arXiv:2101.08779v2 [cs.CV] 2 Feb 2021

Learn to Dance with AIST++: Music Conditioned 3D Dance Generation

Ruilong Li*[1] Shan Yang*[2] David A. Ross[2] Angjoo Kanazawa[2,3]

[1]University of Southern California [2]Google Research [3]University of California, Berkeley

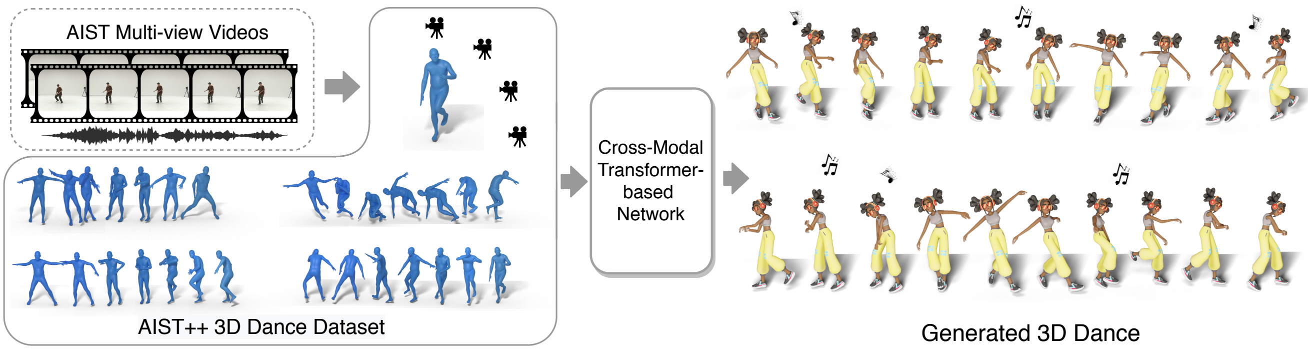

Figure 1: **Learning to Generate 3D Dance with Our AIST++ Dataset.** We present a new 3D dance dataset, AIST++, which contains 3D motion reconstructed from real dancers paired with music (left), and use this to train a novel cross-modal transformer-based architecture that is able to generate realistic 3D dance motion given music. Our approach generates realistic smooth dance motion in 3D with full translation, which allow applications such as automatic motion retargeting to a novel character (right). Here we use a character from Mixamo [1]

Abstract

In this paper, we present a transformer-based learning framework for 3D dance generation conditioned on music. We carefully design our network architecture and empirically study the keys for obtaining qualitatively pleasing results. The critical components include a deep cross-modal transformer, which well learns the correlation between the music and dance motion; and the full-attention with future-N supervision mechanism which is essential in producing long-range non-freezing motion. In addition, we propose a new dataset of paired 3D motion and music called AIST++, which we reconstruct from the AIST multi-view dance videos. This dataset contains 1.1M frames of 3D dance motion in 1408 sequences, covering 10 genres of dance choreographies and accompanied with multi-view camera parameters. To our knowledge it is the largest dataset of this kind. Rich experiments on AIST++ demonstrate our method produces much better results than the state-of-the-art methods both qualitatively and quantitatively. See project page and dataset at https://google.github.io/aichoreographer.

* equal contribution. Work performed while Ruilong was an intern at Google.

1. Introduction

The ability to dance by composing movement patterns that align to musical beats is a fundamental aspect of human behavior. Dancing is an universal language found in all cultures [49], and today, many people express themselves through dance on contemporary online media platforms. The most watched videos on YouTube are dance-centric music videos such as “Baby Shark Dance”, and “Gangnam Style” [74], making dance a more and more powerful tool to spread messages across the internet. However, dancing is a form of art that requires practice—even for humans, professional training is required to equip a dancer with a rich repertoire of dance motions to create an expressive choreography. Computationally, this is even more challenging as the task requires the ability to generate a continuous motion with high kinematic complexity that captures the the non-linear relationship with the accompanying music.

In this work, we address these challenges by presenting a novel cross-modal transformer-based learning framework and a new 3D dance motion dataset called AIST++ that can be used to train a model that generates 3D dance motion conditioned on music. Specifically, given a piece of music and a short (2 seconds) seed motion, our model is able to generate a long sequence of realistic 3D dance motions.

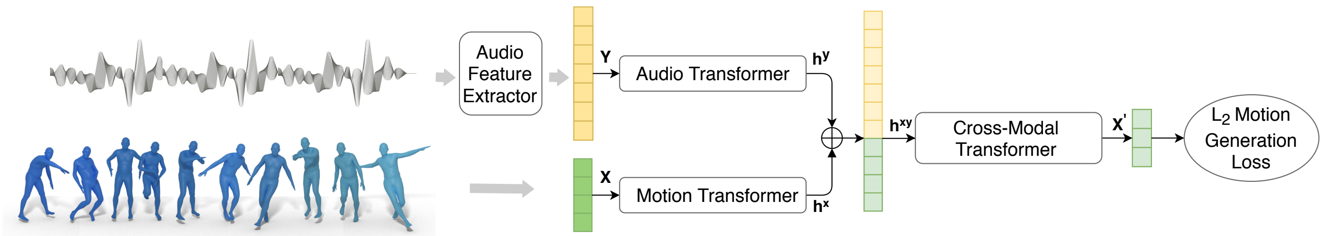

Figure 2: **Cross-Modal 3D Motion Generation Overview.** Our proposed cross-modal learning framework takes in a music piece and a 2-second sequence of seed motion, then generates long-range future motions that correlates with the input music.

Our model effectively learns the music-motion correlation and can generate dance sequences that varies for different input music. We represent dance as a 3D motion sequence that consists of joint rotation and global translation, which enables easy transfer of our output for applications such as motion retargeting as shown in Figure 1.

For the learning framework, we propose a novel transformer based cross-modal architecture for generating 3D motion conditioned on music. We build on the recent attention based networks [16, 65, 3, 75], which have shown to be effective for long sequence generation especially, and take inspiration from the cross-modal literature in vision and language [75] to design a framework that uses three transformers: one for audio sequence representation, one for motion representation and one for cross-modal audio-and-motion correspondence. The motion and audio transformer encode the input sequences, while the cross-modal transformer learns the correlation between these two modalities and generates future motion sequences.

We also carefully design our novel cross-modal transformer to be auto-regressive but with full-attention and future-N supervision, which are shown to be key for preventing 3D motion from freezing or drifting after several iteration as reported in prior works on 3D motion generation [4, 3]. The resulting model generates different dance sequences for different music, while generating long-term realistic motion that does not suffer from freezing of drifting at inference time.

In order to train the proposed model, we also address the problem of data. While there are a few motion capture datasets of dancers dancing to music, collecting mocap data requires heavily instrumented environments making these datasets severely limited in the number of available dance sequences, dancer and music diversity. As such, we propose a new dataset called AIST++, which we build on from the existing multi-view dance video database called AIST [83]. We use the multi-view information to recover reliable 3D motion from this data. Note that while this database has multi-view shots, the cameras are not calibrated, making 3D reconstruction a non-trivial challenge. The resulting AIST++ dataset contains up to 1.1M frames of 3D dance motions accompanied with music, which to our knowledge is the largest dataset of such kind. AIST++ also spans 10 music genres, 30 subjects, and 9 video sequence per dance with recovered camera intrinsics, which has ample potential to be useful for other human body and motion research. This dataset is available at https://google.github.io/aistplusplus_dataset/.

We conduct a thorough set of experiments comparing our approach to two prior works quantitatively, qualitatively, and with user studies. We propose novel metrics to better evaluate 3D motion conditioned on music, and ablate our models to identify key aspects of our architecture. The resulting learning framework has a wide variety of applications, such as the development of exercise tools where users may learn to dance by observing the dance sequence from multiple angles with their own choice of music. Our computational model is also useful for content creation and animation where the generated 3D motion can be directly transferred to a novel 3D character as illustrated in Figure 1.

