Dance2MIDI: Dance-driven Multi-Instruments Music Generation

Bo Han[1], Yuheng Li[1], Yixuan Shen[2], Yi Ren[3], and Feilin Han[4](✉)

Abstract Dance-driven music generation aims to generate musical pieces conditioned on dance videos. Previous works focus on monophonic or raw audio generation, while the multi-instruments scenario is under-explored. The challenges associated with the dance-driven multi-instrument music (MIDI) generation are twofold: 1) no publicly available multi-instruments MIDI and video paired dataset and 2) the weak correlation between music and video. To tackle these challenges, we build the first multi-instruments MIDI and dance paired dataset (D2MIDI). Based on our proposed dataset, we introduce a multi-instruments MIDI generation framework (Dance2MIDI) conditioned on dance video. Specifically, 1) to capture the relationship between dance and music, we employ the Graph Convolutional Network to encode the dance motion. This allows us to extract features related to dance movement and dance style, 2) to generate a harmonious rhythm, we utilize a Transformer model to decode the drum track sequence, leveraging a cross-attention mechanism, and 3) we model the task of generating the remaining tracks based on the drum track as a sequence understanding and completion task. A BERT-like model is employed to comprehend the context of the entire music piece through self-supervised learning. We evaluate the generated music of our framework trained on the D2MIDI dataset and demonstrate that our method achieves State-of-the-Art performance.

Keywords video understanding, music generation, symbolic music, cross-modal learning, self-supervised.

1 College of Computer Science and Technology, Zhejiang University, 310058, China. E-mail: borishan815@zju.edu.cn.

2 National University Singapore, 119077, Singapore. E-mail: yshe0148@gmail.com.

3 Bytedance AI Lab, Speech & Audio Team, Singapore, 048583, Singapore. E-mail: rayeren613@gmail.com.

4 Department of Film and TV Technology, Beijing Film Academy. Email: hanfeilin@bfa.edu.cn(✉)

1 Introduction

As choreographer Zakharov puts it "*Music is the soul of dance; music contains and determines the structure, characteristics, and temperament of dance*". The relationship between music and dance is complementary. Studies have demonstrated that humans utilize the same neural pathways to appreciate both dance and music [1, 2]. Therefore, it is essential for the accompanying music that conform to the fundamental structure, style, and emotional expression to enhance the artistic appeal of the dance videos.

In the era of short videos, sharing dance performances on social media platforms has become a popular trend. Mainstream platforms often provide automatic soundtracks for dances or allow creators to independently select music clips. However, it should be noted that the music available in these libraries is pre-existing and may only be suitable for simple and regular dance movements. Matching complex and diverse movements can be challenging. Additionally, the use of pre-existing music may result in copyright disputes. Manually selecting appropriate music for dance can be a time-consuming process. To improve matching, originality, and efficiency, automatic music generation has emerged as a thriving subject of research in recent years [3–8].

While there has been significant research on music-to-dance generation [9–13], the inverse direction of dance-to-music generation remains underexplored, which is a challenging task for the following reasons:

- Music generation is challenging [14, 15]. In real-world applications, music is often polyphonic and multi-instrumental, requiring harmony and coherence across all instruments. This complexity in music representation makes the generation difficult.
- Conditional music generation is also a challenging task [3, 4, 16]. The correlation between the music and the control signals, such as dance videos, is often weak. For instance, this correlation may include musical and dancing beats, tempo, and emotion [17]. However, there

are many degrees of freedom for each modality (music and dance), which can be regarded as noise and may confuse the generative model during training.
- The lack of publicly available datasets containing paired music and dance videos hinders the development of dance-to-music generation research.

There are only a few works studying dance-to-music generation: Dance2Music [3] takes in the local history of the dance similarity matrix as input and generates monophonic notes. However, the handcrafted features they used may discard much useful information in dance videos and monophonic music is not applicable to the real scenarios. D2M-GAN [8] takes dance video frames and human body motions as input to directly generate music waveforms. While this approach can generate continuous multi-instrumental music, the high variability of waveform data (e.g., variable and high-dynamic phase, energy, and timbre of instruments) makes it difficult to directly model high-quality waveforms. As a result, the generated music often contains strange noise.

This work aims to tackle the challenges of dance-to-music generation and address the issues of previous works. To overcome the scarcity of datasets, we collect and annotate the first large-scale paired dataset of dance and multi-instrument music (D2MIDI), which encompasses six mainstream dance genres: classical, hip-hop, ballet, modern, latin, and house. In total, it contains 71,754 pairs of multi-instrument MIDI data and dance video data. To model the correlation between music and dance, we introduce a multi-instrument MIDI generation framework (Dance2MIDI). It is architecturally designed with three primary modules: a Context Encoder for understanding the dance motion features related to music, a Drum Rhythm Generator for creating a base rhythmic drum track, and a Multi-Track MIDI BERTGen for producing multi-track MIDI track based on the drum track.

