

# Synthesizing Dance Performance Using Musical and Motion Features

Takaaki Shiratori  
Institute of Industrial Science,  
The University of Tokyo  
Tokyo, Japan  
siratori@cvl.iis.u-tokyo.ac.jp

Atsushi Nakazawa  
Cybermedia Center,  
Osaka University  
Osaka, Japan  
nakazawa@cmc.osaka-u.ac.jp

Katsushi Ikeuchi  
Institute of Industrial Science,  
The University of Tokyo  
Tokyo, Japan  
ki@cvl.iis.u-tokyo.ac.jp

**Abstract**—This paper proposes a method for synthesizing dance performance synchronized to played music and our method presents a system that imitates dancers’ skills in performing their motion while they listen to the music. Our method consists of a motion analysis, a music analysis, and a motion synthesis based on results of the analyses. In these analysis steps, motion and music features are acquired. These features are derived from motion keyframes, motion intensity, music intensity, musical beats, and chord changes. Our system also constructs a motion graph to search similar poses from given dance sequences and to connect them as possible transitions. In the synthesis step, the trajectory that provides the best correlation between music and motion features is selected from the motion graph, and the resulting motion is generated. Our experimental results indicate that our proposed method actually creates dance as the system “hears” the music.

## I. INTRODUCTION

Creating a realistic motion sequence is currently a major goal of the computer graphics research and the robotics research. One method often employed is dynamics simulation. First, a programmer manually creates an appropriate motion sequence, and then, the sequence is optimized to maintain balance over the whole motion sequence by using a dynamics simulator. This method is a useful technique; however, it is time-consuming. Another popular method is based on motion capture system. This technique consists of “Motion Graph” [1], [2], [3], or applying statistics to a set of motion capture data to synthesize new motion sequences [4]. However, most of these techniques do not consider human emotion and recognition despite the fact that humans often exhibit behaviors based on emotion and recognition.

There are a few previous methods to consider human emotion when creating motion sequences. Kim et al. [5] proposed a motion enhancement method that could analyze the rhythm of motion and synthesize new motions based on the results of motion rhythm analysis, and by using this method, we could synthesize rhythmic motions. However, for example, for creating new dance performance, the music signal needed to be synchronized to the rhythm of motion. Stone et al. [6] proposed a method that could synthesize new utterance performance synchronized to an input speech signal. However, this method needed considerable manual processing, was not automatically, and required a great deal of time.

The ability to dance to music is a natural born skill for a human. Most people at times experience a desire to move their bodies while listening to a “rhythmic” song. Hip-hop dancers can simultaneously compose a dance to the musical sounds as they hear these sounds. Although this ability may appear amazing, actually these performers do not create these motions, but rather combine appropriate motion segments from their knowledge database with music as their key to perform their unique movements. Considering this ability, we are led to believe that dance motion has strong connections with music in the following two points:

- The rhythm of dance motions is synchronized to that of music.
- The intensity of dance motions is synchronized to that of music.

The first assumption is derived from the fact that almost all people can recognize the rhythm of music and they can clap or wave their hands and dance to music. The second assumption is derived from the fact that people feel quiet and relaxed when listening to relaxing music such as a ballad, and they feel excited when listening to intense music such as hard rock music.

Toward an avatar and robot performance with emotion, we propose a method, based on the assumptions above mentioned, to synthesize new dance performance that are matched to input music. As shown in Fig. 1, our proposed method consists of three steps: a motion analysis step, a music analysis step, and a motion synthesis step based on the results of these analyses. The motion analysis step extracts two motion features, and for dance performance synthesis: the motion keyframe component and the motion intensity component, and we use the motion features at each frame of each set of motion data. The music analysis step consists of a beat tracking part, a chord change analysis part, and a music intensity analysis part to detect characteristics of input music. In addition, we construct a motion database called a “Motion Graph.” A motion graph algorithm synthesizes various motions from a set of motion data by considering the similarity of pose. In our case, input motion data for a motion graph is a combination of primitive motions [7] and the extracted motion features are also recorded in the motion graph. In the motion synthesis step,