2. Related Work

3D Human Motion Synthesis Realistic and controllable 3D human motion synthesis from past motion has long been studied. Earlier works employ statistical models such as kernel-based probability distribution [63, 8, 24, 9] to synthesize motion, but abstract away motion details. Motion graphs [52, 6, 46] address this problem by generating motions in a non-parametric manner. Motion graph is a directed graph constructed on a corpus of motion capture data, where each node is a pose and the edges represent the transition between poses. Motion is generated by a random walk on this graph. A challenge in motion graph is in generating plausible transition that some approaches address via parameterizing the transition [29]. With the development in deep learning, many approaches explore the applicability of neural networks to generate 3D motion by training on a large-scale motion capture dataset, where network architectures such as CNNs [34, 33], GANs [30], RBMs [80] and RNNs [23, 4, 39, 26, 13, 17, 89, 10, 88] have been explored. Auto-regressive models like RNNs are capable of generating unbounded motion in theory, but in practice suffer from regression to the mean where motion "freezes" after several iterations, or drift to unnatural motions [4, 3].

Some works [7, 55, 48] propose to ease this problem by periodically using the network's own outputs as inputs during training. Phase-functioned neural networks and it's variations [95, 32, 72, 73] address this issue via conditioning the network weights on phase, however, they do not scale well to represent a wide variety of motion. In this work, we present a transformer based approach for generating 3D motion conditioned on music. The use of transformers is similar to a recently proposed approach by Aksan *et al*. [3], however we employ a full-attention transformer with future-N supervision which are shown to be key for long-range 3D motion generation in our experiments.

Cross-Modal Sequence-to-Sequence Generation Beyond of the scope of human motion generation, our work is closely related to the research of using neural network on cross-modal sequence to sequence generation task. In natural language processing and computer vision, tasks like text to speech (TTS) [68, 40, 42, 84] and speech to gesture [21, 27, 22], image/video captioning (pixels to text) [11, 43, 58, 47] involve solving the cross-modal sequence to sequence generation problem. Initially, combination of CNNs and RNNs [87, 86, 92, 94] were prominent in approaching this problem. More recently, with the development of attention mechanism [85], transformer based networks achieve top performance for visual-text [96, 76, 18, 53, 37, 75, 75],visual-audio [25, 90] cross-modal sequence to sequence generation task. Our work explores audio to 3D motion in a transformer based architecture. While all cross-modal problems induce its own challenges, the problem of music to 3D dance is uniquely challenging in that there are many ways to dance to the same music and that the same dance choreography may be used for multiple music. We hope the proposed AIST++ dataset advances research in this relatively under-explored problem.

Audio To Human Motion Generation Dance to motion generation has been studied in 2D pose context either in optimization based approach [81], or learning based approaches [51, 71, 50, 66, 67, 20] where 2D pose skeletons are generated from a music conditioning. Training data for 2D pose and music is abundant thanks to the high reliability of 2D pose detectors [12]. However, predicting motion in 2D is limited in its expressiveness and potential for downstream applications. For 3D dance generation, earlier approaches explore matching existing 3D motion to music [70] and motion graph based approach [19]. More recent approach employ LSTMs [5, 79, 91, 98, 41], GANs [50, 77], transformer encoder with RNN decoder [35] or convolutional [2, 93] sequence-to-sequence models. Closest to our work is that of Li *et al*. [54], which also employ transformer based architecture but only on audio and motion. Furthermore, their approach discretize the output joint space in order to account for multi-modality, which generates unrealistic motion. In this work we introduce a novel full-attention based cross-modal transformer for audio and motion, which can not only preserve the correlation between music and 3D motion better, but also generate more realistic long 3D human motion with global translation. One of the biggest bottleneck in 3D dance generation approaches is that of data. Recent work of Li *et al*. [54] reconstruct 3D motion from dance videos on the Internet, however the data is not public. Further, using 3D motion reconstructed from monocular videos may not be reliable and lack accurate global 3D translation information.

In this work we also reconstruct the 3D motion from 2D dance video, but from multi-view video sequences, which addresses these issues. While there are many large scale 3D motion capture datasets [38, 59, 1, 36], mocap dataset of 3D dance is quite limited as it requires heavy instrumentation and expert dancers for capture. As such, many of these previous works operate on small-scale motion capture dataset such as Dance with Melody [79], which is 94 minutes long with 4 types of music, GrooveNet [5], which is 23 minutes long with one dancer and one genre of electronic dance music, and DanceNet [97], which consist of an hour long two sequences of dance. In this paper, we present the AIST++ dataset, a 5 hours long 3D dance dataset with 10 genres of music and 30 dancers.

3. AIST++ Dataset

Here we first discuss the content of the proposed AIST++ dataset, and then the process by which we obtain the 3D motion from the original AIST Dance Database(AIST) [83], which is a non-calibrated multi-view collection of dance videos.

AIST++ is a large-scale 3D human dance motion dataset that contains a wide variety of 3D motion paired with music. It has the following extra annotations for each frame:

- 9 views of camera intrinsic and extrinsic parameters;
- 17 COCO-format[69] human joint locations in both 2D and 3D;
- 24 SMPL [56] pose parameters along with the global scaling and translation.

This dataset is designed to serve as a benchmark for both motion generation and prediction tasks. It can also potentially benefit other tasks such as the 2D/3D human pose estimation. To our knowledge, AIST++ is the largest 3D human dance dataset with $\mathbf{1408}$ sequences, $\mathbf{30}$ subjects and $\mathbf{10}$ dance genres with basic and advanced choreographies. See Table. 1 for comparison with other 3D motion and dance datasets. AIST++ is a complementary dataset to existing 3D motion dataset such as AMASS [59], which contains only 17.8 minutes of dance motions with no accompanying music.

Dataset	Music	3D Joint$_{pos}$	3D Joint$_{rot}$	2D Kpt	Views	Genres	Subjects	Sequences	Seconds
AMASS[59]	-	✓	✓	-	-	-	344	11265	145251
Human3.6M[38]	-	✓	✓	✓	4	-	11	210	71561
Dance with Melody[79]	✓	✓	-	-	-	4	Unknown	61	5640
GrooveNet [5]	✓	✓	-	-	-	1	1	2	1380
DanceNet [97]	✓	✓	-	-	-	2	2	2	3472
AIST++	✓	✓	✓	✓	9	10	30	1408	18694

Table 1: **3D Motion Datasets Comparisons.** Here we present a detailed comparison between our AIST++ dataset against other published 3D motion datasets. Length-wise, our AIST++ dataset rank the third. Motion-wise, our AIST++ dataset has 10 types of different dance motions accompanied with music. Whereas, Human3.6M [38], the second largest, only has simple walking, sitting down etc. motions.

Thanks to AIST, AIST++ contains 10 dance genres: Old School (Break, Pop, Lock and Waack) and New School (Middle Hip-hop, LA-style Hip-hop, House, Krump, Street Jazz and Ballet Jazz) (shown in Figure 3). Please see Appendix 7.1 for more details and statistics. The motions are equally distributed among all dance genres, covering wide variety of music tempos denoted as beat per minute (BPM)[61]. Each genre of dance motions contains 85% of basic choreographies and 15% of advanced choreographies, in which the former ones are those basic short dancing movements while the latter ones are longer movements freely designed by the dancers. However, note that AIST is an instructional database and records multiple dancers dancing the same choreography for different music with varying BPM, a common practice in dance. This posits a unique challenge in cross-modal sequence-to-sequence generation.