In light of the diversity in human skeleton space and the intricacy of associated movement patterns, we represent the human skeleton as a motion graph. To augment the feature extraction efficacy of the Context Encoder, we employ a graph convolutional network [18]. Our approach involves bifurcating the process into two distinct branches, each dedicated to extracting specific features: one for dance movement and the other for dance style. The drum, being fundamental to the establishment of rhythm in music, often serves as the starting point for composers when crafting a new musical piece. This is typically achieved by designing the rhythm for the accompanying drum track. In this context, we utilize a Transformer as the core of the Drum Rhythm Generator. This generator progressively creates drum tracks through autoregression, guided by dance condition control information. For the generation of other audio tracks, considering the unique characteristics of symbolic music in sequence modeling, we reframe this task as a sequence comprehension and completion task. Consequently, we introduce a model akin to BERT [19] to understand the entire MIDI music sequence in a self-supervised manner. Through experimentation, we have found that our method can achieve harmonious and coherent multi-instrument dance-to-music generation and outperforms all baseline methods [3, 4, 8], demonstrating the effectiveness of our dataset and framework. In summary, our main contributions are as follows:

- We construct the first multi-instrument dance-to-music dataset (D2MIDI), which facilitates research in the field of dance-to-symbolic music generation.
- We introduce an effective multi-instrument dance-to-multi track music framework Dance2MIDI, which demonstrates the feasibility of multi-instrument music generation and provides insights into multi-modal symbolic music generation.
- Exhaustive qualitative and quantitative assessment demonstrate that our method achieves State-of-the-Art performance.

2 Background

2.1 Music Generation

While the waveform is the original form of audio, some models generate audio directly in the waveform [20–22]. However, a single second of audio waveform spans tens of thousands of timesteps. As a result, existing non-symbolic music-based generative methods typically employ intermediate audio representations for learning generative models [23–25]. Nevertheless, this does not completely alleviate the dilemma [15]. Consequently, some recent works have adopted a symbolic music modeling approach. MuseGAN [26] employs a multi-track GAN-based model using 1D piano-roll symbolic representations. Music Transformer [27] generates long sequences of music using 2D event-based MIDI-like audio representations. Despite the potential of generative models for long sequence generation, the quality of samples produced by these models often deteriorates significantly. To address this issue, TBPTT [28] employs a Transformer-XL [27] generator in conjunction with a pre-trained Span-BERT [29] discriminator for long symbolic music generation, which enhances training stability. PopMAG [30] proposes a novel Multi-track MIDI representation MuMIDI that enables simultaneous multi-track generation in a single sequence and introduces extra long-context as memory to capture long-

term dependency in music. SymphonyNet [31] introduces a novel Multi-track Multi-instrument representation that incorporates a 3-D positional embedding and a modified Byte Pair Encoding algorithm for music tokens. Additionally, the linear transformer decoder is employed as the backbone for modeling extra-long sequences of symphony tokens.

2.2 Dance To Music

A recent novel approach to dance beat tracking has been proposed [32], which only detects music beats from dance videos. RhythmicNet [16] employs a three-stage model comprising video2rhythm, rhythm2drum, and drum2music. However, it is limited to generating music for only two instruments. CMT [4] establishes three relationships between video and music, including video timing and music beat, motion speed and simu-note density, and motion saliency and simu-note strength. While this approach does not specifically target dance-to-music tasks and fails to fully exploit the human motions present in dance videos. Dance2Music [3] utilizes the local history of both dance similarity matrices to predict notes but is restricted to generating single-instrument music. D2M-GAN [8] takes dance videos and human body motions as input to directly generate music waveforms. However, the generated music often contains noise.

2.3 Symbolic Music Dataset

The Groove MIDI Dataset (GMD) [33] comprises 13.6 hours of aligned MIDI and synthesized audio of human-performed, tempo-aligned expressive drumming, including 1,150 MIDI files and over 22,000 measures of drumming. In contrast, the Lakh MIDI dataset [34] is a collection of 176,581 unique MIDI files, with 45,129 matched and aligned to entries in the Million Song Dataset. The MAESTRO dataset [35] is a dataset composed of 198.7 piano MIDI, audio, and MIDI files aligned with 3 ms accuracy. ADL Piano MIDI [36] is a dataset that is based on the Lakh MIDI dataset. It generates 9,021 pieces of piano MIDI data from the Lakh MIDI dataset and then crawls an additional 2,065 pieces of piano MIDI data from network channels. Both datasets contain only MIDI music of the piano instrument type. All the above datasets are purely symbolic and lack corresponding dance movement annotations. They can't support the task of dance-to-music. The AIST Dance Video Database [37] provides a large-scale collection of dance videos with paired dance action videos and music annotations. However, its paired music is in the form of waveforms and lacks paired symbolic music annotations. Additionally, one piece of music corresponds to multiple videos, and the non-overlapping music clips comprise only 60 pieces.

3 D2MIDI Dataset

In this section, we provide a brief overview of our newly collected dance-to-MIDI dataset (D2MIDI) and the methodology of its acquisition. D2MIDI represents the first multi-instrument dataset of its kind and possesses several notable features:

- High-quality solo dance video: it comprises high-quality solo dance videos that have been carefully curated from internet sources to exclude low-quality footage and videos featuring multiple dancers (Section 3.1)
- Multi-instrument and polyphonic MIDI: the dataset contains multi-instrumental and polyphonic MIDI transcriptions that are temporally synchronized with the corresponding dance videos (Section 3.2)
- Multi-style and large-scale: the dataset is both multi-style and large-scale, encompassing a diverse range of dance styles across 71,754 clips. (Section 3.4).

3.1 Video Crawling and Selection

We manually filter dance videos from various video platforms using the following screening criteria: 1) The video must have a pure background with minimal interference from other characters. 2) Only videos featuring a single dancer are selected. 3) The music and dance movements must be highly synchronized. 4) The background music must be clear, and free of extraneous noise.

3.2 MIDI Transcription and Annotation

To ensure consistency in the Frames Per Second (FPS) with the dance motions in the videos [13], we first standardize the FPS of all dance videos to 20. Additionally, we unify the sample rate of the audio to 10,240Hz and then separate the audio in the video. Next, we utilize the MT3 [38] music transcription model to convert the original audio into multi-instrument MIDI music. However, the MIDI transcribed by MT3 may contain low-quality notes and discrepancies between music tempo changes and character movement changes in dance videos. To address these issues, we enlist professionals to align and label the MIDI music with reference to the context of the video and music. Specifically, professionals adjust the pitch, start time, duration, and instrument type of notes at corresponding positions in the music based on the pleasantness of the music and the context of the video.

