EUROGRAPHICS 2006 / E. Gröller and L. Szirmay-Kalos
(Guest Editors)

Volume 25 (2006), Number 3

Dancing-to-Music Character Animation

Takaaki Shiratori† Atsushi Nakazawa‡ Katsushi Ikeuchi†

† Institute of Industrial Science, The University of Tokyo, Japan
‡ Cybermedia Center, Osaka University, Japan

Abstract

In computer graphics, considerable research has been conducted on realistic human motion synthesis. However, most research does not consider human emotional aspects, which often strongly affect human motion. This paper presents a new approach for synthesizing dance performance matched to input music, based on the emotional aspects of dance performance. Our method consists of a motion analysis, a music analysis, and a motion synthesis based on the extracted features. In the analysis steps, motion and music feature vectors are acquired. Motion vectors are derived from motion rhythm and intensity, while music vectors are derived from musical rhythm, structure, and intensity. For synthesizing dance performance, we first find candidate motion segments whose rhythm features are matched to those of each music segment, and then we find the motion segment set whose intensity is similar to that of music segments. Additionally, our system supports having animators control the synthesis process by assigning desired motion segments to the specified music segments. The experimental results indicate that our method actually creates dance performance as if a character was listening and expressively dancing to the music.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and RealismAnimation; J.5 [Arts and Humanities]: Performing ArtsMusic

1. Introduction

Synthesizing realistic human motion is currently one of the most important topics in computer graphics research. Most of the motion synthesis techniques use motion capture data and synthesize new motion whose features are synchronized with external input signals such as trajectories designed by users [KGP02], environmental obstacles [LCR*02], speech information [SDO*04], motion of another character [HGP04], and so on. The issue surrounding these techniques is what kinds of cues are used to search the appropriate motions from the large amount of data in a motion database. Animators need to choose suitable cues in order to create the motion sequences they really want.

This paper proposes a new approach for synthesizing dance motion well matched to music, and our approach uses music signals as a cue to synthesize new motion. The goal of this approach is a realization of a dance algorithm that mimics human motions. The ability to dance to music is a natural born skill for a human. Everyone has experienced a desire to move their bodies while listening to a rhythmic song. Hip-hop dancers can simultaneously compose a dance motion to the musical sounds they are listening to. 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 two following aspects:

- 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.

Our approach consists of three steps: a motion analysis, a music analysis, and a motion synthesis based on the ex-

tracted features. In the motion analysis step, we analyze rhythm and intensity features of input dance motions, and assign the features to each motion in a database. The analysis methods depend on recent studies about the emotional aspects of human motions. Using these features, our system finds the sequence of motion segments matched to input music sequence with respect to the rhythm and the intensity of the music. In the music analysis step, first, we analyze a structure of input music sequence, and extract music segments based on the structure analysis results. Next, musical rhythm and intensity features are extracted, and are assigned to each music segment. Finally, our method automatically synthesizes new dance motion by interpolating between the motion segments. Additionally, our system has a user interface that enables animators to control the synthesis process by choosing desired motion segments well matched to music segments. For example, animators can set key motions in the motion database for desired music segments, such as setting a jumping motion to the final scene of the song, or a punch motion to a particular sudden sound in the music.

The remainder of this paper is organized as follows: we first present related work on motion capture-based animation and music signal processing in Section 2. Then Section 3 introduces a concept of our approach, which depends on human emotional aspects. Our motion and music analysis methods are described in Section 4 and Section 5, respectively. Section 6 describes a dance motion synthesis algorithm using results of analyses. In Section 7, user interfaces of our system for designing resulting motion are shown. The experimental results are shown in Section 8. Discussion and conclusions are presented in Section 9 and Section 10 respectively.

2. Related Work

2.1. Data-driven Character Animation

In computer graphics, research on motion capture-based character animation has been well studied. To reuse motions efficiently, methods to edit motion data have been proposed using signal processing methods such as a filter bank or dynamic programming [BW95], or warping the motion to satisfy a given time and position [WP95]. Some researchers have proposed a *retargeting* method, in which motion capture data are transferred to new characters while retaining important constraints [Gle98, LS99].

Recently many researchers have focused on using a motion database. One of the representative methods is *Motion Graph*, which is constructed by connecting motion capture data and tracing a motion graph to synthesize new motions depending on the users' input such as path or environmental obstacles [KGP02, PB02, AF02, LCR*02, LWS02]. Motion databases also enable learning of motion patterns for extracting style components [BH00, GMHP04, HPP05], to make path planning easier while considering geometric, kinematic, and posture constraints [YKH04].

Stone et al. [SDO*04] proposed a method whose approach is quite similar to ours in that input sound signals are considered. The purpose of their method is to synthesize utterance performance by extracting emphasis features of motion and speech data and synchronizing them. However, their feature extraction needs many manual processes, and is accordingly a very time-consuming system for synthesizing new utterance motions.

Kim et al. [KPS03] proposed a rhythmic motion synthesis method using the results of motion rhythm analysis. But using their method, music data needs to have a rhythm interval that is similar to that of the resulting motion, and it is quite difficult to apply this method with various kinds of music data.

Müller et al. [MRC05] proposed a motion-retrieval method based on motion contents, which is a similar approach to ours. However, the motion contents considered in our approach are defined based on human emotional aspects, while those in their method were defined based on specified joint position/angle.

2.2. Auditory Scene Analysis

Computational analysis methods for a music scene are important for understanding how humans recognize musical features, and are called Computational Auditory Scene Analysis. [Bre90,CB93]. In particular, for dance motion synthesis, we believe that rhythm features, rhythm structure, and musical intensity are very important.

Most humans have an ability to recognize rhythm and rhythm structure. When people hear music, they tap their feet, wave their hands in time with the music, and discover the ability to dance to the music even if they are children or beginners. Many researchers are working on the rhythm tracking method considering these abilities.

