IEICE TRANS. FUNDAMENTALS, VOL.E107–A, NO.5 MAY 2024

PAPER

Dance-Conditioned Artistic Music Generation by Creative-GAN

Jiang HUANG[†a)], **Xianglin HUANG**[†], **Lifang YANG**[†], ***and*** **Zhulin TAO**[†], ***Nonmembers***

SUMMARY We present a novel adversarial, end-to-end framework based on Creative-GAN to generate artistic music conditioned on dance videos. Our proposed framework takes the visual and motion posture data as input, and then adopts a quantized vector as the audio representation to generate complex music corresponding to input. However, the GAN algorithm just imitate and reproduce works what humans have created, instead of generating something new and creative. Therefore, we newly introduce Creative-GAN, which extends the original GAN framework to two discriminators, one is to determine whether it is real music, and the other is to classify music style. The paper shows that our proposed Creative-GAN can generate novel and interesting music which is not found in the training dataset. To evaluate our model, a comprehensive evaluation scheme is introduced to make subjective and objective evaluation. Compared with the advanced methods, our experimental results performs better in measureing the music rhythm, generation diversity, dance-music correlation and overall quality of generated music.

key words: *dance video, vector quantized, music generation, genre, creative-GAN*

1. Introduction

The famous Russian dance director Zakharov said: "Music is the soul of dance, and dance is the echo of music". It is meaningful to explore the multimodal generation task on automatically generating personalized and creative music from dance. Almost given any dance, our target is to generate a coherent, rich and varied music work. With dance-music technology, users can share creative videos on short-video social media platforms such as YouTube and TikTok. Music generation task based on dance videos will also have several application scenarios in sports and fitness environment. Although some progress has been made, generating music from dance videos is still a challenging task for various technical reasons as follows:

1. It is ambiguous to map between music and dance. Music signals are high-dimensional, and a CD-quality music work owns more than tens of thousands of data points per second. It is a tough task to choose some intermediate audio representation to form an effective audio-visual correlation mapping between low-dimensional motion data and high-dimensional audio data.

2. Compared to the dance video in reality, the generated music underperforms in harmony and richness. Symbolic-based music generation tasks are not very flexible, we observe that the generated music is composed with acoustic sounds from a single instrument of a specific type. What's more, the generated music is equally difficult to keep coherent with different types and beats of dance videos.

3. Music generation is a quite creative and artistic process. If the model simply remembers intrinsic modes of the training data and learns to copy them, we will inevitably fail to create something novel and original, which is impossible to generate music with multiple styles. To deal with the cross-modal mapping problem, we prefer an improved Generative Adversarial Network (GAN) [1]. Similar to the method of VQVAE [2], the generator outputs quantized vectors as intermediate audio representations. Compared to the continuous original audio signal and the classical symbolic representation, the quantized representation can increase the abstraction ability and improve the mapping level to better represent complex real-world music. Vector quantization is performed in the embedding space called music memory, where a finite dictionary of quantized music units is made. We encode and quantize music samples to a codebook in an unsupervised manner. Every prelearned code represent a unique music clip, including basic and reusable musical structural components.

In order to make the generated music more diverse and artistic, it is proposed to add the second discriminator to GAN framework. We call the new framework, Creative-GAN, which consists of two discriminators. One efficiently captures the temporal correlations and rhythms of music to generate complex music. The other classifies the generated music genre as one of the established genres in the training dataset. By this way, the generator is motivated to generate new style of music.

In the complete process of dance to music, we extract visual and motion posture features from dance videos, and then perform features fusion. The generator outputs the music vector as an intermediate audio representation, and then the music vector embeds to the corresponding quantized feature by means of the pretrained music memory codebook. Finally, we adopt a dedicated learned VAE-based decoder for generating music.

In summary, our main contributions are:

1. We propose a novel adversarial, end-to-end framework that takes dance video as input and learns to generate complex and diverse music via quantized feature representations.

Manuscript received May 22, 2023.
Manuscript revised July 19, 2023.
Manuscript publicized August 23, 2023.

[†]The authors are with State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing 100024, China.
a) E-mail: huangjiang@cuc.edu.cn

DOI: 10.1587/transfun.2023EAP1059

2. A musical memory codebook is created to encode and quantize musical samples by VQ-VAE, which is pre-trained on a large-scale musical dataset.
3. We newly introduce Creative-GAN, which extends the original GAN framework to two discriminators, to focus on real/fake and musical style respectively in the stage of generating artistic music.
4. To evaluate our model, we present a comprehensive evaluation scheme to make subjective and objective evaluation. Compared with existing methods, our proposed model shows better performance on the complexity and variety of generated music.

2. Related Work

2.1 Music Generation

Music generation has been a hot research direction in recent years. Most solutions are based on deep learning models and MIDI datasets, which could be divided into several models: recurrent neural networks (RNN, LSTM, GRU), variational encoders (MuseNet [3], MusicVAE [4]), generative adversarial networks (MuseGAN [5], MidiNet [6]). There are some studies attempting to generate music by training on raw audio waveforms. WaveNet [7] has a perfect performance on generating speech and music.

With the development of attention mechanisms, music generation tasks based on transformer have shown exciting results in generating sound and music. Huang et al. [8] came up with a music transformer model to generate piano music from MIDI event. Recently, Dhariwal et al. [9] proposed a large-scale generative model, named jukebox, to generate music with VQ-VAE. The VQ representation has good flexibility and can express complex musical styles with a unified codebook.