Dataset	Dance	Audio	MIDI	Genres	Instrument	Pieces	Hours	Available
Groove MIDI [33]	✗	✗	✓	-	1	1,150	13.6	✓
LMD-aligned MIDI [34]	✗	✓	✓	-	10	45,129	-	✓
MAESTRO Dataset [35]	✗	✓	✓	1	1	1,276	198.7	✓
ADL Piano MIDI [36]	✗	✗	✓	-	10	11,086	-	✓
AIST Database [37]	✓	✓	✗	10	-	60	118.1	✓
Ours	✓	✓	✓	6	13	71,754	**597.95**	✓

Table 1 Symbolic music dataset comparisons. The AIST Database does not contain symbolic music. However, it is often used in dance-to-music tasks for modeling non-symbolic music. In this paper, we also labeled it for comparison experiments.

3.3 Dance Motion Estimation

The movement and posture of the human body are closely related to the fluctuations in the music. Unlike other 2d-keypoints methods [39, 40], we extract the 3d keypoints of the human body including body, hand, and face, allowing densely represented pose features.

3.4 Statistics

We employ the sliding window method to sample data from the video. Each sampling window has a size of 600 frames, equivalent to a 30-second dance video, with a sliding window size of 40 frames. This process resulted in a total of 71,754 pairs of data, in which the dance type includes classical, hip-hop, ballet, modern, latin, and house. The music in each data pair does not repeat each other. In the D2MIDI dataset, the duration in each data pair is 30 seconds, which is guaranteed to generate music with a rhythmic structure. The music in the pair contains up to 13 instrument types, including Acoustic Grand Piano, Celesta, Drawbar Organ, Acoustic Guitar (nylon), Acoustic Bass, Violin, String Ensemble 1, SynthBrass 1, Soprano Sax, Piccolo, Lead 1 (square), Pad 1 (new age) and Drum. We compare our proposed dataset with public datasets in Table 1.

4 Dance2MIDI Framework

The proposed architecture is schematically illustrated in Fig. 1 and comprises three main components: the Context Encoder, the Drum Rhythm Generator, and the Multi-Track BERTGen. In the Context Encoder, we commence by employing the joint point extraction to obtain the spatial coordinates of the human joints within the dance video. Subsequently, via the utilization of two distinct branches, we extract the dance style features and dance movement features. These extracted features are then combined to form a concatenated representation, which serves as a guide for generating conditional control information that corresponds to the MIDI music. It is worth noting that in the realm of MIDI music, drums play a pivotal role in generating fundamental rhythm patterns that underlie the musical composition. Moreover, in the context of composition and improvisation, it is customary for composers to initiate the creation of a new musical piece by designing the rhythm for the accompanying drum track. As the piece progresses, additional instrumental tracks are incrementally layered on top of the drum track, thereby culminating in the production of the final musical composition. So we first leverage Drum Rhythm Generator to incrementally generate drum tracks in an autoregressive manner, thus establishing the foundational melody of the music. Subsequently, we augment the overall music composition by incorporating note information from other tracks and instruments, thereby enhancing its richness and complexity. We conceptualize this process as a sequence completion task, wherein the BERT-like model is employed to enrich the remaining music track, facilitating a comprehensive understanding of the entirety of the musical piece.

4.1 Context Encoder

The Context Encoder primarily comprises two branches designed to extract features related to dance movement and dance style. Initially, the human body motion joint coordinate $X \in \mathbb{R}^{T\times J\times 3}$ is extracted from the original dance video via the Mediapipe framework [41], which is then input into two conditional encoders.

In the dance movement branch, the human body motion joint is modeled as a motion graph. The spatial position of the character's joints in each frame is first aggregated through a spatial Graph Convolutional Network (GCN). Subsequently, timing information across time frames is aggregated via temporal convolution. After this, we obtain the dance movement features $Z_m \in R^{T\times F}$, where T and F represent the number of video frames and the number of feature channels, respectively. It's particularly noteworthy that the relationship between dance movements and the beats of music is intricately linked, as it is the transitions in dance movements that often drive changes in musical beats. In this context, we transform the dance movement feature into a binary detection problem for music beats: Given the dance movement feature $Z_m \in R^{T\times F}$, motion information across both temporal and spatial dimensions is consolidated via attention learning

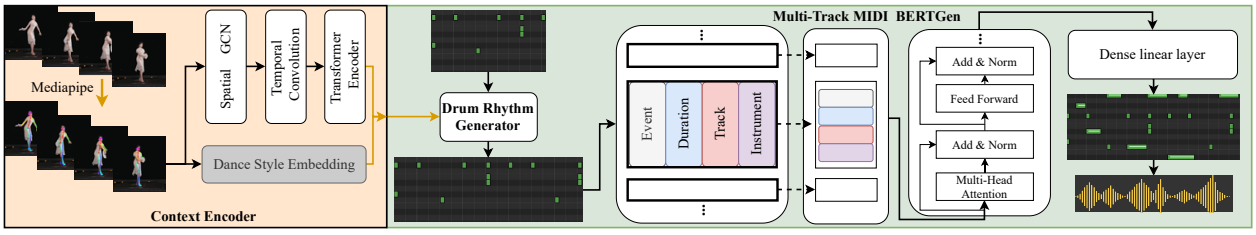


Fig. 1 An overview of our proposed Dance2MIDI model. The dance video is input into the Mediapipe framework [41] to extract the coordinates of the human body's joint points. These coordinates are then used to encode the spatio-temporal features of dance movement and dance style (yellow block). Subsequently, these features serve as conditional information to guide the generation of multi-instrument MIDI music sequences (green block).

within the Transformer encoder, yielding the final beat binary sequence $Z_b \in R^T$ of the same length as Z_m, where each frame is classified as either a beat or non-beat. This approach lays the foundation for the entire musical piece, ensuring consistency in timing and rhythm between the dance movement and the music.