3.1. 3D Motion Reconstruction

Next we describe how we reconstruct 3D motion from the AIST dataset. Although the AIST dataset contains multi-view videos, they are not calibrated meaning their camera intrinsic and extrinsic parameters are not available. Without camera parameters, it is not trivial to automatically and accurately reconstruct the 3D human motion. We start with 2D human pose detection [62] and manually initialized the camera parameters. On this we apply bundle adjustment [82] to refine the camera parameters. With the improved camera parameters, the 3D joint locations $\hat{J} \in \mathbb{R}^{M\times 3} (M = 17)$ are then triangulated from the multi-view 2D human pose keypoints locations. During the triangulation phase, we introduce temporal smoothness and bone length constraints to improve the quality of the reconstructed 3D joint locations. We further fit SMPL human body model [57] in order to obtain 3D joint rotation information, which is the most common form of motion data used in animation and other graphics applications. SMPL can be thought of as a function $\Phi(\theta, \beta, \gamma, \alpha)$ that takes input a 3D joint rotation pose parameters θ, a low-dimensional shape parameters β, a global scale coefficient α and global translation γ and outputs a mesh and a set of joints $J(\theta, \beta, \gamma, \alpha)$. We fit this model to the triangulated

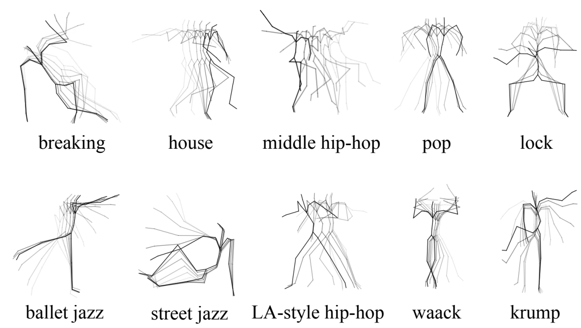

Figure 3: **AIST++ Motion Diversity Visualization.** Here we show the 10 types of 3D human dance motion in our dataset.

joint locations $\hat{J}$ by minimizing an objective with respect to $\Theta = \{\theta_i\}_i^M$, global scale parameter α and global transformation γ for each frame:

$$\min_{\Theta,\gamma,\alpha} \sum_{i=1}^{M} \|\hat{J} - J(\theta_i, \beta, \gamma, \alpha)\|_2. \quad (1)$$

We fix β to the average shape as the problem is under-constrained from 3D joint locations alone. We verify the quality of the recovered 3D motion using multi-view re-projection in Section 5.1.

4. Music Conditioned 3D Dance Generation

With the 3D dataset in hand, next we describe our approach to music conditioned 3D dance generation.

Problem statement Given a 2-second seed sample of motion represented as $\mathbf{X} = (x_1, \ldots, x_T)$ and a music sequence represented as $\mathbf{Y} = (y_1, \ldots, y_{T'})$, the problem is to generate a sequence of future motion $\mathbf{X}' = (x_{T+1}, \ldots, x_{T'})$ from time step $T + 1$ to T', where $T' \gg T$.

4.1. Cross-Modal Motion Generation Transformer

We propose a transformer-based network architecture that can learn the music-motion correlation and generate non-freezing realistic motion sequences. The overview of this architecture is shown in Figure 2. While previous works have leveraged transformers [54, 3], we introduce some critical design choices that assist in learning cross-modal correspondence and, more importantly, prevent generated motion from freezing. These choices include the cross-modal transformer architecture (the number of attention layers), attention mechanism—causal attention [65] vs. full attention [16] for each transformer—and the supervision scheme. Here, we explain our design choices in detail.

We introduce three transformers to our model: the motion transformer $f_{\text{mot}}(\mathbf{X})$, which embeds the motion feature $\mathbf{X}$ into a motion embedding $\mathbf{h}^x{}_{1:T}$; the audio transformer $f_{\text{audio}}(\mathbf{Y})$, which similarly embeds audio feature $\mathbf{Y}$ into an audio embedding $\mathbf{h}^y{}_{1:T'}$, and the cross-modal transformer $f_{\text{cross}}(\mathbf{h}^{xy}{}_{1:T+T'})$, which learns the correspondence between both modalities and generates future motion $\mathbf{X}'$. To better learn the correlation between the two modalities, we employ a deep 12 layer cross-modal transformer. We find that increasing the depth of the cross-modal transformer can greatly help the model to pay attention to both modalities (as shown in Figure. 6).

Transformer network is all about attention [85], and there are two common types of attention: causal and full attention. They differ in the data computation dependencies. Figure 4 illustrates this relation between the inputs (the bottom row), context vectors $\mathbf{C}$ (the middle row) and the outputs (the top row) as a simplified two-layer transformer. The connections (edges) in the Figure 4 represents the computation dependencies. For causal-attention, the context tensors are only computed from the current and past inputs. Similarly, the output tensors in the causal-attention only get to see the current and previous context tensors. But for full-attention, they are fully dependent on each other. Full-attention is more commonly used in a transformer encoder network [16, 14], while the causal-attention is usually used in a transformer decoder network [65, 15]. Specifically, the output of the attention layer, the context vector $\mathbf{C}$ is computed using the query vector $\mathbf{Q}$ and the key $\mathbf{K}$ value $\mathbf{V}$ pair from input with or without a mask $\mathbf{M}$ via .

$$\begin{aligned}\mathbf{C} &= \text{Attn}(\mathbf{Q}, \mathbf{K}, \mathbf{V}, \mathbf{M}) \\ &= \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T + \mathbf{M}}{\sqrt{D}}\right)\mathbf{V},\end{aligned} \tag{2}$$

where D is the number of channels in the attention layer. In causal-attention, the mask $\mathbf{M}$ is a triangular matrix, also referred to as the look-ahead-mask, with non-zero elements close to negative infinity, while in full-attention $\mathbf{M} = \mathbf{0}$. For all three transformers: motion transformer $f_{\text{mot}}(\mathbf{X})$, audio transformer $f_{\text{audio}}(\mathbf{Y})$ and cross-modal transformer $f_{\text{cross}}(\mathbf{h}^{xy}{}_{1:T+T'})$, we apply the full-attention.

Aside from the network architecture, the supervision scheme can critically affect the model's performance as it is tightly related to how the gradients are computed. Many previous works [65, 54] on the sequence to sequence generation task apply the shift-by-1 (or auto-regressive) supervision scheme with the causal-attention transformer. When supervised on shift-by-1 output, the attention layer learns to predict the immediate next frame of the input vector. We find that motions generated using this shift-by-1 supervision are subject to freezing after a few steps. Instead, we combine our full-attention transformer with the proposed future-N supervision scheme. Specifically the output of the transformer is supervised on the future N time steps from the last observed timestamp in the input sequence. At test time, our approach can still be applied in an auto-regressive framework. Compared with the causal-attention (shown on the left of Figure 4) with shift-by-1 supervision, our full-attention transformer with future-N supervision results in a non-freezing, more realistic long motion generation. We show this comparison results in Sec. 5.2.4.

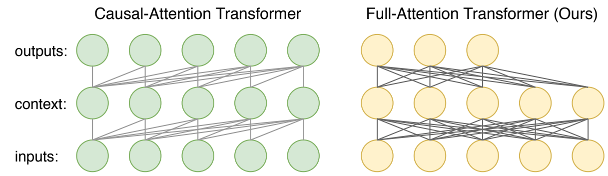

Figure 4: **Attention Mechanism Comparison.** Here we show the data tensor relation for a causal-attention transformer (used in models like GPT [65] and the motion generator of [54]) and our full-attention transformer, as a simplified two-layer transformer for illustration purposes. The dots on the bottom row are the input tensors, which are computed into context tensors through causal (left) and full (right) attention transformer layer. The output (predictions) are shown on the top. While causal models are often supervised to predict the immediate next future for each input tensor, in our work we employ full attention and predict the N future time steps from the last timestamp (3 shown here, 20 used in practice). We empirically show that this results in a non-freezing, more realistic motion generation.