2.2 Cross-Modal Learning

The cross-modal learning in vision and sound fields is a hot research topic. Due to the low dimension of visual data, mapping from sound to motion is relatively easy to realize. Specifically, LSTM model based on autoencoders is used to learn the mapping of music-to-dance [10]. Tendulkar et al. [11] put forward a search-based method to synthesis dance from music, which makes the total spatio-temporal movement pattern match the entire structure of music well. Audio data has poor structure and high dimensionality, so it is hard to align with visual information. Recently, there are few work to demonstrate the strong connection between body movement and audio by jointly learning the visual and audio representation using a visual-audio correspondence task [12].

2.3 Music to Dance

Several researches have focused on music to dance synthesis. Early works usually treat this problem as a mechanical template matching problem. Cardle et al. [13] modified dance motion according to musical features, while Lee et al. [14] manually defined muscial features and generate dance motions according to musical similarity. In recent years, with the emergence of deep learning, many works design a dedicated network structure, including CNNs [15], RNNs [10], GANs [16] and Transformers [17], to map the given music to a joint sequence of the continuous human pose space directly.

2.4 Dance to Music

Dance is a stage performance form that is artistically created by human movements, postures, gestures, and facial expressions. A large amount of existing work is to generate audio from motion with 2D poses or 3D motion. Zhao et al. [18] introduced an end-to-end model to generate sound from motion tracks with a curriculum learning scheme. Di et al. [19] propose the Controllable Music Transformer (CMT) model, which generates music based on the rhythm features of the input video and user-defined style features.

The vector quantized generative model VQ-VAE has demonstrated its feasibility in various generative tasks, such as image and audio synthesis [20]. Specifically, the VQ-VAE [2] is initially tested for generating images, videos, and speech. An improved version of VQ-VAE [21] is proposed with a multi-scale hierarchical organization. Esser et al. [22] apply the VQ representations in the GAN-based framework for generating high resolution images. We take the high abstract representation capability of vector quantization as intermediate audio representation to effectively capture the temporal relevance, rhythm, and genres of music. The dance-to-music problem is a unique challenge, on one hand, a variety of music can be composed by the same dance; on the other hand, the same music can also be used for a variety of choreography.

2.5 GAN and Creativity

GAN [1] is a deep learning model based on adversarial game. The generator outputs fake samples that satisfy real samples distribution to the greatest extent, while the discriminator network tries its best to identify true and false samples. However, the GAN algorithm can't create something new, it just remembers established styles in the dataset and learns to replicate them. If our model can only copy the training dataset, the result will be immutable and may also violate the original music's copyright.

To control what to generate and guide the generation process, Mirza [23] proposed a music genre-conditioned GAN architecture to generate genre-specific rhythmic patterns. Elgammal et al. [24] trained GAN model with a historical and diverse paintings dataset, and generated paintings which do not belong to any genres in the dataset. In order to generate pluralistic dance movement with multiple dance types conditioned on music, Jinwoo Kim [25] newly introduces MNET, a new transformer-based GAN model to

convert one dance type into target dance type and support multiple target dance types.

3. Approach

In this section, a new dance to music method is proposed. An overview of the music generation framework is shown in Fig. 1. The entire model architecture is divided into three modules: feature fusion module, Creative-GAN module and music synthesis module. In Sect. 3.1 we introduce dance and music representations. We do not directly learn a mapping from motion features to music clips. Instead, we construct a Music VQ-VAE generative model in Sect. 3.2, with quantizing music samples into a finite codebook $Z = \{z_i\}_{i=0}^{N-1}$ as musical memory, where N is the codebook length, and each code z_i is shown as a musical unit associated with context. The Creative-GAN framework is introduce in Sect. 3.3 to generate the music vector as an intermediate audio representation, and then adopt our newly designed two discriminators for real/fake and musical style learning. Finally, artistic music is generated by a dedicated VAE decoder.

3.1 Dance and Music Representations

Assuming that there is only one dancer, and we express the dance as a series of combinations of postures, gestures, and facial expressions.

3.1.1 Dance Representation

a) visual feature
We adopt the Temporal Segment Network (TSN) [26] model to obtain visual feature, sparsely sample the video into segments, then extract a snippet from each segment with a convolutional neural network to get spatial features. This structure uses a dual-stream ConvNets network and gives the model the ability to capture long-term information and significantly improves the accuracy of action recognition.
b) motion posture feature
We use a 3D model, named SMPL-X, to compute body posture, hand posture and facial expression from a single frame of RGB image [27]. SMPL-X applies standard vertex-based linear blend skinning to correct blend shapes, which can accurately represent different shapes and poses of the human body. Then motion posture encoder is designed as convolution-based feedforward networks, takes body motion representation as input and extracts motion posture features.

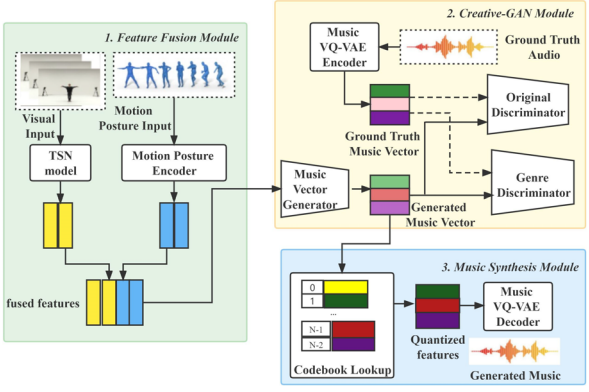


Fig. 1 Overview of the proposed music generation framework. We take the visual and motion posture data as input, and utilize TSN and motion posture encoder to extract features respectively. The generator takes fused features as input and generates music vector as audio representations. Music vector is calibrated by discriminator network, including an original discriminator and a genre discriminator. Afterwards, we adopt music vector to lookup in the pre-learned codebook to quantize vectors. Finally, the quantized features are decoded into generated music through a VQ-VAE decoder.