The dance style branch operates similarly to the Choreo-Mater network [42], utilizing four GCN blocks and two Gated Recurrent Unit (GRU) layers to compress the dance sequence into 32-dimensional embedding vectors Z_s. The vectors are then input into an MLP classifier. The dance style branch is pre-trained on the large-scale annotated dataset D2MIDI. In the final step, the beat binary sequence and style feature are concatenated to derive conditional control information Z, which subsequently guides the multi-instrument MIDI music generation.

4.2 Music Representation

Inspired by SymphonyNet [31], we represent multi-instrument music using quads, which include event, duration, track, and instrument.

- Event: The event attribute comprises four sub-attributes: measure, chord, position, and pitch. A BOM symbol is used to indicate the beginning of each measure, with all symbols in the measure added after the BOM symbol. Beat and note duration are adopted as time units to divide each measure and determine position. The pitch range is divided from 0 to 127 based on the general MIDI design.
- Duration: It represents the duration of each note.
- Track&Instrument: The track and instrument attributes are determined by traversing the music and identifying the track and instrument corresponding to each note.

Unlike natural language sequences, symbolic music sequences exhibit relative position invariance. For instance, a chord C containing the music notes (C, N_1, N_2, N_3) is equivalent to (C, N_2, N_1, N_3). As they comprise the same notes and are controlled by the same chord, the order of the notes does not affect the musical effect. Therefore, we employ relative position encoding for music notes.

4.3 Drum Rhythm Generator

Given that multi-instrument MIDI music can be represented as discrete tokens, it is inherently suitable for sequence modeling in the realm of natural language processing. Consequently, we employ an autoregressive approach to generate drum track notes. More specifically, we utilize a Transformer model [27] to generate drum notes in a step-by-step manner, guided by the dance condition control vector Z. For the key attention module in the model, we adopt the Masked Self-Attention (MSA) module consistent with the Transformer and design the cross-attention module Video Guided MIDI (VGM). The cross-attention mechanism [43] is utilized to blend two distinct sequences of embeddings, where these sequences can represent different modalities. Similarly, one sequence serves as the input query (Q), defining the length of the output sequence, while another sequence provides the input keys (K) and values (V). The MSA and VGM modules are employed in pairs. The VGM module employs conditional control information Z to guide attentional learning in the Drum Rhythm Generator. The attention maps of the VGM block tend to focus on values related to visual information. The specific calculation method is shown in Eq. (1). Among them, the Drum-encoded sequence D is utilized as the query, while the conditional control information Z extracted from the dance video is employed as both the key and value. The parameter matrices $W^q \in \mathbb{R}^{d_{\text{model}} \times d_q}$, $W^k \in \mathbb{R}^{d_{\text{model}} \times d_k}$, and $W^v \in \mathbb{R}^{d_{\text{model}} \times d_k}$. In this work, the number of heads h in the multi-head VGM attention module is 8. For each head, we use $d_q = d_k = d_{model}/h = 64$.

$$\text{VGM}(Q, K, V) = \text{softmax}\left(\frac{DW^q(ZW^k)^T}{\sqrt{d_k}}\right)(ZW^v) \tag{1}$$

4.4 Multi-Track MIDI BERTGen

In the process of enriching the entirety of a musical piece, we generate note information for tracks beyond the drum track. This task bears resemblance to image inpainting in computer vision and context understanding tasks in natural language processing. Given the unique nature of symbolic music as sequence modeling, we incorporate the BERT model [19] to comprehend the entire symbolic music sequence. The audio track that is to be completed is considered a part of the random mask in the BERT model. Unlike the masking strategy used in the BERT model, we have designed a novel masking approach tailored to the characteristics of symbolic music composition. Considering that a piece of music comprises multiple measures, and each measure contains tokens with similar attributes – the values of signature, tempo, and measure attributes remain consistent within each measure, and the types of instruments also follow a similar pattern, restricted within a small-scale range. Within the same measure, the information on position and pitch is also closely related. Therefore, employing a masking strategy within the same measure facilitates the model's learning of musical pattern structures. Specifically, we apply masking to the same type of tokens (events, durations, tracks, instruments) across different measures. In alignment with BERT, we replace 80% of all masked tokens with MASK tokens, substitute 10% with a randomly chosen token, and leave the remaining 10% unaltered. The Multi-Track MIDI BERTGen, a classic multilayer bi-directional Transformer encoder, comprises 12 layers of multi-head self-attention, each with 12 heads, and a hidden space dimension of 768 in the self-attention layers. As a self-supervised method, BERTGen does not require labeled data from downstream tasks for pre-training.

Each input token is initially transformed into a token embedding via an embedding layer, supplemented with a relative positional encoding that corresponds to its time step in the sequence. This is subsequently fed into a stack of 12 self-attention layers to obtain a contextualized representation, known as a hidden vector or hidden states, at the output of the self-attention stack. Owing to the bi-directional self-attention layers, the hidden vector is contextualized in that it has attended to information from all other tokens from the same sequence. Ultimately, the hidden vector of a masked token is fed into a dense layer to predict the missing token. As the vocabulary sizes for the four token types vary, we proportionally weight the training loss associated with tokens of different types to the corresponding vocabulary size to facilitate model training.