In the case of MIDI signals, parameters of various musical features such as onset, pitch, and volume are easily obtained and the most useful in rhythm tracking [DH89, Ros92]. However, it is quite difficult to extract most of these musical features from audio signals, and considerable research has been done on rhythm tracking for audio signals. Most of the rhythm tracking methods for audio signals are based on the knowledge of the onset component [Tod94, LZ03]. Goto [Got01] proposed a real-time rhythm tracking method based on not only the onset component, but also chord changes and drum sounds for rhythm structure analysis. Scheirer [Sch98] proposed an offline rhythm tracking method for music that signifies rhythm changes by notations such as *accel.* and *rit.* There are methods that can predict the musical rhythm by using kalman filtering [CKDH01] and bayesian network [SMS05, NT04].

Most musical songs have repeating patterns and prominent structure, and musical structure analysis methods have

been used to accomplish applications such as music summarization. In general, repeating patterns are considered as melody similarity. In order to extract the melody similarity, musical intensity features that are extracted from spectral components [LC00, WLZ04, SXWS04] or amplitude envelopes [LZ03] are used.

3. Concept of Our Method

Our approach uses musical information as a cue to retrieve motion segments from a motion capture database. We start by discussing a human perceptional model based on the relationship between human motions and music. To define this music and motion relationship model, previous studies of human dance motion analysis are of great help.

Laban, who is famous for his novel dance description method called "Labanotation," is a pioneer in the study of this issue. He has studied human emotional aspects of body movements [LU60]. According to his theory, the emotion of human motion comes from motion features consisting of *"effort"* and *"shape"* components. The effort component is defined as the movements of body portions, and the shape component is defined as the shape of elements he calls "keyposes." More recently, Nakata et al. [NMS02] have tested the validity of Laban's theory by using their small robot and user studies. Although they could not find a significant relationship between the shape component and any emotions, they found that the *"weight effort"* component, one of the effort components, is closely related to the excitement of the motion. Laban defined the weight effort component as the strength of movement, and Nakata considered them physically as the linear sum of rotation velocity of each body joint. We use these metrics to define the motion intensity component F_I^{Motion}.

Additionaly, we have developed a method that analyzes the relationship between stop motions and musical rhythm, and the results indicate that musical rhythm has a strong connection with motion elements called "motion keyposes" [SNI04]. Accordingly, our motion analysis method extracts the local minimums of the weight effort component in order to extract the motion rhythm feature F_R^{Motion}. A motion feature vector for each frame is obtained via the motion feature analysis:

$$\textbf{MotionFeature}(f) = \begin{bmatrix} F_R^{\text{Motion}}(f) \\ F_I^{\text{Motion}}(f) \end{bmatrix}. \qquad (1)$$

The next issue is to extract musical features. We believe that there are three important musical features for dance performance. One is *musical rhythm*. As everyone has experienced, there is a very close relationship between musical rhythm and motion rhythm. We consider musical knowledge about what is called "the onset component" to estimate musical rhythm F_R^{Music}. Another important factor is *music structure*, which consists of several musical phrases. Both musical players and dancers try to keep the structure from being violated during their performances. We extract repeating patterns to detect the musical structure, and obtain music segments from the music sequence. The other important component is *music intensity*. People feel various emotions depending on musical mood, and the same is true for dance performance. For musical mood analysis, we mainly focus on music intensity, one of the effective factors for musical mood. We extract the music intensity component F_I^{Music} using the energy of the melody line. Accordingly, a music feature vector for each music segment $\mathcal{M}$ is obtained:

$$\textbf{MusicFeature}(f;\mathcal{M}) = \begin{bmatrix} F_R^{\text{Music}}(f;\mathcal{M}) \\ F_I^{\text{Music}}(f;\mathcal{M}) \end{bmatrix}. \qquad (2)$$

Our motion synthesis step extracts the most appropriate motion segment sequence by evaluating motion and music features. First, we detect candidate motion segments for each music segment by using rhythm features. Then, connections between neighboring motion segments are analyzed, and motion segment sequences that look like natural motions are obtained. Finally, the best motion sequence is selected and interpolated from the remaining motion sequences by evaluating the similarity between the motion and music intensity components.

4. Motion Feature Analysis

As described in Section 3, our motion analysis method strongly relies on Laban's weight effort component. In this section, we describe our definition of the weight effort component and how to extract the motion features.

4.1. Human Model

We first convert motion capture data into our simple human body model. Figure 1 illustrates our human model. In our model, a human pose at each frame is converted into a *body center coordinate*, in which we set the origin $\mathbf{t}$ to the waist position in the global coordinate, x-coordinate $\mathbf{r}_x$ to the direction from left to the right thigh, y-coordinate $\mathbf{r}_y$ to the forward direction of the body, and z-coordinate $\mathbf{r}_z$ to a vertical upper direction. The length of each coordinate vector is set to 1. $\mathbf{v}_n$ is a unit vector representing the direction of the n-th body link in the coordinate $\{\mathbf{R},\mathbf{t}\}$, and l_n represents the length of the n-th body link.

4.2. Weight Effort

According to Laban's definition, the weight effort component represents the strength of motion. Thus, we define the weight effort component W as the linear sum of approximated instantaneous momentum magnitude calculated from the link and body directions:

$$\begin{aligned} W(f) = \quad & \textstyle\sum_i \alpha_i \arccos\left(\frac{\dot{\mathbf{v}}_i(f)}{|\dot{\mathbf{v}}_i(f)|} \cdot \frac{\dot{\mathbf{v}}_i(f+1)}{|\dot{\mathbf{v}}_i(f+1)|}\right) \\ & + \textstyle\sum_{j\in\{x,y,z\}} \arccos\left(\frac{\dot{\mathbf{r}}_j(f)}{|\dot{\mathbf{r}}_j(f)|} \cdot \frac{\dot{\mathbf{r}}_j(f+1)}{|\dot{\mathbf{r}}_j(f+1)|}\right), \end{aligned} \qquad (3)$$

