TM2D: Bimodality Driven 3D Dance Generation via Music-Text Integration

Kehong Gong[1*] Dongze Lian[1*] Heng Chang[2] Chuan Guo[3]
Xinxin Zuo[3] Zihang Jiang[1] Xinchao Wang[1†]
[1] National University of Singapore [2] Tsinghua University [3] University of Alberta

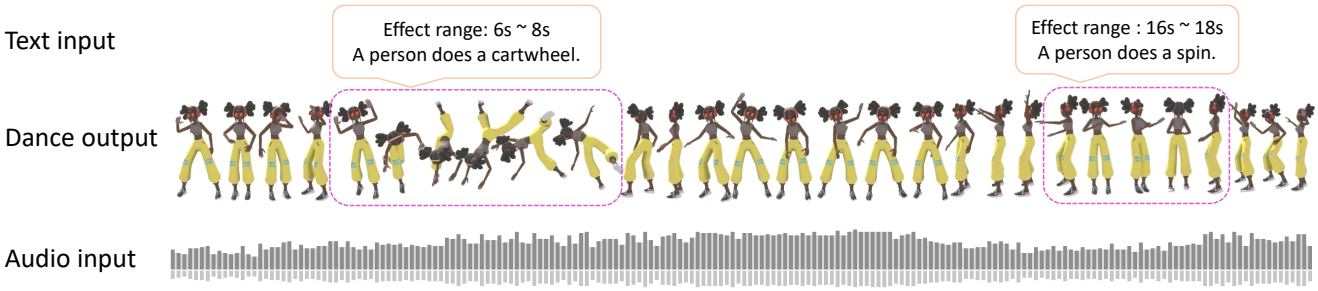


Figure 1: Generated 3D dance examples conditioned music and text with our method. Given an audio input and text input with a specific starting time and duration, our method is able to generate a sequence of dance motions that fit the music and text instruction. The character is from Mixamo [1].

Abstract

We propose a novel task for generating 3D dance movements that simultaneously incorporate both text and music modalities. Unlike existing works that generate dance movements using a single modality such as music, our goal is to produce richer dance movements guided by the instructive information provided by the text. However, the lack of paired motion data with both music and text modalities limits the ability to generate dance movements that integrate both. To alleviate this challenge, we propose to utilize a 3D human motion VQ-VAE to project the motions of the two datasets into a latent space consisting of quantized vectors, which effectively mix the motion tokens from the two datasets with different distributions for training. Additionally, we propose a cross-modal transformer to integrate text instructions into motion generation architecture for generating 3D dance movements without degrading the performance of music-conditioned dance generation. To better evaluate the quality of the generated motion, we introduce two novel metrics, namely Motion Prediction Distance (MPD) and Freezing Score, to measure the coherence and freezing percentage of the generated motion. Extensive experiments show that our approach can generate realistic and coherent dance movements conditioned on both text and music while maintaining comparable performance with the two single modalities. Code will be available at https://garfield-kh.github.io/TM2D/.

1. Introduction

The music-conditioned dance generation has become a topic of great interest in recent years. The ability to generate dance movements that are synchronized with music has numerous applications, such as behavior understanding, simulation, and benefiting the community of dancers and musicians [23, 8, 27, 40]. Although music has been used as a guidance to generate dance movements, another important modality cue, text (or language), which provides richer actions and more flexible motion guidance, and is studied in other tasks such as image classification [36], detection [14], segmentation [46], and text-driven image generation [38], has not been fully explored in dance generation. To this end, we first propose a novel task for generating 3D dance movements that simultaneously incorporate both text and music modalities, enabling the generated human to perform rich dancing movements in accordance with the music and text.

Designing a system pipeline for this bimodality driven 3D dance generation task is non-trivial. There exist two significant challenges to be considered: i) the existing datasets only cater to either music-driven (music2dance) [42, 5, 51, 27] or text-driven (text2motion) [34, 15] human motion generation, and no paired 3D dance generation dataset exists that takes into account both music and text. While building a new large-scale paired 3D dance dataset

*Equal contribution: gongkehong@u.nus.edu, dongze@nus.edu.sg
†Corresponding author: xinchao@nus.edu.sg

based on music and text is possible, it is time-consuming with fully annotated 3D human motion [27]; ii) the integration of text into music-conditioned dance generation requires a suitable architecture. However, existing methods that use music as a driving force to generate dance movements might result in temporal-freezing frames or fail to generalize to in-the-wild scenarios [27, 40]. Therefore, simple integration of text into the existing music-conditioned architecture might pose a risk of degraded dance generation quality in our new task.

To address the first challenge, we take advantage of existing music-dance and text-motion datasets for this new task. However, directly mixing motions from these two datasets would result in inferior performance since the motions from these two datasets are in completely different motion spaces. To overcome this, we propose to utilize a VQ-VAE architecture to project the motions into a consistent and shared latent space. In particular, we build up a shared codebook for all the motions from the training set, and motions from both datasets are now represented as discrete tokens that are implicitly constrained to fall into a shared latent space. For the second challenge, we propose to utilize a cross-modal transformer architecture that formulates both music2dance and text2motion as sequence-to-sequence translation tasks. This architecture directly translates audio and text features into motion tokens and enables bimodality driven ability by introducing a fusion strategy in the latent space with a shared motion decoder for both tasks. With the shared decoder, audio and text information can be efficiently fused during inference. Our entire cross-modal transformer architecture is both effective and efficient, allowing for the integration of text instructions to generate coherent 3D dance motions, as illustrated in Figure 1.

To better evaluate the coherence of generated dance in our task, we propose a new evaluation metric, Motion Prediction Distance (MPD), which measures the distance between the predicted motion and the ground truth at the time of integrating text, thereby providing a more accurate evaluation of the coherence of frames. Additionally, we introduce a Freezing Score that quantifies the percentage of temporal freezing frames in dance generation, which is a common problem in music-conditioned dance generation. To better evaluate the performance of our method in real-world scenarios, we also collect an in-the-wild music test set for evaluation. Our method successfully performs dance generation based on both text and music while maintaining comparable performance on the single modality tasks (music2dance, text2motion) compared with the state-of-the-art.

In summary, our contributions are as follows: i) We propose an interesting task of utilizing both music and text for 3D dance generation and propose a pipeline named TM2D (Text-Music to Dance) for this task. ii) Rather than collecting a new training set, we effectively combine the existing music2dance and text2motion datasets and employ a VQ-VAE framework to encode motions from all training sets to a shared feature space. iii) We propose a cross-modal transformer as well as a bimodal feature fusion strategy to encode both audio and text features, which is both effective and efficient. iv) We propose two new metrics, MPD and Freezing Score, which efficiently reflect the quality of generated motion. v) We successfully generate realistic and coherent dance based on both music and text instructions while maintaining comparable performance on the single modality tasks (music2dance, text2motion).