4.5 Training and Inference

Our model is trained in an end-to-end manner. During training, dance motion features and historical MIDI event sequences are input to predict the probability output of the next music event token. In the inference phase, the model autoregressively predicts the next MIDI event. Notably, at time step 0, the historical MIDI event sequence is empty, meaning that generation begins with an empty token.

5 Experiments

5.1 Datasets

We evaluate the effectiveness of our method through experiments on two datasets with paired dance video and music: the publicly available AIST dataset [37] and our D2MIDI dataset. The non-repetitive music in the AIST dataset comprises only 60 pieces, with one piece of music corresponding to multiple dance segments. The AIST dataset contains a total of 1,618 dance motions. However, many motions are filmed from different camera perspectives, resulting in 13,940 dance videos. Thus, on average, one piece of music corresponds to 232 dance videos. The AIST dataset encompasses ten dance genres: ballet jazz, street jazz, krump, house, LA-style hip-hop, middle hip-hop, Waack, lock, pop, and break. In contrast to, the music corresponding to each dance segment in our D2MIDI dataset is unique, making it more suitable for music generation tasks. The D2MIDI dataset contains 71,754 paired dance videos and MIDI music data, which are not present in the AIST dataset. The dance types of the D2MIDI dataset encompass six major genres: classical, hip-hop, ballet, modern, latin, and house.

5.2 Evaluation Metrics

We evaluate our method both objectively and subjectively using publicly available metrics [4, 8] and compare our model with the three State-of-the-Art models.

5.2.1 Coherence

To assess the coherence between dance beats and generated music rhythms, we utilize two objective metrics: Beats Coverage Score (BCS) and Beats Hit Score (BHS), as employed in previous works [11, 45]. These works have demonstrated that dance motions and music beats are typically aligned, allowing for a reasonable evaluation of music tempo by comparing beats in generated and ground truth music. We denote the number of detected beats in the generated music samples as B_g, the total number of beats in the original music as B_t, and the number of aligned beats in the generated samples as B_a. During inference, the duration of music generated using the

Fig. 2 Visualization result. For the given dance video input, Dance2MIDI generates corresponding MIDI music and converts it into a waveform. The music beat is detected using the public toolbox Librosa [44]. Two pieces of the dance videos are examples, where the blue box indicates the real dance beat (the turning point of the dance motion), and the red box indicates the frame of the dance video corresponding to the timestamp of our audio beat.

different methods may not align precisely with the duration of the real music, resulting in an overflow of detected beats B_g. Consequently, when the duration of music generated by the model does not match or significantly differs from the ground truth, BCS may exceed its value range of [0,1]. To address this issue, we standardize BCS value and propose Beat Average Score (BAS) in conjunction with the BHS. Specifically, as shown in Eq. (2), we employ exponential functions to constrain the BCS values within the range of 0 and 1. If the BCS value is less than 1, we use the exponential function $e^{\mathrm{BCS}-1}$. When BCS is greater than 1, we apply the exponential function $e^{1-\mathrm{BCS}}$.

- **BCS** It is calculated as B_g/B_t, representing the ratio of overall generated beats to total music beats.
- **BHS** It is calculated as B_a/B_t, representing the ratio of aligned beats to total musical beats.
- **BAS** The calculation method is shown in Eq. (2), representing the overall coherence between dance beats and music rhythms.

$$BAS = \begin{cases} 0.5 \times (e^{\mathrm{BCS}-1} + \mathrm{BHS}) & \text{s.t.BCS} < 1 \\ 0.5 \times (e^{1-\mathrm{BCS}} + \mathrm{BHS}) & \text{s.t.BCS} > 1 \end{cases} \quad (2)$$

5.2.2 *Quality*

We employed the objective metrics outlined in [4, 46] to assess the quality of symbolic music. These metrics include Pitch Class Histogram Entropy (PHE) and Grooving Pattern Similarity (GS).

PHE It evaluates the tonal quality of the music. we first gather all notes within each bar and then construct a 12-dimensional pitch class histogram $\overrightarrow{h}$, based on the pitch of all notes. This histogram is normalized by the total note count within the period such that $\sum_i h_i = 1$. Then, we calculate the entropy of $\overrightarrow{h}$ as Eq.(3):

$$\mathcal{H}(\overrightarrow{h}) = -\sum_{i=0}^{11} n_i \log_2(h_i) \quad (3)$$

GS It measures the rhythmicity of the music. In information theory [47], entropy is utilized to measure uncertainty. We apply entropy $\mathcal{H}$ to assess the tonal quality of symbolic music. A lower entropy value $\mathcal{H}$ indicates a clear tonality within a piece of music. For GS, it represents the positions within a bar where at least one note onset occurs, denoted by $\overrightarrow{g}$, a 64-dimensional binary vector. The similarity between a pair of grooving patterns is defined as Eq. (4).

$$GS\left(\overrightarrow{\mathbf{g}}^a, \overrightarrow{\mathbf{g}}^b\right) = 1 - \frac{1}{Q}\sum_{i=0}^{Q-1} \mathrm{XOR}\left(g_i^a, g_i^b\right) \quad (4)$$

Where Q represents the dimension of $\overrightarrow{g}^a$ and $\overrightarrow{g}^b$, XOR denotes the exclusive OR operation. If the music exhibits a distinct rhythmic feel, the groove pattern between each pair of bars should be similar, resulting in a high Groove Similarity score.

