Motion Editing Based on Joint Classification and Discrete Blending

Hyoungseok B. Kim† and Hosook Kim††,

Department of Multimedia Engineering, Dong-Eui University, Busan, Rep. of Korea
Division of Advanced Computer Information, Dong-Eui Institute of Technology, Busan, Rep. of Korea

Summary
In this paper, we present a new method of motion editing based on joint classification and discrete blending in order to efficiently reuse captured motion data. First of all, we define a joint energy of a character in order to measure the similarity of motions, and then apply the clustering technology to the set of joint energy. Hence, we may have two groups of joints: major joint group and minor joint group. If the major groups of two given motions have the same cardinality and the error of their motion energy is within the given threshold, then we classify the motions as similar motions. Otherwise, we consider the given two motions different. Our method may enrich the database of motion capture data, so plays an important role in digital contents such as computer game and computer animation.
Key words:
Motion Editing, Joint-Classification, Motion Synthesis, Blending.

1. Introduction

The realistic motion control of characters plays the important role in 3D computer game and computer animation. In general, there are two methods to control realistic motion of characters: key-frame technology and motion capture technology. The key-frame technology was used as a standard method of cartoon animation because it is very simple and cheap. However, the quality of key-frame animation is poor, so it is not adequate to high-quality animation. Nowadays, most of animations request realistic motion control. The motion capture technology acquires the motion of real actors as it is, so it may satisfy such requirements. But, it needs additional effort to make the captured motion data suitable for virtual spaces such as an animation or a computer game. Moreover, the cost of motion capture systems is very expensive, so user may be hard to obtain a variety of motion data. So users have to modify the captured motion data and obtain wanted motion in order to solve the high-cost problem. There are several researches related to motion synthesis and motion editing. Most of the previous researches focus attention on rearrangement of frames or inverse kinematics. The methods are suitable for the case which makes a character move along a given path or creates a simple motion satisfied with constraints. However, they can not create a motion with postures which are not in real world.

In this paper, we present a new method of motion editing based on joint classification and discrete blending in order to raise the ratio of reuse of given motion data. First of all, we define a joint energy of a character in order to measure the similarity of motions, and then apply the clustering technology to the set of joint energy. Hence, we may have two groups of joints: major joint group and minor joint group. If the major groups of two given motions have the same cardinality and the error of their motion energy is below over the given threshold, then we classify the motions as similar motions. Otherwise, we consider the given two motions different. Therefore, if a user creates a similar motion for a given motion, then we can recommend that he/she has only to modify the joints of the minor group below over the limit of the motion energy.

The remainder of the paper is as follows. In Section 2 we review the existing methods for motion synthesis and motion editing. In Section 3 we define the joint energy and then classify the joints by applying a clustering technology to the joint energy. We explain the motion classification method based on the joint energy in Section 4. An efficient motion editing algorithm based on the discrete blending is shown in Section 5. Section 6 provides a summary and discusses some future work.

2. Related Work

The several researchers expend a great effort to create the realistic motion of 3D characters for a decade. There are rigid methods to control the motion data [1, 2], motion synthesis algorithms to generate a wanted motion by utilizing the given data [3, 4], and physically based motion animation [5]. Moreover, the measurements of the reality of motion and physical correctness were provided in [6, 7]

† Manuscript received February 5, 2007
Manuscript revised February 25, 2007

or the analysis on motion data were presented to construct a larger motion database [8].

The reuse of captured motion data is especially needed in the field of computer game and crowd animation and the skeletal simplification plays an important role in such applications. The previous researches of skeletal simplification were focus on the improvement of performance of physically based simulation. There was an instance to manually simplify the skeletal data for more fast motion blending in physically based motion creation system [9]. Most recently, a variety of researches on skeletal simplification were applied to fast animate the captured motion data. Especially, JPC (joint posture clustering) method in [10] extracts the movement of similar joints for speed improvement of captured motion. The authors analyzed the transformation of joints for each frame into the reason of a drop in performance. In order to solve this problem, they clustered the joints into several groups based on the similarity of joints and minimized the transformation of joints. The goal of this method is to improve the speed of animation, it is not the reuse of a given motion data. Moreover, the method excludes the physical properties in joint-clustering process because it dose not consider the limit of joint's movement.