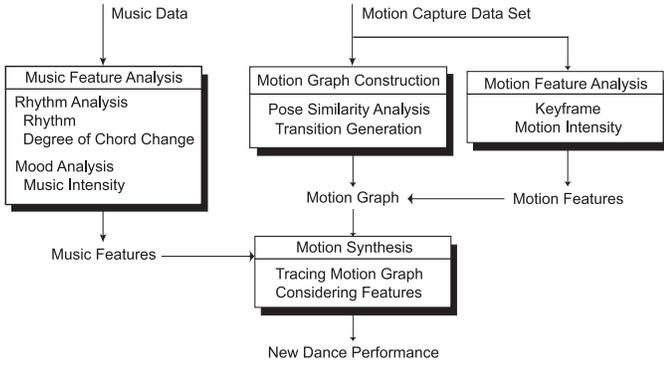


Fig. 1. Algorithm Overview.

we input music and the new dance performance is synthesized by calculating the correlation between motion features and music features and tracing the motion graph based on the correlations. The synthesized dance motion is synchronized with input music with respect to its rhythm and intensity.

The remainder of this paper is organized as follows: Our motion feature analysis and music feature analysis are presented in Section II and Section III, respectively. Section IV explains how to construct a motion graph. In Section V, we present a motion synthesis algorithm using the results of our analyses. Section VI presents our experimental results, and Section VII presents our conclusions.

## II. MOTION FEATURE ANALYSIS

In this section, we describe our method for extracting motion features. We use two components as the motion features: keyframe and intensity. The keyframe component represents whether hands stop moving or not, and the intensity component represents how wild the dance motion is. These motion features are detected for all frames of all data, and recorded in a motion graph for synthesizing dance performance. As we noted above, for synthesizing new dance performance, the motion keyframe component is correlated with the music rhythm component and the motion intensity component is correlated with the music intensity component during the tracing of the motion graph. In the latter part of this paper, we define the motion keyframe component as  $MC_{\text{stop}}$  and the motion intensity component as  $MC_{\text{intensity}}$ .

### A. Motion Keyframe Component

This component is related to the velocities of the hand movements. Many biological surveys [7], [8] say that human motion consists of the repetition of the start and stop frames (of a target position) and their interpolations. According to these ideas, we assume that the keyframes are the ‘stop’ frames of the hand movement. They can be determined by finding the local minimum of the hands’ velocities.

First we convert the hands’ positions into body center coordinate, and calculate their velocities. In the body center coordinate  $\{\mathbf{R}, \mathbf{t}\}$ , its origin is  $\mathbf{t}$  that presents the waist position,  $Z$  axis  $\mathbf{r}_z$  is the direction from waist to body,  $Y$  axis

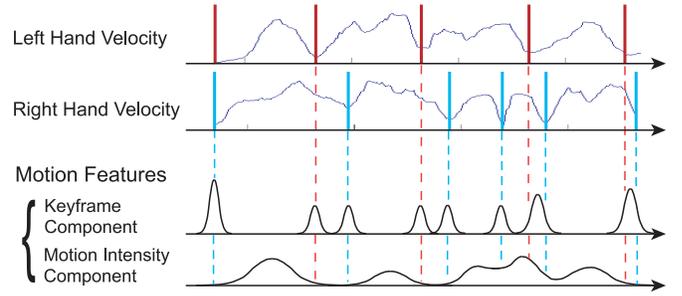


Fig. 2. Velocity graphs of both hands and illustration of motion features.