2. Related Work

2.1. Music to Dance

Music2Dance is typically divided into 2D and 3D dance generation and has been explored for many years. Recent methods model 3D dance generation from the perspective of network architecture. For instance, Lee *et al.* [24] explore the convolutional neural network in casual dilated setting. Lee *et al.* [23] propose a dance unit with Variational Auto-Encoder (VAE). Ren *et al.* [37] employ recurrent neural network (RNN) and the graph convolution network is introduced in [10]. As for the 3D dance generation, [49, 3, 22] implement convolutional neural network, [41, 12] apply adversarial learning in generated dance. [5, 42, 47, 52, 21] implement Long short-term memory (LSTM), and [8] implement motion graph with learned music/dance embedding through matching approach [9]. More recently, transformer has been applied in dance generation [26, 27, 40, 25]. However, the previous methods usually produce temporal freezing frames when generating the long sequences, or are difficult to generalize to in-the-wild music, thereby not satisfiable when directly used for bimodal driven 3D dance generation. In this paper, we design a cross modal transformer which is effective and efficient to integrate text instruction to generate the 3D dance.

2.2. Text to Motion

In addition to music-driven motion generation, text is also utilized as instructions to generate motions. Text2motion can be categorized into action label based motion generation and language description based motion generation. The action label based methods [17, 32] generate motion conditioned on action labels. Action2Motion [17] implement a GRU-VAE to iteratively generate the next frame based on the action label and previous frames. ACTOR [32] implements a transformer-VAE to encode and decode the whole pose sequence in one-shot. Since the action label based methods are restricted in a small set of action labels, language description based methods are proposed for more flexible motion generation. [35, 28, 2, 15, 16] formulate the text-to-motion task as a machine translation prob-

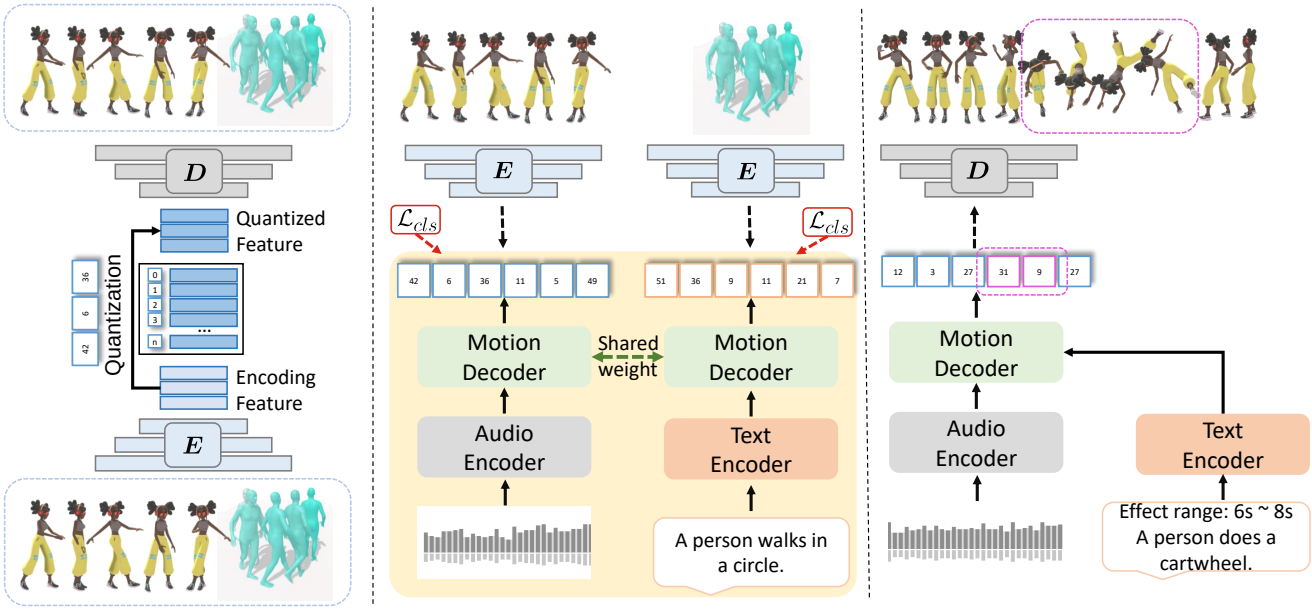


Figure 2: Our proposed pipeline for music-text conditioned 3D dance generation. Three stages from left to right: 3D human motion VQ-VAE, training stage of the cross-modal transformer, and inference stage of our pipeline. In the first stage, a VQ-VAE is trained with both motions from music2dance and text2motion data, which is then used to tokenize all motions. In the second stage, a dual path cross-modal transformer is employed for sequence2sequence translation tasks (*i.e.*, audio to motion tokens, text to motion tokens), with a shared motion decoder. In the third stage, given audio and text inputs, the audio and text encoders first extract the corresponding features, which are then fused (late fusion) in the motion decoder to generate dance condition on both music and text.

lem, others [48, 4, 11, 33] learns a joint embedding space of text and motion. Among them, [35, 28, 2, 48, 4, 11] use RNN encoder-decoder to learn the mapping between text and motion. [16, 15] use transformer encoder with RNN decoder, and TM2T [16] further introduces motion to text as an inverse alignment for text to motion. TEMOS [33] and TEACH [6] use transformer for both encoder and decoder for one-shot generation. Different from these text2motion methods, we focus on how to integrate two modalities (*i.e.*, music and text) together for 3D dance generation.

3. Method

3.1. System Pipeline

The overall pipeline is shown in Figure 2, which consists of three stages. The first stage employs a 3D human motion VQ-VAE to encode both the motions of the music2dance and text2motion datasets to a shared codebook encoded by multiple vectors such that each dance motion can be represented as a discrete motion token. After that, the combinations of music, text, and motion token are fed into a cross-modal transformer for training, which effectively learn how to predict a sequential motion tokens according to the previous those as well as music and text. In the third stage, a random starting motion token is generated and inputted to the cross-modal transformer with the given music and text for the sequential motion token prediction, which will be decoded by the pre-trained 3D human motion VQ-VAE in the first stage to generate 3D dance.

3.2. 3D Human Motion VQ-VAE

Since there are no paired 3D dance motions conditioned on both music and text, we try to tackle our task with the existing music2dance and text2motion datasets. Instead of directly mixing the motions from both datasets for training, we employ a VQ-VAE to project the motions into a consistent and shared latent space, which is represented with a codebook, as illustrated in Figure 2. To be specific, given a 3D human motion $M \in \mathbb{R}^{T \times d_m}$, where T is the time length and d_m is the dimension of the human motion, an encoder of VQ-VAE that consists of several 1-D convolutional layers projects M to a latent vector $z \in \mathbb{R}^{T' \times d}$, where $T' = \frac{T}{t}$ and t is the time interval for downsampling and d is the dimension of the latent vector. A learnable codebook $e \in \mathbb{R}^{K \times d}$ describes all latent variable features of the whole dataset, where K and d are the length and dimension of the codebook, respectively. A quantized latent vector $z_q \in \mathbb{R}^{T' \times d}$

will record the closest vector from codebook e as follows

$$z_{q,i} = \arg\min_{e_j \in e} \|z_i - e_j\| \in \mathbb{R}^d, \quad (1)$$

and the motion token $t_m \in \mathbb{R}^{T' \times 1}$ stores the index of the closest vector

$$t_{m,i} = \arg\min_{j} \|z_i - e_j\| \in \mathbb{R}^1. \quad (2)$$

The quantized z_q will be decoded with stacked convolutional layers to reconstruct the human motion $\hat{M}$, and t_m will be used in the training stage of the cross-modal transformer.