3.1.2 Music Representation

Considering the powerful representation ability of quantized vectors, we utilize quantized vectors as audio representations to decode into music.

3.2 Music VQ-VAE

Our proposed Music VQ-VAE adopts vector quantized as audio representation in discrete latent variable space, and then decodes into music by a dedicated VAE decoder. The whole quantization process is finished by looking up the music memory codebook, each code in codebook represents a music unit, which is not only a meaningful architecture, but also a basic component of music. The process of music generation can be considered as the combination and connection of musical units.

Our target is to gather up these musical units into a plentiful, reusable codebook. After encoding and quantifying meaningful music samples into the codebook, the quantized features are decoded to reconstruct the target music effectively. In the stage of training, the code in music memory is continuously updated. The structure of Music VQ-VAE is shown in Fig. 2. At first, we adopt a one-dimensional temporal convolution network, named music encoder, to encode music samples into context-based features in a latent variable space, and then each latent variable feature e_i is replaced by its closest codebook element z_j to quantize e.

$$e_{q,i} = \arg\min_{z_j \in Z} \left\| e_i - z_j \right\|_2 \tag{1}$$

The Music VQ-9VAE has 2 million parameters and is trained on 2-second audio clips on GPU. The dimension of

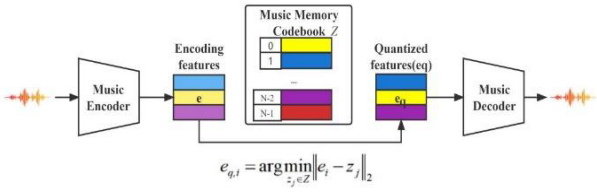


Fig. 2 Structure of Music VQ-VAE. It is learned to encode and gather up significant music units to music memory, and generate the target music clip from quantized features. Musical memory covers patterns of sound, score, melodic harmony and rhythmic structure.

the codebook is 64, which is the same as the generated music vector sequences. The music encoder and decoder are learned with the codebook at the same time by VQ-VAE loss functions:

$$L_{VQ-VAE} = L_{rec}\left(\hat{M}, M\right) + \left\|sg[e] - e_q\right\| + \beta\left\|e - sg[e_q]\right\| \quad (2)$$

$$L_{rec}\left(\hat{M}, M\right) = \left\|\hat{M} - M\right\|_1 \quad (3)$$

L_{rec} is reconstruction loss which is defined as the L1 distance between the generated music M and ground truth $\hat{M}$. The second part is the "codebook loss" for learning codebook entries and sg[·] denotes "stop gradient calculation", meaning that gradients cannot be back-propagated, which makes the vector of the codebook and the output of encoder as close as possible. And the last part is the "commitment loss" with trade off β. Since the quantization operation of Formula 2 is not differentable, to train the whole networks end to end, the back-propagation of this operation is achieved by simply passing the gradient of e_q to e. After training, each quantized feature is decoded as a unique musical clip. The music generation task turns into a process of selecting and ranking quantized features from the learned music memory.

3.3 Creative-GAN for Music Generation

GAN only learns the basic patterns of the training dataset and reproduce them without anything new. Therefore, a new approach of Creative-GAN is proposed. We adopt generated music vectors as input to the original discriminator (D_r) and genre discriminator (D_c) respectively for calibration. As shown in Fig. 3, the generator receives two feedbacks from the discriminator, one comes from the original discriminator's classification of "real or fake", the other is a feedback about whether the genre discriminator could classify the generated music into given styles.

3.3.1 Music Vector Generator

The music vector generator takes visual features, motion posture features as input, and outputs the music vector as audio representations. Therefore, the music generation task is redefined as the generation problem of music vector, and the discriminator network also works in the vector space.

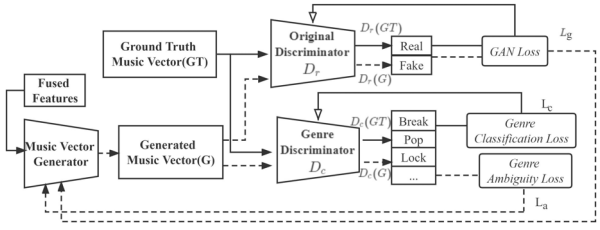


Fig. 3 Block diagram of the Creative-GAN Network. Music Vector Generator takes fused feature as input and output Generated Music Vector (G). The original discriminator distinguishes the generated music from the ground truth, and the genre discriminator is trained to classify the generated music as one of the existing styles in the training data.

The Music vector generator is a convolution based feedforward network. We utilize leaky rectified activation functions in hidden layers, and make use of a tanh activation to improve training stability in last layer. The convolution layer in the generator structure needs to be designed with a relatively large kernel size, thus residual blocks with dilations are added after the conv layer. For a two-second audio sequence with a sampling rate of 22050 Hz, the generated music vector sequences from low-level generators are in dimension of 64 × 1378, where 64 is the dimension of the codebook entry, 1378 are the sequence lengths.

3.3.2 Discriminator Network

The discriminator network includes an original discriminator and a genre discriminator. There are two classification tasks, one is to determine whether it is real music, and the other is to classify music style. We train the generator not only to fool the discriminator to believe that the generated music is a real sample extracted from the training data, but also to confuse the genre discriminator uncertain about the type of patterns generated, which makes classification results as equiprobable as possible.

a) Original discriminator

The original discriminator utilizes an architecture called PatchGAN, which is designed in the form of full convolution. After several convolutional layers, it will not be feeded to the fully connected layer or activation function, but use convolution to map the input into an K*K matrix. We measure the entire piece of music by means of an K*K matrix. Therefore, the discriminator can capture the long-term dependencies of music like the generator.

