J. COMBIN. MATH. COMBIN. COMPUT. 127b (2025) 3771--3789

Journal of Combinatorial Mathematics and Combinatorial Computing
www.combinatorialpress.com/jcmcc

A Modeling Study of Music and Dance Rhythm Matching Using Time Series Analysis

Peipei Liu[1,✉]

[1] *School of Art and Design (Center for Public Art Education), Henan University of Urban Construction, Pingdingshan, Henan, 467002, China*

ABSTRACT
Rhythm matching of music and dance is an important research area in cross-modal analysis. In this paper, a music and dance rhythm matching algorithm based on time series analysis is proposed to extract the time series features of music and dance, and a genetic algorithm is used to determine the correspondence between music and dance movements to reflect the degree of correlation between changes in music and dance rhythm movements. In order to improve the matching and smoothing degree between the dance movement time series and the music time series, a constraint-based dynamic programming algorithm is introduced. The experimental results show that the model performs well in the matching degree and matching efficiency enhancement between dance movement time series and music time series, and its matching efficiency is 2-3 times of the traditional method. It shows high practicality in dance choreography and music matching, and can match any music clip with smooth and beautiful dance movements. The research in this paper provides new technical means for dance choreography and music matching, which will further optimize the transition harmony between music time series and dance movement time series.

Keywords: time series features, dynamic programming algorithm, rhythm matching algorithm, match enhancement

1. Introduction

Any kind of dance, any dancer must be according to the rhythm of the music to dance movement creation, music is like the oxygen of the dance creation environment, all the time reflects its existence. The combination of music and dance has enriched the expression of musical art, and people's various aesthetic needs are satisfied in the splendid and colorful comprehensive art after the combination of the two [1-4].

✉ Corresponding author.
E-mail address: hnlpp812@163.com (P. Liu).
Received 05 August 2024; Revised 25 October 2024; Accepted 15 February 2025; Published Online 16 April 2025.
DOI: 10.61091/jcmcc127b-210

Dance needs music to stimulate emotion and action to strengthen, leaving music, dance is difficult to fully express feelings. Music is the art of sound, is to sound for the performance of the physical form, its main means of expression is melody, rhythm and harmony, while the dance is mainly physical movement for its performance of the physical form, the main means of expression is the dance, dance and modeling [5-8]. Music is the art of time, space and hearing, and dance is the art of space and vision. From the point of view of the combination of dance and music, rhythm is the common ground that exists between dance and music, and rhythm is the basic factor that constitutes the melody of music as well as dance movements [9-11]. Dance as a comprehensive art, the role played by music in it is obvious. Dance belongs to the category of visual, tangible and silent, while music belongs to the category of auditory art, sound and intangible, the two can be organically combined [12-13]. In the art of dance performance, beautiful music with, can help the dance in the whole performance process to express the mood, reflect the personality and atmosphere, so as to shape the perfect dance image [14-16]. Music in dance is an integral part of completing the artistic image and revealing its thematic ideas in dance works, while the emotional changes in dance performance are reflected through the melody, rhythm, speed and intensity of music [17-20].

Literature [21] constructed a model for matching dance movements with music rhythmic features in order to improve the degree of matching between dance movements and music rhythms, revealing that the model has good matching accuracy and significant advantages. Literature [22] proposed a music-driven movement synthesis framework that generates movements controlled by the musical content and is able to produce effective natural movements in a variety of dance genres and respects the overall structure of the dance. Literature [23] proposed two models, timbre similarity and rhythmic similarity, which extract feature values from audio clips and verified that the accuracy of the rhythmic similarity model is very high, while the accuracy of the timbre model is medium. Literature [24] examined the effect of metrical matching between music and dance and the kinematic familiarity of dance movements on the audiovisual integration of rhythm, revealing that the mechanisms of rhythm perception and movement simulation are able to modulate the temporal integration of this audiovisual stimuli. Literature [25] developed a VR application (DRF) that combines frame-by-frame rendering with motion trajectory visualization; the DRF program enabled users to significantly improve choreographic skills and rhythmic accuracy compared to a video-based VR system. Literature [26] analyzed the effect of musical rhythmic complexity on dance performance accuracy and examined the effect of varying degrees of auditory syncopation on the execution of dance sequences by dancers and non-dancers, revealing that the pattern of differences between dancers and non-dancers was similar, a result that has a greater impact on practitioners. Literature [27] introduced MuDaR, a music-dance representational learning framework, and synchronized music and dance rhythms explicitly and implicitly, verifying that the framework largely outperforms other self-supervised methods.

This paper proposes a music and dance rhythm matching method based on time series analysis, aiming to further improve the synchronization between music and dance through more accurate time series feature extraction and matching degree optimization algorithms. The action time series features and music time series features are extracted separately using time series analysis. A genetic algorithm with relationship accuracy as a function of fitness was introduced to construct the correspondence between music and dance movements. In order to optimize the search results, a dynamic programming algorithm is used to minimize the distance between the action time series features and the music time series features, and to complete the improvement of the matching degree between music and dance rhythm. Quantitative analysis experiments and visualization experiments are used to determine the relevant performance of the model in this paper.

2. Music and motion feature extraction and segmentation

Feature extraction and segmentation of music and dance movements is the starting point of music and dance rhythm matching modeling, and common features include spectrum, constant Q transform, intensity, time series features, etc. Feature extraction of dance movements involves the position of joints, movement trajectories and other information. These features provide actionable data for subsequent model training and matching.

2.1. Music feature extraction and music segmentation

In this section, we will introduce music spectrum analysis based on ConstantQ transform [28], music rhythm extraction based on note onset, music intensity extraction based on human ear auditory attributes and music segmentation based on note repetition structure and rhythm.

2.1.1. Music spectrum analysis. Music spectrum analysis provides the basis for music rhythm extraction, intensity extraction and music segmentation. The original music signal is a waveform signal in the time domain, but in order to extract richer information about the music, such as rhythm, intensity, etc., the music signal needs to be transformed from the time domain to the frequency domain and then analyzed.

In this paper, the Constant Q Transform (CQT) algorithm is used to make the music signal decomposed at the frequency points of the twelve mean laws, reflecting the frequency distribution law of music.

The CQT transform of the finite length music signal $x(n)$ is:

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

According to music theory, the frequency of the k st note is calculated as follows:

$$f_k=f_0\cdot 2^{k/b} \tag{2}$$

where f_0 is the minimum frequency, which corresponds to the frequency of the C3 note at 130.8 Hz, and b indicates the number of semitones contained in an octave. a constant factor Q in the CQT transform indicates that the frequency to bandwidth ratio is a constant value:

$$Q=\frac{f_k}{f_{k+1}-f_k}=\frac{1}{2^{1/b}-1} \tag{3}$$

Window size N_k:

$$N_k=\left\lfloor Q\frac{f_s}{f_k}\right\rfloor \tag{4}$$

After the above CQT transformation, the CQT spectrum of the music signal is obtained. In the subsequent sections, $X(n,k)$ denotes the spectral energy of the kth note of the nth frame of the signal.

2.1.2. Music Rhythm Extraction. Music beat extraction is based on the following two principles:

1) Musical beats are periodic in nature.

2) The period of musical beats is equal to the interval of musical start notes.