5.2.3 *Qualitative Evaluation*

We also conduct an audio-visual survey to subjectively compare the different models. We conduct the Mean Opinion Score experiments [8] to assess the quality of the music and the correspondence between the video and music. For each dance genre, 50 samples are evaluated by 5 professional choreographers. That is, a total of 500 evaluation samples are provided for the AIST dataset, and a total of 300 evaluation samples are provided for the D2MIDI dataset. Among the five choreographers involved in this study, three are female and two are male, spanning an age range of 25 to 45 years. They possess extensive experience in choreographing various types of dances. The entire evaluation process was conducted anonymously, ensuring that participants were unaware of which model generated the data samples. In our study, we present human participants with the same video accompanied by music synthesized using different methods. Participants are then asked to rate the music on a scale of 1 to 5, with

Metric	D2MIDI				AIST			
	CMT [4]	Dance2Music [3]	D2M-GAN [8]	ours	CMT	Dance2Music	D2M-GAN	ours
PHE ↑	2.49	2.24	/	**2.89**	2.55	2.26	/	**2.92**
GS ↑	0.62	0.98	/	**0.99**	0.64	0.98	/	**0.99**
BCS	5.11	1.75	0.68	**0.73**	4.87	1.73	0.70	**0.76**
BHS	0.29	0.42	0.45	**0.53**	0.32	0.44	0.48	**0.61**
BAS ↑	0.15	0.44	0.59	**0.65**	0.17	0.46	0.61	**0.69**
Consistency ↑	3.21	2.82	2.55	**3.91**	3.38	2.99	2.62	**3.99**
Noise ↑	3.43	**3.68**	2.68	3.57	3.45	**3.72**	2.82	3.67

Table 2 Objective and subjective evaluation results on the D2MIDI and AIST Dataset. ↑ means the higher the better.

higher scores indicating better performance. The primary evaluation criteria are:

- **Consistency** the degree to which the major stress or boundaries of the generated music aligned with the video boundaries or visual beat. For instance, fast-paced dance movements should be accompanied by major stress to enhance musicality.
- **Noise** noise degree of sounds produced by non-instrumental sources. For instance, a pleasing musical composition should be free of extraneous white noise. The lower the noise level, the higher the score awarded by participants.

5.3 Implement Details

We apply the same processing method described in Sections 3.2 and 3.3 to the AIST dataset to obtain paired dance motion joint data and MIDI music data. Our framework is implemented using PyTorch. The encoder of the graph convolutional network in our framework comprises 10 layers with residual connections. The number of layers in the graph convolution network is set with reference to the ST-GCN network [18], which is specifically designed for action recognition tasks. It consists of 9 layers of spatial-temporal graph convolution operators. The first three layers have an output of 64 channels. The following three layers have an output of 128 channels. The last three layers have an output of 256 channels. Subsequently, we add a layer with 512 channels to align with the dimension of the Transformer Encoder for predicting the beat binary sequence. To prevent overfitting, we apply random affine transformations to the skeleton sequences of all frames during training as a data augmentation technique. Both the encoder and decoder blocks of the Drum Rhythm Generator are set to 6. For each block, the dimensionality of the attention layer and feed-forward network layer are set to 512 and 1024, respectively. The multi-head VGM attention block has 8 heads. For post-processing of the generated MIDI music data, we use the FluidSynth [48] software synthesizer to convert the generated MIDI music into music waveform, consistent with the CMT model [4]. We train our model using the Adam optimizer with parameters $\beta_1 = 0.9$, $\beta_2 = 0.9$, and $\varepsilon = 10^{-9}$. The learning rate is scheduled during training with a warm-up period: it linearly increases to 0.0007 for the first 6000 training steps and then decreases proportionally to the inverse square root of the step number.

5.4 Results

We conduct training and evaluation on the AIST and D2MIDI datasets, respectively. For the state-of-the-art methods, including the CMT [4], Dance2Music [3], and D2M-GAN [8], we use the default parameter settings provided in their open-source code. For different dance genres, we divide the training, validation, and test sets in a ratio of 8:1:1 and use the same dataset settings for all models. As shown in Table 2, our model outperforms existing state-of-the-art methods on the objective metrics of PHE and GS, indicating that the music we generated was slightly better in terms of tone and rhythm. In a similar vein, our method outperforms others on BHS, BCS, and BHS. This demonstrates that the congruence between the music and videos generated by our approach is superior. Our method also achieves the best performance on the subjective metric of Consistency, further indicating that the music and dance videos we generated are highly consistent.

Dance2Music [3] achieves optimal performance on Noise metric, a subjective indicator measuring the purity of music. This is because the Dance2Music [3] model only models piano music and can only generate piano music, resulting in limited scalability. In the case of the CMT model [4], it has been observed that the length of the generated music often does not align with the duration of the corresponding dance. This discrepancy can be attributed to the model's lack of explicit consideration for dance characteristics, resulting in a BCS metric that is significantly greater than 1. Our model is much stronger than that of the CMT model and the duration of the generated music is generally consistent with that of the dance video.

The visualization results are presented in Fig. 2, where two dance videos are depicted. The red box represents the video frame corresponding to the timestamp of the generated music beat, while the blue box represents the real dance beat. It can be observed that there is a difference of only three video frames between them, indicating a high degree of alignment between the generated music and dance movements.

6 Discussion

In this paper, we constructed the first multi-instrument MIDI and dance paired dataset (D2MIDI), which can serve as a benchmark dataset for future research on generating background music for dance videos. We proposed the Dance2MIDI framework for multi-instrument MIDI generation. Dance2MIDI leverages the consistency of paired data to mitigate the weak correlation between music and video. As a two-stage generation framework, Dance2MIDI initially synthesizes a fundamental drum rhythm track utilizing the Transformer cross-attention mechanism, guided by dance condition information. Subsequently, the synthesis of the remaining audio tracks is structured as a sequence completion task. With the aid of the BERT model, we inpaint the remaining audio tracks.