In the stage of training, the discriminator actually operates the generated music vector and ground truth music vector togther. We import the ground truth music into the pre-trained VAE encoder to get ground truth music vector. The discriminator is designed as a structure to discriminate blocks individually, and then the distribution of each music vector block is classified to determine whether it is true or not through a node with a sigmoid activation function. Finally, we average the output to enhance overall consistency.

b) Genre discriminator

We extend the GAN framework by adding a genre discriminator to categorize the styles of generated music. During the training process, the genre discriminator can access lots of musical works with style labels, and the generator is reversely trained to confuse the discriminator about the style of generated music. The genre discriminator consists of two layers of bidirectional LSTM, and shares the same input with the original discriminator, outputing with K nodes by Softmax activation and categorical cross-entropy loss. We add the genre classification loss and the genre ambiguity loss to the cost function, the latter quantifies the ambiguity of genre classification. If the classification is too confident, there will be a penalty. If the output of the classification model is as equal probability as possible, the loss is minimal.

3.4 Training Objectives

In general, the model is trained with an adversarial loss and some additional losses.

a) Adversarial Loss
The training of GAN is in a game process, the discriminator needs to maximize the loss, while the generator wants to minimize the loss. An original adversarial loss is defined as follows:

$$\begin{aligned} \mathrm{L_g} &= \mathrm{E}_{\phi(x_{GT})}\left[\log D_r\left(x_{\mathrm{GT}}\right)\right] \\ &\quad + \mathrm{E}_{(x_v,x_{mp})}\left[\log\left(1-\mathrm{D_r}\left(G\left(x_v,x_{mp}\right)\right)\right)\right] \end{aligned} \tag{4}$$

where $\mathrm{x_v}$ represents visual input, and $\mathrm{x_{mp}}$ represents the motion posture, GT is the original wave music, ϕ means the encode process of pretrained encoder. $\mathrm{x_{GT}}$ is a real music, $\mathrm{D_r}(\cdot)$ is the transformation function that tries to discriminate between ground truth and generated music.

We modified the GAN loss function by adding a style classification loss ($\mathrm{L_c}$) and a style ambiguity loss ($\mathrm{L_a}$). $\hat{c}$ is the style label of real music, $\mathrm{D_c}(\cdot)$ is the function that differentiate between different style categories.

$$\mathrm{L_c} = \mathrm{E}_{\phi(x_{GT})}\left[\log D_c\left(c=\hat{c}\,|x_{GT}\right)\right] \tag{5}$$

$$\begin{aligned} \mathrm{L_a} = \mathrm{E}_{(x_v,x_{mp})}\Bigg[&-\sum_{k=1}^{K}\left(\frac{1}{K}\log\left(D_c\left(c_k\left|G\left(x_v,x_{mp}\right)\right.\right)\right)\right. \\ &\left.+\left(1-\frac{1}{K}\right)\log\left(1-D_g\left(c_k\left|G\left(x_v,x_{mp}\right)\right.\right)\right)\right)\Bigg] \end{aligned} \tag{6}$$

We add the probability of correct classification $\mathrm{D}_c\left(c=\hat{c}\,|x\right)$ to the loss function. In order to fuse style with the data generated by the generator as much as possible, the generator is trained by multi label cross entropy loss with K number of classes. The above two losses correspond to art-style classification and style ambiguity separately. Thus, the style loss can have the following definitions:

$$\begin{aligned} \mathrm{L_{sty}} = L_c + L_a = \mathrm{E}_{\phi(x_{GT})}&\left[\log D_c\left(c=\hat{c}\,\middle|x_{gt}\right)\right] + \\ \mathrm{E}_{(x_v,x_{mp})}\Bigg[&-\sum_{\mathrm{k=1}}^{\mathrm{K}}\left(\frac{1}{K}\log\left(D_c\left(c_k\left|G\left(x_v,x_{mp}\right)\right.\right)\right)\right. \\ &\left.+\left(1-\frac{1}{K}\right)\log\left(1-D_g\left(c_k\left|G\left(x_v,x_{mp}\right)\right.\right)\right)\right)\Bigg] \end{aligned} \tag{7}$$

Consequently, the new adversarial loss can be computed by $\mathrm{L_{adv}} = L_{\mathrm{g}} + L_{sty}$.
b) Feature Matching Loss
A feature matching loss [28] is applied to stabilize the discriminator during training process, calculating the L1 distance between the ground truth VQ features and generated music VQ features.

$$\mathrm{L}_{FM} = \mathrm{E}_{(x_v,x_{mp})}\sum_{i=1}^{T}\left\|D^i\left(\phi\left(x_{GT}\right)\right)-D^i\left(G\left(x_v,x_{mp}\right)\right)\right\|_1 \tag{8}$$

where T and $\mathrm{D^i}$ respectively represents the number of layers and the ith layer of D.
c) Codebook Commitment Loss
We define the codebook commitment loss as the L1 distance between the generated music VQ features and the corresponding codebook entries of the ground truth VQ features after performing the process of codebook lookup.