3. Joint Classification

In this section, we define the joint energy according to a given motion, and then classify the joints to two groups: major joint group and minor joint group. The joints belonging to major group are representatives of the motion and so their change has an important effect on the motion. On the other hand, the change of a joint in the minor group affects little the motion. For example, we know that a "shooting" motion and a "dribbling" motion are different. The reason is that the major groups of two motions have different elements from each other and the joint energies are beyond the limit of the motion energy.

3.1 Joint Energy

First of all, we must have a good grasp of the data structure which represents well the structure of the human body in order to classify the joints of given motion data. There are several data structures such as BVH, HTR, and ASF. In general, BVH is used as a representative data structure. BVH has 18 joints and its root is joint "hip" as shown in Figure 1.

Let J and H be the number of joints in the BVH skeletal structure and the height of the BVH tree, respectively. In general, the height of a node in a tree is the length of a longest path from the node to a leaf, and the height of a tree is the height of the root. The depth of a node is the length of the unique path from the root to the node.

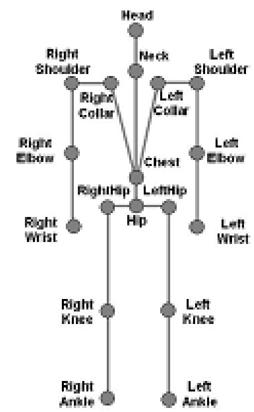


Fig. 1 BVH hierarchical structure

Now we are ready to define the energy of a joint for a given motion M_i with N frames. Each joint j has the three Euler angles $A_j(t) = (x_j(t), y_j(t), z_j(t))$ which represent the relative orientations of the joint to its parent joint. Now, we define a joint energy $E(j)$ of a joint j as follows;

$$E(j : M_i) = w(j)\sum_{k=1}^{3}\left|\sum_{0}^{N-1}\frac{A_j^k(t+1) - A_j^k(t)}{MaxA_j^k}\right| \quad (1)$$

, where $A_j^1(t), A_j^2(t), A_j^3(t)$ are the three Euler angles of joint j at frame t, $MaxA_j^k$ is the maximum Euler angle of the joint j which can be rotated about the k^{th} axis, and $w(j)$ is the weight of the joint j. Here the weight is defined as $w(j) = H - d(j)$, where $d(j)$ is the depth of the joint j.

3.2Joint Grouping

In this subsection, we classify the joints into two groups: the major joint group and the minor joint group. First of all, we compute all of the joint energy for a given motion data , and then sort the values of the joint energy as shown in Figure 2. The filled circles present the values of the joint energy. We apply a general clustering algorithm to the set of values of joint energy, so that we may have several clusters C_{i1}, C_{i2}, $\cdots, C_{iK(i)}$, where $K(i)$ is the number of clusters for a given motion data M_i. The joints of the first cluster C_{i1} have a larger energy than them of the other clusters. We call the first cluster as the major group and the union of the other clusters as the minor group. We define the energy of cluster for C_{ij} as

$$EC_{ij} = \frac{\sum_{j \in C_{ij}} E(j : M_i)}{\left| C_{ij} \right|},$$

where $\left| C_{ij} \right|$ is the cardinality of the cluster C_{ij}.

