Finding Repetitive Patterns in 3D Human Motion Captured Data

Kai-Tai Tang
Department of Computer Science, City University of Hong Kong, Hong Kong
itjeff@cityu.edu.hk

Howard Leung
Department of Computer Science, City University of Hong Kong, Hong Kong
howard@cityu.edu.hk

Taku Komura
School of Informatics, The University of Edinburgh, United Kingdom
tkomura@informatics.ed.ac.uk

Hubert P.H. Shum
School of Informatics, The University of Edinburgh, United Kingdom
hubert.shum@ed.ac.uk

ABSTRACT

Finding repetitive patterns is important to many applications such as bioinformatics, finance and speech processing, etc. Repetitive patterns can be either cyclic or acyclic such that the patterns are continuous and distributed respectively. In this paper, we are going to find repetitive patterns in a given motion signal without prior knowledge about the type of motion. It is relatively easier to find repetitive patterns in discrete signal that contains a limited number of states by dynamic programming. However, it is impractical to identify exactly matched states in a continuous signal such as captured human motion data. A point cloud similarity of the input motion signal itself is considered and the longest similar patterns are located by tracing and extending matched posture pairs. Through pattern alignment and auto-clustering, cyclic and acyclic patterns are identified. Experiment results show that our approach can locate repetitive movements with small error rates.

Keywords

3D human motion capture, pattern discovery, repetitive pattern, cyclic and acyclic patterns, point cloud similarity.

1. Introduction

Repetitive patterns are frequently appearing units commonly found in daily life, for example, keywords in a text paragraph or repetitive logos on the clothes. They can be either cyclic or acyclic. A cyclic pattern repeats continuously while an acyclic pattern is non-continuous and distributed over the time or region. Finding repetitive patterns in either discrete or continuous signals leads to the development of many important applications such as detection of the motifs in DNA sequence, prediction of the trend of the stock price, and mining for desirable segments in speech or motion signals.

Repetitive patterns in discrete signals such as text paragraphs, DNA sequence and music data can be found by exact string matching techniques [1][2][3] or inexact matching with dynamic programming [4][5]. Researchers believe that repetitive tandems in DNA sequence associated with disease syndromes [6]. It has been a popular topic for bioinformatics researchers to repetitive tandems in DNA sequences. Gilbert *et al.* [7][8] discover repeated patterns by extending matched sub-strings. To find repetitive patterns in music data, Hsu and Liu [9] consider exact string matching through the correlation matrix of the music notes in sequence.

Some researchers attempt to match for similar segments in continuous signal such as financial, speech, and motion data. Since continuous signal does not contain exactly matched states, the problem is harder than that in discrete signal for finding similar segments. Wu *et al.* [10] predict the trend of stock price by matching for history signals of similar shapes in a continuous financial data curve. In speech processing, Park [11] considers a point cloud similarity between an input signal and a known template in order to identify some spoken keywords. More specific to human motion studies, Kovar and Gleicher [12] consider the point cloud similarity between a query and the motion data sequence and extract logically related motions for motion blending. They approximate the optimal matching paths by line tracing techniques. However, they focus in motion retrieval so a known template should be provided and the length of result pattern is confined. Forbes and Fiume [13] attempt to improve the work by Kover *et al.* by indexing the point cloud by manually defining key postures in order to speed up the searching. However, the number of available motion database is growing rapidly and it is impractical to spend a lot of manpower to do such pre-processing. Some researchers attempt to detect cyclic patterns in captured human motion. Li and Holstein [14] detect cyclic motion by constructing motion templates of standard movements like walking in frequency domain. Meng *et al.* [15] extend the work by Li and Holstein. However, this approach requires the user to know about the types of input motion in advance. Laptev *et al.* [16] detect cyclic movement by aligning a sequence of space-time corresponding points in video frames. Given a known cyclic motion, Ormoneit *et al.* [17] detect the cycles by folding and overlapping the input motion until the minimal signal-to-noise error is attained. However, detection of acyclic repetitive patterns has received much less attention in the literature.

To break through the limitations of existing template matching approaches, it motivates us to solve an unsupervised pattern discovery problem in which repetitive patterns in 3D human motion captured data are automatically detected without knowing the types of input patterns. A point cloud matrix of posture similarity is considered and the longest similar motion segment pairs are located through tracing for the diagonal patterns. Cyclic and acyclic patterns are then identified by automatic alignment and clustering. Finally, the duration of each cycle is estimated by the auto correlation method. The robustness of our algorithm will be shown in the experiment results. Challenging cases that can be handled by our algorithm will also be demonstrated.

2. Proposed algorithm

In this paper, repetitive patterns including both cyclic and acyclic patterns are discovered in an unsupervised way in which no template query is needed. The input is a 3D human motion captured data, which is a high dimensional continuous signal. The point cloud similarity approach, which has been used for matching similar regions on a continuous signal, is adopted to deal with the problem. Our algorithm aims to relax the assumption in existing methods that require a known template query in order to search for similar patterns.