$$\mathrm{L}_{code} = \mathrm{E}_{(x_v,x_{mp})}\left\|Lookup\left(\phi\left(x_{\mathrm{GT}}\right)-G\left(x_v,x_{mp}\right)\right)\right\|_1 \tag{9}$$

d) Perceptual Loss
In order to develop the realism of the synthetic audio, we use audio classification networks to capture different audio's semantic characteristics. In general, we leverage a pre-trained SoundNet [29], fix its parameters and apply it to encode audio features for both real audio m and generated audio $\hat{\mathrm{m}}$ during training. The perceptual loss is then defined as the L1 distance between real audio features and generated audio features:

$$L_p = \left\|\mathrm{d}\left(\varphi\left(\hat{m}\right)-\varphi(m)\right)\right\|_1 \tag{10}$$

where $\varphi(.)$ means the feedforward feature extraction process of SoundNet, d(.) is a smooth L1 regression loss function proposed by Faster RCNN [30] to calculate the distance of loss, which is shown as below.

$$\mathrm{d}(x) = \begin{cases} 0.5x^2, |x| < 1 \\ |x| - 0.5, |x| \geq 1 \end{cases} \tag{11}$$

x means the numerical difference between predicted and true values. Compared to L1 loss, smooth L1 Loss improves the zero point non-smoothness problem.
e) Full Objective
Our full objective is

$$\arg\min_{G}\max_{D} L_{adv} + w_{FM}L_{FM(G;D)} + w_c L_{code} + w_p L_p \tag{12}$$

where $\mathrm{w_{FM}}$, $\mathrm{w_c}$ and $\mathrm{w_p}$ represent hyperparameters for every loss part.

4. Experiments

4.1 Experimental Setup

4.1.1 Datasets

We use AIST++ dataset [31] to train our model, AIST++ is a massive 3D human dance motion dataset which includes various 3D motion combined with music. It is shot by professional dancers in a clean studio setting with no obstructions. Human motion data is provided in the form of SMPL-X parameters and body key points, which are annotated with different types and styles. The dataset contains 1408 sequences, 30 subjects, over 18k seconds motion data and 10 types of different dance motions accompanied by music. We conduct experiments and evaluations on AIST++ dataset to prove the effectiveness and robustness of our framework.

Table 1 3D motion datasets comparisons.

Dataset	Music	3D Joint Rotation	Genres	Sequences	Seconds
AMASS[32]	-	√	-	11265	145251
Human3.6M[33]	-	√	-	210	71561
GrooveNet[34]	√	-	1	2	1380
DanceNet[35]	√	-	2	2	3472
AIST++	√	√	10	1408	18694

As shown in Table 1, there is a detailed comparison between our AIST++ dataset and other 3D motion datasets. The results show that AIST++ has a comprehensive performance in evaluation metrics, which contains the most genres. Other metrics such as sequences and seconds also rank top 3, accompanying by music and rich motion types.

4.1.2 Implementation Details

In our experiments, the dataset has a total of 1050 data for training, validation, and testing. The number of videos in each split is 990, 30, and 30, respectively. The training set is used for model training, the validation set is used to adjust parameters, and the test set is used to measure the quality of the final model. We use a sampling rate of 22050 Hz for all audio data, and set the batch size to 16. We adopt 2 seconds length video and audio clips to train and test. While training the Music VQ-VAE, we follow a hierarchical structure with two levels for the audio signals, the hop lengths for high and low level are 128 and 32. Meanwhile, we adopt two independently pretrained codebooks for two level corresponding to hop length. The low level model has smaller hop-length, which enables the generation of music with higher fidelity and better quality. While the high level model has fewer parameters, resulting in faster inference. We adopt Adam optimizer with $\beta 1 = 0.5$ and $\beta 2 = 0.9$ to train Music VQ-VAE with learning rate 1e-5. The music VQ-VAE and the music vector generator are trained in turn, and the weights of VQ-VAE are fixed when generator is in the process of learning. After training, we adopt pre-trained music codebook in our model. When training the Creative-GAN, we adopt the Adam optimizer with a learning rate of 1e-4 with $\beta 1 = 0.5$, $\beta 2 = 0.9$ and $\beta 3 = 0.9$ for the generator and two discriminators. Finally, we perform a denoising process on the generated raw music data for better audio quality. Specially, we make use of a Python package of the spectral gating noise reduction algorithm called noisereduce available on GitHub [36].

4.1.3 Baseline

From a cross-modal generation perspective, generating music from dance is a relatively new task. Quite a lot of work is still in the exploratory stage. For the comprehensive assessment, we compare our method with the following methods:
InverseMV [37]: a new attention based model VMT (Video-Music Transformer) which automatically generates piano scores from video frames.
Dance2Music [38]: propose a search-based offline method to generate music after processing the entire dance video and an online method which adopt a deep neural network to generate music as the video proceeds.
Foley Music [39]: a novel Graph–Tansformer model for predicting MIDI events from body pose features, converting the MIDI back to the original waveform with single-instrumental sound.
Controllable Music Transformer (CMT) [19]: a model based on transformer is put forward to enable local and global control of video background music generation via MIDI representation.
Dance2Music-GAN (D2M-GAN) [40]: a new adversarial multimodal framework which generates complex music from dance videos by Vector Quantized representations.