$\mathbf{r}_y$  is the frontal direction, and  $X$  axis  $\mathbf{r}_x$  is perpendicular to both of them. The local minimums are detected from these velocity sequences through simple smoothing and they are registered as keyframes. Suppose we dance to music. We cannot dance to the rhythm of music exactly; we can do so only roughly. So, we calculate Gaussian functions around the registered keyframes by inputting variance  $\sigma_1$ . According to the Gaussian variance  $\sigma_1$ , we allow some time lag. We calculate this feature in both left hand and right hand motion sequences, and add them together. In other words, the motion keyframe component gets high value when both of the hands stop moving at the same time.

$$MC_{\text{stop}} = \sum_{i \in \{L, R\}} \sum_{j=1}^{N^i} \exp\left\{-\frac{(f - f_j^i)^2}{\sigma_1^2}\right\}, \quad (1)$$

where  $N^L, N^R$  are the number of the keyframes and  $f_j^L, f_j^R$  are  $j$ -th keyframes of the left and right hands.

### B. Motion Intensity Component

In order to represent motion intensity, we consider the velocity of both hands between two neighboring keyframes. For example, if the velocity is low, it seems relaxing and the motion intensity becomes low, and if the velocity is high, it seems exciting and the motion intensity becomes high. By considering these features of motion intensity, we use the maximum velocity between two neighboring keyframes (Eq. 2). As we described above, the maximum velocity is in the body center coordinate system.

$$MC_{\text{intensity}}(f) = \sum_{i=L, R} \sum_{j=1}^{N^i-1} v_{\text{max}}^i(j) \exp\left\{-\frac{(f - f_j^c)^2}{\sigma_2^2}\right\}, \quad (2)$$

where  $f_j^c = (f_j + f_{j+1})/2$ .

Fig. 2 shows the velocity graph and illustrations of motion features. The first component is motion keyframe. This increases when  $f$  is close to the local stop frames of the hands. If both hands stop at the same time, these values are summed and increased. Similarly, the second component, the motion intensity component, it increases if hands move fast in the motion segment.

### III. MUSIC FEATURE ANALYSIS

We assume that the two music features, rhythm and musical mood, play the central role in a dance performance by human dancers. We employ these two features as music features for tracing the motion graph.

**Rhythm** People recognize the musical beats in music, and clap or wave their hands in time to the rhythm. It is general knowledge that the beat of the music is one of the most important features for dance performance. Also, western music has rhythm structures, such as the waltz rhythm and four-four time. Accordingly, dance performed to a waltz rhythm differs from that in four-four time. So we must consider the beat times and the rhythm structure of music in order to synthesize a corresponding dance performance.

**Musical Mood** When we listen to soft, relaxing music, such as a ballad, we are relaxed. On the other hand, when we listen to hard rock music, we become excited. So we can have various reactions depending on the mood of music. Further, it is very important to perform dance giving consideration to both the mood and intensity of the music. Generally speaking, when people hear intense music, they tend to perform excited, perhaps even agitated dance movements. When they hear relaxing music, people tend to perform quiet, perhaps slower dance movements.

With these considerations in mind, we extract the following three musical features: music beat and degree of chord changes for beat structure analysis, and music intensity for mood analysis. In the latter part of this paper, we define the music rhythm component as  $ASC_{\text{beat}}$ , which is extracted from music beat and degree of chord change, and the music intensity component as  $ASC_{\text{intensity}}$ .

We will explain how to obtain these components in the following sections.

#### A. Beat Tracking

To extract the music beat, we use the following assumptions:

**Assumption 1** A sound is likely to be produced with the timing of the beat.

**Assumption 2** The interval of the onset component is likely to be equal to that of the beat.

We developed a beat tracking method based on the onset component. Fig. 3 illustrates onset component calculation. By using Assumption 1, we calculate an onset component per frequency [9], the power increase from the previous time frame  $t - 1$  defined as  $d(t, f)$ .