For the training of 3D human motion VQ-VAE, we follow the strategy in [44] and the total loss contains a reconstruction loss for human motion regression, a codebook loss for the dictionary learning, and a commitment loss to stabilize the training process:

$$\mathcal{L}_{vq} = \|\hat{M} - M\|_1 + \|\text{sg}[e] - e_q\|_2^2 + \beta\|e - \text{sg}[e_q]\|_2^2, \quad (3)$$

where $\text{sg}[\cdot]$ is 'stop gradient' and β is the factor term to adjust the weight of the commitment loss. A straight-through estimator is also employed to pass the gradient from the decoder to the encoder in back-propagation.

As we will show in Sec. 4, the motion tokens from both datasets encoded by 3D human motion VQ-VAE fall almost in the shared latent space, which shows the feasibility of using the separate music-conditioned and text-conditioned motion datasets for music-text conditioned 3D dance generation task.

3.3. Cross-modal Transformer

The cross-modal transformer contains an audio encoder, a text encoder, and a motion decoder. Since we use two separate datasets for our task, the cross-modal transformer is divided into two branches, where one for music2dance and the other for text2motion. It takes the audio feature, the text feature, and the motion token t_m encoded by 3D human motion VQ-VAE as inputs, and performs the sequence-to-sequence translation task to generate the future motion tokens.

Attention. Attention is introduced in Transformer [45] for natural language processing. A L-layer transformer typically consists of L transformer blocks, which contains a multi-head self-attention (MSA) and a feed forward network (FFN). Given the input $x \in \mathbb{R}^{N \times c}$, where N is the sequence length and c is the dimension of the input, the transformer block first maps it to keys $K \in \mathbb{R}^{N \times c}$, queries $Q \in \mathbb{R}^{N \times c}$, values $V \in \mathbb{R}^{N \times c}$ with the linear projections, and then a self-attention operation is performed by

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{c}})V. \quad (4)$$

The output is followed by a normalization layer and a FFN.

Audio encoder. Given a sequence of music, the audio encoder first extracts the raw audio features following Bailando [40], where a public audio processing toolbox, Librosa [20], is employed to obtain the *mel frequency cepstral coefficients (MFCC)*, *MFCC delta*, *constant-Q chromagram*, *tempogram* and *onset strength*. After that, an embedded layer followed by several transformer blocks containing self-attention operation *e.g.*, Eq. (4), is used to generate the processed audio features $f_a \in \mathbb{R}^{T' \times d}$.

Text encoder. Given a text instruction, we first extract the text token with a GloVe [31], and then an embedded layer followed by several transformer blocks is performed to obtain a processed text feature $f_t \in \mathbb{R}^{n \times d}$, where n is the length of the text feature.

Motion decoder. Motion decoder takes the motion tokens encoded by the 3D human motion VQ-VAE as inputs and outputs the future motion tokens, which has the same spirit as the sequence-to-sequence translation task. Since the information after time t is unknown at moment t, a masked MSA is first employed to interact with motions at different times. The mask has the shape of a simple upper triangular matrix and is performed in Eq. (4). For the moment t, only the motions before moment t are able to perform self-attention operations. Motion decoder also maps the extracted audio feature f_a and text feature f_t via audio encoder and text encoder to K and Q, to perform cross-modal attention with motion tokens. To enable the music-text conditional dance generation, (*i.e.*, feature fusion, we apply shared parameters), we use the motion decoder with shared parameters for model efficiency, as shown in Figure 2.

3.4. Details of Music-text Fusion

The detailed architecture is listed in Figure 3. Given the audio feature and text feature, the motion decoder first processes the past motion tokens with a self-attention layer followed by addition, normalization, a cross-attention with audio and text features separately, addition, normalization, and a feed-forward layer are performed. Such a procedure is repeated L (L = 6) times to build a typical L-layer transformer. Then we adopt a late fusion strategy to perform a weighted sum of features of audio and text at a specific time (*i.e.*, effect range). The weight curve is shown in Figure 3, where we slightly increase the weight of the text feature by a half cosine curve until a peak value of 0.8 at the beginning (20% time of effect range), and decrease it by a half cosine curve at the end (20% time of effect range). The weight of audio feature W_{audio} is 1 - W_{text} to ensure the feature keeps the same scale. With the fused feature, a linear projection layer and a softmax operation are applied to predict the music-text conditioned motion tokens, which are then decoded by the decoder of human motion VQ-VAE to ob-

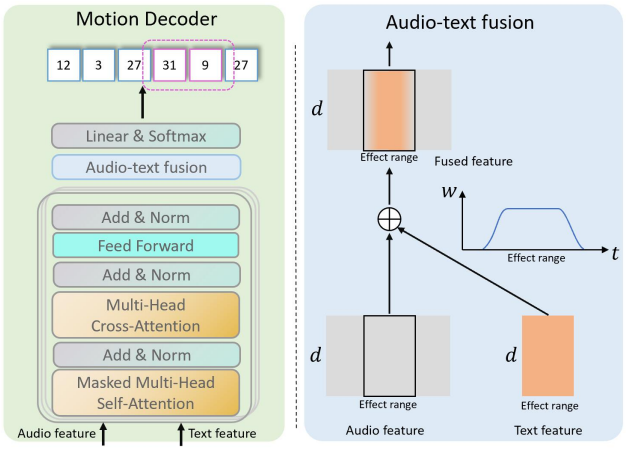


Figure 3: Detail of music-text feature fusion: Given the audio and text features from the encoder, the decoder processes them separately in the early layers. Then the audio-text fusion layer is applied by the weighted sum at the effect range with weight cure. Finally, a linear projection layer and a softmax operation are applied to predict the music-text conditioned motion tokens.

tain the music-text conditioned 3D dance sequences.

3.5. Training and Inference

Training. To train the 3D human motion VQ-VAE, we crop 64 frames with a sliding window from the original motion sequences as inputs, *i.e.*, T = 64, for both music2dance and text2motion datasets. We use a three-layer encoder and decoder so that the time interval of downsampling t is 8. We randomly sample motions from both datasets and employ the Adam optimizer with a batch size of 128 and a learning rate of $1e^{-4}$ to optimizer our 3D human motion VQ-VAE.

To train the cross-modal transformer, we split two branches into two streams to train the music2dance and text2motion tasks on both datasets. For the music2dance task, we use the same frequency as the motion token to sample the music vector so that the music feature has the same time length as the motion token. We perform the sequence to sequence translation with the motion decoder to obtain the prediction of motion tokens conditioned by the music input. As for the text2motion task, since the text and motion token have different lengths, we use a maximum text length of 84 for text2motion translation with a padding strategy. We employ 6 self-attention layers for the audio encoder, text encoder, and motion decoder with the hidden dimension of 512 and 8 heads. The cross entropy loss is adopted for both music2dance and text2motion tasks

$$\mathcal{L}_{cls} = -\frac{1}{m}\sum_{i=0}^{m-1}\sum_{j=0}^{C-1} y_{ij}\log \hat{y_{ij}}, \tag{5}$$

where $\hat{y}$ is the prediction of motion token and y is the ground-truth. m is the length of motion tokens and C is the number of classes of motion tokens, *i.e.*, the length of codebook of 3D human motion VQ-VAE. Both tasks are simultaneously optimized with a batch size of 64 and a learning rate of $1e^{-4}$.