4.2 Evaluation Metrics

We designed and followed a overall evaluation agreement that includes objective and subjective metrics to evaluate the relationship between the generated music and the corresponding dance videos. We measured (1) music rhythm, (2) generation diversity, (3) dance-music correlation, and (4) overall quality of generated music. Comparing our proposed model with other music generation models, the results indicate that our model perform better than the baselines, which is shown in Table 2.
a) Music Rhythm
Music rhythm is an important feature of generated music. The kinematic dance and musical beats are usually aligned as a prior condition, so we only need to compare the beats of the generated music and ground truth music samples to evaluate the music rhythm. Music beats are extracted by the onset strength. The number of detected beats from the generated music samples are defined as B_g, the total beats from ground truth music as B_t, and the number of aligned beats from the generative samples as B_a. We employ two objective scores as evaluation metrics: (i) Beats Coverage Scores B_g/B_t, measure the ratio of total generated beats to total musical beats. (ii) Beats Hit Scores B_a/B_t measures the ratio of aligned beats to total musical beats. As shown in Table 2, compared to other methods, our proposed methods achieve better scores.
b) Generation Diversity
Dance is creative art works, so is music. The generated music should be expected to be harmonious and diverse with dance video. Compared with other methods, our model is able to generate a variety of music with different dances, and different styles of music for a given dance. For the purpose of measuring the musical diversity, we generated 30 pieces of music, and then calculated the diversity by the average feature distance Distm in the Euclidean space. We use Distm to compute the Euclidean distance between audio features extracted by a VGG network [41] which is pretrained on AudioSet [42].

Table 2 demonstrate that our model accomplishes higher Distm than other methods. This is due to two rea-

Table 2 Quantitative results on the AIST++ dataset. Comparing to the four baseline methods, our method generates music that is more diversified when conditioned on different music and more consistently aligned with input dance. ↑ A higher value is better.

Method	Music Rhythm		Generation Diversity	Dance-Music Correlation	Overall Quality	User Study
	Beats Coverage Score↑	Beats Hit Score↑	Distm↑	Beat Alignment Score↑	Mean Opinion Score↑	Our Method Wins
AIST++	–	–	8.12	0.289	4.8	42.1%
InverseMV	72.5	71.4	3.76	0.182	2.9	93.4%
Foley Music	74.6	70.1	4.17	0.214	3.3	85.6%
CMT	85.7	43.6	4.79	0.233	3.2	81.2%
D2M-GAN	88.5	84.9	5.25	0.235	3.5	67.8%
Ours High-level	87.9	84.6	6.75	0.241	3.7	–
Ours Low-level	**92.1**	**91.8**	**7.68**	**0.245**	**4.2**	–

sons. Firstly, since the baseline method relies on MIDI events as audio representations, it can only simply generate music samples with single-instrument sounds. While our generated quantized feature vector can represent complex music instead, this beneficial to develop the diversity of the generated music. Secondly, we extend the GAN framework by adding a genre discriminator, which is trained to classify the types of music clips, and generate music with multiple styles during the inference process.

c) Dance-Music Correlation

When we combine the input dance and output music, they should be harmonious, coherent and consistent. In order to assess the correlation between kinematic beats and the generated music rhythms, we define Beat Alignment Score as the average distance between kinematic beat and its nearest music beat.

$$BeatAlign = \frac{1}{m}\sum_{i=1}^{m}\exp\left(-\frac{\min_{\forall t_j^y \in B^y}\left\|t_i^x - t_j^y\right\|^2}{2\sigma^2}\right) \quad (13)$$

$B^x = \{t_i^x\}$ is the kinematic beats that are the local minima of the kinetic velocity, and $B_y = \{t_j^y\}$ is the music beats extracted with Librosa. While σ is a parameter to normalize sequences. We set σ = 3 in all our experiments.

Beat alignment is an important characteristic in dance-to-music process. We visualize instance of beat alignment between dance and generated music in Fig. 4. Compared the beats extracted from the generated music with the corresponding kinematic beats, it shows that most beats of our generated music happen at kinematic beat times. The results are presented in Table 2, it can be observed that our model obtains better scores compared to other baselines, which validates that our proposed method can capture relations to input dance and generate fairly artistic music that matches well.

d) Overall Quality of Generated Music

Our evaluation metrics not only include objective assessments of dance-music correlations, but also subjective assessments of the overall quality of the generated music. The Mean Opinion Scores (MOS) test is conducted for subjective evaluations. A total of 30 volunteers participated in our MOS test. In the overall quality test, 15 audio samples (without video frames) were played for each volunteer. Afterwards, they were required to score the overall quality of the audio samples in the range 1 to 5. MOS tests in Table 2 reveals that our model can generate music samples with better overall quality.

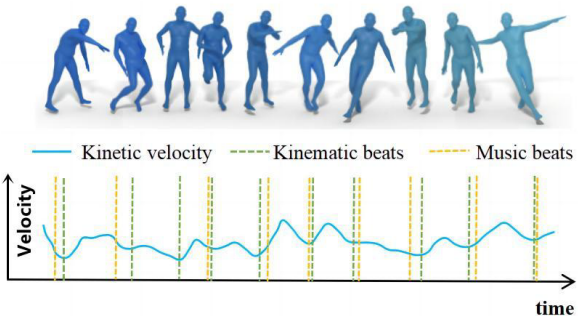


Fig. 4 Instance of beat alignment visualizations between dance and generated music. The blue line curve represents the dynamic velocity of the dance movement, and the green and orange dashed lines respectively represent the extracted kinematic beats and music beats. The kinematic beats are extracted by finding local minima from the kinetic velocity curve.

e) User Study

To learn more about the generated work performance of our approach, we perform a user study to compare the user preference of our method with baseline in AIST++ test set. In user study, we shuffle playback 25 pairs of comparison videos. The participant is asked to select one between our result and the other model's result in the same dance for the question of "which music follows better to the dance". 20 participants take part in the user study. We observe in Table 2, that our approach significantly surpasses all the compared baselines with a win rate of at least 67.8%. Even compared to the ground truth, 42.1% of our generated dances are considered better. According to participant feedback, the music we generated has a higher variety and stable rhythm.

f) Results on the TikTok Dance-Music Dataset

In order to prove the application of this method in real-world scenarios, we collect a paired dance-music dataset from TikTok video. Our dataset contains a total of 112 dance videos with 20 songs and the average length of each

Table 3 Evaluations for the experiments on the TikTok dataset.