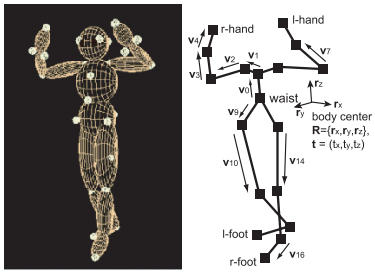


Figure 1: *Our human body model. The shape and pose are described by the base matrix* $\{\mathbf{R},\mathbf{t}\}$ *and the 17 vectors* $\mathbf{v}_n$*. The lengths of the body links are given by* l_n*. Our method converts the pose at each frame into this coordinate.*

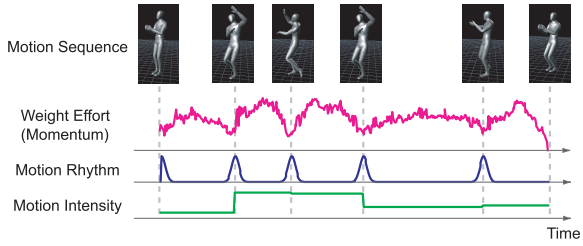


Figure 2: *The motion feature vector of an example motion. Motion rhythm and intensity components are obtained from "weight effort" of body movement. Motion rhythm component is the local minimums of the weight effort component (dashed lines), and motion intensity comes from the average of weight effort and forward translation of the body within the neighboring motion rhythm frame.*

where α_i is a regularization parameter for the i-th link. These regularization parameters depend on which parts we recognize as important for dance expression. For example, if we recognize the hands and feet as important, α corrsponding to them will be greater than those corresponding to other parts.

4.3. Motion Rhythm Feature

Considering the characteristics of the weight effort component, the local minimums of this component indicate stop motions, which are impressive instances for dance performance. We recognize these local minimums as motion "keyposes," and define the motion rhythm features F_R^{Motion} as follows:

$$F_R^{\text{Motion}}(f) = \begin{cases} 1 & \text{if } W(f) \text{ is around the local minimum} \\ 0 & \text{otherwise} \end{cases}. \tag{4}$$

4.4. Motion Intensity Feature

It was validated that motion intensity is related to momentum and forward translation. We obtain instant motion intensity I from the momentum W and the speed of the forward direction $\mathbf{r}_y \cdot \dot{\mathbf{t}}$:

$$I(f) = W(f) \cdot (1.0 + k \cdot \mathbf{r}_y(f) \cdot \dot{\mathbf{t}}(f)), \tag{5}$$

where k is a regularization parameter between the weight effort and the speed. Finally, we calculate the average of the instant motion intensity from the previous motion keypose f_i^R to the next one f_{i+1}^R, and set it to the motion intensity:

$$F_I^{\text{Motion}}(f) = \sum_{i=f_i^R}^{f_{i+1}^R} \frac{I(i)}{f_{i+1}^R - f_i^R + 1}. \tag{6}$$

5. Music Feature Analysis

When people listen or dance to music, they extract some musical features from an audio signal. The important features for dance performance are music structure, rhythm, and intensity. In this section, we describe how to acquire the music segments and to extract the musical rhythm and intensity.

5.1. Constant Q Transform

Music is different from speech in that music consists of a sequence of notes whose frequencies are already defined. Ideally, it is most appropriate for extraction of musical features that music signals are converted into a note sequence. But most of the frequency component extraction methods such as Fourier transform do not consider this musical aspect. In order to extract frequency components representing music notes more accurately, we use constant Q transform (CQT) proposed by Brown et al. [Bro90]. The CQT method sets the bank of filters whose center frequencies represent musical notes, and enables extraction of the spectral energy of each note.

In our approach, we extract the spectral energies of the 37 semi-tones (over three octaves from the C3 note to the C6 note) from audio signal $x(n)$ as follows:

$$X(k) = \frac{1}{N_k} \sum_{n=0}^{N_k-1} x(n) \exp(-j\frac{2\pi Q n}{N_k}), \tag{7}$$

where j represents $\sqrt{-1}$, $X(k)$ represents the spectral power of the k-th note, N_k is the window size, and n represents the sampled frame. According to music theory, the frequency of the k-th note is calculated as

$$f_k = f_0 \cdot 2^{k/N_{\text{octave}}}, \tag{8}$$

where f_0 is the minimal frequency that we are interested in for analysis and is set to 130.8 Hz, the pitch of the C3 note, and N_{octave} denotes the number of semi-tones in one octave and is typically set to 12. Q is a constant ratio of frequency to resolution:

$$Q = f_k/(f_{k+1} - f_k) = 1/(2^{1/N_{\text{octave}}} - 1), \tag{9}$$

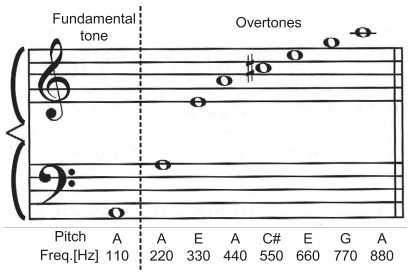


Figure 3: *An example of fundamental tone and its overtones. When a sound 'A' whose frequency is around 110Hz is produced, its overtones, whose frequencies are integral multiples of the fundamental tone, are also produced.*

and accordingly the window size N_k is set as:

$$N_k = \lfloor f_s Q / f_k \rfloor, \tag{10}$$

where f_s represents the sampling rate of the input audio signal. Our method uses the hamming window function, and shifts it by some interval, and then calculates the CQT component until the window reaches the end of the music, like the short-time FFT calculation. In the following section, $X(t,k)$ denotes the spectral power of k-th note at t-th temporal frame.

5.2. Music Segment Retrieval

With respect to musical structure, we use the following knowledge:

Knowledge1 Music structure consists of the repetition of several phrases.

The goal of this analysis is to extract the patterns of the repeating phrases and to segment the music by the extracted repeating patterns.

Some phrases may be repeated, performed by the different instruments (e.g., one phrase is performed by a vocalist, and the repeat is performed by the guitar). However, people can easily recognize that they are the same phrases, and therefore the structure analysis method should depend on the sequence of the notes, but not be affected by the timbre of the instruments.