Figure 1 shows the overview of our algorithm. First, a grayscale point cloud matrix of posture similarity values is generated. Similar postures are then clustered by turning the point cloud into a binary representation. The local minima of each cluster are traced diagonally to form feature points that approximate the duration of each similar motion segments. The longest possible matching paths are obtained by merging feature points with the least dynamic time warping (DTW) cost. Finally, the cyclic and acyclic patterns are identified by pattern alignment and auto-clustering techniques. The period in each cycle pattern will then be estimated by the auto correlation method.

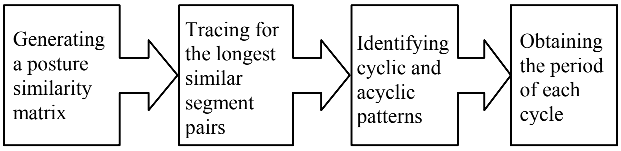
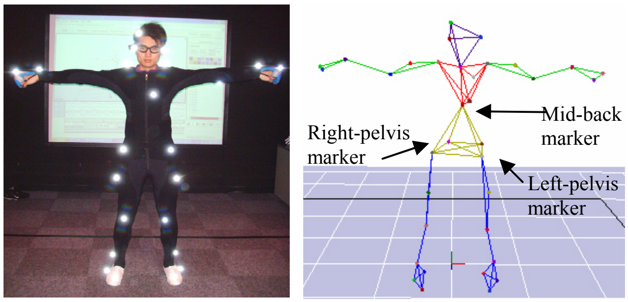


Figure 1. The overview of our proposed algorithm.

2.1 Data acquisition and normalization

The motion data is captured by an optical motion capture system as shown in Figure 2. The capturing area is covered by seven cameras from different view points. During the capture, the actor as shown in Figure 3 should wear a suit with 35 optical markers attached to different body parts such as the head, the torso and the limbs. The 3D motion of the actor is captured as a time sequence of frames, while each frame contains the time stamp and the corresponding 3D posture data in terms of 3D marker coordinates.

Figure 2. The motion capture area.



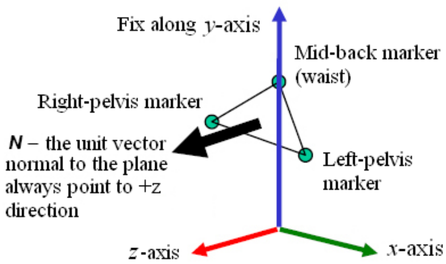
(a) Optical markers adhered to the body (b) Captured 3D posture represented by a skeleton

Figure 3. The actor and corresponding captured skeleton.

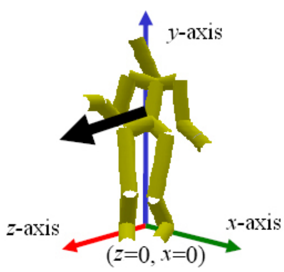
The actor is free to move around the capture area thus global translation and rotation are incorporated into the 3D coordinates of the captured motion. Normalization of horizontal translation and frontal orientation is thus needed to facilitate the posture comparison. The positions of the markers p_i are first translated in order to make the marker of the body center (Mid-back) invariant to the origin of the horizontal plain. The vertical displacement is allowed hence the y coordinate is not normalized. The translation function is given by Equation (1):

$$p_i = \left(x'_i, y'_i, z'_i\right) = \left(x_i - x_{Mid-back}, y_i, z_i - z_{Mid-back}\right) \quad \textbf{(1)}$$

A rotation function is defined to ensure that the actor is always facing the front (the positive z direction) as shown in Figure 4(b). Three markers defining the major orientation of the body is shown in Figure 4(a) and the unit vector $\boldsymbol{N}$ normal to the resulting plane is calculated. The rotation angles θ_z and θ_x are obtained by the dot product between $\boldsymbol{N}$ with the z-axis and with the x-axis respectively. The coordinates of each marker p_i are then rotated by θ_z and θ_x accordingly.



(a) Normalization of frontal orientation



(b) Normalized posture

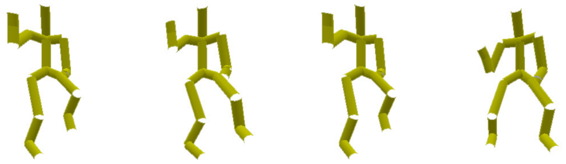
Figure 4. Normalization of a 3D posture.

2.2 Generating a posture similarity matrix

A point cloud similarity matrix locates similar postures in clusters. The similarity between every pair of postures is computed by a similarity cost function.

2.2.1 Posture similarity cost

The perceptual similarity between a pair of postures is modeled by their spatial difference. A pair of perceptually similar postures show a similar concept (e.g. the right hand is up-lifted), and similar normalized coordinates in the 3D space. Figure 5 shows two pairs of spatially similar and dissimilar postures.



(a) Spatially similar postures (b) Spatially dissimilar postures

Figure 5. Similar and dissimilar postures.