In addition, some dance genres such as folk and ballet, may not exhibit the strong rhythmic elements characteristic of pop dance, and often utilize music without drums. In the Dance2MIDI framework, we initially generate the drum track for the entire piece using the Drum Rhythm Generator module. This drum part, serving as the cornerstone of the music's beat, is enriched with the transitions and dynamics of dance movements and assists in the generation of other tracks within the Multi-Track MIDI BERTGen module. For dances like folk and ballet, we opt to remove the drum track in the final stage through post-processing, resulting in music without drums. Adopting this pipeline enhances generalizability, making it applicable to various dance types.

However, there are still limitations to our work: due to the variability in shape, form, and mechanics of drum instruments [16], their performance is a major bottleneck for music quality, which we aim to address in future work. Additionally, Dance2MIDI is currently limited to the generation of soundtracks for single-person dances. Future work will include an in-depth exploration of the application of group dances. For the user study, the current qualitative assessment experiment is indeed limited by the number of participants. We plan to extend our coverage to diverse participant groups in the future. This will be achieved by randomly selecting participants, a method aimed at minimizing self-selection bias.

Acknowledgements

This work was supported by the National Social Science Fundation Art Project (No. 20BC040) and China Scholarship Council (CSC) Grant (No. 202306320525).

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] Cannataro M, Talia D. The knowledge grid. *Communications of the ACM*, 2003, 46(1): 89–93.

[2] Mastroianni C, Talia D, Verta O. A super-peer model for resource discovery services in large-scale grids. *Future Generation Computer Systems*, 2005, 21(8): 1235–1248.

[3] Aggarwal G, Parikh D. Dance2music: Automatic dance-driven music generation. *arXiv preprint arXiv:2107.06252*, 2021.

[4] Di S, Jiang Z, Liu S, Wang Z, Zhu L, He Z, Liu H, Yan S. Video background music generation with controllable music transformer. In *ACM Multimedia*, 2021, 2037–2045.

[5] Gan C, Huang D, Chen P, Tenenbaum JB, Torralba A. Foley Music: Learning to Generate Music from Videos. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI*, 2020, 758–775.

[6] Kao HK, Su L. Temporally guided music-to-body-movement generation. In *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, 147–155.

[7] Li R, Yang S, Ross DA, Kanazawa A. Ai choreographer: Music conditioned 3d dance generation with aist++. In *ICCV*, 2021, 13401–13412.

[8] Zhu Y, Olszewski K, Wu Y, Achlioptas P, Chai M, Yan Y, Tulyakov S. Quantized GAN for Complex Music Generation from Dance Videos. In *The European Conference on Computer Vision (ECCV)*, 2022.

[9] Han B, Peng H, Dong M, Xu C, Ren Y, Shen Y, Li Y. AMD Autoregressive Motion Diffusion. *arXiv preprint arXiv:2305.09381*, 2023.

[10] Kim J, Oh H, Kim S, Tong H, Lee S. A Brand New Dance Partner: Music-Conditioned Pluralistic Dancing Controlled by Multiple Dance Genres. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, 3490–3500.

[11] Lee H, Yang X, Liu M, Wang T, Lu Y, Yang M, Kautz J. Dancing to Music. In *NeurIPS 2019*, 2019, 3581–3591.

[12] Li B, Zhao Y, Zhelun S, Sheng L. Danceformer: Music conditioned 3d dance generation with parametric motion transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 2022, 1272–1279.

[13] Siyao L, Yu W, Gu T, Lin C, Wang Q, Qian C, Loy CC, Liu Z. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, 2022, 11050–11059.

[14] Briot JP, Pachet F. Deep learning for music generation: challenges and directions. *Neural Computing and Applications*, 2020, 32(4): 981–993.

[15] Ji S, Luo J, Yang X. A comprehensive survey on deep music generation: Multi-level representations, algorithms, evaluations, and future directions. *arXiv preprint arXiv:2011.06801*, 2020.

[16] Su K, Liu X, Shlizerman E. How Does it Sound? *NeurIPS*, 2021, 34: 29258–29273.

[17] Wang Z, Ma L, Zhang C, Han B, Wang Y, Chen X, Hong H, Liu W, Wu X, Zhang K. SongDriver2: Real-time Emotion-based Music Arrangement with Soft Transition. *arXiv preprint arXiv:2305.08029*, 2023.

[18] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[19] Devlin J, Chang MW, Lee K, Toutanova K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[20] Engel J, Agrawal KK, Chen S, Gulrajani I, Donahue C, Roberts A. Gansynth: Adversarial neural audio synthesis. *arXiv preprint arXiv:1902.08710*, 2019.

[21] Goel K, Gu A, Donahue C, Ré C. It's Raw! Audio Generation with State-Space Models, 2022.

[22] Oord Avd, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kalchbrenner N, Senior A, Kavukcuoglu K. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016.

[23] Dhariwal P, Jun H, Payne C, Kim JW, Radford A, Sutskever I. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[24] Kumar K, Kumar R, de Boissiere T, Gestin L, Teoh WZ, Sotelo J, de Brébisson A, Bengio Y, Courville AC. MelGAN: Generative Adversarial Networks for Conditional Waveform Synthesis. In *NeurIPS 2019*, 2019, 14881–14892.

[25] Vasquez S, Lewis M. Melnet: A generative model for audio in the frequency domain. *arXiv preprint arXiv:1906.01083*, 2019.