5. Experiments

5.1. AIST++ Motion Quality Validation

We first carefully validate the quality of our 3D motion reconstruction. Possible error sources that may affect the quality of our 3D reconstruction include inaccurate 2D keypoints detection and the estimated camera parameters. As there is no 3D ground-truth for AIST dataset, our validation here is based-on the observation that the re-projected 2D keypoints should be consistent with the predicted 2D keypoints which have high prediction confidence. We use the 2D mean per joint position error MPJPE-2D, commonly used for 3D reconstruction quality measurement [45, 38, 64]) to evaluate the consistency between the predicted 2D keypoints and the reconstructed 3D keypoints

	Motion Quality		Motion Diversity		Motion-Music Correlation		User Study
	Pos. Frechet Dist↓	Vel. Frechet Dist↓	Pos. Var↑	Vel. Var↑	Beat Align. Score↑	Beat DTW Cost↓	Our Winning Rate↑
AIST++	–	–	–	–	0.295	10.51	–
AIST++ (unpaired)	–	–	–	–	0.212	13.54	25.4%
Li *et al.* [54]	5595.91	3.40	0.019	121.36*	0.231	12.56	80.6%
Dancenet[97]	2367.26	1.13	0.215	1.05	0.232	12.17	71.1%
Ours	**113.56**	**0.45**	**0.509**	**6.51**	**0.241**	**12.16**	–

Table 2: **Conditional Motion Generation Evaluation on AIST++ dataset.** Comparing to the two baseline methods, our model generates motions that are more realistic, better correlated with input music and more diversified when conditioned on different music. *Note Li *et al.* [54]'s generated motions are highly jittery making its velocity variation extremely high.

$\hat{J}$ along with the estimated camera parameters. Note we only consider 2D keypoints with prediction confidence over 0.5 to avoid noise. The MPJPE-2D of our entire dataset is 6.2 pixels on the 1920×1080 image resolution, and over $\mathbf{86}\%$ of those has less than 10 pixels of error. Please refer to Appendix 7.1 for the distribution of MPJPE-2D on AIST++.

5.2. Music Conditioned 3D Motion Generation

5.2.1 Experimental Setup

Dataset Split All the experiments in this paper are conducted on our AIST++ dataset, which to our knowledge is the largest dataset of this kind. We split AIST++ into *train* and *test* set, and report the performance on the *test* set only. We carefully split the dataset to make sure that the music and dance motion in the *test* set does not overlap with that in the *train* set. To build the *test* set, we first select one music piece from each of the 10 genres. Then for each music piece, we randomly select two dancers, each with two different choreographies paired with that music, resulting in total 40 unique choreographies in the *test* set. The *train* set is built by excluding all test musics and test choreographies from AIST++, resulting in total 329 unique choreographies in the *train* set. Note that in the test set we *intentionally* pick music pieces with different BPMs so that it covers all kinds of BPMs ranging from 80 to 135 in AIST++.

Implementation Details In all our experiments, the input of the model contains a seed motion sequence with 120 frames (2 seconds) and a music sequence with 240 frames (4 seconds), where the two sequences are aligned on the first frame. The output of the model is the future motion sequence with 20 frames (expect for the attention ablation study experiments in Sec. 5.2.4). During training we supervise the future motion sequence with $L2$ loss. During inference we generate long-range future motions in a auto-regressive way on 60 FPS. We use the public available audio processing toolbox Librosa [60] to extract the music features including: 1-dim *envelope*, 20-dim *MFCC*, 12-dim

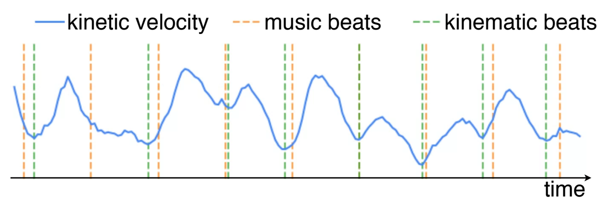

Figure 5: **Beats Alignment between Music and Generated Dance.** Here we visualize the kinetic velocity (blue curve) and kinematic beats (green dotted line) of our generated dance motion, as well as the music beats (orange dotted line). The kinematic beats are extracted by finding local minima from the kinetic velocity curve.

chroma, 1-dim *one-hot peaks* and 1-dim *one-hot beats*, resulting in a 35-dim music feature $\mathbf{Y}$. This 35-dim music feature is the input to the audio transformer $f_{\text{audio}}(\mathbf{Y})$. For the motion feature, we use the 9-dim rotation matrix representation for all 24 joints, along with a 3-dim global translation vector, resulting in a 219-dim motion feature as the input to the motion transformer $f_{\text{motion}}(\mathbf{X})$. All the three (audio, motion, cross-modal) transformers have 10 attention heads. All the layers in each transformer have 800 hidden size. The number of attention layers in each transformer varies based on the experiments, as described in Sec. 5.2.4. Learn-able position encoding is used for motion and audio transformer, respectively. All our experiments are trained with 16 batch size using Adam [44] optimizer. The learning rate starts from $1\text{e}{-4}$ and drops to $\{1\text{e}{-5}, 1\text{e}{-6}\}$ after $\{60k, 100k\}$ steps. The training finishes after $300k$, which takes 3 days on 4 TPUs. For baselines, we compare with Dancenet[97] and Li *et al.* [54], which we train and test on the same dataset with ours using the *official* code provided by the authors.

5.2.2 Evaluation Metrics

Motion Quality Metric We evaluate the motion quality by calculating the distribution distance between the generated motion clips and the ground-truth motion clips inspired by FID [31], which is widely used in image generation evaluations. As there is no standard motion feature extractor like Inception network [78] for motions, we directly calculate the Frechet Distance (FD) in the joint position space $\mathbb{R}^{T\times N\times 3}$ and joint velocity space $\mathbb{R}^{T\times N\times 3}$. All the following experiments using this metric are conducted on 1000 randomly sampled $T = 1$-second motion clips from 20-second generated sequences with all the $N = 24$ joints.

Motion Diversity Metric The diversity of the long-range generated dance motions conditioned on various musics reflects how well the model learns the cross-modal relationship. We use motion *variation* in both joint position space $\mathbb{R}^{T\times N\times 3}$ and joint velocity space $\mathbb{R}^{T\times N\times 3}$ to measure to diversity. All the following experiments using this metric are conducted on $T = 20$-seconds sequences with all the $N = 24$ joints.

Motion-Music Correlation Metric Furthermore, we propose two metrics: Beat Alignment Score and Beat Dynamic-Time-Warping (DTW) Cost to evaluate how well the generated motion is correlated with the input music. The correlation is defined through the similarity between kinematic beats and music beats. The kinematic beats are computed as the local minima of the kinetic velocity, as shown in Figure 5. The Beat Alignment Score is the average distance between every kinematic beat and its nearest music beat. Specifically, our Beat Alignment Score is defined as:

$$\text{score} = \frac{1}{m}\sum_{i=1}^{m} \exp\left(-\frac{\min_{\forall t_j^y \in B^y} ||t_i^x - t_j^y||^2}{2\sigma^2}\right), \tag{3}$$

where $B^x = \{t_i^x\}$ is the kinematic beats, $B^y = \{t_j^y\}$ is the music beats and σ is a parameter to normalize sequences with different FPS. We set $\sigma = 3$ in all our experiments as the FPS of all our experiments sequences is 60. A similar metric Beat Hit Rate was introduced in [50, 35], but this metric requires a dataset dependent handcrafted threshold to decide the alignment ("hit") while ours directly measure the distances. To measure the similarity between music beats and kinematic beats, we employ the Dynamic-Time-Warping (DTW) [28]. The Beat DTW Cost is thus computed as, $\frac{1}{s}\sum_{k=1}^{s}||t_k^x - t_k^y||$, where $P_k = (t_k^x, t_k^y), k \in \{1, ..., s\}$ represents the k^{th} paired frame indexes on the warping path.