$$d(t, f) = \begin{cases} \max(p(t, f), p(t + 1, f)) - \text{PrevPow} \\ (\min(p(t, f), p(t + 1, f)) \geq \text{PrevPow}), \\ 0 \quad (\text{otherwise}) \end{cases} \quad (3)$$

where

$$\text{PrevPow} = \max(p(t - 1, f), p(t - 1, f \pm 1)), \quad (4)$$

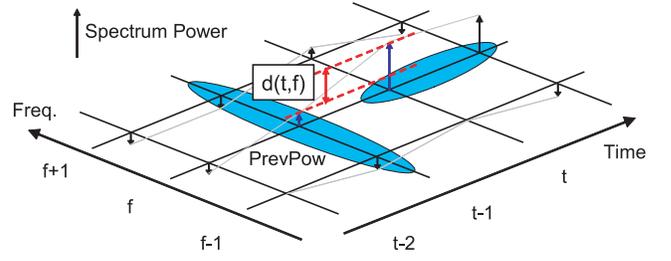


Fig. 3. An illustration of onset component extraction.

and  $p(t, f)$  is the spectral power at time  $t$  and frequency  $f$ . By calculating total onset component  $D(t) = \sum_f d(t, f)$ , we can determine the intensity of the produced sound at time frame  $t$ .

Then, by using Assumption 2, we calculate the auto-correlation function of  $D(t)$  to estimate the average beat interval. And, to estimate timing of the beat start, we calculate the cross-correlation function between  $D(t)$  and pulse sequence whose interval is the estimated beat interval. However, in practice, a beat interval sometimes changes slightly due to the performers' sensibilities, etc., and errors caused by these slight rhythm changes make beat tracking impossible. So our method determined the local maximum around the estimated rhythm based on Assumption 1.

#### B. Degree of Chord Change

To estimate beat structure, we use the following assumption:

**Assumption 3** Chord changes are likely to appear with the timing of measure lines.

Chord changes make it possible to recognize a higher beat structure and to synthesize a dance performance synchronized with the beat structure.

A chord consists of a bass sound and its overtones whose frequencies are integral multiples of bass frequency. These frequencies tend to change significantly with a chord change, and remain relatively stable when there is no change in chord. The frequency components dominant between successive beats can be roughly identified by using a histogram of frequency components. The frequency spectrum is therefore sliced into strips at the estimated beat times, and the dominant frequencies of each strip are estimated by using a histogram of frequency components in the strip, in other words, by noting the peaks of a histogram of frequency components [9]. We define these peaks as  $P_{\text{peak}}$ . The degree of chord change  $P_{\text{diff}}$  is then obtained by comparing dominant frequencies among 0 to 500Hz range, based on the frequency of bass sounds and their overtones, between adjacent strips as follows:

$$P_{\text{diff}}(\tau, f) = \begin{cases} P_{\text{peak}}(\tau, f) - P_{\text{peak}}(\tau - 1, f) \\ (P_{\text{peak}}(\tau, f) > P_{\text{peak}}(\tau - 1, f)), \\ 0 \quad (\text{otherwise}) \end{cases} \quad (5)$$

where  $\tau$  is the estimated beat time.

Because the method does not require musical notes to be identified, it can detect chord changes in real-world audio signals, where chord identification is generally difficult.

At each estimated beat time,  $P_{\text{diff}}$  is considered to evaluate the beat time:

$$ASC_{\text{beat}}(t) = \begin{cases} \sum_{f=0\text{Hz}}^{500\text{Hz}} P_{\text{diff}}(t, f) & (\text{if } t = \text{estimated beat time}). \\ 0 & (\text{otherwise}) \end{cases} \quad (6)$$

### C. Music Intensity

To extract music intensity, we use the following assumption:

**Assumption 4** The spectral power of a melody line is likely to increase during increased intensity in the music.

In most music, especially songs, the melody line is played between the frequency range of 250 to 500Hz, the range people can hear most easily. Accordingly, we use the frequency component dominant  $P_{\text{peak}}$ , as described in III-B, and extract all the maximum peaks among 250 to 1kHz range, which includes tones from base tone of melody to the 3rd overtone. After extracting these peaks, we calculate the degree of intensity by summing them:

$$ASC_{\text{intensity}}(t) = \sum_{f=250\text{Hz}}^{1\text{kHz}} P_{\text{peak}}(t, f). \quad (7)$$