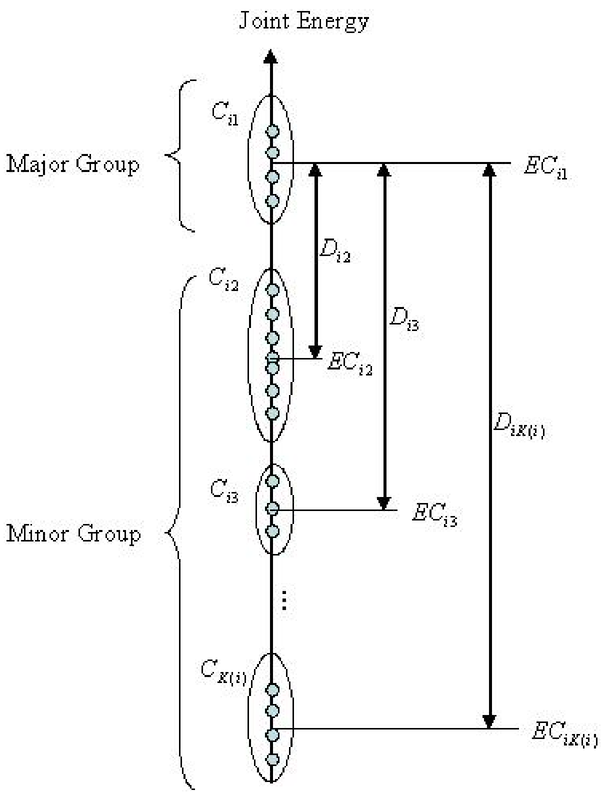


Fig. 2 Joint Energy Classification

Now, we are ready to recommend the joints to be edited by the joint energy classification. If user wants to get a similar motion to a given motion datum, then we recommend the user to select a joint k from the minor group and may modify the joint within the limit D_{ij} of its cluster energy, where the joint belongs to the cluster C_{ij}. If user wants to get a different motion to a given motion datum, then we may recommend the user to select a joint k from the major group.

4. Motion Similarity

We exploit the motion classification based on the joint classification. Assume that there are two motion data M_1 and M_2. First of all, we apply the joint classification to the motion data. If their major groups have the same joints and the difference between their cluster energies is within the minimum of the standard deviations of C_{11} and C_{21}, then the two motions are regarded to be similar to each other.

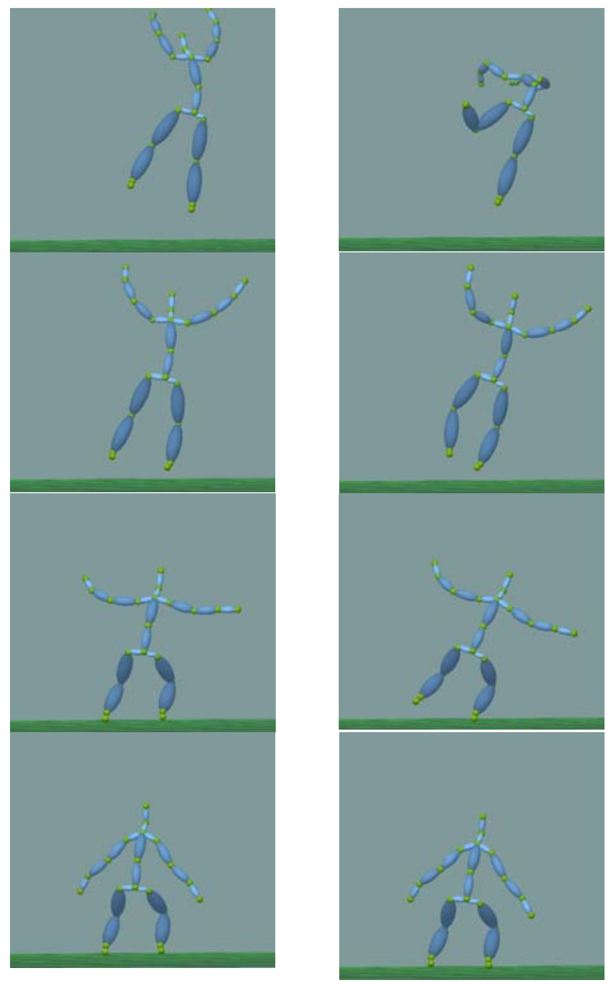
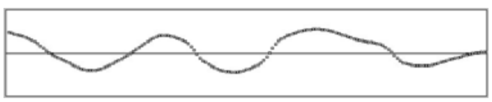


Fig. 3 TALCHUM : Two sequences of Korean dancing

Figure 3 shows two scenes of “TALCHUM”, which is a traditional Korean dancing. Two motions have the same major joint groups, but the major group energy of the left TALCHUM are beyond the standard deviation of the major group energy of the right motion. So we may classify the motions to be different.

