A Framework of Mocap Data Editing with Labanotation

Xiaojie Shen, Weidong Geng, Tao Yu
State Key Lab. of CAD&CG
Zhejiang University
Hangzhou, 310027, PR China
gengwd@cs.zju.edu.cn

Abstract

The artists and dancers prefer to edit and plan motions via notations, as it can provide an intuitive meaningful visualization of motion sequence. It will make the motion editing tasks more intuitive if the motion manipulation can be implemented via the preview interface of notations. In order to explore this approach, we implemented a framework that can edit the motion captured data with Labanotation, a well-developed notation language that records human movements. Its motion editing pipeline consists of three major steps. It firstly converts the motion captured data into Labanotation representation, and then the user makes changes directly on the Labanotation sequence. Lastly, these modified Labanotation will be transformed back into the resulting motion sequence.

Key words: Labanotation, LND (Labanotation data), motion capture, motion edit, motion plan

1. Introduction

Motion capture is one of the most promising technologies bringing realistic and natural motions into character animation. The use of motion capture is currently most widespread and well accepted in video game and animation [1] [2]. However, motion capture also has its share of weakness [3] [4] [5]. Motion capture systems are still too expensive to use. The motion capture process is labor intensive and time-consuming for actors, directors and technicians. In order to make motion capture widely available, the motion capture data needs to be made reusable [6]. This means that we may create the needed motions by reusing pre-recorded motion capture data.

Unfortunately, motion capture data has proved to be difficult to modify due to its specificity [7] [8]. A captured motion specifically records the performance. It encodes a specific performer, performing a specific action, in a specific way. Moreover, the “essence” of the motion is not distinguished from the large number of potentially irrelevant details. This makes its editing and modification a challenging problem.

Up to now, much of the research has been dedicated to developing various tools to generate a convincing motion sequence from prerecorded motion clips, including editing motions by signal processing, constrained optimization, Statistical models, and Interpolation[1]. However, all these methods make changes on motion captured data merely by mathematical parameters and numerical values. The user can not have the intuitive understanding of the resultant motion sequence. It means that even if the user already knows how to correctly setup the editing parameters, the resulting motion sequence may not be the motion desired in the user’s mind.

Therefore, some researchers in computer graphics community brought up the idea of manipulating motion sequence via notations such as Benesh movement notation [9]. But the notations need to be marked manually based on original motion data. In this paper, we developed a framework that can automate the process of motion editing with notations. We choose the Labanotation as the understandable preview interface to edit the motion sequence.

Labanotation is a notation language that describes the motion of human body graphically in the form of vertical notes, where each column represents motion of a part of a body, similar to music score that represents the music graphically. We read the notations from bottom to top, and horizontal bars are placed regularly which represent ticks of time just like vertical bars in the musical notes. A column of the notation consists of three vertical lines. The centerline represents the center of the body, and the right side and the left side from the centerline are used for describing the motion of the right and left side of the body, respectively. Labanotation is represented by graphical symbols, thus easy for us to understand the denoted motions. Moreover, directors who are specialized in dance and performing arts are very familiar with Labanotation, since it was published as a universal Language for recording and analyzing human movements, and has already been widely used and well developed. With this notation language, choreographers obtain a powerful tool for documenting and representing the dance in paper form, which had been an arduous task before. Symbols in Labanotation can contain three basic kinds of information about certain parts of body. Fig. 1 shows an instance of the structure of symbols.

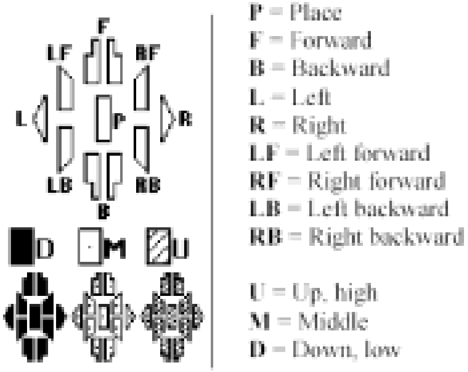


Fig. 1 Structure of symbols in Labanotation

Horizontal direction – figured by its shape;
Vertical height (level) – figured by its inside shading;
Duration – figured by its length;
And columns between staffs figure out which part of body is being described. Fig. 2 gives a simple example of the standard structure of Labanotation.