The input audio signal $x(n)$ is transformed by CQT to obtain the CQT spectrum $X\left(n,k\right)$. Using the distance between adjacent frames of the same frequency, and denoising optimization at the same time, the calculation formula can be obtained as follows:

$$d\left(n,k\right)=\begin{cases}\max(X(n,k),X(n+1,k))-\Pr ePow(n,k)\\(\min(X(n,k),X(n+1,k))\geq \Pr ePow(n,k)\\0(otherwise)\end{cases} \tag{5}$$

The final value of the mutation point function, $D(n)$, is obtained by summing the increments of all frequencies at the current moment:

$$D(n) = \sum_k d(n,k) \tag{6}$$

Then, beat period prediction is performed. Based on the cycle characteristics of musical beats, an autocorrelation function is defined to predict the beat length:

$$AC(\tau) = \sum_{n=1}^{len/2} D(n) * D(n+\tau), \tau \in \left(0, \frac{len}{2}\right) \tag{7}$$

Finally, the beat point evaluation is performed. Assuming that the position of the current beat is T_n, the position of the next beat can be predicted based on the beat cycle:

$$T'_{n+1} = T_n + \tau_{\max} \tag{8}$$

There is a small error between the predicted value T'_{n+1} and the true rhythm point T_{n+1} due to the error in the actual performance. In this paper, the moment near the prediction value T'_{n+1} such that $D(n)$ is a local maximum is taken as the rhythm point.

From the above, the rhythmic characteristics of music clip M_i are defined as follows:

$$F_R^{Music}(f) = \begin{cases} 1 & \begin{array}{l} if \text{ f is the moment of} \\ \text{the beat of the assessment}, f \in M_i \end{array} \\ 0 & otherwise \end{cases} \tag{9}$$

2.1.3. Music intensity extraction:

The extraction of musical intensity is based on the following three principles;

1) Spectral energy increases with increasing tone intensity.

2) Musical melodies use frequencies above C4.

3) The human ear readily recognizes sounds with frequencies much higher than adjacent frequencies in the spectrum.

Based on this principle, the transformed CQT spectrum is utilized to first find the average energy of the k nd note of music clip M_i:

$$\bar{X}(k) = \frac{1}{|M_i|} \sum_{n \in M_i} X(n,k) \tag{10}$$

Calculate the local peak of the average energy:

$$X_{peak}(k) = \begin{cases} \bar{X}(k), \bar{X}(k) \geq \bar{X}(k \mp 1) \\ 0, \; otherwise \end{cases} \tag{11}$$

Considering the auditory properties of the human ear and the frequency of the signal, a musical intensity signature is obtained:

$$F_I^{Music}(f) = \log_{10}\left(\sum_{k=C4}^{c6} X_{peak}(k)^2 \cdot f_k^2\right), f \in M_i \tag{12}$$

where f_k is the frequency value corresponding to C4-C6.

2.1.4. Music segmentation. Two approaches to music segmentation are considered, one approach is based on the principle that music is structured by the repetition of several musical segments. Therefore, the goal of music segmentation is to extract patterns of repeating music segments and then segment the music based on the repeating patterns. The other approach is to simply take music segments of 16 rhythmic lengths.

In this paper, a music segmentation algorithm with 16 rhythmic lengths is used. The music segmentation algorithm that takes 16 beat lengths is mainly used to segment music uniformly where the melody is not strictly repetitive. When segmenting music, it is segmented at intervals of $16\tau_{\max}$ according to the beat length $\tau_{\max}$ obtained from the rhythm extraction analysis. For the audio Antrikos2 with a duration of 35.2s, according to the corresponding beat length of 0.2s, 11 music segments can be obtained, with an average length of 3.2s. It can be seen that the music segments obtained by this segmentation method are more uniform and not too short, which is conducive to the subsequent matching of features and the synthesis of dance articulation.

2.1.5. Music Time Series Extraction. The database in this paper consists of a collection of musical melody fragments. The input search fragment will be matched with every melody fragment in the database. In general, subsequence retrieval is slower than whole sequence retrieval because the number of possible candidate sequences is much larger for subsequence retrieval. For files, a melodic fragment consists of a sequence of binary groups (notes, durations). Because we are using a monophonic melody, there is only one note at a moment in time. We use a sequence of binaries to represent a piece of musical melody that begins with a note of duration 1, followed by a note of duration 2, and so on. A binary sequence can be converted into a time sequence.

To obtain segments of music, this paper uses a standard speech file streaming class library to store music segments in wav format. This acoustic input is segmented into frames of 10ms length, and then each meal is converted to a pitch using a pitch extraction algorithm. This melodic fragment is then processed in the same steps as a database file. This creates a time series of pitches. Since there is no highly accurate algorithm for splitting notes, the time series method avoids note slicing. Note that this does not include information about the rests of the melody.

2.2. Dance feature extraction and dance segmentation

Since the feature matching of music and dance rhythmic movements is based on the common features of music and movements - rhythm and intensity, the rhythmic features and intensity features of movements need to be extracted. In order to fully mine the features at a finer granularity and at the same time facilitate the reuse of movement data, the movements are segmented. In this paper, all features are defined on the scale of action/music segments, which is the basic data unit for model analysis.

The action sequence N is segmented to obtain multiple action segments $N_1, N_2, \cdots$. All subsequent references to action segment $N_i(1 \le i \le n)$ refer to action segments in action sequence N, abbreviated as action segment N_i. The action features extracted from action segment N_i are denoted as:

$$Motion\ Feature(f) = \begin{bmatrix} F_R^{Motion}(f) \\ F_I^{Motion}(f) \end{bmatrix}, f \in N_i \tag{13}$$

where f is the frame number of action segment N_i, and F_R^{Motion} and F_I^{Motion} are the rhythm and intensity characteristics of the action segment, respectively.

2.2.1. Dance Rhythm Extraction. For heel-stepping beat type of dance, this paper proposes a rhythm extraction method based on foot joints and hip joints. The foot joint-based rhythm extraction considers the positions of the great and small values of the vertical displacement of the foot joint as the rhythm points.

From the above, the rhythm extraction algorithm based on foot vertical displacement threshold for action segment N_i is as follows:

$$F_{Rfoot}^{Motion}(f)=\begin{cases}1 & if\ z'_{foot}(f)=0, z''_{foot}(f)<0, z_{foot}(f)\geq \varepsilon_{\max} \\ & or\ if\ z_{foot}(f)=0, z_{foot}(f)>0, z_{foot}(f)\leq \varepsilon_{\min} \\ 0 & otherwise\end{cases} \tag{14}$$

The formula for rhythm extraction based on the vertical displacement of the arm is as follows:

$$F_{Rhip}^{Motion}(f)=\begin{cases}1 & ifz'_{hip}(f)=0, z''_{hip}(f)>0, z_{hip}(f)\leq \varepsilon_{\min 2} \\ 0 & otherwise\end{cases} \tag{15}$$

In summary, the rhythm extraction formula can be obtained by combining the rhythm extraction of the foot joints and arm joints:

$$F_{R}^{Motion}(f)=\begin{cases}1 & ifz'_{foot}(f)=0, z''_{foot}(f)<0, z_{foot}(f)\geq \varepsilon_{\max} \\ & orifz'_{foot}(f)=0, z''_{foot}(f)>0, z_{foot}(f)\leq \varepsilon_{\min} \\ & orifz'_{hip}(f)=0, z''_{hip}(f)>0, z_{hip}(f)\leq \varepsilon_{\min 2} \\ 0 & otherwise\end{cases} \tag{16}$$