Figure 3 shows a mechanism of timbre. The timbre of every instrument has a basic characteristic that it always consists of a fundamental tone and its overtones, whose frequencies are integral multiples of the fundamental frequency, but the energies of the overtones differ from one instrument to another. That is, it is difficult to extract accurate repeating patterns directly in the frequency domain.

In order to find the repeating patterns, we use CQT feature vectors, and evaluate them with a structure-based similarity measurement that is independent of the timbre effects proposed by Lie et al. [WLZ04]. First, we calculate the autocorrelation of the elements of difference vector:

$$r_{ij}(m) = \sum_{n=0}^{N-m-1} \Delta v_{ij}(n+m) \cdot \Delta v_{ij}(n), \tag{11}$$

where $\Delta v_{ij}(n)$ is the absolute difference of the n-th CQT feature vector element between the i-th and j-th temporal frames:

$$\Delta v_{ij}(n) = |X(i,n) - X(j,n)|, \tag{12}$$

and N is the number of the elements of CQT feature vectors. If the CQT feature vectors contain the same pitch sound, the peaks of $r_{ij}(m)$ will have *harmonic intervals* that are based on the characteristics of the overtones, and if not, the peaks will appear without this interval. In detail, if the vectors contain the same pitch, the peak of $r_{ij}(m)$ will strongly appear at $m = 0, 12, 19, 24, 29$ etc., which represent the fundamental frequency f_b and its integral multiples $2f_b, 3f_b, 4f_b, 5f_b$. This characteristic is modeled as the spiral array [Che01], and the elements of the weighting vector $w(m)$ for $\mathbf{r}(i,\ j) = [r_{ij}(0), r_{ij}(1), \cdots, r_{ij}(N)]^T$ are represented as

$$w(m) = \frac{1}{A} |\mathbf{p}(7m \bmod 12) - \mathbf{p}(0)|, \tag{13}$$

where A is a normalization factor to satisfy $\sum_m w(m) = 1$, and

$$\mathbf{p}(m) = [\sin\frac{m\pi}{2}, \cos\frac{m\pi}{2}, \frac{m\pi}{2}]^T. \tag{14}$$

Accordingly, the distance D between two CQT feature vectors is considered the neighboring frames and evaluated as follows:

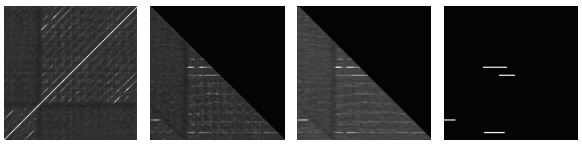
$$D(i,j) = \frac{1}{2N_r} \sum_{k=-N_r}^{N_r-1} \mathbf{w} \cdot \mathbf{r}(i+k,\ j+k), \tag{15}$$

where $\mathbf{w}$ represents the weighting vector, and $2N_r$ is the range for the distance calculation.

Once the distance function is defined, we can get the similarity matrix $\mathbf{S}$ whose elements are the similarity measurements $1/D(i,j)$, and then convert it to *time-lag matrix* $\mathbf{T}$:

$$T_{ij} = S_{i,i+j} = \frac{1}{D(i,i+j)}. \tag{16}$$

Figure 4 shows examples of these matrices. In this figure, the brighter regions show the greater similarity, and several white horizontal lines appear clearly in the time-lag matrix. These lines denote the repeating patterns. By extracting them, we can acquire the repeating phrases, and analyze the structure of the input music. In detail, erosion and dilation operators that are often used in image processing are applied to make the lines more clear, and then the lines can be extracted with a thresholding process. Finally, music segments are extracted by dividing the music sequence at the boundaries of resulting repeating phrases. The other musical features are extracted and assigned to each music segment.



(a) (b) (c) (d)

Figure 4: *An example of repeating pattern analysis steps. (a) Similarity matrix, (b) time-lag matrix, (c) time-lag matrix after erosion and dilation operations, and (d) result of repeating phrases extraction.*

5.3. Music Rhythm Feature

To extract the musical rhythm, we use the following knowledge:

Knowledge2 A sound is likely to be produced with the timing of the rhythm.

Knowledge3 The interval of the onset component is likely to be equal to that of the rhythm.

So we consider the onset component for estimating the musical rhythm. Figure 5 illustrates the onset component extraction. First, using Knowledge2, we calculate the onset component of the k-th note, which is the power increase from the previous temporal frame $t-1$ defined as $d(t,k)$ [Got01].

$$d(t,k) = \begin{cases} \max(X(t,k), X(t+1,k)) - \text{PrevPow} \\ \quad (\min(X(t,k), X(t+1,k)) \geq \text{PrevPow}), \\ 0 \quad (\text{otherwise}) \end{cases} \tag{17}$$

where

$$\text{PrevPow} = \max(X(t-1,k), X(t-1,k\pm 1)). \tag{18}$$

By calculating total onset component $D(t) = \sum_k d(t,k)$, we can determine the total intensity of the sounds produced at the t-th temporal frame.

Using Knowledge3, we calculate the auto-correlation function of $D(t)$ to estimate the average rhythm interval. Then, the starting time is estimated by calculating the cross-correlation function between $D(t)$ and pulse sequence whose interval is the estimated rhythm interval. However, in practice, a rhythm interval may change slightly due to the performers' sensibilities, etc., and errors caused by these changes make rhythm tracking impossible. So, considering Knowledge2 again, our method tracks the local maximum around the estimated rhythm. The musical rhythm feature F_R^{Music} is defined as follows:

$$F_R^{\text{Music}}(f;\mathcal{M}) = \begin{cases} 1 & \text{if } f \text{ in } \mathcal{M} \text{ is estimated rhythm time} \\ 0 & \text{otherwise} \end{cases}. \tag{19}$$

5.4. Music Intensity Feature

To extract music intensity, we use the following knowledge:

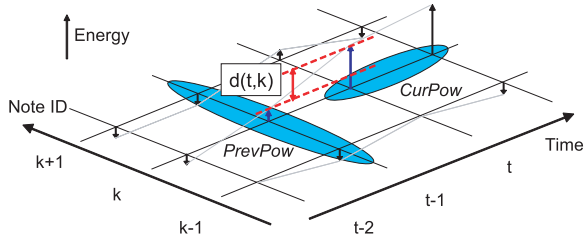


Figure 5: *An illustration of onset component extraction. First, the maximum among $X(t-1,k)$ and $X(t-1,k\pm 1)$ described as* PrevPow*, and the minimum between $X(t,k)$ and $X(t,k+1)$ described as* CurPow *are extracted. Then, the difference between CurPow and PrevPow is calculated. If the difference is larger than 0, $d(t,k)$ is the difference; otherwise, $d(t,k)$ is 0.*

Knowledge4 The spectral power of a melody line is likely to increase during increasing intensity in the music.

Knowledge5 A melody line is likely to be performed using a higher range than the C4 note.

Many surveys on auditory psychology [Roa96] say that our ears tend to recognize only the sound whose spectral power is the strongest among the neighboring frequency sounds, which is often used in many audio signal compression algorithms such as MP3. Accordingly, a temporally average spectral power $\bar{X}$ of k-th note within a music segment $\mathcal{M}$ is calculated to figure out which note sounds are produced in the music segment:

$$\bar{X}(\mathcal{M},k) = \frac{1}{|\mathcal{M}|} \sum_{t\in\mathcal{M}} X(t,k), \tag{20}$$

where $|\mathcal{M}|$ denotes the number of the CQT feature vectors in $\mathcal{M}$, and then the local peaks X_{peak} of each average CQT feature vectors are picked up: $X_{\text{peak}}(\tau,k) = \bar{X}(\mathcal{M},k)$ if $\bar{X}(\mathcal{M},k) > \bar{X}(\mathcal{M},k\pm 1)$, otherwise $X_{\text{peak}}(\mathcal{M},k) = 0$. In order to extract music intensity feature F_I^{Music}, we approximately calculate the *Sound Pressure Level*, which considers the humans' auditory properties and is related to both the amplitude and the frequency:

$$F_I^{\text{Music}}(f;\mathcal{M}) = \log_{10}\Big(\sum_{k\in[\text{C4,C6}]} X_{\text{peak}}(\mathcal{M},k)^2 \cdot f_k^2\Big). \tag{21}$$

6. Motion Synthesis Considering Motion and Music Features

The final step of our approach is to synthesize new dance motions considering both the motion and music feature vectors. The main purpose *and* problem of this step are to select the motion segment set from the motion database with as low a loss of correlation as possible. In order to achieve this, we perform three steps to synthesize a new dance motion. Figure 6 gives an overview of our motion synthesis algorithm. First, we evaluate the similarity of the rhythm components,

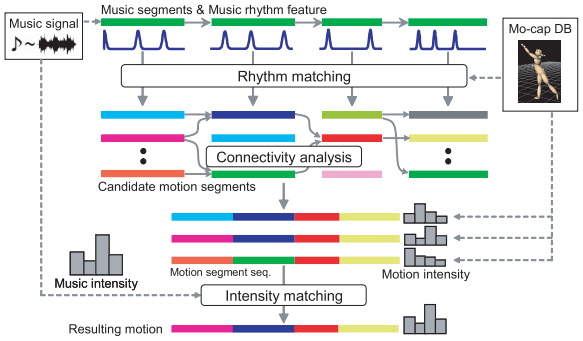


Figure 6: *Overview of our motion synthesis algorithm. For each music segment, candidate motion segments are obtained from a motion-capture database by evaluating the similarity with music rhythm components. Then, all possible motion segment sequences can be acquired by connectivity analysis between neighboring motion segments. Finally, we evaluate the similarity of the intensity components between the motion segments and the music segments.*

and detect the candidate motion segments strongly corresponding to each music segment. Then, we apply connectivity analysis, which checks if synthesized transition motion between the neighboring motion segments looks natural, and extract the possible sequences of motion segments. Finally, we analyze the similarity of their intensity components between the music segments and the selected motion segment sequences, and synthesize new dance motions by connecting the motion segments with each other.

6.1. Similarity Measurement of Rhythm Components

This step extracts the candidate motion segments from every input motion sequence, considering motion and music rhythm components. To include more detail, we focus on one input motion sequence whose length is L_{motion} and a music segment $\mathcal{M}$ whose length is L_{music}. In our method, we allow a slight stretching of the duration of the input motion sequence. Thus, on the similarity measurement of their rhythm components, we consider not only the rhythm components but the scaling parameter $s \in [0.9\ 1.1]$ and the offset parameter f_o, which represents the frame from which a motion segment starts. We extract the scaling parameter $\hat{s}$, which maximizes the similarity measurement

$$\hat{s} = \arg\max_{s} \sum_{f=0}^{L_{\text{music}}} \frac{F_R^{\text{Music}}(f;\mathcal{M}) \cdot F_R^{\text{Motion}}(s \cdot f + f_o)}{F_R^{\text{Music}}(f;\mathcal{M}) + F_R^{\text{Motion}}(s \cdot f + f_o)} \quad (22)$$

for each $f_o \in [0,\ L_{\text{motion}} - L_{\text{music}}]$.

We extract all possible sets of (s, f_o) for each input of motion sequence, and apply a simple thresholding process to the parameter sets. Using the remaining parameters, we can extract candidate motion segments for each music segment.