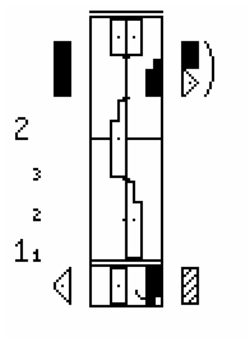


Fig. 2 Example of Labanotation

These are basic symbols and structure used frequently in Labanotation, and there are some other signs related to some particular motion. But example shown in figure 2 can represent human motion independently, since Labanotation fulfils the demand that a language for describing human motion has to be as easy as possible and as complex as necessary. More information about Labanotation please refers to [10] [11].
In the following section, we will present an overview of motion capture data editing with Labanotation. The detailed process of conversion between motion captured data and Labanotation will be described in section 3 and 4. In section 5, some experiment results and a brief discussion are given in section 6.

2. Overview of Motion Editing via Labanotation

In our framework, the motion editing pipeline consists of three major steps (Fig. 3):

1) It imports and converts the motion captured data into Labanotation representation.
2) The user makes changes on the Labanotation sequence.
3) The modified Labanotation is transformed back into the motion sequence.

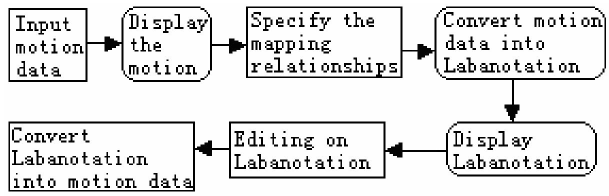


Fig. 3 Flow chart of the editing pipeline

It is necessary to let the user to specify the mapping relationships between joints in motion capture data and those in Labanotation, because joints in motion captured data are randomly named without any naming regulations. For example, root joint and stomach joint in two different motion sequences may denote the same joint. Moreover, in the same motion sequence, there would occasionally be several nodes with the same name such as end-site. Furthermore, joints in motion data are usually more than those in Labanotation. Therefore the mapping relationships can only be specified manually by users.
When the conversion process starts, the motion capture data in BVH format is at first transformed into positional data of joints, instead of rotational angles. This is the forward translation in our framework. Labanotation data (LND) is the internal representation of Labanotation. All the editing operations are performed on LND, which will be translated into the resultant motion via the backward translation process later on. The conversion process of forward and backward translation is shown in figure 4.

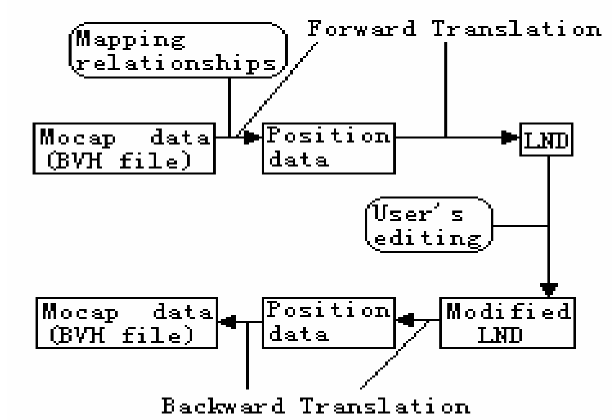


Fig. 4 The conversion process of motion capture data and Labanotation

The major issue in this translation (either forward translation or backward translation) process is the discrepancy of redundancy in motion capture data and Labanotation. Motion capture data is famous for its high accuracy of the data, thus having much more redundancy than Labanotation. Generating Labanotation from motion data has already been realized by other researchers such as LabanEditor [12] [13], which can not only edit, display and print Labanotation scores, but also output a VRML-based file corresponding to the LND. On the other hand, transforming Labanotation into motion sequence (backward translation) is not easy. We have to additionally extrapolate the movements needed by motion capture data from Labanotation.