2.2.2. Dance intensity extraction. For the dances used in this paper, the momentum depends mainly on the vertical movement of the feet and the horizontal movement of the center of the human body. The instantaneous intensity of the current movement segment N_i can be obtained as:

$$I(f)=v_{foot}(f)+k\cdot v_{root}(f) \tag{17}$$

By averaging the intensities of neighboring frames in the same rhythmic cycle in which the current frame is located, the intensity features of action segment N_i can be extracted as follows:

$$F_{I}^{Motion}(f)=\sum_{i=R_s^f}^{R_e^f}\frac{I(i)}{R_e^f-R_s^f+1} \tag{18}$$

Wherein, R_s^f denotes the start frame number of the same rhythmic cycle in which the action of the current frame f is located, and R_e^f denotes the end frame number of the same rhythmic cycle in which the action of the current frame f is located.

2.2.3. Dance segmentation. Dance movement segmentation segments movements based on rhythmic extraction of movements. In this paper, the action segmentation is segmented by every 16 rhythmic lengths. According to the music and action feature matching model proposed in this paper, when selecting action segments from the action database for matching the input music, the length of the action segment is required to be greater than or equal to that of the music segment, and then the part of the action segment that is equal to the length of the music is intercepted and matched with the music segment for feature matching, based on the fact that the matching of this action that is equal to the length of the music is better than that of the other actions that are equal to the length of the music. The reason is that the matching of the action with the music length is better than other actions with the music length. Therefore, the segmentation algorithm based on 16 rhythmic lengths in this paper can easily and effectively segment actions, and at the same time ensure that the length of the segmented action meets the music matching requirements.

2.2.4. Dance movement time series extraction. In this paper, the sequence of frames to frames before and after is used as temporal information, rather than points in time. That is, the focus here is on which pose is located in front of a pose and which pose is located behind a pose during behavior recognition, rather than at what moment a pose occurs and at what moment another pose occurs before or after that moment. The order of frames and frames on the timeline is important for behavior recognition, especially if the action poses are reversible. For example, the actions of standing up and sitting down are a reversible pair of poses on the time axis.

A comparison of the two actions reveals that, time aside, the postures on the crossed lines are similar. Thus the order of the postures in time can be important information for recognition. In order to preserve the order information of the time series, this paper achieves this by adding a regular term that controls the temporal order. This added regular term enables the sparse coefficients to reflect the temporal order of the original feature vectors, whereas the traditional sparse coefficients just select the feature base vectors that best express the content of the feature vectors themselves in a dictionary, and the closer the feature bases correspond to the original feature vectors, the larger the coefficients.

3. Determining music-dance correspondence

This paper proposes a method for determining music-movement correspondences for music and dance of a specific style, which first performs a correlation analysis of music-movement features and excludes some corresponding feature pairs, and then uses a genetic algorithm to optimally select correspondences that satisfy the matching accuracy and computing speed. When performing correlation analysis, in order to better reflect the influence of feature changes on music-movement matching, this paper uses the rate of change of feature values instead of the feature values themselves. When performing genetic algorithm training, in order to facilitate the calculation, this paper uses the correspondence accuracy as the fitness function of the genetic algorithm.

3.1. Correspondence

In music feature vector $M:\left(m_1,m_2,\ldots,m_p\right)$ and action feature vector $A:\left(a_1,a_2,\ldots,a_q\right)$, m_i and a_i correspond to a certain music feature and action feature respectively. For any m_i and a_i, if the change of m_i has an effect on a_j, then m_i and a_i are said to be correlated, denoted as $m_i \to a_j$. The set of correlated feature pairs is called a correspondence, i.e., $\left\{\left(m_1^f,a_1^f\right),\left(m_2^f,a_2^f\right),\ldots,\left(m_i^f,a_i^f\right)\right\}$.

3.2. Correlation analysis

3.2.1. Calculation of correlation coefficients. After extracting the music and action features, each music segment and action segment can be described using feature vectors. For example, the feature vectors of the i music segment M_i and the j action segment A_j are $M_i=\left(m_1,m_2,\ldots,m_p\right)$ and $A_j=\left(a_1,a_2,\ldots,a_q\right)$, respectively, where m_p denotes the p th music feature value and a_q denotes the q th action feature value. Then, the music-action segment training samples that satisfy the matching relationship can be represented in the form of $\left(M_i,A_j\right)$.

According to the music beat slicing algorithm, it can be seen that the music and action in the training samples are best matched by default, so we consider that the feature correlation coefficients calculated from the training samples are accurate. For music feature m_i and action feature a_j, the correlation coefficient $r\left(m_i,a_j\right)$ of both is calculated as:

$$r\left(m_i,a_j\right)=\frac{n\sum m_i a_j-\left(\sum m_i\right)\left(\sum a_j\right)}{\sqrt{n\sum\left(m_i\right)^2-\left(\sum m_i\right)^2}\sqrt{n\sum\left(a_j\right)^2-\left(\sum a_j\right)^2}} \tag{19}$$

3.2.2. Rate of change of eigenvalues. After the calculation of correlation coefficients for all music-movement features is completed, a matrix of correlation coefficients is obtained. A high music-movement feature correlation coefficient is manifested by a high synchronization of the change rates of the two eigenvalues.

In this paper, the formula for calculating the correlation coefficients of music feature m_i and action feature a_j is improved as follows:

$$r^{\Delta}\left(m_i, a_j\right) = \frac{n\sum \Delta m_i \Delta a_j - \left(\sum \Delta m_i\right)\left(\sum \Delta a_j\right)}{\sqrt{n\sum\left(\Delta m_i\right)^2 - \left(\sum \Delta m_i\right)^2}\sqrt{n\sum\left(\Delta a_j\right)^2 - \left(\sum \Delta a_j\right)^2}} \tag{20}$$

In the above equation, n indicates the number of training samples. The value of $r^{\Delta}\left(m_i, a_j\right)$ ranges from $[-1,1]$, $r^{\Delta}\left(m_i, a_j\right) > 0$ means positive correlation, $r^{\Delta}\left(m_i, a_j\right) < 0$ means negative correlation, $r^{\Delta}\left(m_i, a_j\right) = 0$ means no correlation, and the larger absolute value of $r^{\Delta}\left(m_i, a_j\right)$ means higher correlation.

For features that can be calculated within one frame, such as fundamental frequency, frequency flow, attitude characteristics, etc., the calculation method of the eigenvalue change rate Δm, Δa is to compare the difference between the features of frame 7 and frame 1 with the feature value of frame 1, and the calculation formula is as follows:

$$\Delta m = \frac{m_i(\alpha,7) - m_i(\alpha,1)}{m_i(\alpha,1)} \tag{21}$$

$$\Delta a = \frac{a_j(\beta,7) - a_j(\beta,1)}{a_j(\beta,1)} \tag{22}$$

where $m_i(\alpha,k)$ represents the i th feature in the k rd frame of the α nd music clip and $a_j(\beta,k)$ represents the j th feature in the k th frame of the β th action clip.