5.2.3 Quantitative Evaluation

In this section, we report the quantitative evaluation results of our method compared with the two baselines: Li *et al*. [54] and Dancenet [97] on out AIST++ *test* set. The results are shown in Table 2.

Motion Quality In this experiment, we first generate 20-seconds motion sequences using the paired data from our AIST++ *test* set. Then we compute our motion quality metrics on these generated motion sequences. As shown in Table 2, our generated motion sequence joint and velocity distributions are much closer to ground-truth motions compared with the two baselines. We also visualize the generated sequences from the two baselines in our supplemental video[1]. The participants of our user study (Sec. 5.2.5) often commented *"too jittery"* on the results from Li *et al*. [54] and *"very limited movements"* on the results from Dancenet [97].

Motion Diversity We evaluate our model's ability of generating diverse dance motions when given various input musics compared with the two baseline methods. In this experiment, we pair all the 10 music pieces with the 40 unique seed motions from the AIST++ *test* set as the input to generate 400 sequences of motion. Then, we compute the diver-

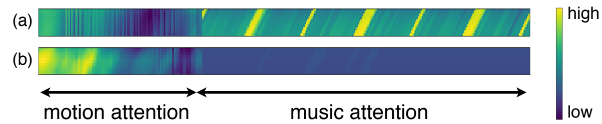

Figure 6: **Attention Weights Visualization.** We compare the attention weights from the last layer of the (a) 12-layer cross-modal transformer and (b) 1-layer cross-modal transformer. Deeper cross-modal transformer pays equal attention to motion and music, while a shallower one intends to pay more attention to motion.

Figure 7: **Motion Diversity Results.** Here we visualize 4 different dance motions generated using our proposed model when given different music but same seed motion. Each row is a sampled 2 second (from a 20 seconds sequences) generated 3D dance motion. Especially, on the second row, when given a more modern dance music but hip-hop seed motion, our model can generate ballet motions (refer to our supplementary video).

[1] https://www.youtube.com/watch?v=VrVsAcgFK_4

Cross-Modal Transformer	Vel. Var↑	Pos. Var↑	Beat Align. Score↑	Beat DTW Cost↓
w/o Audio	–	–	0.227*	12.64*
w/ Audio + CM-1	0.480	0.78	0.233	12.59
w/ Audio + CM-12	**0.509**	**6.51**	**0.241**	**12.16**

Table 3: **Ablation Study on Cross-modal Transformer.** The deeper cross-modal transformer the better it learns motion-music correlation so it more follows the beats and generate more diverse motion from different musics. *Note those numbers are calculated using the music paired with the input motion.

Attn-Supervision	Pos. Frechet Dist↓	Vel. Frechet Dist↓
Causal-Attn-Shift-by-1	206.7	1.60
Full-Attn-F1	188.3	2.34
Full-Attn-F10	142.4	0.50
Full-Attn-F20	**113.6**	**0.45**

Table 4: **Ablation Study on Attention and Supervision Design Choices.** Causal-attention transformer with shift-by-1 supervision intends to generate freezing motion in long-term. Full-attention transformer supervised with more future frames boost the ability of generating more realistic dance motion.

sity metrics over the generated motions from the same seed motion but different music. Table 2 shows our method generates more diverse dance motions comparing to the baselines. Based on our model ablation study (Sec. 5.2.4), our careful network design, particularly the cross-modal transformer is the main reason for this difference. In addition, we also visualize our generated diverse motions in Figure 7.

Motion-Music Correlation Further, we evaluate how much the generated 3D motion correlates to the input music. To calibrate the results, we compute the correlation metrics on the entire AIST++ dataset (upper bound) and on the random-paired data (lower bound). As shown in Table 2, our generated motion is better correlated with the input music compared to the baselines. We also show one example in Figure 5 that the kinematic beats of our generated motion align well with the music beats. However, when comparing to the real data, all three methods including ours have a large space for improvement. This reflects that music-motion correlation is still a challenging problem.

5.2.4 Ablation Study

Cross-modal Transformer We study the functionality of our cross-modal transformer using three different settings: (1) 14-layer motion transformer only; (2) 13-layer motion/audio transformer with 1-layer cross-modal transformer; (3) 2-layer motion/audio transformer with 12-layer cross-modal transformer. For fair comparison, we change the number of attention layers in the motion/audio transformer and the cross-modal transformer simultaneously to keep the total number of the attention layers the same. Table 3 shows that the cross-modal transformer is critical to generate motions well correlated with input music. Also as shown in Figure 6, deeper cross-modal transformer pays more attention to the music, thus it learns better music-motion correlation.

Causal-Attention or Full-Attention Transformer Here we study the effectiveness of our full-attention mechanism and future-N supervision scheme. We set up four experiments with different settings: causal-attention with shift-by-1 supervision, and full-attention with future-1/10/20 supervision. Qualitatively, we find the motion generated by the causal-attention mechanism with shift-by-1 supervision (similar to the GPT model [65]) starts to freeze after several seconds. Similar problem was reported using this type of setting in motion prediction works [4, 3]. Quantitatively, as shown in the Table 4, there is a huge distribution difference between the generated motion and ground-truth motion in 20-seconds long-range generation when using causal-attention mechanism. For the full-attention mechanism with future-1 supervision setting, the results rapidly drift during long-range generation. However, when the model is supervised with 10 or 20 future frames, it can generate good quality (non-freezing, non-drifting) long-range motion.

5.2.5 User Study

Finally, we perceptually evaluate the motion-music correlation with a user study to compare our method with the two baseline methods and the "random" baseline, which randomly combines AIST++ motion-music. (refer to Appendix 7.2 for user study details.) In this study, each user is asked to watch 10 random videos out of 120, and answer the question *"which person is dancing more to the music? LEFT or RIGHT"* for each video. There are in total of 90 participants in this user study, ranging from professional dancers to people rarely dance. We analyze the feedback and the results are: (1) 81% of our generated dance motion is better than Li *et al.* [54]; (2) 71% of our generated dance motion is better than Dancenet [97] (3) 75% of the unpaired AIST++ dance motion is better than ours. Clearly we surpass the baselines in the user study. But because the "random" baseline consists of real advanced dance motions that are extremely expressive, participants are biased to prefer it over ours. However, quantitative metrics show that our generated dance is more aligned with music.

6. Conclusion and Discussion

In this paper, we present a cross-modal transformer-based neural network architecture that can not only learn the audio-motion correspondence but also can generate non-freezing high quality 3D motion sequences conditioned on music. As generating 3D movement from music is a nascent area of study, we hope our work will pave the way for future cross-modal audio to 3D motion generation. We also construct the largest 3D human dance dataset: AIST++. This large, multi-view, multi-genre, cross-modal 3D motion dataset can not only help research in the conditional 3D motion generation research but also human understanding research in general. While our results shows a promising direction in this problem of music conditioned 3D motion generation, there are more to be explored. While our approach can generate realistic motion, the model is currently deterministic. Exploring how to generate multiple realistic dance per music is an interesting direction.

7. Appendix

7.1. AIST++ Dataset Details

Statistics We show the detailed statistics of our AIST++ dataset in Table 5. Thanks to the AIST Dance Video Database [83], our dataset contains in total 5.2-hour (1.1M frame, 1408 sequences) of 3D dance motion accompanied with music. The dataset covers 10 dance genre and 60 pieces of music. For each genre, there are 6 different pieces of music, ranging from 29 seconds to 54 seconds long, and from 80 BPM to 130 BPM (except for House genre which is 110 BPM to 135 BPM). Among those motion sequences, 120 (85%) of them are *basic* choreographies and 21 (15%) of them are *advanced*. Advanced choreographies are longer and more complicated dances improvised by the dancers. Note for the *basic* dance motion, dancers are asked to perform the same choreography on all the 6 pieces of music with different speed to follow different music BPMs. So the total *unique* choreographies in for each genre is $120/6 + 21 = 41$. In our experiments we split the AIST++ dataset such that there is no overlap between *train* and *test* for both music and choreographies (see Sec. 5.2.1 in the paper).