Inference. Our aim is to generate 3D dance with music-text integration. Therefore, at the inference stage, we first fuse the music feature and text feature extracted by the audio encoder and text encoder with a weighted sum. Specifically, given an audio feature with a time length of T' and a text feature with the length of n, we first feed them into the motion decoder for future motion token prediction. Then we adopt the late fusion strategy to have a weighted sum of the generated motion tokens from both audio and text features at the duration of the integrated text, followed by a linear projection layer and a softmax operation, to obtain the combined motion tokens. Finally, the combined motion tokens are decoded into a 3D dance sequence after going through the decoder of our 3D human motion VQ-VAE model.

4. Experiments

4.1. Experimental Settings

Datasets. Since there are no paired 3D dance motions conditioned on both music and text for our task, we combine the existing music2dance and text2motion datasets. For the music2dance dataset, we employ the AIST++ dataset [27]. For the text2motion dataset, we employ the HumanML3D [15] dataset. Additionally, we also collect a new in-the-wild music dataset to evaluate the generalization ability of our method, which contains 82 wild music clips from Youtube. The total of 53 minutes duration (8x larger size than the AIST++ test set) and various styles and content of music, which faithfully lies out of the distribution of AIST++ (explained in supplementary material). The evaluation on our dataset better reflects the generalization ability of the models in real-world scenarios.

Baselines. Since we are the first to propose this bimodality driven 3D dance generation task, there are no existing methods for comparisons. We implement a traditional motion editing algorithm, slerp [39], which aligns the last generated frame of the dance with the first frame of the motion by text with a transition window of 10 frames, and applies spherical interpolation of quaternion to fill in the transition in between. To validate the effectiveness of our method on music2dance task, we compare our method to the previous music2dance works [26, 51, 19, 27, 40]. To validate the effectiveness of our method on text2motion task, we compare our method to the previous text2motion works [28, 4, 7, 11, 43, 23, 16].

Evaluation metrics. We adopt the same evaluation settings as suggested by FACT [27] and Bailando [40] to evaluate the dance generation quality, including Fréchet Inception Distances (FID) [18], Diversity, and Beat Align Score. For

Data	Method	Motion Quality		Motion Diversity		Freezing		Beat Align Score ↑	User Study
		FID_k ↓	$FID_g^\dagger$ ↓	Div_k ↑	$Div_g^\dagger$ ↑	PFF↓	AUC_f ↓		Our Method Wins
AIST ++	Ground-truth	17.10	10.60	8.19	7.45	0.00	0.00	0.2374	41.9%
	Li *et al*. [26]	86.43	43.46	6.85*	3.32	**0.00**	**0.00**	0.1607	98.3%
	DanceNet [51]	69.18	25.49	2.86	2.85	**0.00**	0.98	0.1430	87.2%
	DanceRevolution [19]	73.42	25.92	3.52	4.87	11.01	12.22	0.1950	70.5%
	FACT [27]	35.35	22.11	5.94	6.18	25.29	21.59	0.2209	83.3%
	Bailando [40]	28.16	9.62	7.83	6.34	14.91	13.25	**0.2332**	65.0%
	ours (only dance data)	23.94	**9.53**	7.69	4.53	**0.00**	**0.00**	0.2127	-
	ours	**19.01**	20.09	**9.45**	**6.36**	**0.00**	**0.00**	0.2049	–
Wild Audio	FACT [27]	70.36	20.23	7.33	**6.34**	32.64	27.21	0.2211	89.6%
	Bailando [40]	50.56	22.55	3.80	6.04	17.55	14.14	0.2166	70.8%
	ours (only dance data)	43.85	**13.08**	**8.515**	4.762	0.78	0.76	0.1998	–
	ours	**27.65**	20.34	7.88	5.27	**0.12**	**0.11**	**0.2290**	–

Table 1: Music Conditioned Dance Generation: quantitative results on AIST++ and Wild Audio test set. The best and runner-up values are bold and underlined, respectively. Among compared methods, "Li *et al*.", DanceNet and FACT are multiplexing the same results of AIST++ benchmark [26], while DanceRevolution [19] is followed from Bailando [40]. † FID_k and DIV_k are fetched from [27] while FID_g and DIV_g are fetched from [40].

the text2motion quality, we adopt the same evaluation settings as suggested by TM2T [16], including R-precision, Multimodal-Dist, FID, Diversity, and MultiModality.

In addition, we also propose two new evaluation metrics, Motion Prediction Distance (MPD) and Freezing Score. The former reflects the coherence of frames at the time of integrating text, and the latter reacts to the percentage of temporal freezing frames in the generated dance. For the Freezing Score, we introduce the Percentage of Freezing Frame (PFF) and AUC_f, where the PFF is defined by measuring the percentage of frozen frames with two criteria: 1) the maximum joint velocity blew a threshold (0.015m/s). 2) its duration exceeds a certain period (3s). The AUC_f is defined by area under curve of PFF in the threshold range from 0 to 0.03. As for MPD, it is defined as

$$\text{MPD} = \min_i \|f_i(M_{t_0 \to t_1}) - M_{t_1 \to t_2}\|_2, \qquad (6)$$

where M is the dance motions, f is a motion prediction model, $f_i(M_{t_0 \to t_1})$ is the i-th predicted possible future motion, and t_0, t_1, and t_2 are the timestamps. It means that the model predicts various potential possible motion from t_1 to t_2 with the motion from t_0 to t_1. If this distance is small (*i.e*., the generated dance $M_{t_1 \to t_2}$ lays in the possible future), then the generated dance motion $M_{t_1 \to t_2}$ is more coherent at the time of integrating text. We adopt the motion prediction model DLow [50] in this evaluation metric as it is designed for diverse potential furture motion prediction.

Perceptual Evaluation. Besides the above metric measuring, we also conduct extensive user studies on Amazon Mechanical Turk (AMT) to perceptually evaluate the visual effects of our generated 3D dance results. Particularly, given each pair of dance movements sampled from our method and others with the same music clip, we request 3 distinct users on AMT to present their preference regarding the music-dance alignment, motion realism, and mobility. We further set the bar of involved users that only the ones with Master recognition who also have finished more than 1,000 tasks with over 98% approve rate are considered. Overall, there are 55 users employed in our user studies that come from various regions, ages, races and gender. The results of the user study are more representative to show the effect of the generated dance in practice.

Data	Method	Motion Prediction Distance			User Study
		ft=10	ft=20	ft=30	Our Method Wins
AIST ++	GT	0.048	0.072	0.088	70.0%
	a2d	0.052	0.084	0.108	66.6%
	slerp [39]	0.088	0.122	0.135	73.3%
	ours	**0.049**	**0.080**	**0.102**	-
Wild Audio	a2d	0.052	0.088	0.113	60.0%
	slerp [39]	0.104	0.148	0.171	70.0%
	ours	**0.048**	**0.079**	**0.107**	-

Table 2: Music-text conditioned Dance Generation: quantitative results on AIST++ and wild audio test set. The best and runner-up values are bold and underlined, respectively.