For the feature values such as velocity, acceleration and direction change that need to be calculated between different frames, the calculation method is to calculate the mean value of the features in the first 4 frames and the last 4 frames respectively, and then calculate the rate of change. The formula is as follows:

$$\Delta m = \frac{\left(m_i(\alpha,7) - m_i(\alpha,4)\right) - \left(m_i(\alpha,4) - m_i(\alpha,1)\right)}{\left(m_i(\alpha,4) - m_i(\alpha,1)\right)} \tag{23}$$

$$\Delta a = \frac{\left(a_j(\beta,7) - a_j(\beta,4)\right) - \left(a_j(\beta,4) - a_j(\beta,1)\right)}{a_j(\beta,4) - a_j(\beta,1)} \tag{24}$$

3.3. Correspondence optimization

After calculating the correlation coefficients of music and movement, selecting which features are used to construct music-movement correspondences is the key to this paper. It is not reasonable to construct music-movement correspondences based solely on the size of the correlation coefficient, and the correspondences constructed from the music and movement features with the highest correlation coefficients do not have the highest correspondence accuracy. Moreover, when music-movement matching is performed, if the correspondence containing all features is used for the matching calculation, the computation amount is huge and the computation speed is low.

Another reason for optimization is that music-movement correspondences in different styles of music-dance contain different pairs of features.

3.3.1. Genetic algorithms. From the optimization point of view, constructing music-action correspondences is actually a combinatorial optimization problem. The methods to solve this problem include traversal search, random search and heuristic search. Genetic algorithm has a wide range of applications in combinatorial optimization problems and belongs to a stochastic search method. For the selection of complex feature sets, genetic algorithms are superior search algorithms for feature selection because of their global implicit parallelism and ease of jumping out of local extremes.

A complete genetic algorithm [29] mainly includes the following steps: gene coding, population initialization, selection operation, crossover operation, mutation operation, end condition judgment, etc.

Genetic Coding. Genetic coding is the coding of features with 0 or 1, with 0 representing unselected and 1 representing selected. The correspondence between the composition of the selected features is represented as an $\{0,1\}$ binary string.

Population initialization. Considering that in the action generation stage, the feature pairs included in the correspondence for music-action matching are 10 appropriate. So after analyzing the correlation of music-action features, this paper selects 40 feature pairs with the highest correlation as the feature pairs to be selected to train the correspondence for music-action matching.

Population evolution includes three ways: selection operation, crossover operation, and mutation operation. The population evolution strategy adopted in this paper is: adaptive crossover and adaptive mutation operations are performed on each population, and the 10 individuals with the highest matching accuracy are selected to form a new population after the iteration is completed. The adaptive crossover and adaptive mutation operations are performed iteratively in the new population until the end condition is satisfied. The end condition of the genetic algorithm can be judged according to the evaluation function or the number of iterations.

After the gene encoding and population initialization, operations are performed on the individuals of the population until the end condition is met and the feature selection operation is stopped.

All correspondences $\left(M^f, A^f\right)$ are trained by a genetic algorithm to obtain an optimal music-movement correspondence $T:\left(M^{ff}, A^{ff}\right)$. Correspondence $\left(M^{ff}, A^{ff}\right)$ is used in the automatic dance movement generation phase for music-movement matching computation. Where M^{ff} is $\left(m_1^{ff}, m_2^{ff}, \ldots, m_n^{ff}\right)$, A^{ff} is $\left(a_1^{ff}, a_2^{ff}, \cdots, a_n^{ff}\right)$, and M^{ff} is in one-to-one correspondence with the features in A^{ff}.

3.3.2. Adaptation function. The design of the fitness function [30] is the key of the genetic algorithm, in order to facilitate the calculation as a starting point, this paper adopts the correspondence accuracy $m\_accuracy$ as the evaluation function, i.e. the fitness function. The higher the correspondence accuracy indicates that T has a greater influence on the degree of music and dance matching.

The matching criterion used in this paper is: if the difference between the correlation value $D_{\cos}\left(M_{i,T^f}^f, A_{i,T^f}^f\right)$ calculated by using the new music-movement correlation T^f for the i rd music clip Mi and the i th movement clip Ai and the correlation value $D_{\cos}\left(M_{i,T}, A_{i,T}\right)$ calculated by using the initial music-movement correlation T is no more than 10%, then the correspondence is considered to be able to accurately match the music clip and the movement clip. If the music and dance movement segments obtained from the correspondence are considered to be accurately matched only when the correlation values are exactly the same, then the correspondence accuracy will be lower due to the influence of data errors. The formula for calculating the correspondence accuracy is as follows:

$$m\_accuracy = \frac{1}{n}\sum_{n} f_{i,T} \tag{25}$$

$$f_{i,T} = \begin{cases} 1 & if \left| \dfrac{D_{\cos}\left(M^{f}_{i,T^{f}}, A^{f}_{i,T^{f}}\right)}{D_{\cos}\left(M_{i,T}, A_{i,T}\right)} - 1 \right| \leq 0.1 \\ 0 & other \end{cases} \tag{26}$$

A correspondence is a collection of music, action feature pairs, each consisting of music features and action features. The correspondence to be analyzed is $\left(M^{f}, A^{f}\right)$, where music feature vector M^{f} is $\left(m_1^f, m_2^f, \ldots, m_k^f\right)$ and action feature vector A^{f} is $\left(a_1^f, a_2^f, \ldots, a_k^f\right)$. Once the sequences of music and action feature values are represented as vectors, respectively, the degree of match between the music and action segments can be obtained by vector computation methods. There are many methods to calculate the degree of matching, and in general, the classical vector cosine distance can solve the above problem well. The calculation formula is as follows:

$$D_{\cos}\left(M_{\alpha,T}, A_{\beta,T}\right) = \frac{\sum_{N} \Delta m_i(\alpha) \Delta a_j(\beta)}{\sqrt{\sum_{N}\left(\Delta m_i(\alpha)\right)^2 \times \sum_{N}\left(\Delta a_j(\beta)\right)^2}} \tag{27}$$

In the above equation, $\Delta m_i(\alpha)$ denotes the rate of change of the i rd eigenvalue of the α nd music clip, $\Delta a_j(\beta)$ denotes the rate of change of the j th eigenvalue of the β th action clip, and N is the number of eigenpairs.

4. Music-driven dance movement matching model

In this chapter, a constraint-based dynamic planning process is introduced in order to synthesize the dance movement sequence that best matches the input music. The process considers both the degree of match between the dance movements a music and the degree of smoothing of the resulting movement sequence.

4.1. Model preprocessing

In order to prepare the pipeline for dance movement synthesis, we first construct a movement graph that contains all the candidate movement segments. We use the action graph construction algorithm proposed by Zhao and Safonova to construct the action graph, because the action graph obtained using this algorithm has better coherence. In order to obtain candidate actions for action graph construction, those action data equipped with background music are sliced into segments based on the music tempo; while action segments without background music are sliced by the action tempo analysis algorithm. Each action segment in the action graph is labeled with information about the dance genre to which it belongs. The additional information recorded on these nodes allows for a complete traversal of the action graph or only the nodes belonging to a specific dance genre. This feature is introduced because our model assumes that the action-music relationship is dance-genre dependent, and because dance-genre based node selection and filtering is a practical way to design. Note that action nodes in our action graph do not have to carry matching background music. This relaxed constraint effectively increases the source of candidate action segments, since in practice there exists a large amount of action data that does not come with background music.