Let $p_i(x_m, y_m, z_m)$ and $p_j(x_m, y_m, z_m)$ be the m-th marker coordinates of the posture pair (i, j) respectively, the similarity cost $C(i, j)$ is defined by averaging the Euclidean distances between the 3D coordinates of M pairs of corresponding markers as shown in Equation (2). In our setting, M=35 markers are considered. The more similar the posture pair, the smaller the similarity cost.

$$C(i,j) = \frac{\sum_{m=1}^{M} \left\| p_i(x_m, y_m, z_m) - p_j(x_m, y_m, z_m) \right\|}{M} \quad \textbf{(2)}$$

2.2.2 The point cloud similarity matrix

Given a motion signal that contains n frames, a $n \times n$ point cloud matrix of posture similarity costs is constructed. The similarity costs are normalized to a range between 0 and 1, which is easier to be compared and visualized by a grayscale bitmap. Figure 6 shows a grayscale point cloud similarity matrix of a motion. A point with a darker color exhibits a higher similarity between the corresponding pair of postures. Only the lower half of the matrix is considered because the matrix is symmetric along the diagonal axis ($x = y$).

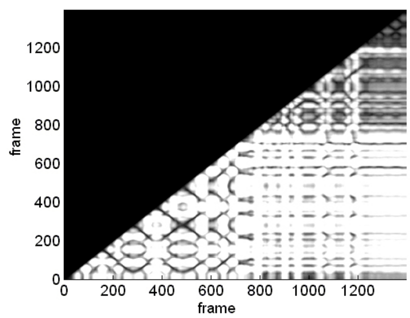
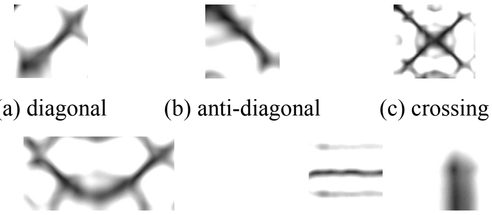


Figure 6. The similarity matrix is visualized by a grayscale point cloud bitmap.

In the point cloud similarity matrix, five types of dark pattern are observed: (a) diagonal, (b) anti-diagonal, (c) crossing, (d) V-shape, and (e) horizontal / vertical. Diagonal pattern is running from bottom-left to top-right in which the corresponding signal segment pairs are mapped along time sequence. Anti-diagonal pattern is invalid because one segment of the matched pair is time-reversed. Cross and V-shape patterns are just valid for the diagonal portion. Horizontal or vertical pattern is more or less a one-to-many mapping of postures such as a stationary posture.



(a) diagonal (b) anti-diagonal (c) crossing

(d) V-shape (e) horizontal / vertical

Figure 7. Dark patterns observed in the point cloud.

2.3 Tracing for the longest similar segment pairs

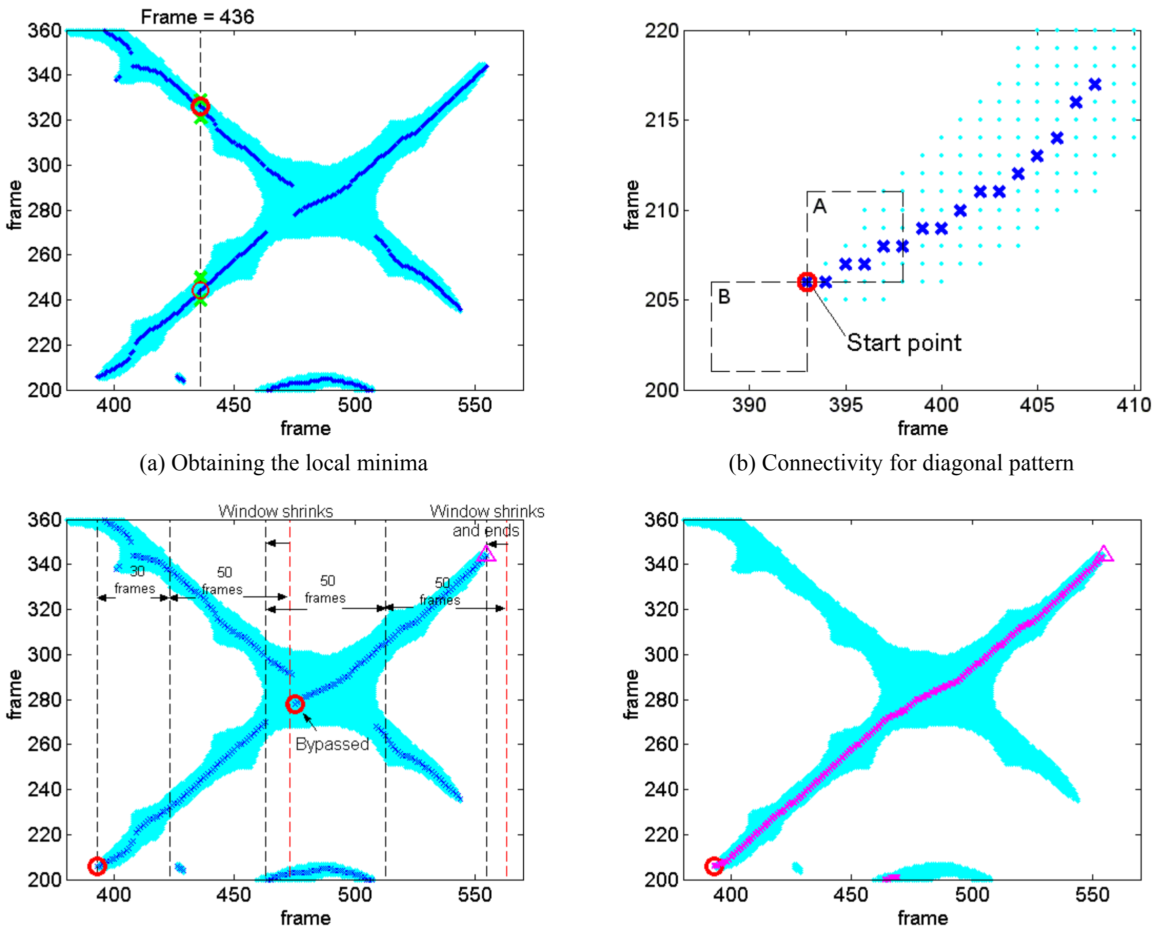
It is non-trivial to search for similar signal pairs from the grayscale point cloud because some regions are quite ambiguous. Hence, as the first step, a binary point cloud is obtained by filtering dissimilar posture pairs. Start points of the valid patterns described in section 2.2.2 are then located. The possible continuation of each start point is determined by dynamic time warping (DTW) with a shrinking window technique, which will be described in later paragraphs.