Validation As described in Sec. 5.1 in the paper, we validate the quality of our reconstructed 3D motion by calculating the overall MPJPE-2D (in pixel) between the re-projected 2D keypoints and the predicted 2D keypoints. We provide here the distribution of MPJPE-2D among all motion sequences and all images (Figure 8)).

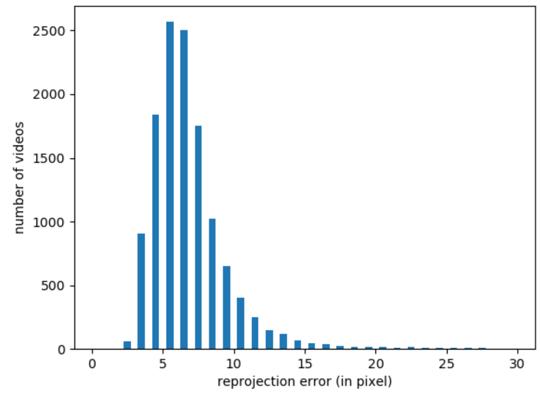

Figure 8: **AIST++ 3D Keypoints Re-projection Error Distribution.** We compare the average pixel distance of the re-projected 2D keypoints and the detected 2D keypoints [62] for each video on 1920x1080 resolution.

7.2. User Study Details

7.2.1 Comparison User Study

As mentioned in Sec. 5.2.5 in the main paper, we qualitatively compare our generated results with several baselines in a user study. Here we describe the details of this user study. Figure 9 shows the interface that we developed for this user study. We visualize the dance motion using stick-man and conduct side-by-side comparison between our generated results and the baseline methods. The left-right order is randomly shuffled for each video to make sure that the participants have absolutely no idea which is ours. Each video is 10-second long, accompanied with the music. The question we ask each participant is *"which person is dancing more to the music? LEFT or RIGHT"*, and the answers are collected through a Google Form. At the end of this user study, we also have an exit survey to ask for the dance experience of the participants. There are two questions: *"How many years have you been dancing?"*, and *"How often do you watch dance videos?"*. Figure 10 shows that our 90 participants ranges from professional dancers to people rarely dance, with majority with at least 1 year of dance experience.

7.2.2 Realism User study

Here we further provide a second-round user study that focus on the *realism* of the generated dance motion. Our ultimate goal is to generate dance motions that from human's perception are realistic. In this user study, we ask each participant to watch *one* 5-second dance video with music (also in stick-man) at a time, and answer the question *"Is it a real dance or a machine synthesized one?"*. The 3D dance motions are randomly selected from a pool with 40 real dances

Genres	Musics	Music Tempo	Motions	Choreographs	Motion Duration (sec.)	Total Seconds
ballet jazz	6	80 - 130	141		7.4 - 12.0 basic / 29.5 - 48.0 adv.	1910.8
street jazz	6	80 - 130	141		7.4 - 12.0 basic / 14.9 - 48.0 adv.	1875.3
krump	6	80 - 130	141		7.4 - 12.0 basic / 29.5 - 48.0 adv.	1904.3
house	6	110 - 135	141		7.1 - 8.7 basic / 28.4 - 34.9 adv.	1607.6
LA-style hip-hop	6	80 - 130	141	85% basic +	7.4 - 12.0 basic / 29.5 - 48.0 adv.	1935.8
middle hip-hop	6	80 - 130	141	15% advanced	7.4 - 12.0 basic / 29.5 - 48.0 adv.	1934.0
waack	6	80 - 130	140		7.4 - 12.0 basic / 29.5 - 48.0 adv.	1897.1
lock	6	80 - 130	141		7.4 - 12.0 basic / 29.5 - 48.0 adv.	1898.5
pop	6	80 - 130	140		7.4 - 12.0 basic / 29.5 - 48.0 adv.	1872.9
break	6	80 - 130	141		7.4 - 12.0 basic / 23.8 - 48.0 adv.	1858.3
total	60		1408			18694.6

Table 5: AIST++ Dataset. We reconstruct these subset of AIST Dataset that are single-person dance sequences

from AIST++ and 40 generated dances from our method. So 50% of the time the participants would see the real dance and 50% of the time they would see the generated dance. Each participant is asked to watch 10 videos selected randomly out of the 80 video pool. We use the similar interface as our first user study, as shown in Figure 9, except for a different Google Form and *one* video at a time. In this user study, there are in total 56 participants with various years of dance experience and computer vision experience (see Figure 11). The feedback shows that on average 34% of our generated dance motion have been selected as *REAL*. Interestingly, only 67% of the real dance motion sequences have been selected as *REAL*. This shows participants have a quite high standard about identifying a video to be a real dance.

7.3. Motion Genre Analysis

Genre Similarity in Real Dance AIST++ contains ten different genres of dance motion. To study how different the dance motions are among different genres, we calculate the *motion similarity matrix* between different genres of dance motion in AIST++ (see Figure 12). The similarity is calculated based on the Frechet Distance in the joint space (see Sec. 5.2.3 in the main paper) between each two sets of motion sequences. Specifically, we define the similarity between two sets of motion $\mathbf{M}_i$ and $\mathbf{M}_j$ as

$$S_{ij} = e^{-\alpha \mathbf{D}(\mathbf{M}_i, \mathbf{M}_j)} \tag{4}$$

where $\mathbf{D}(\mathbf{M}_i, \mathbf{M}_j)$ is the Frechet Distance and α is a dataset dependent scale factor which we set to 0.003 for AIST++. As shown in Figure 12, motions in the same genre have high similarity and motions in different genres are less similar to each other. Specially the *ballet jazz* dance motion is quite different to any other kind of dance in AIST++. This poses a unique challenge for cross-modal learning between music and motion.

Genre Consistency of Generated Dance A good model should not only generated long-range, non-freezing dance

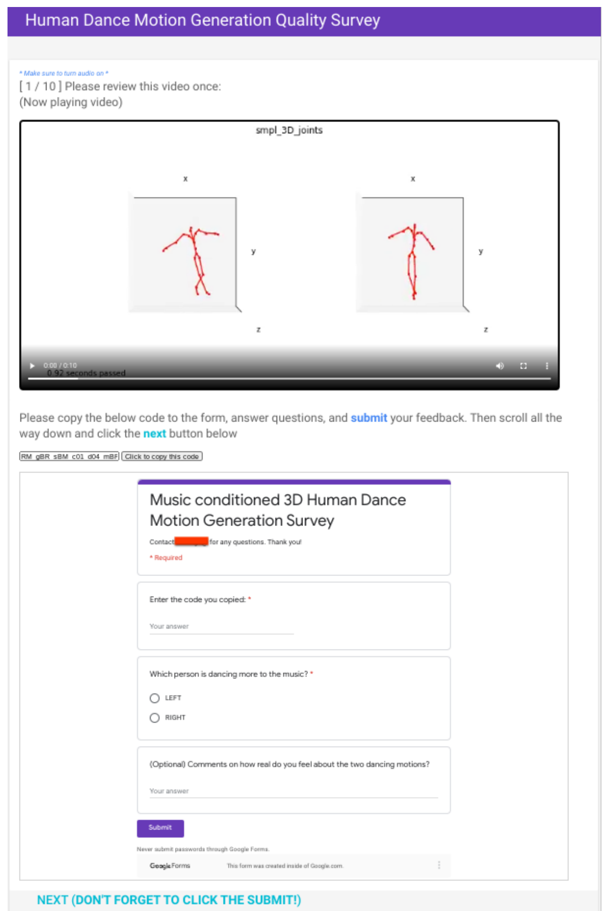

Figure 9: **User study interface.** The interface of our User study. We ask each participant to watch 10 videos and answer the question *"which person is dancing more to the music? LEFT or RIGHT"*.