4.2. Music-driven action synthesis objective function

In order to synthesize matching actions for the input music clips, we consider two objectives: the quality of the action-music match and the smoothness of the preceding and following action clips. Relatively, the objective function used to guide action synthesis is divided into two parts, each corresponding to one of the above objectives.

Given a piece of music data input to the model, which is sliced into n music clip $\{A_1,\ldots,A_n\}$ according to the tempo, our goal is to find an optimized sequence of action clips $\{M_{u_1},\ldots,M_{u_n}\}$, where M_{u_1} is the action piece clip corresponding to A_i, the following minimization objective function is obtained:

$$F_{(n)} \square \sum_{j=1}^{n} R\left(j,u_j\right)+\gamma\sum_{j=2}^{n} T\left(u_{j-1},u_j\right) \tag{28}$$

In the above equation, $R\left(j,u_j\right)$ is a simplified representation of $S\left(A_j,M_{uj}\right)$, which measures the matching error between music clip A_i and action clip M_{u_j}, while $T\left(u_{j-1},u_j\right)$ is the smoothness score for migration from action clip $M_{u_{j-1}}$ to clip M_{u_j}, which can be obtained by querying the action graph. γ is used to balance the weight between the two. This parameter can either be manually adjusted by the user or set based on the model default value. In our experiments, we set its default to 0.6.

4.3. Objective function optimization based on dynamic programming

Assume that there are a total of w candidate movement segments in the target dance category. The input music contains n music clips. In order to find the optimal sequence of movements, a brute force global search would result in a search in a space of up to w^n. In order to achieve optimal search results while avoiding such a huge amount of computation, we instead introduce the following dynamic programming [31] framework to synthesize the resultant action sequence with the goal of minimizing the objective function.

Assume that $F_{\min}\left(i\right)$ is the minimum value of the objective function $F(i)$ conditional on when the music segments from 1 to i have been matched by i action segments. We introduce the auxiliary function $H(i,j)$ to represent the minimum of the objective function conditional on the fact that the music segments from 1 to i have been matched by i action segments and the last current action segment is M_j. We can now define the following iterative formula to find $F_{\min}(n)$:

$$F_{\min}(n) = \min\{H(n,j) \mid j = 1,\ldots,w\} \tag{29}$$

$$H(1,i) = R(1,i)(i = 1,\cdots,w) \tag{30}$$

$$\begin{gathered} H(j+1,i) = R(j+1,i) + \min_{t:M_t \in V(i,\varepsilon)}(H(j,t)+\gamma T(t,i)) \\ \left(j = 1,\ldots,n-1; i = 1,\ldots,w\right) \end{gathered} \tag{31}$$

In the above equation, $V(i,\varepsilon)$ denotes the set of action segments with smoothing degree scores less than ε from M_i. We restrict the value of t to this set instead of $1,\cdots,w$, thus reducing unnecessary search branches. In addition, reinforcing these constraints ensures the coherence of the entire action sequence, and in general, this dynamic planning process yields optimized action synthesis results. The time complexity of the process is $0(n\times w\times e)$, where e is the average size of the set of marquee action segments. It can be seen that the time complexity of the dynamic programming algorithm without additional constraints attached will be $0\left(n\times w^2\right)$, which is obviously a large number, especially when the candidate action fragments, i.e., w, are relatively large.

Once the optimal action sequence $\{M_{u_1},\ldots,M_{u_n}\}$ has been determined, we are able to synthesize an entire coherent action for the input music by joining these segments together. Afterwards, we perform two operations to further improve the visualization of the synthesized action sequence. First, for each action segment A_{u_1} whose length corresponds to that of its music segment A_{u_1}, we scale the action time sequence segment to the same length as the music time sequence segment. After that, the action fusion technique is applied to make the transition between neighboring action segments smoother.

4.4. General framework of the model

The music-driven dance movement matching model based on time series described in this paper contains two phases of data training and movement generation its algorithmic flow is shown in Fig. 1.

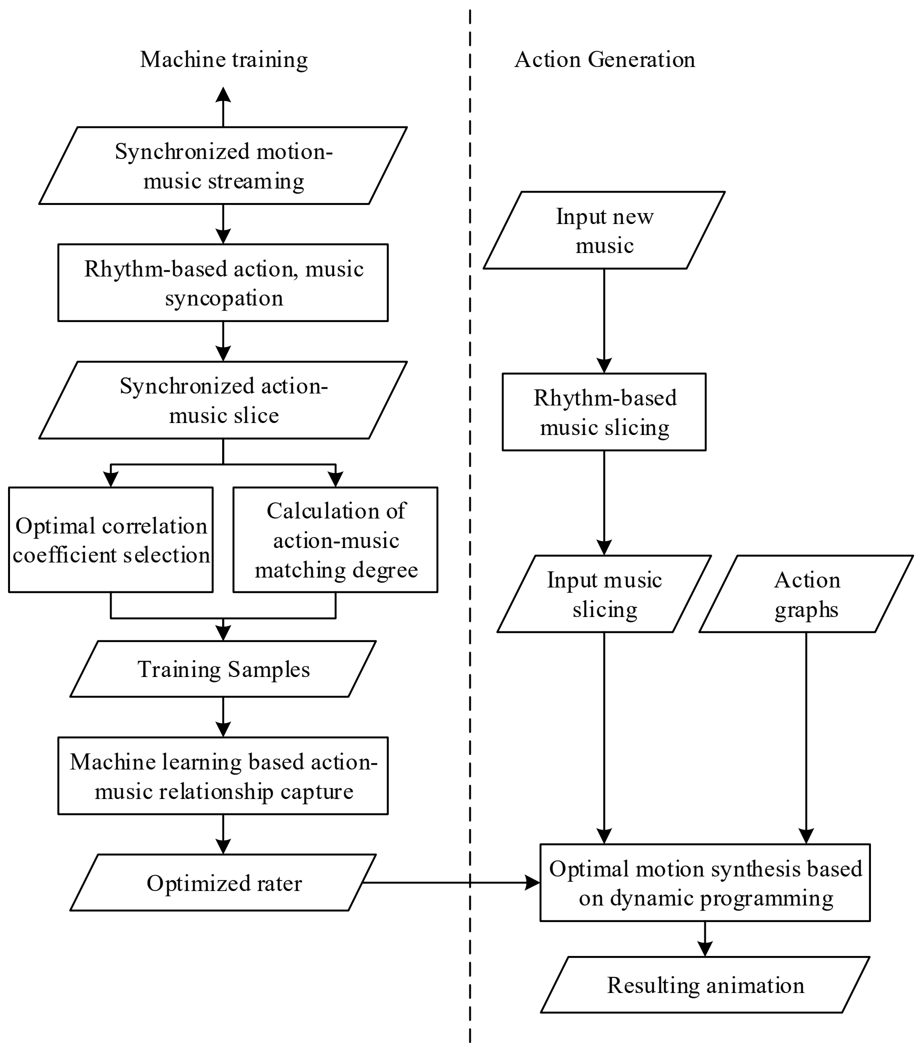

Fig. 1. System algorithm process

4.5. DTW distance

Dynamic time-bending distance has been extensively studied in the field of speech processing and introduced to the field of data mining. Unlike the Euclidean distance, the dynamic time warping (DTW) distance does not require one-to-one matching of points between two time series, allowing the sequence points to self-replicate before alignment matching. When a time series undergoes time axis bending, self-replication can be performed on the bending portion, allowing similar waveforms between two time series to be aligned and matched.