4.2. Evaluation on Music-text Conditioned Dance Generation

We validate the results of music-text conditioned dance generation on the test set of AIST++ and our dataset, as illustrated in Table 2. Here we mainly evaluate the coherence of generated dance with MPD metric at the time where music meets text, *i.e.*, the time point where text starts to take effect. We measure the coherence of pure music2dance generation as a baseline named a2d to reflect the influence of with/without text instruction. We use the past 25 motion frames to predict the future 30 frames, and calculate the MPD from future frame (ft) = 10 to ft = 30, respectively. Our method consistently outperforms slerp [39] and a2d baseline in both datsets, and gains a similar result compared to the ground-truth shown in Table 2, which verifies the naturalness of our generated motion. More importantly, user study experiments show that our method generates more realistic 3D dance compared to dance generation only conditioned music, and 70.0% Win rate even compared to the ground-truth shown in Table 2. The results of our user study indicate that the music-text conditioned dance generation received high ratings from participants, highlighting the importance of considering audience preferences in evaluating dance quality.

4.3. Evaluation on Music Conditioned Dance Generation

To validate the effectiveness of our architecture, we quantitatively compare our method with state-of-the-art those for music conditioned 3D dance generation. The results on the test set of AIST++ are shown in Table 1. We can find that our method outperforms the previous ones in terms of motion diversity (Div_k and Div_g), while the performance of the motion quality (FID_g) and beat align score is inferior under the condition that we do not use seed motion from ground-truth compared with FACT [27] and Bailando [40].

The existing evaluation metrics are not sufficient to reflect the quality of generated dance in practice, which motivates us to propose a PFF and AUC_f to evaluate the percentage of freezing frames. It is worth noting that our architecture outperforms FACT [27] and Bailando [40] in terms of PFF and AUC_f, which shows that our method rarely generates frozen frames. Meanwhile, Li *et al*. [26] and DanceNet [51] also gain nearly zero freezing in PFF and AUC_f. The reason is that the generated dances of Li *et al*. [26] are highly jittery making its velocity variation extremely high, which is also reported in [27, 40], and leads to the non-freezing issue. And for DanceNet [51], it generates dance in a repeat motion pattern, nearly zero freezing but with low diversity. Furthermore, the results of user study show that the generated 3D dance is more visually realistic compared to other methods. Even in comparison to the ground truth, 41.9% of our generated dance is voted as the better in average. We also report the result trained with dance-only data, which shows comparable performance.

Methods	R Precision↑	FID↓	MM Dist↓	Diversity→	MModality↑
Real motions	0.511	0.002	2.974	9.503	-
Seq2Seq [28]	0.180	11.75	5.529	6.223	-
Language2Pose [4]	0.246	11.02	5.296	7.676	-
Text2Gesture [7]	0.165	5.012	6.030	6.409	-
Hier [11]	0.301	6.532	5.012	8.332	-
MoCoGAN [43]	0.037	94.41	9.643	0.462	0.019
Dance2Music [23]	0.033	66.98	8.116	0.725	0.043
TM2T baseline(T) [16]	0.351	1.669	4.046	9.632	4.352
TM2T [16]	**0.424**	1.501	3.467	8.589	2.424
TM2D (t2m)	0.319	**1.021**	4.098	**9.513**	4.139
TM2D (LFR 0.8)	0.300	1.105	**4.307**	8.887	**4.443**

Table 3: The result of text2motion, only mean reported.

In addition to the evaluation on the test set of AIST++, we also show the experimental results on our in-the-wild dataset in Table 1. Our method outperforms FACT [27] and Bailando [40] in almost all metrics except for FID_g Div_g. We can observe that both FACT [27] and Bailando [40] shows a large performance drop in terms of FID_k, while ours maintain a small change. This is because FACT [27] and Bailando [40] requires seed motion, however, there is no ground-truth for in-the-wild scenario. With random sampled seed motion or token, their methods are not adapted well for the in-the-wild scenarios. One can also notice that there exists freezing in our method on the in-the-wild dataset but the percentage of frozen frames is zero on the test set of AIST++, which results from the different distributions of AIST++ and our dataset. It also shows that our dataset is more challenging. We also report the result trained with dance-only data, which shows the advantage of our mix-training strategy in the wild scenario.

4.4. Evaluation on Text Conditioned Motion Generation.

We evaluate our text2motion approach in two different scenarios: inference with text only (t2m), and inference with text and music feature fusion. Table 3 shows the results. In the text-only setting, our approach achieves comparable performance with TM2T baseline(T) [16], which demonstrates that the mixed data/tasks training does not affect the quality of text2motion generation. We then apply our late fusion method by randomly sampling a music clip (from the whole AIST++ test set) and a text (from the whole t2m test set), and evaluate the generated dance clips following the same protocol as TM2T [16]. As Table 3 indicates, the text-dance consistency remains acceptable with a late fusion ratio of 0.8. For more details on the relation between the late fusion rate and text2motion result, please refer to the supplementary material.

4.5. Qualitative Results

We also show the qualitative results of our method for music-text conditioned 3D dance generation in Figure 4,

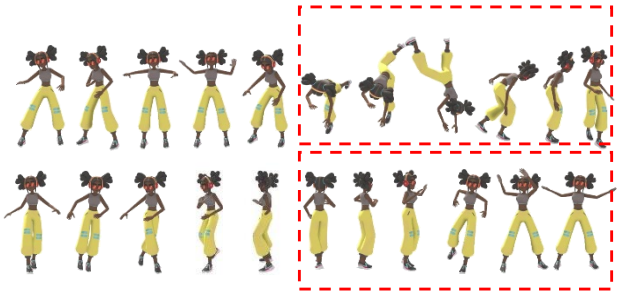


Figure 4: **Same music different action.** These two generated dance share same music and effect range (from 6s to 8s), with different text instruction: "A person does a cartwhee" (top), "A person is spinning with arms spread out" (bottom).

Figure 5 and Figure 6. Specifically, Figure 4 shows the generated 3D dance with the same music sequence but with different text instructions. Figure 5 shows the generated 3D dance with the same text instruction but with a different start time. Figure 6 shows the generated 3D dance with the same text instruction but with different durations. As one can see, our approach maintains plausible visual results according to the text instructions for all three cases, which confirms that our approach is more flexible.

4.6. The Impact of Mixed Data

We employ two datasets (AIST++ and HumanML3D) to train our 3D human motion VQ-VAE so that the motion tokens drop in the same latent space. To empirically show this point, we conduct experiments to count the number of shared motion tokens. For a trained 3D human motion VQ-VAE with AIST++ and HumanML3D, there are 855 vectors and 912 vectors from codebook used to construct the motion token in AIST++ and HumanML3D. The total number of vectors contained in the codebook is 1024. among them, 846 vectors (98.9% in AIST++ and 92.8% in HumanML3D) are shared to generate the motion tokens, which shows the feasibility of using the separate music-conditioned and text-conditioned motion datasets for music-text conditioned 3D dance generation task. We further verify the latent space of the motions by VQ-VAE, and we perform t-SNE visualizations of the raw motion distribution (before VQ-VAE) and feature distribution encoded by VQ-VAE (after VQ-VAE) from a2d motion and t2m motion in Figure 7. Two motion datasets are mixed successfully, which provides the potential integration of bimodality dance generation. Refer to the supplementary materials for more details.

4.7. Model Efficiency

We also show the efficiency of our architecture in Table 4. In the inference stage, the complete model parameters

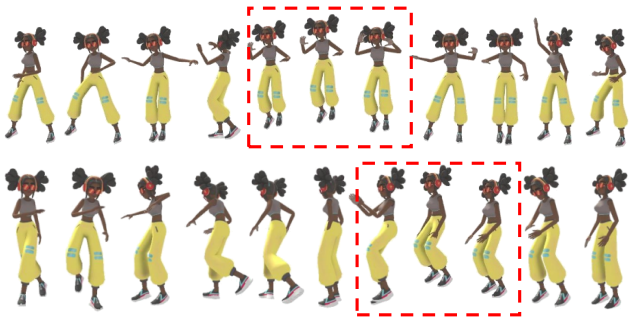


Figure 5: **Same action different start time point.** These two generated dance share the same music and text instruction ("A person jumps up and down") but different effect range: from 6 to 8s (top), from 7 to 9s (bottom).