2.3.1 Obtaining the binary point cloud

To obtain the binary point cloud, only similar posture pairs are kept as pattern points. A classifier is trained to determine whether a pair of input postures is perceptually similar. The ground truth similar and dissimilar pairs are determined by some users through subjective evaluation. Equal number of samples in each class is selected randomly and a total of 100 ground truth similar/dissimilar pairs are obtained. From the distribution of matching costs exhibited by similar and dissimilar pairs, the false matched rate and false non-matched rate can be observed. The equal error rate, with which the false matched rate is equal to the false non-matched rate, is used to determine the threshold of the similarity cost to classify between similar/dissimilar pairs.

2.3.2 Locating start points of valid patterns

After the binary point cloud is obtained, the start points of all valid patterns are then estimated. The binary point cloud is thinned by considering the vertical local minima in terms of similarity cost. For each frame, the local minima are identified as shown in Figure 8(a). The dark thin lines illustrate the thinned patterns. However, we only chose the valid portion as described in section 2.2.2. Therefore, there are a few constraints for the selection of start points as shown in Figure 8(b). First, a start point should be the bottom-leftmost point of the pattern and hence has no minima points appearing in the preceding time (region B). Also, it should have neighboring points in next frames (region A).



(a) Obtaining the local minima

(b) Connectivity for diagonal pattern

(c) Pattern tracing

(d) The optimal path of the traced pattern

Figure 8. Pattern tracing procedures.

2.3.3 Tracing patterns

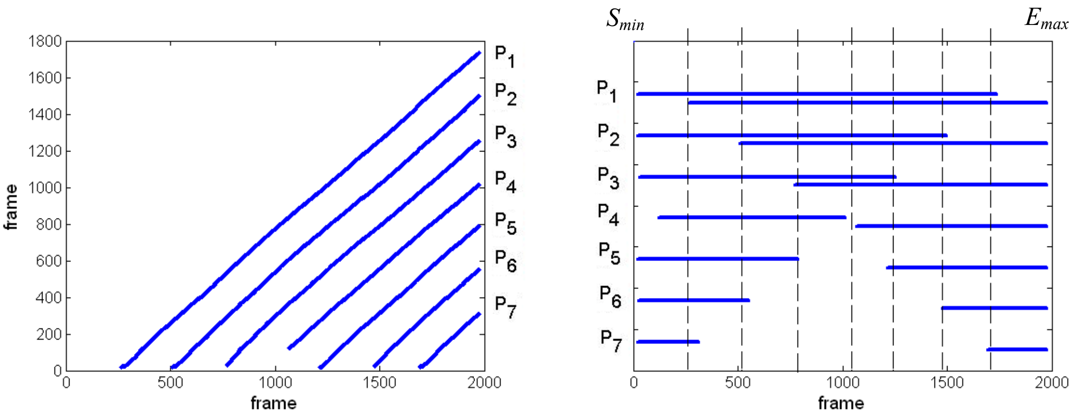
The pattern tracing starts from the bottom-left corner of the point cloud. Figure 8(c) shows the procedures of tracing a diagonal pattern of a selected region of the point cloud. Suppose that the start point (in circle) near to the bottom-left corner is considered, possible continuations of a pattern are determined by DTW. A window of certain size slides across the horizontal axis without overlapping is used to locate candidate of continuation points by cutting across the vertical axis for minima points. A valid candidate should make a slope with the start point no greater than 2 and no smaller than 1/2. There are some possible cases: (1) cuts exactly one point, (2) cuts more than one points, and (3) cuts no points. For the first case, the candidate is obviously the continuation point. If more than one candidate is obtained, the one with the minimal DTW cost is selected. If the window cuts no points, the window shrinks progressively until a valid candidate is found. Once a candidate is found, it becomes the new start point for next iteration until no further continuation points are found. The DTW path is safeguarded by a threshold of 0.5 (note that the DTW cost is normalized by the length of the optimal path and falls into the range [0, 1]). When the DTW cost is greater than the threshold, the candidate will be ignored and this ensures the continuation of pattern is valid. In our setting, an initial window size of 30 frames is used as we assume that a valid pattern should have at least 30 frames. A more aggressive window size of 50 frames will be tried for next iterations. Although there may have some gaps between pattern lines, our algorithm can verify whether the gaps are acceptable for a longer pattern. If yes, the start point of the pattern next to the gap is simply bypassed. Figure 8(d) shows the optimal path determined by DTW after the pattern tracing steps.