Because the DTW distance defines the optimal alignment matching relationship between two sequences, it supports the time axis bending well, and it can perform the similarity metric between two non-equal-length sequences, so it is also supported for time axis stretching. Because of the characteristics of the time series itself, the two deformations of time axis bending and stretching are prevalent, and they cannot be eliminated by general normalization or preprocessing methods, so DTW distance is widely adopted in the similarity metric of time series. In this paper, DTW distance is adopted as a matching metric for music and dance rhythms.

5. Analysis of dance rhythm matching results

5.1. Assessment of the quality of matching in the same dance category

In practice, the action map portion and the music map portion of the action-music mapping graph are mostly composed of the same dance species, so the matching quality of the resultant animations created by the model when the action map and the music map are constructed from the same dance species is first evaluated. Taking the Korean dance action maps and Korean dance music maps as an example, this paper evaluates them using 2 scoring methods:.

1) Using the DTW algorithm to calculate the distance between the music and dance rhythms in the resultant animation as a score for the degree of matching merit.

2) In order to test the audience's feelings about the matching quality of the resultant animation, 10 graduate students with certain knowledge of music theory were invited to manually score the matching quality of the dance rhythm and music in the resultant animation.

In the DTW scoring system, the higher the score, the lower the degree of matching. In the manual scoring system, the degree of matching was categorized into 5 levels of scores from 1 to 5, with higher scores indicating a higher degree of matching. Each user is asked to rate the resultant animation according to its visual and auditory smoothness. The model was fed with 5 music sequences for testing the choreography and 5 movement sequences for testing the soundtrack. The DTW scores and manual scoring values for the test data are given in Table 1, where 1, 2 and 3 in the horizontal table numbers represent the matching scores obtained by random traversal, the matching scores obtained by model traversal, and the matching scores obtained by model traversal with editing and adjustment, respectively. As can be seen from Table 1, according to the 2 scoring methods, the matching quality of the results generated by the model has a significant advantage over random traversal, and the degree of matching can be substantially improved after editing the dance tempo or music result data. In addition, the relationship between manual scoring and DTW scoring shows a strong positive correlation. When the number of feature points is 1, the manual scoring increases from 1.4 to 4.2, which represents that the degree of matching is improved after model editing, and the DTW scoring also decreases from 38.97 to 4.36, which represents that the degree of matching is substantially improved, which is in line with the results of manual scoring. This also shows that the DTW algorithm does effectively reflect the user's perception of the degree of matching between the music time series segments and the dance rhythm time series segments, and that the matching effect of the segments matched by the model in this paper is better than that of random traversal.

Table 1. The choreography of Korean dance and the quality of the score match

	Serial number	Characteristic point	DTW score			Manual scoring		
			1	2	3	1	2	3
Dance	1	24	38.97	18.94	4.36	1.4	2.8	4.2
	2	25	36.56	18.78	4.21	1.4	2.8	4.3
	3	26	41.53	16.95	3.92	1.3	3.2	4.6
	4	27	41.22	18.29	4.46	1.2	2.4	3.8
	5	28	39.45	15.89	3.58	1.2	3.4	4.4
Music	6	22	29.81	10.28	2.98	1.4	2.8	4.2
	7	21	25.46	9.78	2.16	1.3	3.1	4.7
	8	20	29.54	11.64	2.97	1.3	3.1	4.4
	9	18	29.54	12.89	2.44	1.3	2.8	4.2
	10	19	28.23	12.92	2.83	1.3	2.8	4.6

5.2. *Assessment of the quality of matching across dance genres*

When the dance rhythm action figure part and the music figure part are composed of different dance genres, the difference between the dance action bars and the music bars due to the style may lead to a change in the performance of the algorithm, and this paper also evaluates the quality of the matching for this case, and the results of the cross-dance genres matching quality evaluation are shown in Table 2. For each given combination of dance rhythmic movement charts and music charts, five Korean dance music sequences were input for testing the choreography and five Korean dance movement sequences for testing the soundtrack according to the methodology of this paper. The difference between the horizontal numbers 2 and 3 in Table 2 and the corresponding items in Table 1 is that the items in Table 2 are the average scores of the 5 resultant animations. As can be seen from Table 2, the DTW scoring results do not change significantly when the action and music diagrams consist of different dance styles, which indicates that the different dance styles do not affect the matching process of the DTW algorithm. However, the manual scoring results decreased, and the feature matching degree of dance rhythm decreased from 3.2 to 2.9, which indicates that the stylistic difference between dance rhythm and music has a certain negative impact on the audience's perception. How to better deal with the stylistic differences between dance rhythm and music is also an aspect of the rhythm matching model proposed in this paper that deserves further in-depth research.

Table 2. Cross-dancing, music and music match result quality

	Database		Method			
	Action chart	Music chart	DTW		Manual	
			2	3	2	3
Dance	Korean	Uygur	17.76	4.08	3.2	4.3
	Uygur	Uygur	16.57	3.88	3.6	4.6
	Korean	Uygur	17.87	4.36	2.8	3.9
	Uygur	Korean	18.31	4.59	2.9	4.0
Music	Korean	Uygur	11.53	2.69	2.8	4.4
	Uygur	Uygur	10.23	2.46	3.4	4.7
	Korean	Uygur	12.34	2.92	2.6	3.9
	Uygur	Korean	13.39	3.05	2.7	4.0

5.3. Time performance evaluation

The experimental data used in Section 5.1 is used as an example for the time performance review, and this test compares the time performance difference between the traditional method and the method in this paper. The time performance differences are shown in Table 3. In the traditional method, for each action or music bar, it is necessary to traverse the music or action graph to find a best matching music node for it. Whereas in this paper's method only the best time series matching node needs to be queried. As can be seen from Table 3, the matching times of this paper's methods are all smaller than the traditional methods, which are 2-3 times more efficient than the traditional methods, and the search speed of this paper's methods is significantly higher than that of the traditional search methods.