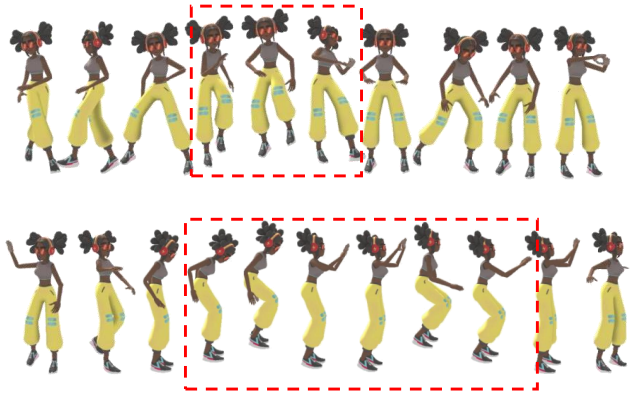
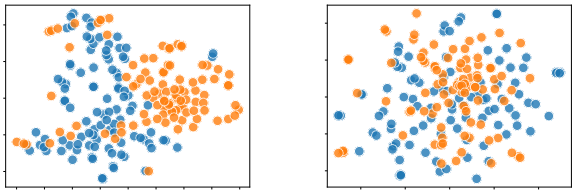


Figure 6: **Same action different duration.** These two generated dance share the same music and text instruction ("A person is keeping jumping") and action start time point (*i.e.*, 6s) but different effect duration: 2s (top), 3s (bottom, *i.e.*, jumped twice).



(a) Before VQ-VAE (b) After VQ-VAE

Figure 7: Motion t-SNE before and after VQ-VAE (orange: motions in AIST++, blue: motions in HumanML3D).

and the inference time for 1s music are compared. Our architecture requires only about half model parameters and inference time compared to Bailando [40], which attributes that we do not need to encode separate features for the upper and lower half body. Compared to FACT [27], our architectures significantly reduce the inference time. The main reason is that FACT employs two transformers for the mu-

Method	Parameters	Inference time
FACT [27]	120M	8.300s/(1s music)
Bailando [40]	152M	0.236s/(1s music)
ours	72M	0.143s/(1s music)

Table 4: Comparisons of parameters and inference time.

sic, dance motion, and a cross-modal transformer, but we formulate it as the standard sequence-to-sequence translation task with one transformer only, which establishes the advantage of our architecture.

5. Conclusion

In this paper, a new task that simultaneously integrates both music and text instruction for 3D dance generation is proposed. Due to the lack of the paired 3D dance motions conditioned on both music and text, we resort to two existing datasets, *i.e.*, music2dance and text2motion, to perform this task and employ a 3D human motion VQ-VAE to project the motions of the two datasets into a latent space so that the two datasets with different distributions can be effectively mixed for training. Moreover, we also propose a cross modal transformer architecture to generate 3D dance without degrading the performance of music conditioned dance generation. A new in-the-wild dataset is collected to evaluate the model's generalization ability to real-world scenarios and an evaluation metric is specifically designed for our task. Extensive experiments show that our method can generate dance motion that matches both music and text in a realistic and coherent way while maintaining comparable performance on two single modalities, music2dance and text2motion).

References

[1] mixamo. https://www.mixamo.com/. Accessed: 2023-03-07. 1

[2] Hyemin Ahn, Timothy Ha, Yunho Choi, Hwiyeon Yoo, and Songhwai Oh. Text2action: Generative adversarial synthesis from language to action. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5915–5920. IEEE, 2018. 2, 3

[3] Hyemin Ahn, Jaehun Kim, Kihyun Kim, and Songhwai Oh. Generative autoregressive networks for 3d dancing move synthesis from music. *IEEE Robotics and Automation Letters*, 5(2):3501–3508, 2020. 2

[4] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019. 3, 5, 7

[5] Omid Alemi, Jules Françoise, and Philippe Pasquier. Groovenet: Real-time music-driven dance movement generation using artificial neural networks. In *Workshop on Machine Learning for Creativity, 23rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, volume 8, page 26, 2017. 1, 2

[6] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. Teach: Temporal action compositions for 3d humans. In *International Conference on 3D Vision (3DV)*, 2022. 3

[7] Uttaran Bhattacharya, Nicholas Rewkowski, Abhishek Banerjee, Pooja Guhan, Aniket Bera, and Dinesh Manocha. Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. In *IEEE Virtual Reality and 3D User Interfaces (VR)*, pages 1–10. IEEE, 2021. 5, 7

[8] Kang Chen, Zhipeng Tan, Jin Lei, Song-Hai Zhang, Yuan-Chen Guo, Weidong Zhang, and Shi-Min Hu. Choreomaster: choreography-oriented music-driven dance synthesis. *ACM Transactions on Graphics (TOG)*, 40(4):1–13, 2021. 1, 2

[9] Rukun Fan, Songhua Xu, and Weidong Geng. Example-based automatic music-driven conventional dance motion synthesis. *IEEE transactions on visualization and computer graphics*, 18(3):501–515, 2011. 2

[10] Joao P Ferreira, Thiago M Coutinho, Thiago L Gomes, José F Neto, and et al. Learning to dance: A graph convolutional adversarial network to generate realistic dance motions from audio. *Computers & Graphics*, 94:11–21, 2021. 2

[11] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1396–1406, 2021. 3, 5, 7

[12] Shiry Ginosar, Amir Bar, Gefen Kohavi, Caroline Chan, Andrew Owens, and Jitendra Malik. Learning individual styles of conversational gesture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3497–3506, 2019. 2

[13] Deepak Gopinath and Jungdam Won. fairmotion - tools to load, process and visualize motion capture data. Github, 2020. 12

[14] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. In *International Conference on Learning Representations*, 2022. 1

[15] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5152–5161, 2022. 1, 2, 3, 5, 12

[16] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *European Conference on Computer Vision*, pages 580–597, 2022. 2, 3, 5, 6, 7, 12

[17] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 2

[18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, volume 30, 2017. 5, 12

[19] Ruozi Huang, Huang Hu, Wei Wu, Kei Sawada, Mi Zhang, and Daxin Jiang. Dance revolution: Long-term dance generation with music via curriculum learning. In *International Conference on Learning Representations*, 2021. 5, 6

[20] Yanghua Jin, Jiakai Zhang, Minjun Li, Yingtao Tian, Huachun Zhu, and Zhihao Fang. Towards the automatic anime characters creation with generative adversarial networks. *arXiv preprint arXiv:1708.05509*, 2017. 4

[21] Hsuan-Kai Kao and Li Su. Temporally guided music-to-body-movement generation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 147–155, 2020. 2

[22] Kosmas Kritsis, Aggelos Gkiokas, Aggelos Pikrakis, and Vassilis Katsouros. Danceconv: Dance motion generation with convolutional networks. *IEEE Access*, 10:44982–45000, 2022. 2

