

# Transflower: probabilistic autoregressive dance generation with multimodal attention

Guillermo Valle Perez, Jonas Beskow, Gustav Eje Henter, Andre Holzapfel, Pierre-Yves Oudeyer, Simon Alexanderson

▶ **To cite this version:**

Guillermo Valle Perez, Jonas Beskow, Gustav Eje Henter, Andre Holzapfel, Pierre-Yves Oudeyer, et al.. Transflower: probabilistic autoregressive dance generation with multimodal attention. SIGGRAPH Asia 2021 - 14th ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques, Dec 2021, Tokyo, Japan. 10.1145/3478513.3480570 . hal-03449915

**HAL Id: hal-03449915**
**https://hal.science/hal-03449915**

Submitted on 25 Nov 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

arXiv:2106.13871v2 [cs.SD] 28 Oct 2021

# Transflower: probabilistic autoregressive dance generation with multimodal attention

GUILLERMO VALLE-PÉREZ, Inria, Ensta ParisTech, University of Bordeaux, France
GUSTAV EJE HENTER, KTH Royal Institute of Technology, Sweden
JONAS BESKOW, KTH Royal Institute of Technology, Sweden
ANDRE HOLZAPFEL, KTH Royal Institute of Technology, Sweden
PIERRE-YVES OUDEYER, Inria, Ensta ParisTech, University of Bordeaux, France
SIMON ALEXANDERSON, KTH Royal Institute of Technology, Sweden

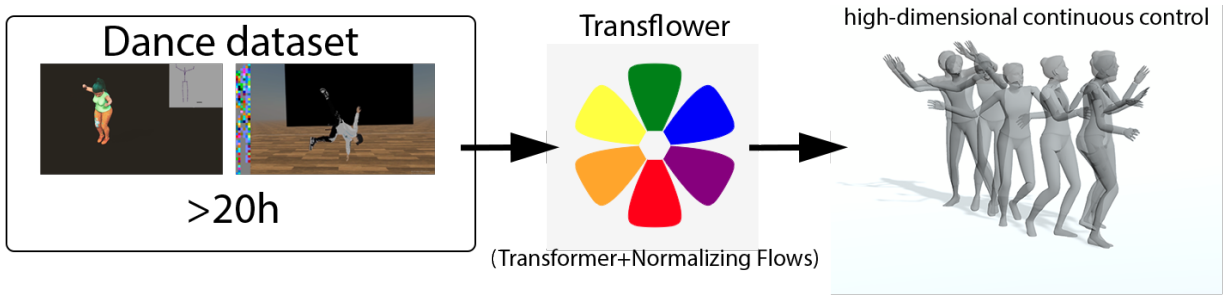



Fig. 1. We have aggregated the largest dataset of 3D dance motion, and used it to train Transflower, a new probabilistic autoregressive model of motion. As a result we obtained a model that can generate dance to any piece of music, which ranks well in terms of appropriateness, naturalness, and diversity.

Dance requires skillful composition of complex movements that follow rhythmic, tonal and timbral features of music. Formally, generating dance conditioned on a piece of music can be expressed as a problem of modelling a high-dimensional continuous motion signal, conditioned on an audio signal. In this work we make two contributions to tackle this problem. First, we present a novel probabilistic autoregressive architecture that models the distribution over future poses with a normalizing flow conditioned on previous poses as well as music context, using a multimodal transformer encoder. Second, we introduce the currently largest 3D dance-motion dataset, obtained with a variety of motion-capture technologies, and including both professional and casual dancers. Using this dataset, we compare our new model against two baselines, via objective metrics and a user study, and show that both the ability to model a probability distribution, as well as being able to attend over a large motion and music context are necessary to produce interesting, diverse, and realistic dance that matches the music.

Authors' addresses: Guillermo Valle-Pérez, guillermo-jorge.valle-perez@inria.fr, Inria, Ensta ParisTech, University of Bordeaux, Bordeaux, France; Gustav Eje Henter, ghe@kth.se, KTH Royal Institute of Technology, Stockholm, Sweden; Jonas Beskow, beskow@kth.se, KTH Royal Institute of Technology, Stockholm, Sweden; Andre Holzapfel, holzap@kth.se, KTH Royal Institute of Technology, Stockholm, Sweden; Pierre-Yves Oudeyer, pierre-yves.oudeyer@inria.fr, Inria, Ensta ParisTech, University of Bordeaux, Bordeaux, France; Simon Alexanderson, simonal@kth.se, KTH Royal Institute of Technology, Stockholm, Sweden.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
© 2021 Copyright held by the owner/author(s).
0730-0301/2021/12-ART1
https://doi.org/10.1145/3478513.3480570

CCS Concepts: • **Computing methodologies** → **Animation**; *Neural networks*; *Motion capture*.