## IV. MOTION GRAPH CONSTRUCTION

A motion graph connects similar poses among existing motion data, and indicates all possible transitions among the existing motion data, not only existing transitions but also new synthesized transitions [1] [2] [3]. The analysis step generates such possible new transition paths, and the synthesis step chooses appropriate ones according to the features of input music. In our case, we use a set of motion data that consists of a few primitive motions [7] of dance performance. First, we calculate pose similarity between each set of motion data and connect them based on the degree of pose similarity by creating transition motions. This graph structure of a motion data set is called a motion graph. We set the extracted motion features in the motion graph, and a new dance motion is synthesized by calculating the correlation of the music features and the motion features, and tracing the motion graph based on the correlation results. In this section, we describe how to construct a motion graph.

### A. Pose Similarity and Connectivity

We define the human body of captured data as Fig. 4. The body shape and pose for each frame  $f$  is described as 17 body shape vectors  $\mathbf{v}_n$ , which represent the direction of the  $n$ -th link in the body center coordinate  $\{\mathbf{R}, \mathbf{t}\}$ , and 9 scalar parameters  $l_n$ , which represent the length of the  $n$ -th link.

$$S(f) = \{\mathbf{R}, \mathbf{t}, \mathbf{v}_1, \dots, \mathbf{v}_{16}, l_0, l_1, \dots, l_8\}. \quad (8)$$

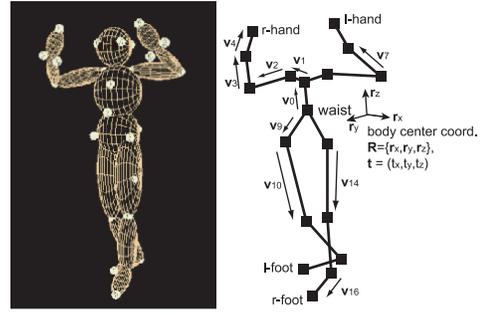


Fig. 4. Human body model.

The pose similarity distance between frame  $f_A$  in the motion sequence  $S^A$  and frame  $f_B$  in the motion sequence  $S^B$  is given by following:

$$\text{Dist}(S^A(f_A), S^B(f_B)) = \sum_i \{ \mathbf{v}_i^A(f_A) \cdot \mathbf{v}_i^B(f_B) + \alpha_i \cdot \dot{\mathbf{v}}_i^A(f_A) \cdot \dot{\mathbf{v}}_i^B(f_B) \}. \quad (9)$$

The first term can calculate the similarity of pose, and the second term can calculate the similarity of movement of the links.  $\alpha_i$  are the regularization parameters indicating the importance of  $i$ -th body portion. The value of the distance function Eq. 9 is maximized if the poses and movements at  $f_A$  and  $f_B$  are similar. In order to detect the connection frames, we apply thresholding to the value of the distance function and generate a new transition motion between selected frames.

### B. Synthesizing Transition Motion

For creating transition motions, we use 3rd order interpolation of body links, which can consider the smoothness of position, velocity, and acceleration. Assume that we would like to interpolate the motion between the pose  $S^A(f^A)$  and the pose  $S^B(f^B)$  with given duration  $T$  (frames), the vectors  $\{\mathbf{v}_i(f) | 0 \leq f \leq T, 0 \leq i \leq 16\}$  are given by the following:

$$\begin{aligned} \mathbf{v}_i(f) &= f^3 \cdot \mathbf{a}_i + f^2 \cdot \mathbf{b}_i + f \cdot \mathbf{c}_i + \mathbf{d}_i, \\ \mathbf{a}_i &= \{T(\dot{\mathbf{v}}_i^B - \dot{\mathbf{v}}_i^A) - 2 \cdot (\mathbf{v}_i^B - \mathbf{v}_i^A - \dot{\mathbf{v}}_i^A T)\}/T^3, \\ \mathbf{b}_i &= \{-T(\dot{\mathbf{v}}_i^B - \dot{\mathbf{v}}_i^A) + 3 \cdot (\mathbf{v}_i^B - \mathbf{v}_i^A - \dot{\mathbf{v}}_i^A T)\}/T^2, \\ \mathbf{c}_i &= \dot{\mathbf{v}}_i^A, \quad \mathbf{d}_i = \mathbf{v}_i^A. \end{aligned} \quad (10)$$