[23] Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. Dancing to music. *Advances in neural information processing systems*, 32, 2019. 1, 2, 5, 7

[24] Juheon Lee, Seohyun Kim, and Kyogu Lee. Listen to dance: Music-driven choreography generation using autoregressive encoder-decoder network. *arXiv preprint arXiv:1811.00818*, 2018. 2

[25] Buyu Li, Yongchi Zhao, Shi Zhelun, and Lu Sheng. Danceformer: Music conditioned 3d dance generation with parametric motion transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1272–1279, 2022. 2

[26] Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. Learning to generate diverse dance motions with transformer. *arXiv preprint arXiv:2008.08171*, 2020. 2, 5, 6, 7

[27] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13401–13412, 2021. 1, 2, 5, 6, 7, 8, 9, 12

[28] Angela S. Lin, Lemeng Wu, Rodolfo Corona, Kevin W. H. Tai, Qixing Huang, and Raymond J. Mooney. Generating animated videos of human activities from natural language descriptions. 2018. 2, 3, 5, 7

[29] Meinard Müller, Tido Röder, and Michael Clausen. Efficient content-based retrieval of motion capture data. In *SIGGRAPH*, pages 677–685. 2005. 12

[30] Kensuke Onuma, Christos Faloutsos, and Jessica K Hodgins. Fmdistance: A fast and effective distance function for motion capture data. In *Eurographics*, pages 83–86, 2008. 12

[31] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 4

[32] Mathis Petrovich, Michael J Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10985–10995, 2021. 2

[33] Mathis Petrovich, Michael J Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision*, page 480–497, 2022. 3

[34] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big data*, 4(4):236–252, 2016. 1

[35] Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems*, 109:13–26, 2018. 2, 3

[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1

[37] Xuanchi Ren, Haoran Li, Zijian Huang, and Qifeng Chen. Self-supervised dance video synthesis conditioned on music. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 46–54, 2020. 2

[38] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1

[39] Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 245–254, 1985. 5, 6, 7, 14

[40] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11050–11059, 2022. 1, 2, 4, 5, 6, 7, 8, 9, 12, 14

[41] Guofei Sun, Yongkang Wong, Zhiyong Cheng, Mohan S Kankanhalli, Weidong Geng, and Xiangdong Li. Deepdance: music-to-dance motion choreography with adversarial learning. *IEEE Transactions on Multimedia*, 23:497–509, 2020. 2

[42] Taoran Tang, Jia Jia, and Hanyang Mao. Dance with melody: An lstm-autoencoder approach to music-oriented dance synthesis. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1598–1606, 2018. 1, 2

[43] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1526–1535, 2018. 5, 7

[44] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 4

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[46] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18134–18144, 2022. 1

[47] Nelson Yalta, Shinji Watanabe, Kazuhiro Nakadai, and Tetsuya Ogata. Weakly-supervised deep recurrent neural networks for basic dance step generation. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019. 2

[48] Tatsuro Yamada, Hiroyuki Matsunaga, and Tetsuya Ogata. Paired recurrent autoencoders for bidirectional translation between robot actions and linguistic descriptions. *IEEE Robotics and Automation Letters*, 3(4):3441–3448, 2018. 3

[49] Zijie Ye, Haozhe Wu, Jia Jia, Yaohua Bu, Wei Chen, Fanbo Meng, and Yanfeng Wang. Choreonet: Towards music to dance synthesis with choreographic action unit. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 744–752, 2020. 2

[50] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, pages 346–364. Springer, 2020. 6

[51] Wenlin Zhuang, Congyi Wang, Jinxiang Chai, Yangang Wang, Ming Shao, and Siyu Xia. Music2dance: Dancenet for music-driven dance generation. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 18(2):1–21, 2022. 1, 5, 6, 7

[52] Wenlin Zhuang, Yangang Wang, Joseph Robinson, Congyi Wang, Ming Shao, Yun Fu, and Siyu Xia. Towards 3d dance motion synthesis and control. *arXiv preprint arXiv:2006.05743*, 2020. 2

APPENDIX:

Abstract

This supplementary material provides more details on the following aspects of our study: i) The dataset we used; ii) The evaluation metrics we employed; iii) The impact of mixed data for shared motion token space; iv) The collected data distribution; v) The effect of music-text fusion weight; vi) The reason why our dance has less freeze issue; vii) More visualizations of our results.

A. Detail of dataset

For the music2dance dataset, we employ the AIST++ dataset [27], which contains 30 subjects and 10 dance genres. There are 992 pieces of 3D human pose sequence, of which 952 are used for training and the rest are used for evaluation.

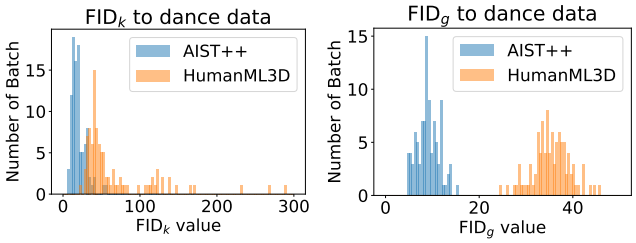
For the text2motion dataset, we employ the HumanML3D [15] dataset, which is a large-scale 3D human motion dataset that covers a broad range of human actions such as locomotion, sports, and dancing. It consists of 14,616 motions and 44,970 text descriptions. Each motion clip comes with at least 3 descriptions. For the joint training of both datasets, we sample the motions with 60 frames per second (FPS) to keep the time consistency with the AIST++ dataset, resulting in duration ranges from 2 to 10 seconds.

To evaluate the generalization ability of our method, we also collect a new dataset of music clips from Youtube that are not included in AIST++. This dataset consists of 82 clips with a total duration of 53 minutes, which is eight times larger than the AIST++ test set. The clips cover various styles and content of music, which are out of the distribution of AIST++.

B. Evaluation metrics.

We follow FACT [27] and Bailando [40] to quantitatively measure the quality of generated dances, the diversity of motions and the beat alignment of the music and the generated motions. In concrete, for the dance quality, we calculate the Fréchet Inception Distances (FID) [18] between the generated 3D dance and all motions of the AIST++ dataset on kinetic features [30] (denoted as 'k') and geometric features [29] (denoted as 'g') extracted by [13] to measure the quality of generated dances. We also follow [27] to calculate the average feature distance of generated motion to measure the diversity of motions. The average distance between the music beat and its closest dance beat is defined as the Beat Align Score as follows

$$\frac{1}{|B^m|}\sum_{b^m \in B^m} \exp\left\{-\frac{\min_{b^d \in B^d} \|b^d - b^m\|^2}{2\sigma^2}\right\}, \quad (7)$$



(a) Difference in FID$_k$. (b) Difference in FID$_g$.

Figure 8: FID$_k$ and FID$_g$ with difference batches in Experiment A.

where B^d and B^m are the dance beats and music beats, respectively. σ is a normalized parameter that we set to be $\sigma = 3$ in our experiments.