3. Motion Capture Data to Labanotation

In this forward translation, the system will convert motion capture data firstly into positional data, and then into LND. Since motion capturing data (BVH file) and positional data (position-data file) are equivalent in essence, we pay attention mainly on the process from positional data to Labanotation. The overall process of generating LND from positional data is shown in fig 5.

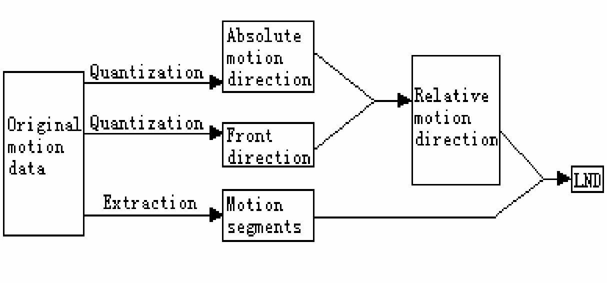


Fig. 5 Process of generating LND from motion data

Step1: Quantization of absolute motion direction.
Step2: Extraction of motion segments
Step3: Quantization of front direction
Step4: Quantization of relative motion direction
Step5: Generation of LND

Quantization of global motion direction. In Labanotation, the horizontal direction of movement is represented by 9 orientations (including place) and vertical directions by 3 orientations. So we divide the space into 27 3-dimensional sectors, and decide which sector the child joint exists by placing the parent joint at the origin of the space. Thus the absolute direction of the child joint (without consideration of front direction) is represented by 27 quantized orientations. This process is called a quantization of absolute motion direction.

Extraction of motion segments. The motion sequence of each joint should been divided into several segments, but we can't simply divide them evenly. Because some movements are slow while the others are fast. In Labanotation, they are distinguished by the length of the relevant notes that determine how many frames in original position data mapping to this symbols.

To divide the motion sequence properly, Nakamura and Hachimura [12] proposed a method named extracting period – segmenting motion data into several sections by eliminating resting periods from motion data in which human movement is regarded as stationary. However this algorithm fails at times, since the value of the threshold which decides if human body moves is selected only by user's experience.

We bring up another method as substitution – the detection of "critical orientation". Critical orientation is the edge between two neighboring sectors of direction, figure 6 gives an example of critical orientation. By checking absolute motion directions of each pair of neighboring frames about a joint, we can select those pairs with different values of absolute motion direction, in which the later frame passes through the critical orientation. The beat where the later frame locates must be the "terminating beat" of a symbol. By marking all of these "terminating beats", the motion could be divided into several segments properly. This method also benefits in another aspect – describing the motion in more detail, compared with algorithm of extracting period proposed in [12]. Besides, the redundancy in Labanotation generated by our algorithm is also obviously much more than the former one, which favors our backward translation.

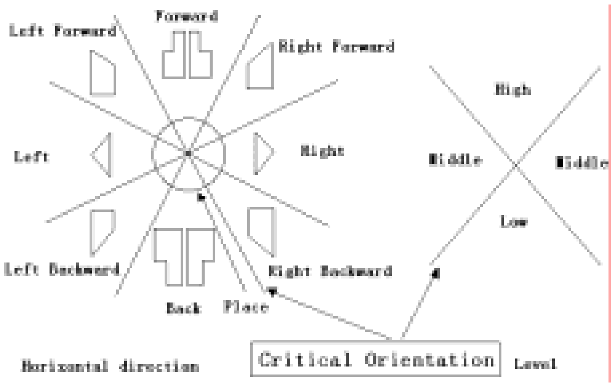


Fig. 6 Critical Orientation

Quantization of front direction. In Labanotation, every horizontal direction of symbols is referred to actor's front direction. But in motion capturing process, actors often turn around, resulting in his/her frequent variation in front direction. So quantization of front direction of the performer is indispensable in the forward translation. However, in LabanEditor [13], this point has never been taken into account. We select 3 points – root, left hip and right hip – from the whole body to form a plane, and the horizontal projection of the normal direction of this plane can be regarded as the front direction. The system will automatically calculate front direction of every frame based on this definition.

Quantization of relative motion direction. System will automatically turn absolute motion direction into relative motion direction according to front direction calculated above.