6.2. Connectivity Analysis of Motion Segments

Whether or not synthesized motion looks natural strongly depends on connectivity analysis. In this step, we consider both the posture similarity S_{pose} and movement similarity S_{move}. Posture similarity S_{pose} between the $i^{\mathcal{A}}$-th frame of the motion segment $\mathcal{A}$ and the $j^{\mathcal{B}}$-th frame of the motion segment $\mathcal{B}$ is defined as the angular similarity of the link direction vectors:

$$S_{\text{pose}}(i^{\mathcal{A}}, j^{\mathcal{B}}) = \sum_l \beta_l \cdot \mathbf{v}_l(i^{\mathcal{A}}) \cdot \mathbf{v}_l(j^{\mathcal{B}}), \quad (23)$$

where β_l is a regularization factor for the l-th link. With regard to movement similarity S_{move}, we use velocity vectors in homogeneous coordinates, since the angular distance measure of their unit vectors in the homogeneous coordinates account for the differences in both direction and magnitude. Specifically, movement similarity S_{move} is calculated as follows:

$$\begin{aligned} S_{\text{move}}(i^{\mathcal{A}}, j^{\mathcal{B}}) = {} & \textstyle\prod_l g[h(\mathbf{v}_l(j^{\mathcal{B}}) - \mathbf{v}_l(i^{\mathcal{A}})) \cdot h(\dot{\mathbf{v}}_l(i^{\mathcal{A}}))] \\ & \cdot g[h(\mathbf{v}_l(j^{\mathcal{B}}) - \mathbf{v}_l(i^{\mathcal{A}})) \cdot h(\dot{\mathbf{v}}_l(j^{\mathcal{B}}))], \end{aligned} \quad (24)$$

where $g[x]$ denotes x if $x \geq 0$, otherwise 0, and $\dot{\mathbf{v}}$ is calculated from the original input motion sequence, not the candidate motion segment. Through h, an input 3D vector $(x,y,z)^T$ is converted to the 4D unit vector $(x,y,z,1)^T/|x,y,z,1|$. That is, Eq. 24 evaluates the similarity of the directions between the original movement in the input motion sequence and the motion to be synthesized. Finally, connectivity is analyzed from both S_{pose} and S_{move} between the end frame of one motion segment and the beginning frame of the neighboring motion segments. From the results of the connectivity evaluation, we obtain the candidate sequences of the motion segments that satisfy the requirements for similarity with the rhythm features and naturalness of the synthesized motion.

6.3. Similarity Measurement of Intensity Components

Next, we evaluate the intensity components of the candidate sequences of the motion segments and input music. In order to find the globally optimal solution, we consider the time series of the intensity features as a histogram, and the Bhattacharyya coefficient [Kai67] is considered to relatively evaluate the similarity between the motion and music intensity histograms. Hence, we finally obtain the motion segment sequence $\hat{D}$ that maximizes the Bhattacharyya coefficient:

$$\hat{D} = \arg\max_{D \in \mathcal{CS}} \sum_j \sqrt{\frac{F_I^{\text{Music}}(j)}{\sum_k F_I^{\text{Music}}(k)} \cdot \frac{F_I^{\text{Motion}}(j)}{\sum_{k \in D} F_I^{\text{Motion}}(k)}}, \quad (25)$$

where $\mathcal{CS}$ represents the candidate sequences of the motion segments after the analyses of rhythm similarity and connectivity.

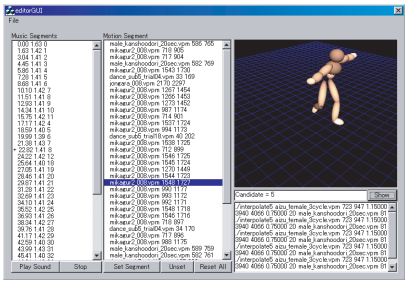


Figure 7: *Our user interface for designing motion. A user can confirm the music and motion segments by selecting and double-clicking an item out of the lists in which the music segments and their corresponding motion segments are displayed from left to center, respectively. The process of designing motion is just assigning the desired motion segment to the music segment. The resulting motion is displayed in the top-right window.*

6.4. Transition Motion Generation

The resulting motion sequence is acquired by connecting the best matched motion segment sequence. For posture, we use a spline function considering the first and second order differential to interpolate the motion segments. For the position of a character, we pay attention to the position and posture relative to the ground in order to avoid effects such as sliding or being stuck in one position.

7. Interface for Designing Dance Motions

Our method can synthesize new dance performances well matched to input music. However, the resulting motion sequence does not reflect the animators' design. For example, some animators may want a character to jump when vocal input music says, "*Jump!*"

Our system supports the designs animators often have. Figure 7 shows our interface that enables animators to design motions. The left list shows the music segment sequence, and the central list shows the extracted motion segments corresponding to the currently assigned music segment. A user can confirm the music segments and the motion segments by selecting and double-clicking an item out of the lists. Using our system, the desired motion segment can be assigned to the music segment as animators want. It is conceivable that there are no candidate sets of the motion segments that satisfy the assigned design. If so, our system re-evaluates the motion and music features under this constraint.

8. Experiments

We have experimented in our proposed method with our motion database consisting of break dance, Indian dance, and simple dance motion, which are all downloaded from the CMU Motion Capture Database. They were all captured with an optical motion-capture system produced by Vicon, and their sampling rate is 120Hz. The length of music data used for our experiments was about 60 seconds, and the sampling was 16bit stereo at 44kHz.

Table 1: *Results of Rhythm Tracking*

Title (Genre)	Rhythm [sec] ([bpm])
Again (pops)	0.459 (131)
Tonite (pops)	0.476 (126)
Carmen Suite (classic)	0.417 (144)
Nutcracker Suite (classic)	0.714 (84)

Results of Music Analysis We first show the results of the music feature analysis. We applied the rhythm tracking method to 13 music data sets that contain classical music, rock, jazz, and so on. Ten out of them correctly tracked the rhythm. The accompanying music data show that our rhythm tracking method can estimate music rhythm correctly. Table 1 shows a part of the successful rhythm tracking results. Additionally, the music intensity analysis was also successful. However, the structure analysis sometimes failed, especially when it was applied to jazz music. This is because jazz music often contains *ad-lib* whose melody lines rarely repeat.

Results of Dance Performance Synthesis Figure 8 shows the synthesized motion for popular music "*Again*." The accompanying video shows that the rhythm and intensity of the resulting motion are well matched to those of the input music.