As for the root motion, we also use 3rd order interpolation for body center coordinates  $\{\dot{\mathbf{t}}^A, \mathbf{R}^A, \dot{\mathbf{R}}^A\}$  and  $\{\dot{\mathbf{t}}^B, \mathbf{R}^B, \dot{\mathbf{R}}^B\}$ . In order to keep the relative posture and feet in a state of contact to the ground, vertical translation  $t_z$  and vertical angle  $\theta_z = \arccos(\mathbf{r}_z \cdot \mathbf{z})$  of the body center coordinate, where  $\mathbf{z}$  is the vertical axis in the global coordinate, must be kept at the destination frame. So we first determine the frontal direction of the body center coordinate by using 3rd order interpolation, then tilt and translate these parameters to satisfy the condition.

The duration for the transition is determined by the angular distance of the concatenated frames and the maximum velocity in the concatenated motions. In the motion analysis step, we get the maximum angular velocities of all body portions

for all motion data. The duration is determined within the range where the angular velocities during transition do not exceed these maximums. This process is for avoiding unnatural transitions such that the hands move too fast compared to the neighboring motion sequences.

## V. DANCE PERFORMANCE SYNTHESIS

Now we have the motion graph and motion features from a set of motion data, and musical features from input music data. The final steps for dance performance synthesis are to evaluate the correlation between the music beat component and the motion keyframe component, to evaluate the correlation between the music intensity component and the motion intensity component, and to trace the motion graph by considering these evaluations. Our algorithm depends on the following relationship between music and human motion:

- The keyframes exist on the music beat frames.
- The greater the music intensity, the greater the motion intensity.

These assumptions are very natural. We can easily find these relationships when we view dancers' musicality in a performance.

First, we choose paths of motion graph by calculating the correlation between the music beat component  $ASC_{beat}$  and the motion keyframe component  $MC_{stop}$ . This is because the rhythm is the most important factor for dance performances and if the beat component is not considered, the synthesized motion will not be synchronized to music. In theory, all frames of the motion data in the motion graph and the music frames should be considered to detect the best motion graph path. But this involves heavy computational cost. So our algorithm considers every motion graph path from the current time  $t$  to  $t + T$ , where  $T$  is the search range and is set to 3 seconds in our experiments. The motion and music correlation of rhythm component  $MatchEval_{beat}$  is described as follows:

$$MatchEval_{beat}(t, path) = \sum_{\tau=0}^T \{MC_{stop}(t + \tau; path) \cdot ASC_{beat}(t + \tau)\}, \quad (11)$$

where  $MC_{stop}(t + \tau; path)$  is the motion keyframe component along the motion graph path  $path$ . We choose several motion graph paths whose  $MatchEval_{beat}$  are highest. We define these chosen motion graph paths as "BP."

The final step in synthesizing a new dance performance is to detect the best motion graph path from BP. This step is searching the highest evaluation path and produces the resulting motions. However, this is NP complete problem because the evaluation values on each branch depends on the past branches selected. So this is done by calculating the following correlation function between the music intensity component  $ASC_{intensity}$  and the motion intensity component  $MC_{intensity}$

$$\arg \max_{p \in BP} \sum_{\tau=0}^T \{MC_{intensity}(t + \tau; p) \cdot ASC_{intensity}(t + \tau)\}. \quad (12)$$