Generation of LND. With some parameters inputted from users (such as how many frames per beat and how many beats per bar), we can ultimately transform frames of motions into symbols, and then assign them with attributes such as relative motion direction and the length of the symbols.

Instead of using normal structure of Labanotation, we utilize a expanded version of its structure[14] – which describes not only arms and legs, but also includes more joints such as elbow, wrist, knee and so on. Examples are showed in figure 7.

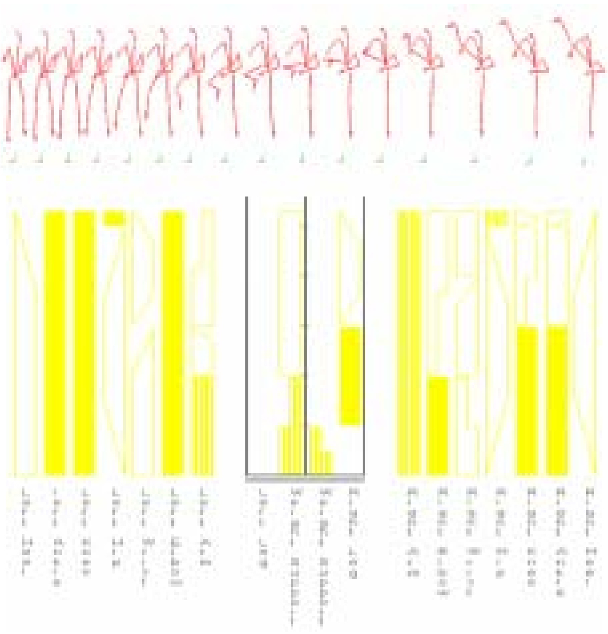


Fig. 7 Original motion data and its corresponding Labanotation generated by our system

This improvement will contribute in two ways:
1 The motion can be described more precisely;
2 The redundancy in Labanotation is enlarged which favors the backward translation.

4. Labanotation to Motion Sequence

In backward translation, our goal is to generate new motion data from modified Labanotation. However, as mentioned before, Labanotation focuses on the abstract qualitative description of movement which is not sufficient to directly convert it into concrete motion data. We take the approach that associates the original motion capture data with LND, and manages to map the modification of LND into the spatial and temporal constraints. Then we employ the constraint-based algorithms to synthesize novel movements and accordingly make changes on the original motion data. This resulting motion sequence is taken as the desired motion converted from the modified Labanotation.

During the conversion process, a copy of original data from position-data file is loaded into memory first, and every symbol in Labanotation is mapped to their corresponding frames in motion sequence, since each note symbol is generated directly from relevant frames in the forward translation. This mapping put strong restrictions in the backward translation especially for those unchanged Labanotation. In most cases, modification operation is performed only on a small part of the entire Labanotation, leaving most of it unchanged. Generation of motion clips from these unchanged parts merely needs copying original data from position-data file into the resulting motion sequence.

However, for those modified symbols in Labanotation, we have no shortcut for it. A Labanotation symbol denotes a specific pose at the instant, and the whole Labanotation describes a pose of the body at each instant just like key frame animation. Thus, Labanotation data is very similar to key framing. Therefore in our system, a conversion template of these basic rules is used to realize the conversion from those modified symbols. The motion conversion template describes the mapping relationship between direction of a particular joint's symbol and the relative position coordinate of this joint with reference to its parent joint. These rules are based on the assumption that distance between every pair of a child joint and its parent joint is constant. By multiplying these vectors with distance from certain joint to its parent joint which can be obtained from original positional data, the relative position coordinate of this joint would be generated. The format and example of this template file is shown in figure 8. We can easily modify the rules of this template if needed.

Format	Example1	Example 2
Direction,Level	Left, Middle	Right forward, high
Joint name	Left arm	Right leg
Unitizational relative position coordinate	[1,0,0]	[-0.447,0.774,0.447]

Fig. 8 Format and examples of motion conversion template

Nevertheless, there would be several gaps between those newly generated frames because of only one frame is obtained from one notation symbol which consists of data of several original frames. In order to make sure that not only the number of frames equals to that in original one, but also the quality of the new motion would be as smooth as possible, we should interpolate some frames as transitional motions for these gaps, taking frames generated from both modified Labanotation and original position-data file as key frames. The whole process of backward translation is shown in figure 9.