[26] Dong HW, Hsiao WY, Yang LC, Yang YH. Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[27] Huang CZA, Vaswani A, Uszkoreit J, Shazeer N, Simon I, Hawthorne C, Dai AM, Hoffman MD, Dinculescu M, Eck D. Music transformer. *arXiv preprint arXiv:1809.04281*, 2018.

[28] Muhamed A, Li L, Shi X, Yaddanapudi S, Chi W, Jackson D, Suresh R, Lipton ZC, Smola AJ. Symbolic music generation with transformer-gans. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 2021, 408–417.

[29] Joshi M, Chen D, Liu Y, Weld DS, Zettlemoyer L, Levy O. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 2020, 8: 64–77.

[30] Ren Y, He J, Tan X, Qin T, Zhao Z, Liu TY. Popmag: Pop music accompaniment generation. In *Proceedings of the 28th ACM international conference on multimedia*, 2020, 1198–1206.

[31] Liu J, Dong Y, Cheng Z, Zhang X, Li X, Yu F, Sun M. Symphony Generation with Permutation Invariant Language Model. In *ISMIR 2022*, 2022, 551–558.

[32] Pedersoli F, Goto M. Dance Beat Tracking from Visual Information Alone. In *ISMIR*, 2020, 400–408.

[33] Gillick J, Roberts A, Engel J, Eck D, Bamman D. Learning to Groove with Inverse Sequence Transformations, 2019.

[34] Raffel C. Learning-Based Methods for Comparing Sequences, with Applications to Audio-to-MIDI Alignment and Matching. 2016, 2016.

[35] Hawthorne C, Stasyuk A, Roberts A, Simon I, Huang CZA, Dieleman S, Elsen E, Engel J, Eck D. Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset. In *International Conference on Learning Representations*, 2019.

[36] Ferreira LN, Lelis LH, Whitehead J. Computer-Generated Music for Tabletop Role-Playing Games, 2020.

[37] Tsuchida S, Fukayama S, Hamasaki M, Goto M. AIST Dance Video Database: Multi-Genre, Multi-Dancer, and Multi-Camera Database for Dance Information Processing. In *ISMIR 2019*, 2019, 501–510.

[38] Gardner J, Simon I, Manilow E, Hawthorne C, Engel J. Mt3: Multi-task multitrack music transcription. *arXiv preprint arXiv:2111.03017*, 2021.

[39] Cao Z, Simon T, Wei SE, Sheikh Y. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017, 7291–7299.

[40] Sun K, Xiao B, Liu D, Wang J. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, 5693–5703.

[41] Lugaresi C, Tang J, Nash H, McClanahan C, Uboweja E, Hays M, Zhang F, Chang CL, Yong MG, Lee J, et al.. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019.

[42] Chen K, Tan Z, Lei J, Zhang SH, Guo YC, Zhang W, Hu SM. Choreomaster: choreography-oriented music-driven dance synthesis. *ACM Transactions on Graphics (TOG)*, 2021, 40(4): 1–13.

[43] Chen CFR, Fan Q, Panda R. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, 357–366.

[44] McFee B, Raffel C, Liang D, Ellis DP, McVicar M, Battenberg E, Nieto O. librosa: Audio and music signal analysis in python.

In *Proceedings of the 14th python in science conference*, volume 8, 2015, 18–25.

[45] Davis A, Agrawala M. Visual Rhythm and Beat. *ACM Trans. Graph.*, 2018, 37(4), doi:10.1145/3197517.3201371.

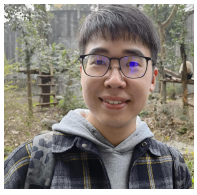
[46] Wu S, Yang Y. The Jazz Transformer on the Front Line: Exploring the Shortcomings of AI-composed Music through Quantitative Measures. In *Proceedings of the 21th International Society for Music Information Retrieval Conference, ISMIR 2020, Montreal, Canada, October 11-16, 2020*, 2020, 142–149.

[47] Shannon CE. A mathematical theory of communication. *The Bell system technical journal*, 1948, 27(3): 379–423.

[48] Newmarch J, Newmarch J. Fluidsynth. *Linux Sound Programming*, 2017: 351–353.

Author biography

Bo Han is a Ph.D. student in the College of Computer Science & Technology, Zhejiang University. His primary research interests are in computer vision and multimedia generation. Email: borishan815@zju.edu.cn



Yi Ren graduated from Chu Kochen Honors College, Zhejiang University with a bachelor's degree and from the Department of Computer Science and Technology, Zhejiang University with a master's degree, advised by Zhou Zhao. He won the Baidu Scholarship and ByteDance Scholars Program in 2020 and was selected as one of the top 100 AI Chinese New Stars and AI Chinese New Star Outstanding Scholar (10 candidates worldwide each year). He has published 50+ papers (citation 4113) at the top international AI conferences such as NeurIPS, ICML, ICLR, and KDD. He currently works at Bytedance AI Lab, Speech & Audio Team as a research scientist in Singapore, leading a fundamental audio/talkingface research group. His primary research interests are in TTS, music generation, speech translation and audio-driven talking face generation research.

Feilin Han graduated from the Department of Computer Science and Technology at Zhejiang University in 2019, with a Ph.D. degree majoring in Digital Art and Design. She graduated from Shandong University with a B.Eng. degree in 2014. Her research interests mainly lie in 3D Human Digitalization (Motion Capture, Animation, Modeling), Cinematic Virtual Reality (Virtual Cinematography, VR Animation), and Human-AI interaction solutions for Film PreViz and Virtual Production. She is currently an M.A. Supervisor in the Department of Film and TV Technology, Beijing Film Academy.