Table 3. System choreography and score time efficiency

	Serial number	Characteristic point	Traditional method	This method
Dance	1	24	37.5	13.84
	2	25	39.56	14.45
	3	26	43.86	15.16
	4	27	41.95	15.43
	5	28	43.69	15.69
Music	6	22	29.81	10.12
	7	21	26.19	9.29
	8	20	28.09	10.06
	9	18	27.98	10.48
	10	19	29.56	10.98

5.4. Long Dance Sequence Matching Effect

In order to verify that the dynamic programming-based dance action matching model in this paper has good performance in terms of long dance sequences, this section plots the FID curves of each time period in a long dance sequence. The specific implementation process is as follows: firstly, a piece of music with a duration of one minute is randomly selected from the YouTube-Dance3D test set and input to the corresponding dance sequence in the matching algorithm model and other algorithm models in this chapter. Then each dance sequence is evenly divided into 15 segments, and the duration of each segment is 4 seconds. Finally, the FID between each dance segment and the real dance is calculated.The experimental results are shown in Fig. 2.The algorithms in this chapter and the DanceRevolution algorithm have a slow-growing and smoother FID curve compared to the other algorithms. Among other methods, TSMT and Without Scheduled Sampling algorithms have steeper FID curves with significant fluctuations. This is due to the accumulation of errors in long sequences, which leads to a lower confidence in the predictions of the model at the beginning, and therefore a lower quality of the dances generated by matching. Although the FID curve of the DanceRevolution algorithm model is also smoother, the FID curve is higher than the other algorithms, and the FID curve of this paper's algorithm is lower than the other algorithms, with an average FID value of 35 and the overall smoothest. The quality of the dances generated by matching is optimal.

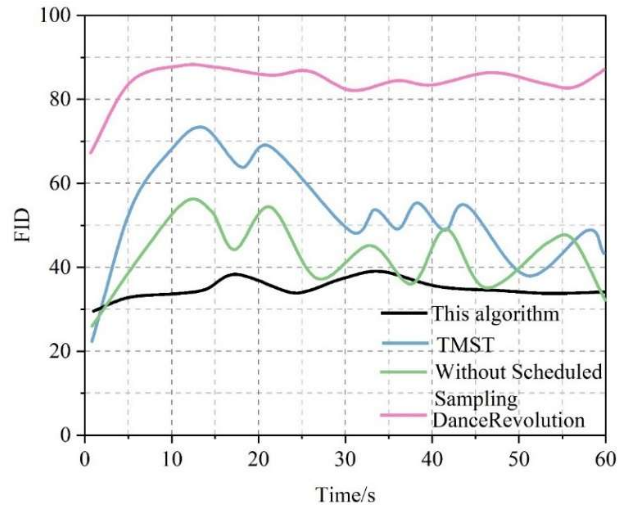

Fig. 2. FID curves of different approaches over time

6. Visualization of the results of matching music and dance rhythms

In this section, the 3D character animation software Motionbuilder is firstly used to demonstrate the 3D visualization effect of the music and dance movement matching segment generated by this model, and then the choreography result of this model is compared with the choreography result based on FCRBM-RNN algorithm, and the dance rhythm matching result of this model is evaluated according to the visual effect of the synthesized dance.

A modern dance choreography segment matched by this model is selected for comparison with it. A segment of target music in the style of modern dance outside the training set is selected, and the music choreography is performed on it using this model, and the choreography results are shown in Fig. 3, and Fig. 4 shows that the target music has appeared in the training set. Observing the video of the dance clip and the snapshots of the action poses, it can be seen that in terms of visual effect, the choreography result of this paper's model has better authenticity and coherence, whether it is the trained music or the newly appeared music clip, this paper's dance tempo matching model can present a better effect of matching the music and dance tempo to complete the choreography task.

The FCRBM-RNN algorithm has mixed results in matching choreography moves to target music using trained models. When the target music is outdated in the training set, more realistic and coherent movements can be generated, and the choreography results are shown in Figure 5. When the target music is not outdated in the training set, it is almost impossible to generate a watchable dance, and the choreography results are shown in Figure 6. It can be seen that for a brand new piece of music, the generated dance movements are extremely discontinuous, and some of the movements obviously do not conform to the physics of the human skeletal structure, which are impossible for human beings to perform, i.e., they are less coherent and realistic. For a music choreography model built based on time series analysis, the music in the test set used to test the model results should not appear in the training set, so although the choreography results shown in Fig. 5 are more effective, the choreography results in Fig. 6 are poor in realism and coherence, indicating that the method of the FCRBM-RNN algorithm is not practical, and it is not possible to generate the ideal choreography results for a brand new piece of music. In summary, from an intuitive visual perspective, for a brand new piece of target music, the music choreography model in this paper can generate more realistic and coherent dance movements, while the authenticity and coherence of the dance movements matched by the FCRBM-RNN algorithm approach cannot meet the general dance requirements.

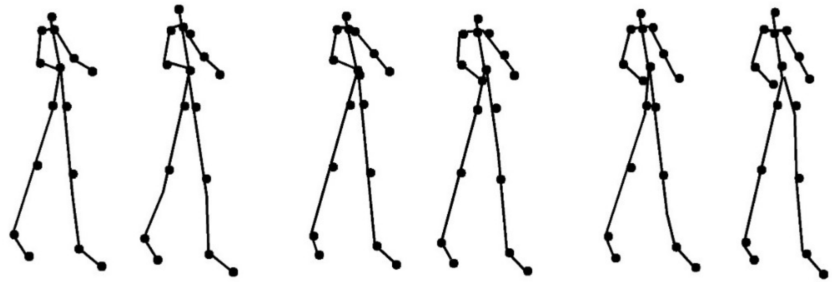

Fig. 3. Training the target music of the outside style

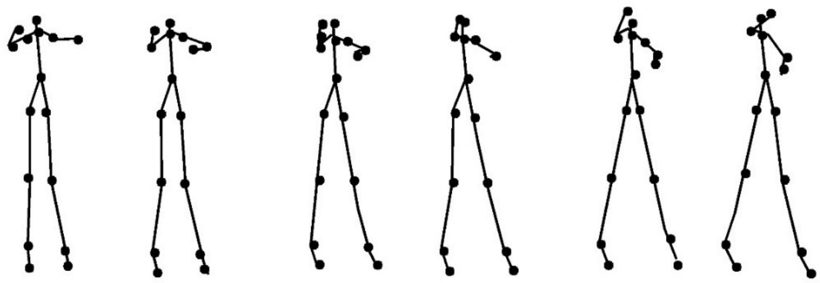

Fig. 4. The goal music of the training set

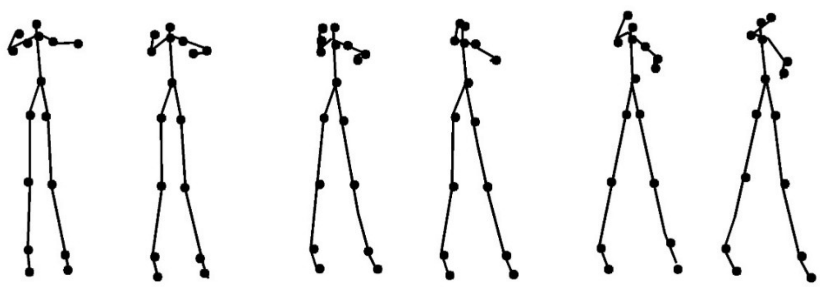

Fig. 5. Training the target music of the outside style

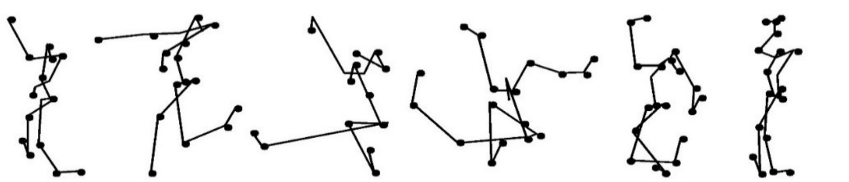

Fig. 6. Training the target music of the outside style

7. Conclusion

This paper proposes a modeling method for matching music and dance rhythms based on time series analysis, which achieves efficient matching of music and dance rhythms through the steps of feature extraction, correlation analysis of music movement features, and dynamic time regularization.