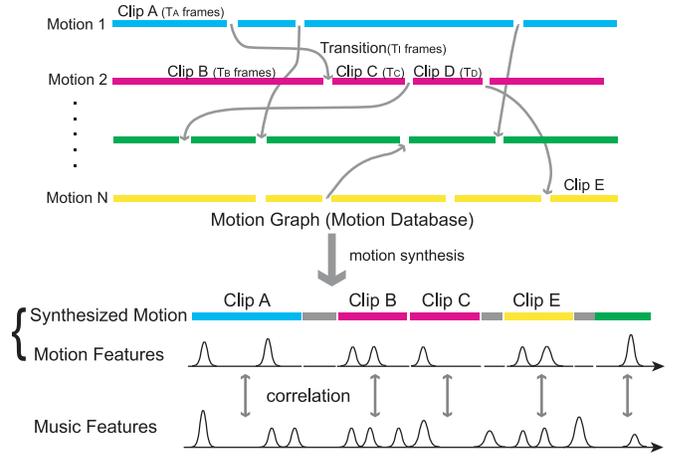


Fig. 5. Motion synthesis algorithm.

TABLE I  
RESULTS OF BEAT TRACKING

title	beat interval[sec] ([bpm])
<i>La Bamba</i>	0.598 (100)
<i>Do Me</i>	0.447 (134)
<i>Nutcracker Suite</i>	0.714 (84)

On each transition branch, we calculate the matching evaluation between the music feature and the motion feature of the destination branch, and finally the nearly optimized path is obtained.

## VI. EXPERIMENTAL RESULTS

We used a CMU Motion Capture Database [10] for the evaluation. The motion data set for our experiments consisted of 26 motion sequences. 19 of which included two or three ballet primitive motions, and the rest of which included simple dance motions that everyone can perform easily. These sets of motion data were captured by an optical motion capture system (VICON8). They contained the position data of 32 markers and were recorded at 120 Hz. The length of every motion data was about 5 seconds.

We constructed a motion graph using this database. Dead links and endless loops were carefully removed in constructing the motion graph. Also, motion features were calculated for each motion data and required less than two minutes using a PENTIUM4 1.7GHz PC.

As for the music analysis, all sounds were saved in WAV format. Their length was about 60 seconds, and the sampling was 16bit stereo at 44.1KHz. It took less than two minutes to extract all the music feature vectors using a PENTIUM-M 1.8GHz PC. We initially prepared 13 examples and checked the beat tracking results. 10 of the 13 examples appeared to successfully track. We decided to use the successful examples for motion synthesis. Table I shows the average beat duration of a part of the examples.

We experimented our synthesis method with the database. Fig. 6 is synthesized dance motion synchronized to the popular



Fig. 6. Synthesis results for “*La Bamba*.”

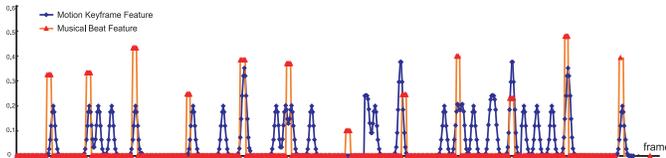


Fig. 7. Feature matching results of “*Do Me*.” The red line and the blue lines show the music beat feature and the motion keyframe feature respectively. We can confirm that most of the music beat frames match to the motion keyframes.

music “*La Bamba*.” This result can confirm that the resulting motion’s rhythm matches the music beat very well. Then we tested our synthesis method using the dance music “*Do Me*.” Fig. 7, in which the music beat feature and the motion keyframe feature are drawn, shows the result of this experiment. Most of the peaks for both features match throughout the motion sequence.

To evaluate our algorithm, we tested another dance performance synthesis method that considers only the rhythm component, using “*The Nutcracker Suite*.” Fig. 8 shows comparing the result of this method with that of our proposed method and the effectiveness of considering the intensity component. Both this result can confirm that the result of our method is synchronized to both music rhythm and music intensity, and that the intensity component is also important for dance performances.