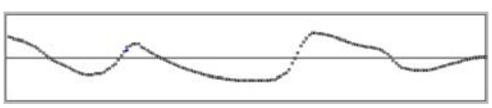
5. Discrete Blending

Human's posture may be represented by the translations and rotations of joints which well resemble the properties of human structure. Motion is a sequence of postures of human, so the motion capture data may be represented by the functions of a time for such transformations of joints. Figure 4(a) shows a part of motion data, where the parameters of the horizontal axis and the perpendicular axis are time and the angle of a joint, respectively. In general, the motion data may be edited by copy-and-paste operator to enrich a database of the motion captured data. This operator surely gives rise to a serious trouble such as discontinuity in motion editing as shown in Figure 4(b). The discontinuity of motion data produces the non-realistic animation of a virtual character. So this problem has to be resolved in the field of motion synthesis and motion editing.



(a) Original motion data

(b) Modified motion data



(c) Blended motion data

Fig. 4 Motion Editing Process

In general, the solution of discontinuity is to apply blending technologies to given two curves of continuous type. However, the type of motion data is discrete and so the general blending functions are not adequate to this motion data. We will present a discrete blending method which can be applied to motion data editing. Figure 4(c) shows the result of our discrete blending method. The graph of motion is represented by a dotted curve. That means data of a discrete type. Figure 5 is a window of our motion editing tool.

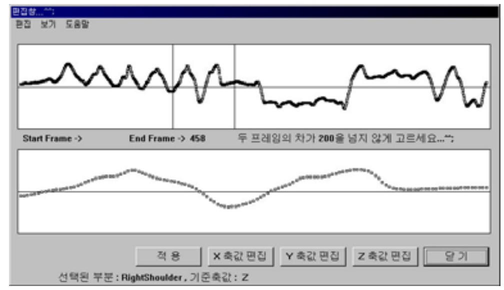


Fig. 5 Motion Editing Tool

Now, we are ready to explain our discrete blending method. Let C_1 and C_2 be two curves which are discontinuous at a time instance t_0 as shown in Figure 6. Let t_1 and t_2 be the time instances where the blending function starts and ends, respectively. That is, t_1 is less than t_0 and t_2 is larger than t_0. The curve C_1 has a point C_{P1} at the time instance t_1, $C_1(t_1) = C_{P1}$, and the curve C_2 has a point C_{P2} at the time instance t_2, $C_2(t_2) = C_{P2}$. The discontinuity appears at time t_0 as follows;

$$C_1(t_0) = P_1,\ C_2(t_0) = P_2.$$

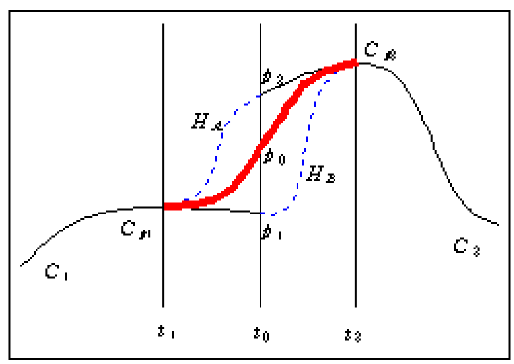


Fig. 6 Hermit Interpolation

We generate a Hermit curve H_A which connects two points C_{P1} and P_2. In general a Hermit curve is defined by two end points and their tangent vectors. Here, we already got two end points, but their tangent vectors are not defined yet. Hence, we have to find out a discrete tangent vectors, which are the key points of our method. Assume that there are N_A time samples between t_1 and t_0. We define the tangent vector of time t_1 and time t_0 as