2.4 Identifying cyclic and acyclic patterns

Repetitive patterns can be either cyclic or acyclic and sometimes both of them may appear in the input motion. Figure 9 shows the character sequences that demonstrate the modes of repetitive patterns. Existing approaches assume that the input motion should be cyclic only and they cannot detect cycles in the mixed mode because the signal-to-noise ratio is too small.

…CCCCC…	…ADBAEGH…	...ADCAFCCCC...
(a) A cyclic pattern	(b) An acyclic pattern	(c) Mixed cyclic and acyclic patterns

Figure 9. Modes of repetitive patterns.



(a) A family of cyclic patterns (b) Illustration of the aligned patterns

Figure 10. Alignment of patterns.

The property of cyclic pattern can be observed through the optimal paths located on the point cloud. Figure 10(a) shows the result of a sample dance motion, which consists of a continuous sequence of waltz steps. This motion consists of cyclic motion only that makes the problem simpler to be illustrated. Each diagonal line P_i represents a pair of similar segments. Because our algorithm obtains the segment pairs as long as possible, for a cyclic motion there exists certain degree of pattern overlapping. Moreover, the patterns form a right-angled triangular region and these patterns belong to the same family of cyclic pattern.

Suppose the patterns are transformed from 2D domain into 1D time line as shown in Figure 10(b). The durations of each pair of segments can be observed. It is clear that for each pair of segments, the minimum of the start points (S_{min}) and the maximum of end points (E_{max}) are likely to be aligned together. It gives us a good classification feature to distinguish whether a pattern belongs to the same family as others.

An auto-clustering method is introduced to classify cyclic and acyclic patterns based on the alignment feature. The procedure is shown in Figure 11. Suppose there are three resulting patterns P_1, P_2, and P_3 and their relationship is unknown. One of the patterns is picked as the initial set, say P_2. Next, each remaining pattern is aligned with the patterns in the existing sets. Suppose P_1 is considered, its S_{min} and E_{max} values will be compared with the set in P_2. A clustering cost C_c is defined in Equation (3) to quantify the measurement, in which N represent the total number of patterns in each set to be examined. If C_c is smaller than 5% of the average pattern duration D (Equation (4)) in each set, the pattern is put into the set. If there is more than one set, the set with the minimal C_c is chosen to enter.

$$C_c = \frac{\sum_{i=1}^{N} \left| E_{\max} - E_{\max_i} \right| + \left| S_{\min} - S_{\min_i} \right|}{N} \tag{3}$$

$$D = \frac{\sum_{i=1}^{N} \left| E_{\max_i} - S_{\min_i} \right|}{N} \tag{4}$$

After the clustering, it is ready to distinguish between cyclic and acyclic patterns. A cyclic set contains more than one patterns (N>1) while an acyclic set contains one pattern (N=1) only.

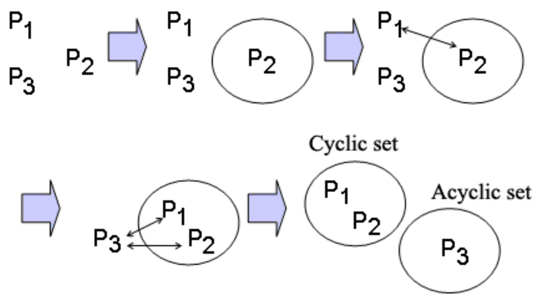


Figure 11. Auto-clustering into cyclic and acyclic patterns.

2.5 Obtaining the period of each cycle

By observation, there should be N+1 cycles for a family having N patterns, assumed that there are no missing patterns and wrong patterns included. However, it is not reliable to simply count the number of patterns in each family because the assumption may not hold all the time. Consider a sinusoidal signal of period P, the signal can perfectly overlap to itself with a phase difference of the multiples of P. Similarly, the movement of each marker can be resolved into x, y, and z displacements. If the displacement signal is cyclic, it can overlap with itself at particular phase differences. An auto correlation method is used to detect the boundaries of each cycle.