For the text2motion quality, we follow the same setting suggested by TM2T [16]: R-precision and Multimodal-Dist quantify the relevancy between the generated motions and the input prompts; FID computes the distance between the generated and ground truth distributions (in latent space); Diversity evaluates the variation of the generated motions; and MultiModality estimates the variance for a single prompt

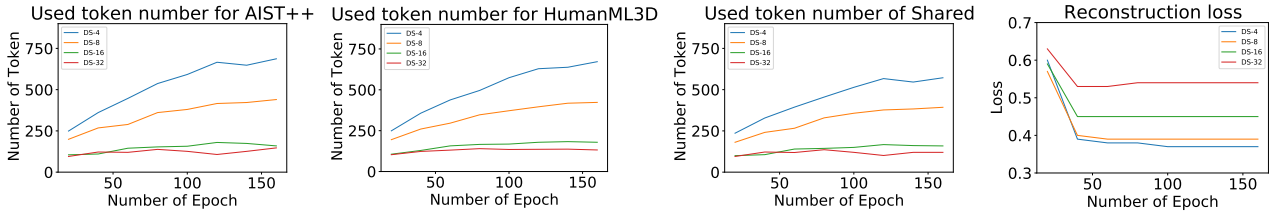
We also introduce two new evaluation metrics: Percentage of Freezing Frame (PFF) and Motion Prediction Distance (MPD). PFF measures the degree of freezing in the generated dance, while MPD assesses the coherence of frames when text is integrated.

C. The Impact of Mixed Data

As mentioned in the main text, a direct combination of the music2dance (AIST++ [27]) and text2motion (HumanML3D [15]) in the motion space might be sub-optimal for training because the motions from these two datasets fall in completely different spaces. In contrast, we project the motions into a consistent and shared latent space with a human motion VQ-VAE architecture. To show the effectiveness of the proposed method quantitatively, we design two experiments as follows.

- Experiment A: we random sample 100 batches of data (same size as AIST++ test set) from both datasets, and measure the FID between the random batch and the whole dance data.
- Experiment B: we sample 30% of the original data from both datasets and train them with a human motion VQ-VAE of different downsample rate (4, 8, 16, 32).

In experiment A, Figure 8 shows the distribution of FID result from both datasets. From Figure 8, we can observe



(a) token-of-AIST++ (b) token-of-ml3d (c) token-of-share (d) val-loss

Figure 9: Shared tokens (latent space) with a human motion VQ-VAE architecture in Experiment B.

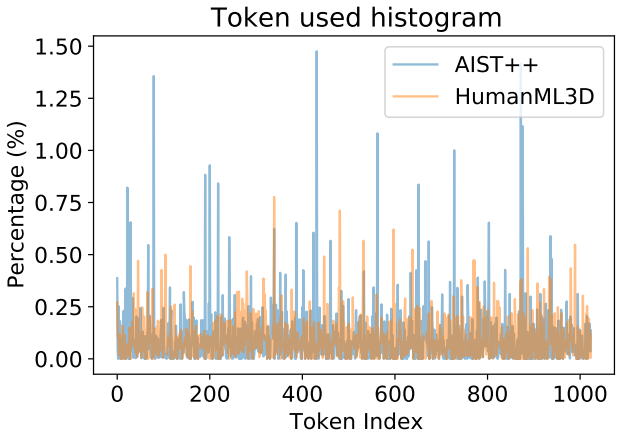


Figure 10: Token used histogram, histogram are normalized by the total frame from each dataset.

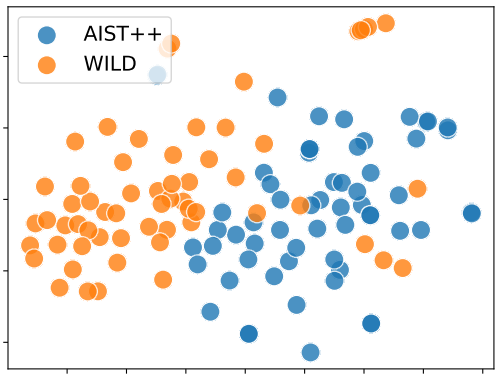


Figure 11: audio-t-SNE of datasets (orange: AIST++, blue: our dataset)).

that there is a distinct difference between two datasets on geometric feature, and a small overlap in kinetic feature.

In experiment B, from the Figure 9, we have the following three findings: i) Figure 9 (a) and Figure 9 (b) show that the tokens used of each dataset will be increasing with the training epoch. ii) In Figure 9 (c), the shared token number is also increasing together with it from both dataset. iii) The lower the downsample rate, the higher the used token number and shared token number, with smaller reconstruction loss (val loss). Consider that the lower the downsample rate, the longer the tokenized sequence for transformer in the second step of our pipeline. We choose downsample rate of 8, (a relatively small val loss, rich shared token number, and relatively short tokenized sequence length).

From Figure 10, we can see that both datasets almost share one codebook when motions are encoded with a VQ-VAE. Specifically, the total number of vectors contained in the codebook is 1024, 855 vectors and 912 vectors of which are used to construct the motions in AIST++ and HumanML3D, respectively. 846 vectors (98.9% in AIST++ and 92.8% in HumanML3D) are shared to generate the motion tokens, which is much better than the feature distance from Figure 8.

D. The analysis of the collected dataset

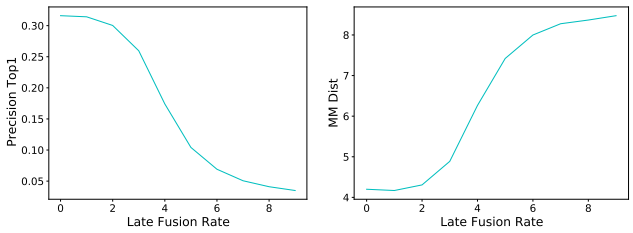
To verify the domain gap between source music and wild music, We sample the music features extracted by the Librosa (used in framework training) and plot a t-SNE in Figure 11. Two music datasets lay on two different distributions with a few overlaps, which shows the generalization ability of our method. The inferior results in Table 1 (main text) compared with our mix training show that mix gains better generalization performance.

E. The fusion weight and text2motion result

We further explore the effect of late fusion rate (LFR), as shown in Fig 12, with the increasing of LFR, the MM distance and Top 1 precision get worse. To balance the feature content, we choose late fusion rate of 0.8.

F. Analysis of the freeze improvement.

Since our method gains better result in freeze issue, we hyposthesis the improvement is bring by both the archtecture design and mix training method. We report the PFF in



(a) top-1 (b) MM dist

Figure 12: The effect of LFR with t2m resulst.

train	AIST++		mix data	
test	AIST++	wild	AIST++	wild
1	4.08 / 4.08	3.76 / 3.40	5.67 / 4.88	1.21 / 0.87
10	1.31 / 1.38	0.61 / 0.63	2.58 / 2.10	0.10 / 0.12
100	0.00 / 0.00	0.78 / 0.75	0.00 / 0.00	0.12 / 0.11

Table 5: PFF/AUC$_f$ with topk=1, 10, 100.

Table 5. In architecture, we sample tokens from the top-k tokens with the highest probability, instead of choosing the one with maximum probability as Bailando [40], which reduces the PFF. With extra HumanML3D data, the share motion decoder learns more motion sequence statics. Thus the PFF further improved. Thus both architecture and extra data mix training improve the PFF (AUC same).

G. More Visualizations of Our Results

We also show more visualizations of our results in the attached 'demo.mp4' file, which contains the following contents.

- Comparisons with other music2dance methods in AIST++ test set and our in-the-wild dataset.
- Our results with the same music, different actions / time / durations.
- Comparisons with Slerp [39] for music-text conditioned dance generation.

From these videos, we can find that our results outperform other methods and are more realistic.