Additional Key Words and Phrases: Generative models, machine learning, normalising flows, Glow, transformers, dance

**ACM Reference Format:**
Guillermo Valle-Pérez, Gustav Eje Henter, Jonas Beskow, Andre Holzapfel, Pierre-Yves Oudeyer, and Simon Alexanderson. 2021. Transflower: probabilistic autoregressive dance generation with multimodal attention. *ACM Trans. Graph.* 40, 6, Article 1 (December 2021), 13 pages. https://doi.org/10.1145/3478513.3480570

## 1 INTRODUCTION

Dancing – body motions performed together with music – is a deeply human activity that transcends cultural barriers, and we have been called "the dancing species" [LaMothe 2019]. Today, content involving dance is some of the most watched on digital video platforms such as YouTube and TikTok. The recent pandemic led dance – as other performing arts – to become an increasingly virtual practice, and hence an increasingly digitized cultural expression.

However, good dancing, whether analog or digital, is challenging to create. Professional dancing requires physical prowess and extensive practise, and capturing or recreating a similar experience through digital means is labour intensive, whether done through motion capture or hand animation. Consequently, the problem of automatic, data-driven dance generation has gathered interest in recent years [Li et al. 2021b, 2020, 2021a; Zhuang et al. 2020]. Access to generative models of dance could help creators and animators, by speeding up their workflow, by offering inspiration, and by opening

up novel possibilities such as creating interactive characters that react to the user's choice of music in real time. The same models can also give insight into how humans connect music and movement, both of which have been identified as capturing important and inter-related aspects of our cognition [Bläsing et al. 2012].

The kinematic processes embodied in dance are highly complex and nonlinear, even when compared to other human movement such as locomotion. Dance is furthermore multimodal, and the connection between music and dance motion is extremely multifaceted and far from deterministic. Generative modelling with deep neural networks is becoming one of the most promising approaches to learn representations of such complex domains. This general approach has already made significant progress in the domains of images [Brock et al. 2018; Karras et al. 2019; Park et al. 2019], music [Dhariwal et al. 2020; Huang et al. 2018], motion [Henter et al. 2020; Ling et al. 2020], speech [Prenger et al. 2019; Shen et al. 2018], and natural language [Brown et al. 2020; Raffel et al. 2019]. Recently, multimodal models are being developed, that learn to capture the even more complex interactions between standard data domains, such as between language and images [Ramesh et al. 2021], or between language and video [Wu et al. 2021]. Similarly, dance synthesis sits at the intersection between movement modelling and music understanding, and is an exciting problem that combines compelling machine-learning challenges with a distinct sociocultural impact.

In this work, we tackle the problem of music-conditioned 3D dance motion generation through deep learning. In particular, we explore two important factors that affect model performance on this difficult task: 1) the ability to capture patterns that are extended over longer periods of time, and 2) the ability to express complex probability distributions over the predicted outputs. We argue that previous works are lacking in one of these two properties, and present a new autoregressive neural architecture which combines a transformer [Vaswani et al. 2017] to encode the multimodal context (previous motion, and both previous and future music), and a normalizing flow [Papamakarios et al. 2021] head to faithfully model the future distribution over the predicted modality, which for dance synthesis is the future motion. We call this new architecture *Transflower* and show, through objective metrics and human evaluation studies, that both of these factors are important to model the complex distribution of movements in dance as well as their dependence on the music modality. Human evaluations are the gold standard to evaluate the perceptual quality of generative models, and are complementary to the objective metrics. Furthermore, they allow us to evaluate the model on arbitrary "in-the-wild" songs downloaded from YouTube, for which no ground truth dance motion is available.

One of the biggest challenges in learning-based motion synthesis is the availability of large-scale datasets for 3D movement. Existing datasets are mainly gathered in two ways: in a motion capture studio [CMU Graphics Lab 2003; Ferstl and McDonnell 2018; Lee et al. 2019a; Mahmood et al. 2019; Mandery et al. 2015; Troje 2002], which provides the highest quality motion, but requires expensive equipment and is difficult to scale to larger dataset sizes, or via monocular 3D pose estimation from video [Habibie et al. 2021; Peng et al. 2018b], which trades off quality for a much larger availability of videos from the Internet.