Recall the 1D alignment of patterns as shown in Figure 10(b), the cyclic patterns are overlapped at different time. The duration of each cyclic family is hence obtained from the alignment result. Next, the duration is selected as an independent motion sequence $\{x_i\}$ of length L that contains only one kind of cyclic movement. The correlation coefficient $r_m(x)$ of the sequence $\{x_i\}$ with itself at different lags without warping $\{x_{i+lag}\}$ is computed as equation (5). The correlation-lag information of x, y and z coordinates of all 35 markers m are combined linearly into r with equal weighting.

$$r_m(x) = \frac{\sum_{i=0}^{L-1}\left[(x_i - \bar{x})(x_{i+lag} - \bar{x})\right]}{\sigma_x^{\ 2}} \quad \textbf{(5)}$$

In cyclic motion segment, the input signals $\{x_i\}$ and $\{x_{i+lag}\}$ can overlap with each other somewhere, and hence a peak will be obtained in the correlation-lag graph. Figure 12 shows the correlation-lag graph of a motion data with eight waltz dance steps only. Imagine that a motion trajectory slides to the left and overlaps with a motion trajectory without lag. The correlation coefficient decreases and then reaches a peak again after certain lag value. The peaks with sufficiently high correlation coefficients are the boundaries between every two adjacent cycles, and the number of peaks plus one is hence the number of estimated cycles. As future work, the start / end points of the member patterns in a cyclic family will be also considered, which approximates the period of each cycle and enhances the searching for best cutting point of each cycle.

Finally, the cyclic patterns are cut accordingly and the period of each cycle is determined. They are grouped together as the same repetitive patterns. The distributed acyclic patterns are then grouped according to the overall similarity of the motion segments and their durations.

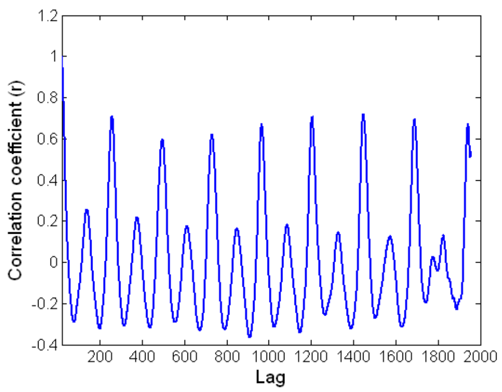


Figure 12. Similarity of stating frame and the frames in the whole cyclic duration.

3. Experiments and results

Our experiment data contains 22 motion clips with 4 types of dances: Waltz, Pop dance, Hip hop dance and House dance as shown in Table 1.

Table 1. Experiment dataset.

Motion type	Number of samples	Average number of frames
Waltz dance	1	1975
Pop dance	1	1397
Hip hop dance	10	2505.6
House dance	10	2109.3

The dance motions are performed by two dancers containing both cyclic and acyclic movements. The motion data contains complicated movements and with long duration which can test the robustness of our algorithm. To evaluate the performance of our algorithm, both type I and type II errors will be checked. Type I error is the false positive rate that measures how many repetitive patterns located by our algorithm are actually not repetitive according to human perception. Type II error is the false negative rate that measures how many missing repetitive patterns that are not detected by our algorithm.

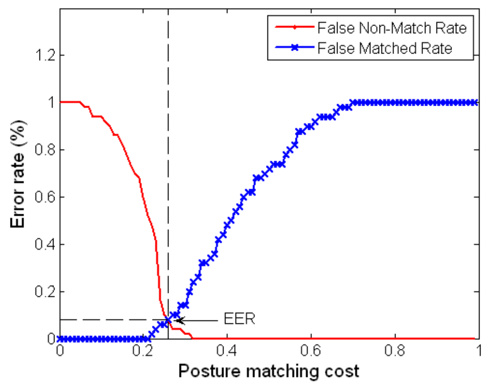


Figure 13. Training result of the threshold value.

Figure 13 shows the training result of similarity threshold. The value 0.26 that yields the equal error rate (EER) is chosen as the threshold T to distinguish similar and dissimilar postures because at this point both the false matched rate and false non-matched rate are low. The pairs with similarity cost smaller than T are classified as similar pair, otherwise are dissimilar. This result is used in the binary point cloud formation and pattern tracing.