Models	Beats Coverage	Beats Hit
High-level	**88.1**	82.3
Low-level	86.9	**83.8**

Table 4 Results for ablation study of sample length.

Sample length	Beats Coverage	Beats Hit	$Dist_m$	Beat Alignment
2s	**92.1**	**91.8**	**7.68**	**0.245**
3s	89.9	88.4	6.35	0.234
4s	87.8	84.7	6.17	0.224
5s	84.1	79.9	5.32	0.207

Table 5 Results for ablation study of loss function.

Losses	Beats Coverage	Beats Hit	$Dist_m$	Beat Alignment
A Adversarial Loss L_{ori}	84.8	84.1	4.37	0.206
B + Style Loss L_{sty}	85.2	84.5	7.53	0.214
C + Feature Matching Loss L_{FM}	87.4	86.9	7.58	0.237
D + Codebook Commitment Loss L_{code}	88.4	87.1	7.61	0.241
E + Perceptual Loss L_p	**92.1**	**91.8**	**7.68**	**0.245**

video is approximately 12.5 seconds. Table 3 shows the quantitative evaluation results of experiments on the TikTok dataset, demonstrating the overall robustness of the proposed method.

4.3 Ablation Study

In this section, we perform the ablation experiments in sample length and loss study.

a) Sample Length

In our experiment, we use 2-second samples for training and testing, our model can also be designed in terms of a longer sample length for ablation studies as shown in Table 4.

b) Loss Study

We consider the effect of objective function in generation model. Specifically, we fixed the original adversarial loss and added the proposed loss function continuously. The results are listed in Table 5, the corresponding results make a contribution to each loss term. Firstly, we observe that genre ambiguity loss contributes to the diversification of musical styles, this is because Creative-GAN can learn all musical styles in the original database. Secondly, we also find that perceptual loss contributes to the generation of music rhythm, as it captures audio's semantic characteristics which are used to focus on auditory quality.

5. Conclusion

We put forward a novel Creative-GAN architecture to generate dance-conditioned artistic music via vector quantized audio representations with multiple genres. Our model also supports converting one genre of music to multiple target genres. Experiments on the standard benchmark observe that our model achieves promising performance in both quantitative metrics and human evaluation. We can generate complex and diverse high-quality music with strong correlation to dance. In the future, we plan to propose a music evaluation function from the perspective of music composition rules, such as music structure, to realize real-time supervision and adjustment of generated music, another interesting future research direction is about generating music when multiple people dance.

Acknowledgments

This research is supported by the National Key Research and Development Program of China (No.2019YFB1406201), the National Natural Science Foundation of China under Grant (No.62071434), and the Fundamental Research Funds for the Central Universities (Grant No.CUC21GZ010).

References

[1] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Advances in Neural Information Processing Systems, vol.3, pp.2672–2680, 2014.

[2] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," NIPS, pp.6309–6318, 2017.

[3] PAYNE, CHRISTINE, "MuseNet," OpenAI [EB/OL]. openai.com/blog/musenet.

[4] A. Roberts, J. Engel, C. Raffel, et al., "A hierarchical latent vector model for learning longterm structure in music," International Conference on Machine Learning, PMLR, 2019.

[5] H.W. Dong, W.Y. Hsiao, L.C. Yang, et al., "MuseGAN: Multitrack sequential generative adversarial networks for symbolic music generation and accompaniment," AAAI Conference on Artificial Intelligence, 2017.

[6] L.C. Yang, S.Y. Chou, and Y.H. Yang, "MidiNet: A convolutional generative adversarial network for symbolic-domain music generation," arXiv preprint arXiv:1703.10847, 2017.

[7] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," ICLR, 2016.

[8] C.Z.A. Huang, A. Vaswani, J. Uszkoreit, et al., "Music transformer," arXiv preprint arXiv:1809.04281, 2018.

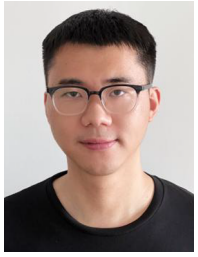
[9] P. Dhariwal, H. Jun, C. Payne, J.W. Kim, A. Radford, and I. Sutskever, "Jukebox: A generative model for music," arXiv preprint arXiv:2005.00341, 2020.

[10] T. Tang, J. Jia, and H. Mao, "Dance with melody: An LSTM-autoencoder approach to music-oriented dance synthesis," ACM Multimedia, pp.1598–1606, 2018.

[11] P. Tendulkar, A. Das, A. Kembhavi, and D. Parikh, "Feel the music: Automatically generating a dance for an input song," arXiv preprint arXiv:2006.11905, 2020.

[12] B. Korbar, D. Tran, and L. Torresani, "Co-training of audio and video representations from self-supervised temporal synchronization," arXiv preprint arXiv:1807.00230, 2018.

