. Gaurav N.P., Balakrishnan P. Indexing 3D human motion repositories for content-based retrieval. IEEE Trans. Inf. Technol. Biomed. 2009;13:802-809.
5. Gu Q., Peng J., Deng Z.S. Compressions of human motion capture data using motion pattern indexing. Comp. Graph Forum 2009;28:1-12.
6. Keogh E., Palpanas T., Zordan V., Gunopulos D., Cardle M. Indexing large human–motion databases. In: Proceedings of the 30th VLDB Conference, 2004;30:780-791.
7. Knuth D.E, Morris J.H, Pratt V.B. Fast pattern matching in strings. SIAM J. Comp. 1977;6:323-350.
8. Li C., Zheng S.Q., Prabhakaran B. Segmentation and recognition of motion streams by similarity search. ACM Trans. Multimedia Computing, Communications, and Applications, 2007;3:Article No. 16.
9. Muller M., Roder T., Clausen M. Efficient content-based retrieval of motion capture data. ACM Trans. Graphics 2005;24:67-685.
10. Sonoda M., Tsuruta S., Yoshimura M., Hachimura K. Segmentation of dancing movement by extracting features from motion capture data. J. Inst. Image Electronics Eng. of Japan 2008;37:303-311.
11. Worawat C., Woong C., Kozaburo H. A quick filtering for similarity queries in motion capture databases. In: P. Muneesawang et al. (eds.) PCM 2009, LNCS 2009;5879:404-415.
12. Wu S., Xia S., Wang Z.Q., Li C. Efficient motion data indexing and retrieval with local similarity measure of motion strings. Vis Computer 2009;25:499-508.
13. Xiang J., Guo T., Wu F., Zhuang Y.T., Ye L. Motion retrieval based on large-scale 3D human motion database by double-reference index (in Chinese). J. Comp. Res. Develop. 2008;45 2145-2153.
14. Xing W., Zhiwen Y., Hausan W.P. 3D motion sequence retrieval base on data distribution. In: Proceeding of IEEE International Conference on Multimedia and Expo, ICME 2008. 2008;pp.1229-1232.
15. Yamasaki T., Aizawa K. Content-based cross search for human motion data using time-varying mesh and motion capture data. In: Proceeding of IEEE International Conference on Multimedia and Expo, ICME 2007. Beijing, China 2007;pp.2007-2009.
16. Yan G., Lizhuang M.T. Content-based human motion retrieval with automatic transition. In: H.-P. Seidel, T. Nishita, Q. Peng (eds.) CGI 2006, LNCS 2006;4035:360-371.
17. Yi L.T. Efficient motion search in large motion capture database. In: G. Bebis et al. (eds.) ISVC 2006, LNCS. 2006;4291:151-160.