In our experiment, U pairs of similar motion segments are first identified by our algorithm. The animation of each pair of motion segments is played at the same time and evaluated by some observers subjectively. By the human judgment, u out of U pairs may be considered as false positive. The remaining frames that are regarded as non-repetitive by our algorithm are then collected into a new sequence R. It is then segmented into s_r segments of window size w. A window size of $w = 30$ frames has been chosen because we only accept valid repetitive patterns of more than 30 frames. Each window slides across the entire sequence R and the cross correlation coefficient r_R is computed by Equation (6). The correlation coefficient of each segment pair is given by the average of correlation coefficients r_x, r_y, and r_z of x, y, and z axis respectively. We set a rather loose condition that the pairs with correlation higher than 0.5 are accepted as candidate false negative pairs. Finally, the candidate false negative pairs are evaluated subjectively by some observers and a total of v false negative pairs are obtained.

$$r_R = \frac{\sum_{i}^{30}\left[(R_i - \bar{R})(R_i' - \bar{R}')\right]}{\sigma_R \sigma_{R'}} \quad \textbf{(6)}$$

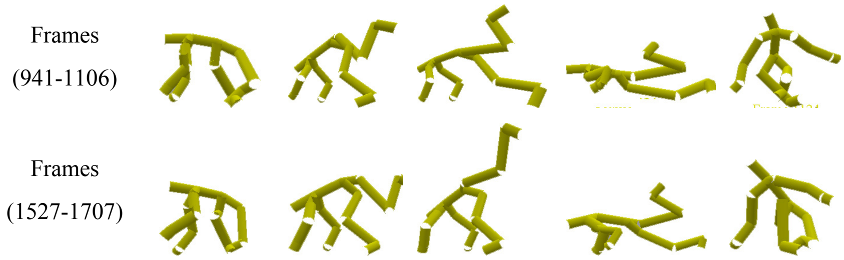


Figure 14. A complicated movement.

There are *U-u+v* pairs of ground truth repetitive pairs obtained per observer. Hence, the type I error rate is given by *u*/(*U-u+v*) while the type II error is given by *v*/(*U-u+v*). Table 2 shows the experiment result. The false negative rates are 0% for all input dance motions. It shows that our algorithm can locate all possible repetitive patterns without prior knowledge of the input motion. In our experiment, only segment pairs with cross correlation coefficient greater than 0.5 are considered as missing cases. This value is sufficiently low for a missing similar pair because similar motion segment always has a correlation coefficient higher than 0.6. The 0% false negative rate shows that our algorithm is capable to detect all possible repetitive patterns without missing.

On the other hand, a relatively low average false positive rate (7.7%) is obtained. Table 2 also shows the false positive rates of each type of dance motion. Our algorithm gives the best estimation in waltz dance (0% in both false positive and false negative rates) while the worse in hip-hop dance (12.8% in false positive rate and 0% in false negative rate). According to the comments from the observer, the hip-hop dance is the most complicated while the waltz dance is quite straight forward. It shows that complicated movements are likely to have observable differences. In most of the erroneous cases, the motion pairs may look alike with difference in positions of a particular limb. It shows that averaging the Euclidean distances of joint positions is not enough because it is unfair to treat the displacement of the joints of end effectors equally to relatively static joints such as the shoulders. Moreover, variation in body size and limb lengths for different dancers may lower the accuracy. To be more generic, joint angles could be considered as a feature in the similarity function in the future. Also, the movements of more active limbs could be boasted by a larger weight in order to make the comparison more conformed to the human perception. Our experiment is ongoing and more motion data of different types will be studied later.

Table 2. The experiment results.

Motion type	Type I error (False positive rate)	Type II error (False negative rate)
Waltz dance	0.0%	0.0%
Pop dance	12.5%	0.0%
Hip hop dance	12.8%	0.0%
House dance	5.5%	0.0%

Figure 14 shows a pair of similar motion with difficult movements, which is a challenging case that our algorithm is able to handle. The movement is relatively fast and it involves movements of the whole body. It shows that our algorithm is robust enough to catch such difficult movement with rapid change.

4. Conclusion

We proposed a method to locate repetitive patterns in captured 3D human motion without prior knowledge about the input pattern. Patterns of different lengths can be discovered by considering the input signal alone without a query. Repetitive patterns are traced from a point cloud of similar postures. Complete pattern is obtained by joining points according to their connectivity along the diagonal. Finally, cyclic and acyclic patterns are identified by pattern alignment and auto-clustering. The period of each cycle is also estimated successfully. Experiment result shows that our method has low false positive and false negative rates and is able to handle complicated cases.

As future work, the repetitive patterns discovered by our proposed method can be used for summarizing a piece of captured motion. By the way, the discovered repetitive patterns can be applied to generate dance lesson automatically. The ubiquitous dance education system developed by our group [18] required teachers to design dance courses manually. Our algorithm can group repetitive movements that are likely the theme movements of the captured motion performed by teacher. The student can learn frequently appearing movements first and then their variations. Our proposed method can also be applied in motion data retrieval by considering the repetitive pattern and periodicity etc. as index features that may uniquely define a motion clip.

5. ACKNOWLEDGMENTS

The work described in this paper was fully supported by a grant from the Research Grants Councils of the Hong Kong Special Administration Region, China (Project No. CityU 1167/05E).