In this paper we present the largest dataset of 3D dance motion, combining different sources and motion capture technologies. We introduce a new approach to obtain large-scale motion datasets, complementary to the two mostly used in previous works. Specifically, we make use of the growing popularity and user base of virtual reality (VR) technologies, and of VR dancing in particular [Lang 2021], to find participants interested in contributing dance data for our study. We argue that, while consumer-grade VR motion capture does not produce as high quality as professional motion capture, it is significantly better and more robust than current monocular 3D pose estimation from video. Furthermore, it is poised to improve both in quality and availability as the VR market grows [Statista 2020], offering potential new avenues for participatory research.

We also collect the largest dataset of dance motion using professional motion capture equipment, from both casual dancers and a professional dancer. Finally, to train our models, we combine our new data, with two existing 3D dance motion datasets, GrooveNet [Alemi et al. 2017] and AIST++ [Li et al. 2021a], which we standardize to a common skeleton. In total, we have over 20 h of dance data in a wide variety of dance styles, including freestyle, casual dancing, and street dance styles, as well as a variety of music genres, including pop, hip hop, trap, K-pop, and street dance music. Furthermore, although all of our data sources offer higher quality motion capture than 3D motion data estimated from monocular video, the different sources offer different levels of quality, and different capture artifacts. We find that this diversity in data sources, on top of the large diversity in dance styles and skill levels, makes deterministic models unable to converge to model the data faithfully, while probabilistic models are able to adapt to such heterogeneous material.

Our contributions are as follows:

- We present a novel architecture for autoregressive probabilistic modelling of high-dimensional continuous signals, which we demonstrate achieves state of the art performance on the task of music-conditioned dance generation. Our architecture is, to the best of our knowledge, the first to combine the benefits of transformers for sequence modelling, with normalizing flows for probabilistic modelling.
- We introduce the largest dataset of 3D dance motion generated with a variety of motion capture systems. This dataset also serves to showcase the potential of VR for participatory research and democratizing mocap.
- We evaluate our new model objectively and in a user study against two baselines, showing that both the probabilistic and multimodal attention components are important to produce natural and diverse dance matching the music.
- Finally, we explore the use of fine-tuning and "motion prompting" to attain control over the quality and style of the dance.

Our paper website at metagen.ai/transflower provides data, code, pre-trained models, videos, supplementary material, and a demo for testing the models on any song with a selection of starting motion.

## 2 BACKGROUND AND PRIOR WORK

### 2.1 Learning-based motion synthesis

The task of generating 3D motion has been tackled in a variety of ways. The traditional approaches to motion synthesis were based

on retrieval from motion databases and motion graphs [Arikan and Forsyth 2002; Chao et al. 2004; Kovar and Gleicher 2004; Kovar et al. 2002; Lee et al. 2002; Safonova and Hodgins 2007; Takano et al. 2010]. Recently, there has been more interest in statistical and learning-based approaches, which can be more flexible and scale to larger datasets and more complex tasks. Holden et al. [2020] explored a continuum between the traditional retrieval-based approaches to motion synthesis and the more scalable deep learning-based approaches, showing that combining ideas from both may be fruitful.

Among the learning-based techniques, most works follow an autoregressive approach, where either the next pose or the next key pose in the sequence is predicted based on previous poses in the sequence. For dance, the prediction is also conditioned on music features, typically spanning a window of time around the time of prediction. We refer to both the previous poses and this window of music features together as the *context*. We can categorize the autoregressive methods along the factors proposed in the introduction, i.e., the way the autoregressive model handles its inputs (context), and the way it handles its outputs (predicted motion). Later in this section, we also compare approaches according to the amount of assumptions they make, and the type of learning algorithm.

**Context-dependence**. We have seen an evolution towards models that more effectively retain and utilize information from wider context windows. The first works applying neural networks to motion prediction relied on recurrent neural networks like LSTMs, which were applied to unconditional motion [Fragkiadaki et al. 2015; Zhou et al. 2018] and dance synthesis [Crnkovic-Friis and Crnkovic-Friis 2016; Tang et al. 2018]. LSTMs represent the recent context in a latent state. However, this latent state can act as a bottleneck hindering information flow from the past context, thus limiting the extent of the temporal correlations the network can learn. Different architectures have been used to tackle this problem. Bütepage et al. [2017]; Holden et al. [2017]; Starke et al. [2020] directly feed the recent history of poses through a feedforward network to predict the next pose, while Zhuang et al. [2020] use a WaveNet-style architecture to extend the context even further for dance synthesis. Recently, Li et al. [2021a] introduce a cross-modal transformer architecture (extending [Vaswani et al. 2017]) that learns to attend to the relevant features over the last 2 seconds of motion, as well as the neighbouring 4 seconds of music, for dance generation.