$1/N_A(P_1 - C_{P1})$ and $1/N_A(C_{P2} - P_2)$, respectively. Then we have a Hermit curve samples between t_1 and t_0 as follows;

$$H_A(s) = C_{P1}H_0^3(s) + \frac{1}{N_A}(P_1 - C_{P1})H_1^3(s) + \frac{1}{N_A}(C_{P2} - P_2)H_2^3(s) + P_2H_3^3(s),$$

We can generate a Hermit curve H_B which connects two points P_1 and C_{P2} by the above process. Assume that there are N_B time samples between t_0 and t_2. We define the tangent vector of time t_0 and time t_2 as $1/N_B(P_1 - C_{P1})$ and $1/N_B(C_{P2} - P_2)$, respectively. Then we have a Hermit curve samples between t_0 and t_2 as follows;

$$H_B(s) = P_1H_0^3(s) + \frac{1}{N_B}(P_1 - C_{P1})H_1^3(s) + \frac{1}{N_B}(C_{P2} - P_2)H_2^3(s) + C_{P2}H_3^3(s),$$

There are two curves between t_1 and t_0. The one is the original curve C_1, the other is a Hermit curve H_A. Now, we blend the two curves between t_1 and t_0 as follows;

$$B(t) = H_A(\alpha_1(t))\omega(t) + (1 - \omega(t))C_1(\beta_1(t)), \quad 0 \le t \le 1/2,$$

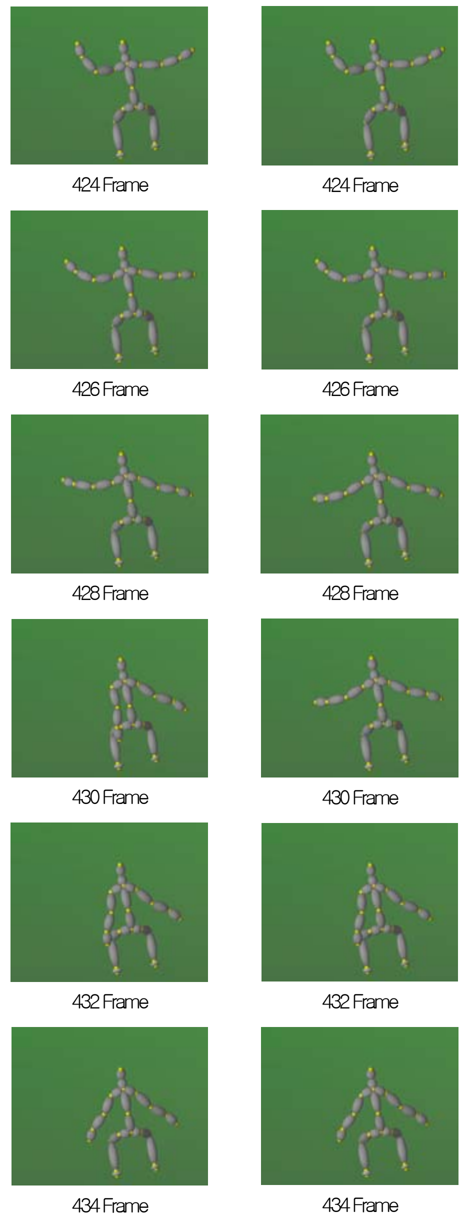
By the similar method, we may another blending function of the original C_2 curve and a Hermit curve H_B as follows;

$$B(t) = H_B(\alpha_2(t))(1 - \omega(t) + \omega(t)C_2(\beta_2(t)) \quad 1/2 \le t \le 1.$$

Here,

$$\omega(t) = t,$$
$$\alpha_1(t) = 2t,$$
$$\alpha_2(t) = 2t - 1,$$
$$\beta_1(t) = 2(t_0 - t_1)t + t_1,$$
$$\beta_2(t) = 2(t_2 - t_0)(t - 1) + t_2.$$

This blending method has several advantages. Even though the numbers of time samples for two time intervals are different, we can generate smooth animation by the blending function which reflects the property of a discrete type. This method can give a rise to a variety of blending functions according to a weight function $\omega(t)$. Figure 7 shows the result of our discrete blending method.