Figure 9 shows another synthesized motion for popular music "*Tonite*." The accomponying video also shows that our method works well. Figure 10 shows the features of the synthesized motion and the input music. In this figure, the yellow line and the light blue line show the motion rhythm component and the music rhythm component respectively, and the blue line and the red line are the intensity histograms of motion and music segments. We can easily confirm that most of the musical rhythm is matched to the motion rhythm, and the distributions of the intensity components are quite similar.

Computational Cost The synthesis step takes much longer than the other analysis steps. Especially, the connectivity analysis between neighboring candidate motion segments is the most time-consuming process, because all possible sets of the neighboring segments are checked. In the case of using the music *Again*, it takes around 10 minutes to synthesize motion from input music data one minute long and around 27 motion data sets (about 520 sec in total) with Pentium-4 2.8GHz PC.

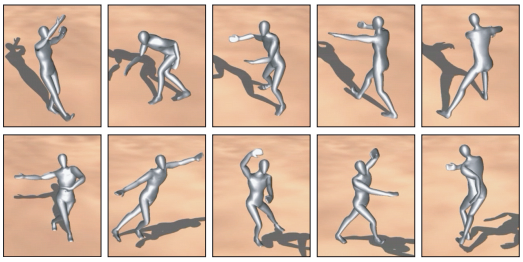


Figure 8: *The synthesis result for music "Again."*

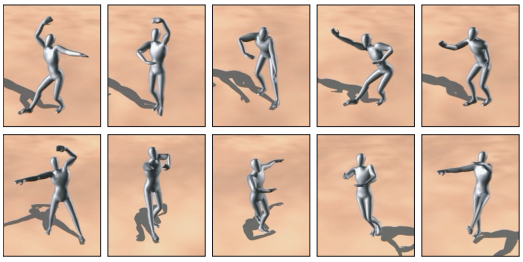


Figure 9: *The synthesis result for music "Tonite."*

9. Discussion

Our algorithm can synthesize new dance motion considering musical and motion rhythms, and musical and motion intensities. This is based on two ideas: 1. motion rhythm is correlated with musical rhythm, and 2. music intensity and motion intensity have a direct correlation. Our contribution is, with regard to CG animation, to automatically synthesize motion that synchronizes input music signals, and to take motion expressions extracted from Laban's weight effort component into consideration. With regard to artificial intelligence, we have been able to imitate the simple models of human emotional aspects and the human ability to recognize music features for dance performance while listening to music.

We believe that it is possible to introduce other features for matching, such as relationships between a music chord or key (major/minor) and mood of motion, or a category of music and its appropriate expression in dance. For example, people tend to feel gloomy when listening to music in a minor key, and happier when listening to music in a major key. To improve our approach, music psychology could be incorporated. Additionally, motion expressions, which have not been well studied in CG animation, are also important factors. As future work, we will develop a motion expressions analysis method, and introduce them into our method with corresponding music psychology.

Additionally, we are now developing another application to synthesize dance motions in real time: a character composes new dance motion while listening to music. The purpose of this application is to imitate the ability of *ad-lib* dance which all people, and particularly children, have. This application will also enable a humanoid robot to dance to music as an entertainment robot.

10. Conclusion

This paper presented a method for synthesizing new motion synchronized to music. Our idea is to consider the musical rhythm and intensity components to be matched to motion rhythm and intensity components. This is an imitation of a dancer's skill in performing motions as they listen to music. Our method can automatically retrieve music features from input music signals and motion features from motion sequence, and synthesize new dance motions whose features are closely matched to those of the music. We have presented results from which we can confirm that our method can synthesize expressive dance performance.

Acknowledgement

This work is supported in part by Ministry of Education, Culture, Sports, Science and Technology under the "Development of fundamental software technologies for digital archives" project. Takaaki Shiratori is supported by the Japan Society for the Promotion of Science. The motion data used in this project was obtained from http://mocap.cs.cmu.edu. The database was created with funding from NSF EIA-0196217. The music data used in this project was obtained from http://www.freeplaymusic.com.

References

[AF02] ARIKAN O., FORSYTH D. A.: Interactive motion generation from examples. *ACM Trans. on Graphics 21*, 3 (2002), 483–490.

[BH00] BRAND M. E., HERTZMANN A.: Style machines. *ACM Trans. on Graphics 22*, 3 (2000), 402–408.

[Bre90] BREGMAN A. S.: *Auditory Scene Analysis: The Perceptual Organization of sound*. The MIT Press, 1990.

[Bro90] BROWN J. C.: Calculation of a constant Q spectral transform. *Journal of Acoustic Society of America 89*, 1 (1990), 425–434.

[BW95] BRUDERLIN A., WILLIAMS L.: Motion signal processing. In *Proc. ACM SIGGRAPH 95* (1995), pp. 97–104.

[CB93] COOKE M., BROWN G.: Computational auditory scene analysis: Exploiting principles of perceived continuity. *Speech Communication 13* (1993), 391–399.

[Che01] CHEW E.: Modeling tonality: Applications to music cognition. In *Proc. Annual Conf. of the Cognitive Science Society* (2001), pp. 206–211.

[CKDH01] CEMGIL A. T., KAPPEN B., DESIAN P., HONING H.: On tempo tracking: Tempogram representation and kalman filtering. *Journal of New Music Research 29*, 4 (2001), 259–273.

[DH89] DESAIN P., HONING H.: The quantization of musical time: A connectionist approach. *Computer Music Journal 13*, 3 (1989), 56–66.