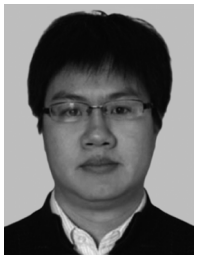
[13] M. Cardle, L. Barthe, S. Brooks, and P. Robinson, "Music-driven motion editing: Local motion transformations guided by music analysis," Proc. 20th Eurographics UK Conference, IEEE, pp.38–44, 2002.
[14] M. Lee, K. Lee, and J. Park, "Music similarity-based approach to generating dance motion sequence," Multimed. Tools Appl., vol.62, no.3, pp.895–912, 2013.
[15] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," ACM Trans. Graph., voo.35, no.4, pp.1–11, 2016.
[16] H.-Y. Lee, X. Yang, M.-Y. Liu, T.-C. Wang, Y.-D. Lu, M.-H. Yang, and J. Kautz, "Dancing to music," NeurIPS, 2019.
[17] J. Li, Y. Yin, H. Chu, Y. Zhou, T. Wang, S. Fidler, and H. Li, "Learning to generate diverse dance motions with transformer," ArXiv, abs/2008.08171, 2020.
[18] H. Zhao, C. Gan, W.-C. Ma, and A. Torralba, "The sound of motions," ICCV, 2019.
[19] S. Di, Z. Jiang, S. Liu, Z. Wang, L. Zhu, Z. He, H. Liu, and S. Yan, "Video background music generation with controllable music transformer," ACMMM, pp.2037–2045, 2021.
[20] V. Iashin and E. Rahtu, "Taming visually guided sound generation," British Machine Vision Conference (BMVC), 2021.
[21] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fifidelity images with VQ-VAE-2," NIPS, 2019.
[22] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," CVPR, 2021.
[23] M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784, 2014.
[24] A. Elgammal, B. Liu, M. Elhoseiny, and M. Mazzone, "CAN: Creative adversarial networks, generating "art" by learning about styles and deviating from style norms," arXiv preprint arXiv:1706.07068, 2017.
[25] J. Kim, H. Oh, S. Kim, H. Tong and S. Lee, "A brand new dance partner: Music-conditioned pluralistic dancing controlled by multiple dance genres," CVPR, 2022.
[26] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks for action recognition in videos," IEEE Trans. Pattern Anal. Mach. Intell., vol.41, no.11, pp.2740–2755, 2018.
[27] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M.J. Black, "SMPL: A skinned multi-person linear model," ACM Trans. Graphic, vol.34, no.6, pp.1–16, 2015.
[28] A.B.L. Larsen, S.K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," International Conference on Machine Learning, PMLR, 2016.
[29] Y. Aytar, C. Vondrick, and A. Torralba, "SoundNet: Learning sound representations from unlabeled video," NIPS, pp.892–900, 2016.
[30] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp.580–587, 2014.
[31] R. Li, S. Yang, D.A. Ross, and A. Kanazawa, "AI choreographer: Music conditioned 3D dance generation with AIST++," ICCV, 2021.
[32] N. Mahmood, N. Ghorbani, N.F. Troje, G. Pons-Moll, and M.J. Black, "AMASS: Archive of motion capture as surface shapes," Proc. IEEE International Conference on Computer Vision, pp.5442–5451, 2019.
[33] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments," IEEE Trans. Pattern Anal. Mach. Intell., vol.36, no.7, pp.1325–1339, July 2014.
[34] O. Alemi, J. Françoise, and P. Pasquier, "GrooVenet: Real-time music-driven dance movement generation using artifificial neural networks," Networks, vol.8, no.17, 26, 2017.
[35] W. Zhuang, C. Wang, S. Xia, J. Chai, and Y. Wang, "Music2Dance: Music-driven dance generation using wavenet," arXiv preprint arXiv:2002.03761, 2020.
[36] T. Sainburg, timsainb/noisereduce: v1.0.1; 2019. https://github.com/timsainb/noisereduce. Available from: https://doi.org/10.5281/zenodo.3243589
[37] C.-T. Lin and M. Yang, "InverseMV: Composing piano scores with a convolutional video-music transformer," ISMIR, 2020.
[38] G. Aggarwal and D. Parikh, "Dance2Music: Automatic dance-driven music generation," arXiv preprint arXiv:2107.06252, 2021.
[39] C. Gan, D. Huang, P. Chen, J.B. Tenenbaum, and A. Torralba, "Foley music: Learning to generate music from videos," ECCV, pp.758–775, 2020.
[40] Y. Zhu, K. Olszewski, Y. Wu, P. Achlioptas, M. Chai, Y. Yan, and S. Tulyakov, "Quantized GAN for complex music generation from dance videos," ECCV, pp.182–199, 2022.
[41] S. Hershey, S. Chaudhuri, D.P.W. Ellis, J.F. Gemmeke, A. Jansen, R.C. Moore, M. Plakal, D. Platt, R.A. Saurous, B. Seybold, M. Slaney, R.J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," ICASSP, IEEE, 2017.
[42] J.F. Gemmeke, D.P.W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R.C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and humanlabeled dataset for audio events," ICASSP, IEEE, 2017.



Jiang Huang received B.S. degree and M.S. degree from East China Jiaotong University and Communication University of China in 2010 and 2013, respectively. He is now working towards Ph.D. degree with the School of Computer Science, Communication University of China. His main research interest include music artificial intelligence and multimedia content computing.

Xianglin Huang received B.S. degree and M.S. degree from Jilin University in 1990 and 1998, and the Ph.D. degree in Beijing University of Technology, China in 2002. He is currently a professor at School of Computer Science and Cybersecurity, Communication University of China. His research interests include image and video intelligent processing.

Lifang Yang received B.S. degree from Qingdao University, China in 2005, M.E. degree and Ph.D. degree from Communication University of China in 2007 and 2012, respectively. She is currently an associate professor in Communication University of China. Her research interests include Intelligent retrieval and High-dimensional index structure.



Zhulin Tao received Ph.D. degree from Communication University of China in 2019. Now, he is a lectorate at Communication University of China. His research interests include graph neural network and personal recommendation system.