436 Frame 436 Frame

Fig. 7 Motion editing results: (Left) original motion data, (Right) edited motion data based on our method.

6. Conclusion

The contribution of this paper is to present an efficient method to reuse motion data. Our method of motion editing is based on joint classification and discrete blending. First of all, we defined a joint energy of a character so that the similarity of motions can be measured, and then applied the general clustering technology to the set of joint energy. Exploiting this joint energy and cluster algorithms, we could classify the joints to major group and minor group. The total energy of joints in the major group may play a role as a measure of motion classification. Moreover, we present a discrete blending method which is useful to motion editing since our blending function well reflects the shape of the original curves. In future, we will develop a motion editing algorithm which is applicable to motion data captured from only two cameras.

Acknowledgments

This work was supported by Dong-Eui University Grant (2003AA193).

References

[1] A. Bruderlin and L. Williams, Motion signal processing, ACM SIGGRAPH '95, pp.97-104, Los Angeles, 1995.

[2] J. Lee and S.Y. Shin, A hierarchical approach to interactive motion editing for human-like Figures, Proc. Of the 26th Annual Conference on Computer Graphics and Interactive Techniques, ACM Press / Addison Wesley Publishing Co., New York, NY, pp 39-48, 1999.

[3] L. Kovar, M. Gleicher, and F. Pighin, Motion Graphs, Proc. of the 29th Annual Conference on Computer Graphics and Interactive Techniques(San Antonio, Texas, July 23-26, 2002), SIGGRAPH '02, pp 473-482, 2002.

[4] O. Arikan, D. A. Forsyth, and J. F. O'Brien, Motion Synthesis from annotations, ACM Trans. Graph., 22, 3 (Jul. 2003), pp 402-408, 2003

[5] A. Witkin and M. Ksss, Spacetime constraints, Proc. of the 15th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '88, pp 159-168, 1988

[6] L. Ren, A. Patrick, A. A. Efros, J. K. Hodgins, and J. M. Rehg, A data-driven approach to quantifying natural human motion, ACM Trans. Graph., 24, 3(Jul. 2005), 1090-1097, 2005.

[7] A. Safonova and J. K. Hodgins, Analyzing the physical correctness of interpolated human motion, Proceedings of '05 ACM SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation, pp 171-180, 2005.

[8] T. Kwon and S.Y. Shin, Motion modeling for on-line locomotion synthesis, Proceedings of '05 ACM SIGGRAPH/EUROGRAPHICS Symposium on Computer Animation, pp 29-38, 2005.

[9] Z. Popovic and A. Witkin, Physically based motion transformation, ACM SIGGRAPH '99, pp 11-20, 1999.

[10] J. Ahn and K. Wohn, Motion level of detail: a simplification method on crowd scene, CASA 2004.

Hyoungseok B. Kim received the M.S. and Ph.D. degrees in department of Mathematics from Korea Advanced Institute of Science and Technology in 1992 and 1998, respectively. During 1998-1999, he was a researcher in Electronics and Telecommunications Research Institute (ETRI). He is an associate professor in department of Multimedia Engineering at Dong-Eui University from 1999. His interesting research fields are computer graphics, computer game engine, and applied mathematics.

Hosook Kim received the M.S. and Ph.D. degrees in department of Computer Engineering from Ewha Womans University in 1999 and 2005, respectively. She is an assistant professor in division of Advance Computer Information at Dong-Eui Institute of Technology from 2001. Her interesting research fields are database, data mining, GIS, and mobile technology.