6. REFERENCES

[1] Qu, Y., Wang, C., and Wang, X. S. 1998. Supporting fast search in time series for movement patterns in multiple scales. In *Proceedings of the Seventh international Conference on information and Knowledge Management* (Bethesda, Maryland, United States, Nov. 02 - 07, 1998). CIKM '98. ACM, New York, NY, 251-258.

[2] Agrawal, R., Psaila, G., Wimmers, E. L., and Zaït, M. 1995. Querying Shapes of Histories. In *Proceedings of the 21th international Conference on Very Large Data Bases*

(September 11 - 15, 1995). U. Dayal, P. M. Gray, and S. Nishio, Eds. Very Large Data Bases. Morgan Kaufmann Publishers, San Francisco, CA, 502-514.

[3] Huang, Y. and Yu, P. S. 1999. Adaptive query processing for time-series data. In *Proceedings of the Fifth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining* (San Diego, California, United States, Aug. 15 - 18, 1999). KDD '99. ACM, New York, NY, 282-286.

[4] Wagner, R. A. and Fischer, M. J. 1974. The String-to-String Correction Problem. J. ACM 21, 1 (Jan. 1974), 168-173.

[5] Shatkay, H. and Zdonik, S. B. 1996. Approximate Queries and Representations for Large Data Sequences. In *Proceedings of the Twelfth international Conference on Data Engineering* (Feb. 26 - Mar. 01, 1996). S. Y. Su, Ed. ICDE. IEEE Computer Society, Washington, DC, 536-545.

[6] Sutherland, G.R. and Richards, R.I. 1995. Simple tandem DNA repeats and human genetic disease. *Proc. Natl Acad. Sci.* USA, 92, 3636–3641.

[7] Gilbert, D. and Viksna, J. 1999. Pattern discovery methods for protein topology diagrams, *German Conference on Bioinformatics*, (1999), 194-196.

[8] Gilbert, D., Westhead, D., and Viksna, J. 2003. Techniques for comparison, pattern matching and pattern discovery: From sequences to protein topology, in *Artificial Intelligence and Heuristic Methods in Bioinformatics*, Paolo Frasconi and Ron Shamir (Eds), IOS Press, ISBN 1-58603-294-1, (2003), 128-147.

[9] Hsu, J.-L. and Liu, C.-C. 2001. Discovering Nontrivial Repeating Patterns in Music Data. IEEE Transactions on multimedia, 3 (3), (Sep. 2001) 311-325.

[10] Wu, H., Salzberg, B., and Zhang, D. 2004. Online event-driven subsequence matching over financial data streams. In *Proceedings of the 2004 ACM SIGMOD international Conference on Management of Data* (Paris, France, Jun 13 - 18, 2004). SIGMOD '04. ACM, New York, NY, 23-34.

[11] Park, A. and Glass, J. R., Towards Unsupervised Pattern Discovery in Speech. Proc. ASRU, (San Juan, Puerto Rico, 2005), 53-58.

[12] Kovar, L. and Gleicher, M. 2004. Automated extraction and parameterization of motions in large data sets. In *ACM SIGGRAPH 2004 Papers* (Los Angeles, California, August 08 - 12, 2004). J. Marks, Ed. SIGGRAPH '04. ACM, New York, NY, 559-568.

[13] Forbes, K. and Fiume, E. 2005. An efficient search algorithm for motion data using weighted PCA. In *Proceedings of the 2005 ACM Siggraph/Eurographics Symposium on Computer Animation* (Los Angeles, California, Jul. 29 - 31, 2005). SCA '05. ACM, New York, NY, 67-76.

[14] Li, B. and Holstein, H. 2002. Recognition of Human Periodic Motion - A Frequency Domain Approach. In *Proceedings of the 16th International Conference on Pattern Recognition (Icpr'02) Volume 1 - Volume 1* (August 11 - 15, 2002). ICPR. IEEE Computer Society, Washington, DC, 311-314.

[15] Meng, Q., Li, B., and Holstein, H. 2006. Recognition of human periodic movements from unstructured information using a motion-based frequency domain approach. Image and Vision Computing, 24 (8) (2006), 795-809.

[16] Laptev, I., Belongie, S. J., Perez, P., and Wills, J. 2005. Periodic Motion Detection and Segmentation via Approximate Sequence Alignment. In *Proceedings of the Tenth IEEE international Conference on Computer Vision (Iccv'05) Volume 1 - Volume 01* (Oct. 17 - 20, 2005). ICCV. IEEE Computer Society, Washington, DC, 816-823.

[17] Ormoneit, D., Black, M.J., Hastie, T., and Kjellström, H. 2005. Representing cyclic human motion using functional analysis, Image and Vision Computing 23 (14) (2005), 1264-1276.

[18] Howard Leung, Jacky Chan, Kai-Tai Tang and Taku Komura. 2007. Ubiquitous Performance Training Tool Using Motion Capture Technology. In *Proc. 1st International Conference on Ubiquitous Information Management and Communication* (Suwon, Korea, Feb. 8-9, 2007), 185-194