motion that follows the beats of the music, but also follow the genre of the music. Thus we further analyze the *genre consistency* between the generated dance motion and the conditioned music. As motion and music are in two different domains, there is no direct way to calculate the genre

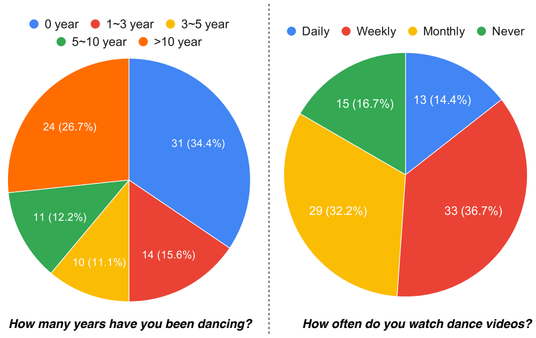

Figure 10: **Participant Demography of the *Comparison* User Study.**

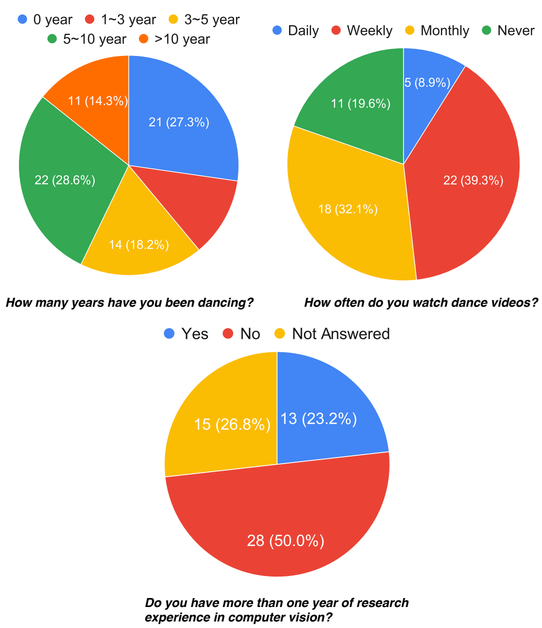

Figure 11: **Participant Demography of the *Realism* User Study.**

similarity between them. Fortunately AIST++ has paired motion-music data, thus we can safely assume the dance motion in AIST++ can also well represent the genre of the paired music. Based on this assumption, the genre consistency is then defined as the similarity between the generated dance motion and the real dance motion in AIST++ with the same genre. Here we also use Equation 4 to calculate the similarity between each two sets of motion sequences. Figure 13 shows the genre consistency of the generated results from our method as well as two baselines Dancenet [97] and Li *et al*. [54]. Clearly our method can generate motions more correlated with the genre of the music, which also demonstrates that our model can better learn the audio-motion correspondence.

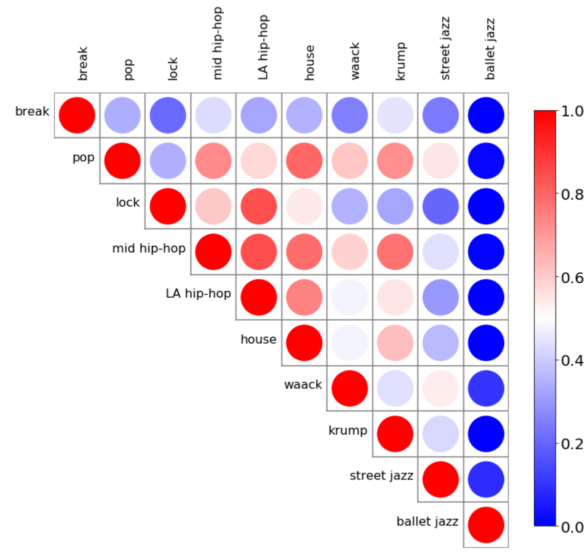

Figure 12: **Motion Similarity in AIST++.** We analyze the similarity between different genres of dance motion in AIST++, which shows distribution difference among genres. For example, ballet jazz is quite different than any other kind of dance.

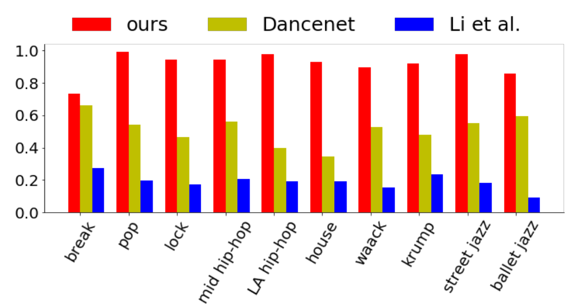

Figure 13: **Genre Consistency between Generated Dance and Music.** We analyze the genre consistency between the generated dance motion and the input music, of all three methods, on all ten genres. Higher the better.

References

[1] Mixamo. https://www.mixamo.com/.

[2] 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):3500–3507, 2020.

[3] Emre Aksan, Peng Cao, Manuel Kaufmann, and Otmar Hilliges. Attention, please: A spatio-temporal transformer for 3d human motion prediction. *arXiv preprint arXiv:2004.08692*, 2020.

[4] Emre Aksan, Manuel Kaufmann, and Otmar Hilliges. Structured prediction helps 3d human motion modelling. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7144–7153, 2019.

[5] Omid Alemi, Jules Françoise, and Philippe Pasquier. Groovenet: Real-time music-driven dance movement generation using artificial neural networks. *networks*, 8(17):26, 2017.

[6] Okan Arikan and David A Forsyth. Interactive motion generation from examples. *ACM Transactions on Graphics (TOG)*, 21(3):483–490, 2002.

[7] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in neural information processing systems*, 2015.

[8] Richard Bowden. Learning statistical models of human motion. In *IEEE Workshop on Human Modeling, Analysis and Synthesis, CVPR*, volume 2000, 2000.

[9] Matthew Brand and Aaron Hertzmann. Style machines. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 183–192, 2000.

[10] Judith Bütepage, Michael J Black, Danica Kragic, and Hedvig Kjellström. Deep representation learning for human motion prediction and classification. In *CVPR*, page 2017, 2017.

[11] Slides by Saheel. Baby talk: Understanding and generating image descriptions.

[12] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018.

[13] Hsu-kuang Chiu, Ehsan Adeli, Borui Wang, De-An Huang, and Juan Carlos Niebles. Action-agnostic human pose forecasting. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1423–1432. IEEE, 2019.

[14] Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7059–7069, 2019.

[15] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[17] Xiaoxiao Du, Ram Vasudevan, and Matthew Johnson-Roberson. Bio-lstm: A biomechanically inspired recurrent neural network for 3-d pedestrian pose and gait prediction. *IEEE Robotics and Automation Letters*, 4(2):1501–1508, 2019.

[18] Xuguang Duan, Wenbing Huang, Chuang Gan, Jingdong Wang, Wenwu Zhu, and Junzhou Huang. Weakly supervised dense event captioning in videos. In *Advances in Neural Information Processing Systems*, pages 3059–3069, 2018.

[19] 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.

[20] Joao P Ferreira, Thiago M Coutinho, Thiago L Gomes, José F Neto, Rafael Azevedo, Renato Martins, and Erickson R Nascimento. Learning to dance: A graph convolutional adversarial network to generate realistic dance motions from audio. *Computers & Graphics*, 94:11–21.

[21] Ylva Ferstl and Rachel McDonnell. Investigating the use of recurrent motion modelling for speech gesture generation. In *Proceedings of the 18th International Conference on Intelligent Virtual Agents*, pages 93–98, 2018.

[22] Ylva Ferstl, Michael Neff, and Rachel McDonnell. Adversarial gesture generation with realistic gesture phasing. *Computers & Graphics*, 2020.

[23] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4346–4354, 2015.