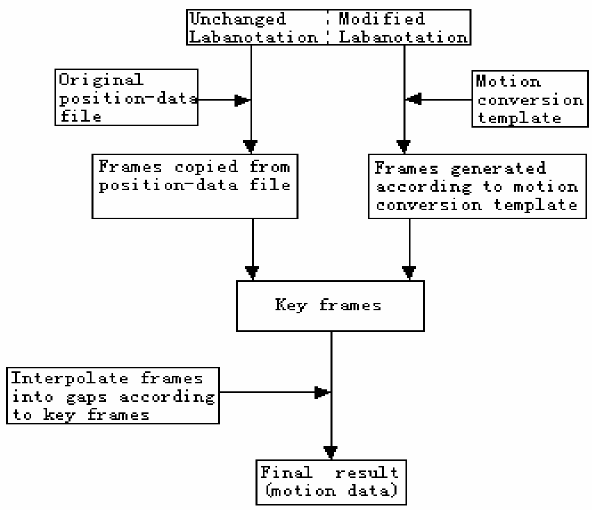


Fig. 9 Backward Translation (translate Labanotation into motion sequence after editing)

5. Results and Discussion

Our framework is implemented in Windows platform using Visual C++ 6.0, and both the source and the resultant motions are represented in BVH file format, while the modifications are based on Labanotation. So users who are only familiar with Labanotation can easily manipulate the motion editing with intuitive preview.

In figure 10, original motion sequence of a movement named "GoalKicking" is shown at the top. Labanotation before and after editing are shown in the middle part, separately. Finally, the resultant motion sequence generated from modified Labanotation is given at the bottom of this figure.

The future work will focus on providing automated methods for extracting motions with the closest similarity to that specified by users from large motion capture data sets. With the high volume of data sets the motions we easily can create the high quality motion sequence by authoring Labanotation.

Reference

[1]. Weidong Geng and Gino Yu: "Reuse of Motion Capture Data in Animation: A Review," ICCSA 2003, LNCS 2669, pp. 620-629, 2003.

[2]. Menache, A.: "Understanding motion capture for computer animation and video games," Morgan Kaufman (Academic Press), San Diego, USA (2000)

[3]. Gleicher, M. and Ferrier, N.: "Evaluating Video-Based Motion Capture," Proceedings of Computer Animation 2002, 75-80

[4]. Kovar, L., Gleicher, M., Pighin, F.: "Motion graphs, SIGGRAPH," 2002, 473 – 482

[5]. Pullen, K. and Bregler, C.: "Motion Capture Assisted Animation: Texturing and Synthesis," SIGGRAPH 2002, 501-508

[6]. Arikan, O. and Forsyth, D.A.: "Interactive Motion Generation from examples," SIGGRAPH 2002, 483-490

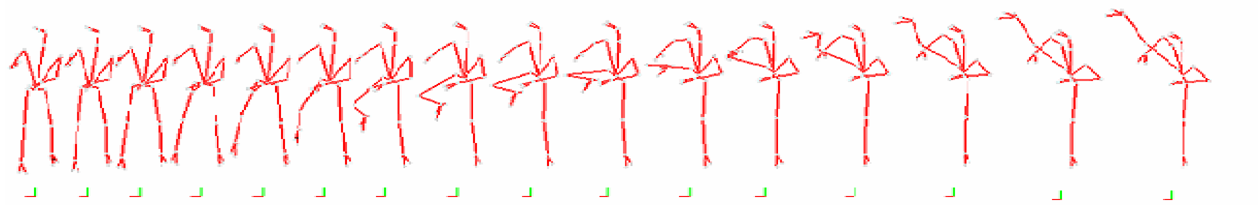
[7]. Weidong Geng,Yan Huang, Yunhe Pan.: "Step/stance planning and hit-point repositioning in martial arts choreography," Proceedings of 17th International Conference on Computer Animation & Social Agents, 2004,95-102

[8]. L. Kovar, M. Gleicher, and F. Pighin.: "Motion graphs," In Proceeding of ACM SIGGRAPH 2002, pages 473–482. ACM, 2002.