## VII. CONCLUSION

In this paper, we described a new motion synthesis method synchronized to input music. Our method is based on two ideas: 1. dance keyframes coincide with music beat frames and 2. music intensity and motion intensity have direct correlation. Our method automatically retrieves the music beat component and the music intensity component from input music and the motion keyframe component and the motion intensity component from motion data, and concatenates corresponding motion data whose features most closely match to those of the music.

For future work, this method will be extended to real-time systems. We suppose that they retrieve exact or similar example cases, with expected rhythm and intensities, from their knowledge. So far our system does not have such a real-time functionality, but we would like to provide for this

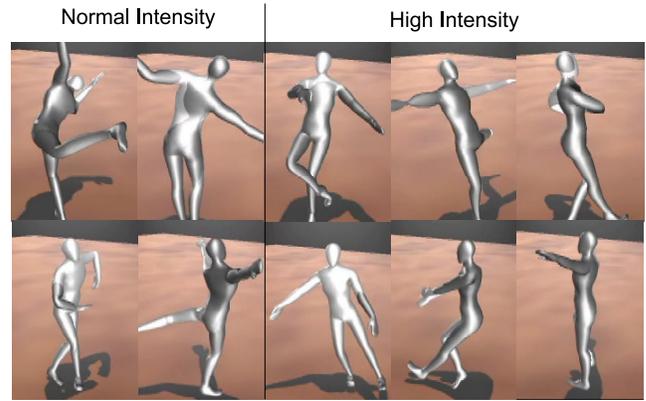


Fig. 8. Synthesis results for “*The Nutcracker Suite*.” Top: The rhythm and intensity components are considered, and bottom: only the rhythm component is considered.

capability in the future through further knowledge of music and analysis of dance motions. Additionally, we plan to equip our human-size dancing humanoid robot with this ability. We expect that the robot will be able to create and perform its own original dances to music it hears.

## ACKNOWLEDGEMENT

This work is supported in part by the Japan Science and Technology Corporation (JST) under the CREST project, and in part by the Grant-in-Aid for Scientific Research on Priority Areas (c) 16016218 of the Ministry of Education, Culture, Sports, Science and Technology, and in part by Ministry of Education, Culture, Sports, Science and Technology under the “Development of fundamental software technologies for digital archives” project.

## REFERENCES

- [1] L. Kovar, M. Gleicher, and F. Pighin, “Motion graphs,” *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 473–482, 2002.
- [2] K. Pullen and C. Bregler, “Motion capture assisted animation: Texturing and synthesis,” *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 501–508, 2002.
- [3] O. Arikan and D. A. Forsyth, “Interactive motion generation from examples,” *ACM Transactions on Graphics*, vol. 21, no. 3, pp. 483–490, 2002.
- [4] K. Grochow, S. L. Martin, A. Hertzmann, and Z. Popović, “Style-based inverse kinematics,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 522–531, 2004.
- [5] T. Kim, S. I. Park, and S. Y. Shin, “Rhythmic-motion synthesis based on motion-beat analysis,” *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 392–401, 2003.
- [6] M. Stone, D. DeCarlo, I. Oh, C. Rodriguez, A. Stere, A. Lees, and C. Bregler, “Speaking with hands: Creating animated conversational characters from recordings of human performance,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 506–513, 2004.
- [7] T. Flash and H. Hogan, “The coordination of arm movements: an experimentally confirmed mathematical model,” *Journal of Neuroscience*, vol. 5, pp. 1688–170, 1985.
- [8] M. Y. Uno and R. Suzuki, “Formation and control of optimal trajectory in human multi-joint arm movement—minimum torque change model,” *Biological Cybernetics*, vol. 61, pp. 89–101, 1989.
- [9] M. Goto, “An audio-based real-time beat tracking system for music with or without drum-sounds,” *Journal of New Music Research*, vol. 30, no. 2, pp. 159–171, 2001.
- [10] CMU Graphics Lab Motion Capture Database. [Online]. Available: <http://mocap.cs.cmu.edu/>