[24] Aphrodite Galata, Neil Johnson, and David Hogg. Learning variable-length markov models of behavior. *Computer Vision and Image Understanding*, 81(3):398–413, 2001.

[25] Chuang Gan, Deng Huang, Peihao Chen, Joshua B Tenenbaum, and Antonio Torralba. Foley music: Learning to generate music from videos. *arXiv preprint arXiv:2007.10984*, 2020.

[26] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. In *2017 International Conference on 3D Vision (3DV)*, pages 458–466. IEEE, 2017.

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

[28] Toni Giorgino et al. Computing and visualizing dynamic time warping alignments in r: the dtw package. *Journal of statistical Software*, 31(7):1–24, 2009.

[29] Rachel Heck and Michael Gleicher. Parametric motion graphs. In *Proceedings of the 2007 symposium on Interactive 3D graphics and games*, pages 129–136, 2007.

[30] Alejandro Hernandez, Jurgen Gall, and Francesc Moreno-Noguer. Human motion prediction via spatio-temporal inpainting. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7134–7143, 2019.

[31] 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*, pages 6626–6637, 2017.

[32] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 36(4):1–13, 2017.

[33] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 35(4):1–11, 2016.

[34] Daniel Holden, Jun Saito, Taku Komura, and Thomas Joyce. Learning motion manifolds with convolutional autoencoders. In *SIGGRAPH Asia 2015 Technical Briefs*, pages 1–4. 2015.

[35] 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.

[36] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. Deep inertial poser learning to reconstruct human pose from sparseinertial measurements in real time. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 37(6):185:1–185:15, Nov. 2018.

[37] Vladimir Iashin and Esa Rahtu. Multi-modal dense video captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 958–959, 2020.

[38] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.

[39] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 5308–5317, 2016.

[40] Ye Jia, Melvin Johnson, Wolfgang Macherey, Ron J Weiss, Yuan Cao, Chung-Cheng Chiu, Naveen Ari, Stella Laurenzo, and Yonghui Wu. Leveraging weakly supervised data to improve end-to-end speech-to-text translation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7180–7184. IEEE, 2019.

[41] 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.

[42] Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyan Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, et al. A comparative study on transformer vs rnn in speech applications. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 449–456. IEEE, 2019.

[43] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.

[44] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[45] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation, 2019.

[46] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion graphs. In *ACM SIGGRAPH 2008 classes*, pages 1–10. 2008.

[47] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE international conference on computer vision*, pages 706–715, 2017.

[48] Jogendra Nath Kundu, Himanshu Buckchash, Priyanka Mandikal, Anirudh Jamkhandi, Venkatesh Babu RADHAKRISHNAN, et al. Cross-conditioned recurrent networks for long-term synthesis of inter-person human motion interactions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2724–2733, 2020.

[49] Kimerer LaMothe. The dancing species: how moving together in time helps make us human. *Aeon*, June 2019.

[50] Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. Dancing to music, 2019.

[51] 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.

[52] Jehee Lee and Sung Yong Shin. A hierarchical approach to interactive motion editing for human-like figures. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 39–48, 1999.

[53] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled transformer for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8928–8937, 2019.

[54] 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.

[55] Zimo Li, Yi Zhou, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. *ICLR*, 2018.

[56] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *SIGGRAPH Asia*, 2015.

[57] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):248, 2015.

[58] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7219–7228, 2018.

[59] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5442–5451, 2019.

[60] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, 2015.

[61] Dirk Moelants. Dance music, movement and tempo preferences. In *Proceedings of the 5th Triennial ESCOM Conference*, pages 649–652. Hanover University of Music and Drama, 2003.

[62] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4903–4911, 2017.

[63] Katherine Pullen and Christoph Bregler. Animating by multi-level sampling. In *Proceedings Computer Animation 2000*, pages 36–42. IEEE, 2000.

[64] Haibo Qiu, Chunyu Wang, Jingdong Wang, Naiyan Wang, and Wenjun Zeng. Cross view fusion for 3d human pose estimation. In *International Conference on Computer Vision (ICCV)*, 2019.

[65] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training, 2018.

[66] Xuanchi Ren, Haoran Li, Zijian Huang, and Qifeng Chen. Music-oriented dance video synthesis with pose perceptual loss. *arXiv preprint arXiv:1912.06606*, 2019.

[67] 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.

[68] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech: Fast, robust and controllable text to speech. In *Advances in Neural Information Processing Systems*, pages 3171–3180, 2019.

[69] Matteo Ruggero Ronchi and Pietro Perona. Benchmarking and error diagnosis in multi-instance pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[70] Takaaki Shiratori, Atsushi Nakazawa, and Katsushi Ikeuchi. Dancing-to-music character animation. In *Computer Graphics Forum*, volume 25, pages 449–458. Wiley Online Library, 2006.

[71] Eli Shlizerman, Lucio Dery, Hayden Schoen, and Ira Kemelmacher-Shlizerman. Audio to body dynamics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7574–7583, 2018.

[72] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Trans. Graph.*, 38(6):209–1, 2019.

[73] Sebastian Starke, Yiwei Zhao, Taku Komura, and Kazi Zaman. Local motion phases for learning multi-contact character movements. *ACM Transactions on Graphics (TOG)*, 39(4):54–1, 2020.

[74] Statista. https://www.statista.com/statistics/249396/top-youtube-videos-views/, 2020. Accessed: 2020-11-09.

[75] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*, 2019.

[76] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7464–7473, 2019.

[77] 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*, 2020.

[78] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[79] 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.

[80] Graham W Taylor and Geoffrey E Hinton. Factored conditional restricted boltzmann machines for modeling motion style. In *Proceedings of the 26th annual international conference on machine learning*, pages 1025–1032, 2009.

[81] Purva Tendulkar, Abhishek Das, Aniruddha Kembhavi, and Devi Parikh. Feel the music: Automatically generating a dance for an input song. *arXiv preprint arXiv:2006.11905*, 2020.

[82] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer, 1999.

[83] Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto. Aist dance video database: Multi-genre, multi-dancer, and multi-camera database for dance information processing. In *Proceedings of the 20th International Society for Music Information Retrieval Conference, ISMIR 2019*, pages 501–510, Delft, Netherlands, Nov. 2019.

[84] Jean-Marc Valin and Jan Skoglund. Lpcnet: Improving neural speech synthesis through linear prediction. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5891–5895. IEEE, 2019.

[85] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[86] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

[87] Subhashini Venugopalan, Huijuan Xu, Jeff Donahue, Marcus Rohrbach, Raymond Mooney, and Kate Saenko. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014.

[88] Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. Neural kinematic networks for unsupervised motion retargetting. In *CVPR*, 2018.

[89] Borui Wang, Ehsan Adeli, Hsu-kuang Chiu, De-An Huang, and Juan Carlos Niebles. Imitation learning for human pose prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7124–7133, 2019.

[90] Haoming Xu, Runhao Zeng, Qingyao Wu, Mingkui Tan, and Chuang Gan. Cross-modal relation-aware networks for audio-visual event localization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3893–3901, 2020.

[91] 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.

[92] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE international conference on computer vision*, pages 4507–4515, 2015.

[93] 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.

[94] Haonan Yu, Jiang Wang, Zhiheng Huang, Yi Yang, and Wei Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4584–4593, 2016.

[95] He Zhang, Sebastian Starke, Taku Komura, and Jun Saito. Mode-adaptive neural networks for quadruped motion control. *ACM Transactions on Graphics (TOG)*, 37(4):1–11, 2018.

[96] Luowei Zhou, Yingbo Zhou, Jason J Corso, Richard Socher, and Caiming Xiong. End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8739–8748, 2018.

[97] Wenlin Zhuang, Congyi Wang, Siyu Xia, Jinxiang Chai, and Yangang Wang. Music2dance: Music-driven dance generation using wavenet. *arXiv preprint arXiv:2002.03761*, 2020.

[98] 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.