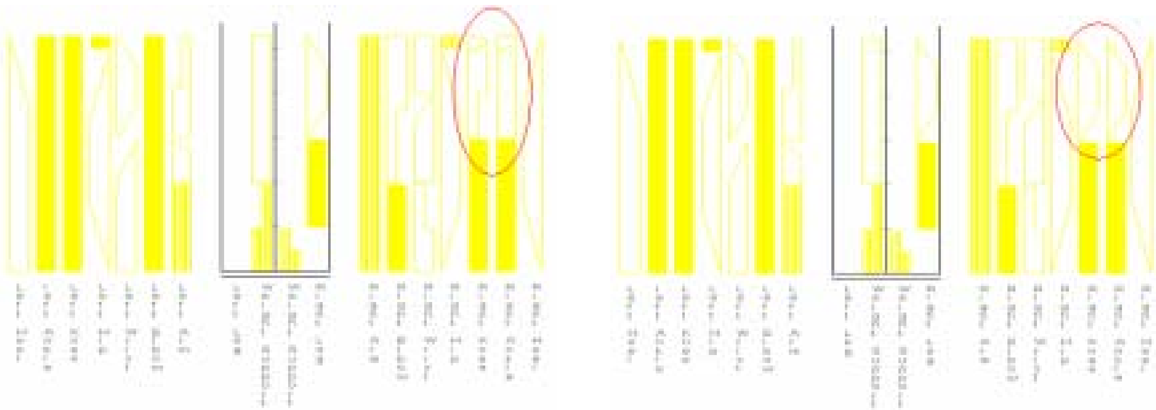
[9]. B. Singh, JC Beatty, KS Booth, and R. Ryman.: "A graphics editor for benesh movement notation," Computer Graphics, 17(3):51--62, July 1983.

[10] Hutchinson Guest, Ann Hutchinson, Doug Anderson: "Labanotation: The System of Analyzing and Recording Movement"

[11] Christian Griesbeck: "Introduction to Labanotation," http://www.uni-frankfurt.de/~griesbec/LABANE.HT

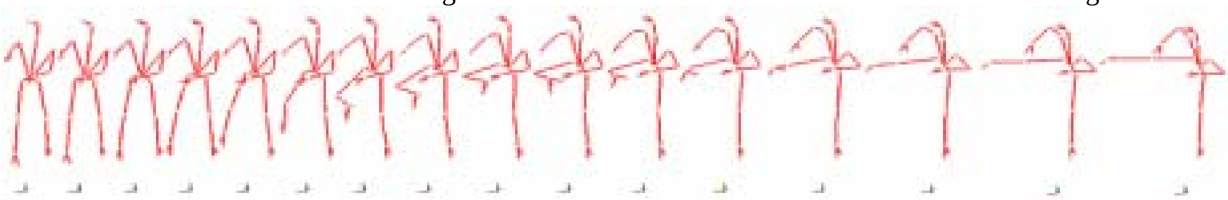


Initial motion sequence stored in position-data file



Labanotation before editing

Labanotation after editing



New motion sequence generated from modified Labanotation

Fig. 10 some experiment results of

ML
[12] Minako Nakamura and Kozaburo Hachimura, "Development of Multimedia Teaching Material for Labanotation", ICKL(International Council of Kinetography Laban)Proceedings of the Twenty-second Biennial Conference, 2002, pp150-160
[13] Kojima, K.; Hachimura, K.; Nakamura, M., "LabanEditor: Graphical editor for dance notation", In Proceeding of 11th IEEE International Workshop on Robot and Human Interactive Communication, Pages: 59- 64
[14] Calvert, T. W., Chapman, J., and Patla, A.1980. "The integration of subjective and objective data in the animation of human movement." In Computer Graphics (SIGGRAPH '80 Proceedings), volume 14, pages 198-203