Through time series analysis, the time series features of music rhythm and dance movements can be accurately extracted, which provides a solid data foundation for matching music and dance rhythm.

The dynamic programming algorithm plays a key role in this paper, minimizing the time difference between the music time sequence segments and the action time sequence segments, dynamically adjusting the time sequence of the dance action to achieve an accurate match between music and dance rhythm, the matching effect is better than random traversal, and the search time is lower than that of the traditional method, 1/3-1/2 of the traditional method search time.

The model constructed in this study is applicable to a variety of music styles and dance types, and can match dance movements to randomly generated music clips, and the generated dance movements have better fluency and realism, which is more widely used than the FCRBM-RNN algorithm, and can be adapted according to specific needs for different application scenarios. It provides new technical support for the creation of music and dance, and is expected to be applied in the fields of music production, dance choreography, and human-computer interaction.

References

[1] Hamilton, A. (2020). Rhythm and Movement: The Conceptual Interdependence of Music, Dance and Poetry. Midwest Studies in Philosophy, 44(1), 161-182.

[2] Xiu, Y., Liu, X., Yuan, Y., Zhao, H., & Zhang, C. (2023). A new method of dance rhythm training based on an immersive cave automatic virtual environment. IEEE Access.

[3] Breska, A., & Deouell, L. Y. (2017). Dance to the rhythm, cautiously: Isolating unique indicators of oscillatory entrainment. PLoS biology, 15(9), e2003534.

[4] Brown, S. (2024). The performing arts combined: the triad of music, dance, and narrative. Frontiers in Psychology, 15, 1344354.

[5] Charnavel, I. (2023). Moving to the rhythm of spring: a case study of the rhythmic structure of dance. Linguistics and Philosophy, 46(4), 799-838.

[6] Levitin, D. J., Grahn, J. A., & London, J. (2018). The psychology of music: Rhythm and movement. Annual review of psychology, 69(1), 51-75.

[7] Moore, P. A., & Moore, P. A. (2019). The Rhythm of the Dance. Into the Illusive World: An Exploration of Animals' Perception, 207-212.

[8] Zhuang, W., Wang, C., Chai, J., Wang, Y., Shao, M., & Xia, S. (2022). Music2dance: Dancenet for music-driven dance generation. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 18(2), 1-21.

[9] Guo, X., Zhao, Y., & Li, J. (2021). DanceIt: music-inspired dancing video synthesis. IEEE Transactions on Image Processing, 30, 5559-5572.

[10] Kong, S. (2024). A deep learning model of dance generation for young children based on music rhythm and beat. Concurrency and Computation: Practice and Experience, 36(13), e8053.

[11] Bresnahan, A. (2019). Dance rhythm (pp. 91-98). Oxford: Oxford University Press.

[12] Davis, A., & Agrawala, M. (2018). Visual rhythm and beat. ACM Transactions on Graphics (TOG), 37(4), 1-11.

[13] Loo, F. C., Loo, F. Y., & Chua, Y. P. (2019). Congruence in music and movement enhances the perception of sports routine quality. Revista Música Hodie, 19.

[14] Santos, A. D. P. D., Tang, L. M., Loke, L., & Martinez-Maldonado, R. (2018, June). You are off the beat! is accelerometer data enough for measuring dance rhythm?. In Proceedings of the 5th International Conference on Movement and Computing (pp. 1-8).

[15] Su, Y. H., & Keller, P. E. (2020). Your move or mine? Music training and kinematic compatibility modulate synchronization with self-versus other-generated dance movement. Psychological Research, 84(1), 62-80.

[16] Han, B., Li, Y., Shen, Y., Ren, Y., & Han, F. (2024). Dance2MIDI: Dance-driven multi-instrument music generation. Computational Visual Media, 10(4), 791-802.

[17] Tang, T., Jia, J., & Mao, H. (2018, October). Dance with melody: An lstm-autoencoder approach to music-oriented dance synthesis. In Proceedings of the 26th ACM international conference on Multimedia (pp. 1598-1606).

[18] Svobodová, L., Skotáková, A., Hedbávný, P., Vaculíková, P., & Sebera, M. (2016). Use of the dance pad for the development of rhythmic abilities. Science of Gymnastics Journal, 8(3), 283-293.

[19] Toussaint, G. T. (2019). The geometry of musical rhythm: what makes a" good" rhythm good?. CRC Press.

[20] Ye, Z., Wu, H., Jia, J., Bu, Y., Chen, W., Meng, F., & Wang, Y. (2020, October). Choreonet: Towards music to dance synthesis with choreographic action unit. In Proceedings of the 28th ACM International Conference on Multimedia (pp. 744-752).

[21] Jiang, D. (2022). Matching Model of Dance Movements and Music Rhythm Features Using Human Posture Estimation. Computational Intelligence and Neuroscience, 2022(1), 7331210.

[22] Aristidou, A., Yiannakidis, A., Aberman, K., Cohen-Or, D., Shamir, A., & Chrysanthou, Y. (2022). Rhythm is a dancer: Music-driven motion synthesis with global structure. IEEE Transactions on Visualization and Computer Graphics, 29(8), 3519-3534.

[23] Panteli, M., Rocha, B., Bogaards, N., & Honingh, A. (2017). A model for rhythm and timbre similarity in electronic dance music. Musicae Scientiae, 21(3), 338-361.

[24] Su, Y. H. (2018). Metrical congruency and kinematic familiarity facilitate temporal binding between musical and dance rhythms. Psychonomic Bulletin & Review, 25(4), 1416-1422.

[25] Tokida, S., & Ishiguro, Y. (2024, May). Dance with Rhythmic Frames: Improving Dancing Skills by Frame-by-Frame Presentation. In Proceedings of the 9th International Conference on Movement and Computing (pp. 1-6).

[26] Karageorghis, C. I., Lyne, L. P., Bigliassi, M., & Vuust, P. (2019). Effects of auditory rhythm on movement accuracy in dance performance. Human movement science, 67, 102511.

[27] Yu, J., Pu, J., Cheng, Y., Feng, R., & Shan, Y. (2023). Learning Music-Dance Representations through Explicit-Implicit Rhythm Synchronization. IEEE Transactions on Multimedia.

[28] Salim Shinimol & Ahmad Waquar. (2023). Constant Q Cepstral Coefficients for Automatic Speaker Verification System for Dysarthria Patients. Circuits, Systems, and Signal Processing(2),1101-1118.

[29] Romero-Tello Pablo,Lorente-López Antonio José & Gutiérrez-Romero José Enrique. (2025). Structural design optimization of pressure hull using genetic algorithm and finite element analysis. International Journal of Structural Integrity(1),60-84.

[30] Haiyan Wang. (2025). Several fitness functions and entanglement gates in quantum kernel generation. Quantum Machine Intelligence(1),7-7.

[31] Kunyu Wang,Hao Song,Zhiqiang Guo & Xuemin Zhang. (2025). Automated multi-dimensional dynamic planning algorithm for solving energy management problems in fuel cell electric vehicles. Energy134408-134408.