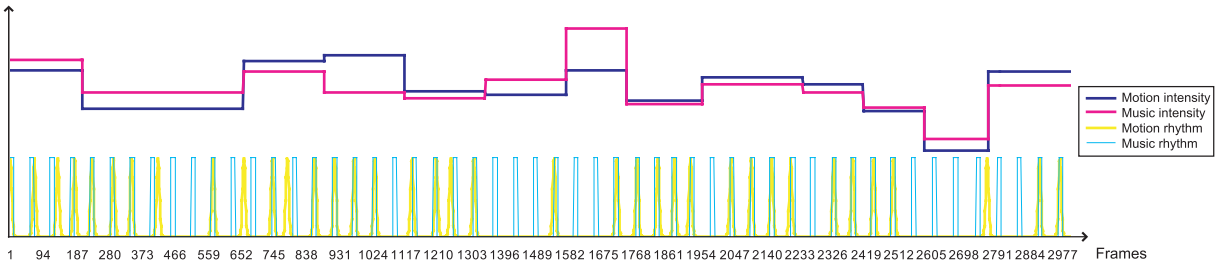


Figure 10: *The feature matching result for music "Tonite." Yellow and light blue lines represent motion and music rhythm components, and blue and red lines represent motion and music intensity components.*

[Gle98] GLEICHER M.: Retargetting moiton to new characters. In *Proc. ACM SIGGRAPH 98* (1998), pp. 33–42.

[GMHP04] GROCHOW K., MARTIN S. L., HERTZMANN A., POPOVIĆ Z.: Style-based inverse kinematics. *ACM Trans. on Graphics 23*, 3 (2004), 522–531.

[Got01] GOTO M.: An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research 30*, 2 (2001), 159–171.

[HGP04] HSU E., GENTRY S., POPOVIĆ J.: Example-based control of human motion. In *Proc. SIGGRAPH/Eurographics Symposium on Computer Animation 2004* (2004), pp. 69–77.

[HPP05] HSU E., PULLI K., POPOVIĆ J.: Style translation for human motion. *ACM Trans. on Graphics 24*, 3 (2005), 1082–1089.

[Kai67] KAILATH T.: The divergence and Bhattacharyya distance measures in signal selection. *IEEE Trans. on Communication Technology COM-15* (1967), 52–60.

[KGP02] KOVAR L., GLEICHER M., PIGHIN F.: Motion graphs. *ACM Trans. on Graphics 21*, 3 (2002), 473–482.

[KPS03] KIM T., PARK S. I., SHIN S. Y.: Rhythmic-motion synthesis based on motion-beat analysis. *ACM Trans. on Graphics 22*, 3 (2003), 392–401.

[LC00] LOGAN B., CHU S.: Music summarization using key phrases. In *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing* (2000).

[LCR*02] LEE J., CHAI J., REITSMA P. S. A., HODGINS J. K., POLLARD N. S.: Interactive control of avatars animated with human motion data. *ACM Trans. on Graphics 21*, 3 (2002), 491–500.

[LS99] LEE J., SHIN S. Y.: A hierarchical approach to interactive motion editing for human-like figures. In *Proc. ACM SIGGRAPH 99* (1999), pp. 39–48.

[LU60] LABAN R., ULLMANN L.: *Mastery of Movement*. Princeton Book Company Publishers, 1960.

[LWS02] LI Y., WANG T., SHUM H.-Y.: Motion texture: a two-level statistical model for character motion synthesis. *ACM Trans. on Graphics 21*, 3 (2002), 465–472.

[LZ03] LU L., ZHANG H.-J.: Automated extraction of music snippets. In *Proc. ACM Multimedia* (2003), pp. 140–147.

[MRC05] MÜLLER M., RÖDER T., CLAUSEN M.: Efficient content-based retrieval of motion capture data. *ACM Trans. on Graphics 24*, 3 (2005), 677–685.

[NMS02] NAKATA T., MORI T., SATO T.: Analysis of impression of robot bodily expression. *Journal of Robotics and Mechatoronics 14*, 1 (2002), 27–36.

[NT04] NAVA G. P., TANAKA H.: Finding music beats and tempo by using an image processing technique. In *Proc. Int'l Conf. on Information Technology for Application* (2004).

[PB02] PULLEN K., BREGLER C.: Motion capture assisted animation: Texturing and synthesis. *ACM Trans. on Graphics 21*, 3 (2002), 501–508.

[Roa96] ROADS C.: *The Computer Music Tutorial*. The MIT Press, 1996.

[Ros92] ROSENTHAL D.: *Machine Rhythm: Computer Emulation of Human Rhythm Perception*. PhD thesis, Massachusetts Institute of Technology, 1992.

[Sch98] SCHERIER E. D.: Tempo and beat analysis of acoustic musical signals. *Journal of the Acoustic Society of America 103*, 1 (1998), 588–601.

[SDO*04] STONE M., DECARLO D., OH I., RODRIGUEZ C., STERE A., LEES A., BREGLER C.: Speaking with hands: Creating animated conversational characters from recordings of human performance. *ACM Trans. on Graphics 23*, 3 (2004), 506–513.

[SMS05] SETHARES W. A., MORRIS R. D., SETHARES J. C.: Beat tracking of musical performance using low-level audio features. *IEEE Trans. on Speech and Audio Processing 13*, 2 (2005).

[SNI04] SHIRATORI T., NAKAZAWA A., IKEUCHI K.: Detecting dance motion structure through music analysis. In *Proc. IEEE Int'l Conf. on Automatic Face and Gesture Recognition* (2004), pp. 857–862.

[SXWS04] SHAO X., XU C., WANG Y., SKANKANHALLI M.: Automatic music summarization in compressed domain. In *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing* (2004).

[Tod94] TODD N. P. M.: The auditory primal sketch: A multiscale model of rhythmic group. *Journal of New Music Research 23*, 1 (1994), 25–70.

[WLZ04] WANG M., LU L., ZHANG H.-J.: Repeating pattern discovery from acoustic musical signals. In *Proc. IEEE Int'l Conf. on Multimedia and Expo* (2004), pp. 2019–2022.

[WP95] WITKIN A., POPOVIĆ Z.: Motion warping. In *Proc. ACM SIGGRAPH 95* (1995), pp. 105–108.

[YKH04] YAMANE K., KUFFNER J., HODGINS J. K.: Synthesizing animations of human manipulation tasks. *ACM Trans. on Graphics 23*, 3 (2004), 532–539.