**Output modelling**. Most works in motion synthesis have treated the next-pose prediction as a deterministic function of the context. This includes the earlier work using LSTMS [Fragkiadaki et al. 2015; Tang et al. 2018; Zhou et al. 2018], and some of the most recent work on dance generation [Li et al. 2021b,a]. However, in many situations, the output motion is highly unconstrained by the input context. For example, there are many plausible dance moves that can accompany a piece of music, or many different gestures that fit well with an utterance [Alexanderson et al. 2020]. Earlier approaches to model a probabilistic distribution over motions include Gaussian mixture models [Crnkovic-Friis and Crnkovic-Friis 2016] and Gaussian processes latent-variable models [Grochow et al. 2004; Levine et al. 2012; Wang et al. 2008]. VAEs weaken the assumption of Gaussianity, and have been applied to motion synthesis [Habibie et al. 2017; Ling et al. 2020]. Recently, Petrovich et al. [2021] used VAEs combined with a transformer for non-autoregressive motion synthesis – they predict the whole motion "at once", as the output of a full-attention transformer. Their architecture allows for learning complex multimodal motion distributions, but their non-autoregressive approach limits the length of the generated sequences. MoGlow [Henter et al. 2020] models the motion with an autoregressive normalizing flow, allowing for fitting flexible probability distributions, with exact likelihood maximization, producing state of the art motion synthesis results. In Li et al. [2020], they discretize the joint angle space. However, their multi-softmax output distribution assumes independence of each joint, which is unusual in many cases. Lee et al. [2019b] develop a music-conditioned GAN to predict a distribution over latent representations of the motion. However, in order to stabilize training, they apply an MSE regularization loss on the latents which may limit its ability to model complex distributions. Li et al. [2021b] also introduce an adversarially-trained model which, however, does not have a noise input – no source of randomness – and thus cannot be said to model a music-conditioned probability distribution.[1]

**Domain-specific assumptions**. A third dimension in which we can compare the different approaches to motion synthesis is the amount of domain-specific assumptions they make. To disambiguate the predictions from a deterministic model, or to increase motion realism, different works have added extra inputs to the model, tailored at the desired types of animations, including foot contact [Holden et al. 2016], pace [Pavllo et al. 2018], and phase information [Holden et al. 2017; Starke et al. 2020]. For dance synthesis, Lee et al. [2019b] and Li et al. [2021b] use the observation that dance can often be fruitfully decomposed into short movement segments, whose transitions lie at music beats. Furthermore Li et al. [2021b] develop an architecture that includes inductive biases specific to kinematic skeletons, as well as a bias towards learning local temporal correlations. On the other end of the spectrum, Henter et al. [2020] presents a general sequence prediction model that makes few assumptions about the data. This allows it to be applied to varied tasks like humanoid or quadruped motion synthesis, or gesture generation [Alexanderson et al. 2020] without fundamentally changing the model, but it has not yet been applied to dance. For dance synthesis, Li et al. [2021a] demonstrate that a similarly general-purpose model can produce impressive results.

**Learning algorithm**. An alternative approach to learn to generate realistic movements from motion capture data is to use a set of techniques known as imitation learning (IL). Merel et al. [2017] use generative adversarial imitation learning (GAIL), a technique closely related to GANs, to learn a controller for a physics-based humanoid to imitate different gait motions. Peng et al. [2018a] and Peng et al. [2021] extend this work to characters that learn a diversity of skills, with a variety of morphologies. This approach can learn to imitate mocap data in a physics-based environment, so that the character movement is automatically physically realistic, which is necessary but not sufficient for natural motion. Ling et al. [2020] also found combining IL approaches with previously described supervised and self-supervised learning approaches to be a promising direction to get the benefits of both.

[1] However, it is possible that the music input itself could serve as a kind of noise source, allowing the model to effectively learn a probability distribution. This deserves further investigation.

Overall, we expect that learning-based motion synthesis will follow a similar trend as in other areas where machine learning is applied to complex data distributions: models that can flexibly attend over a large context, like transformers, while being able to model complex distributions over its outputs, produce the best generative results, when enough data is available [Brown et al. 2020; Dosovitskiy et al. 2020; Henighan et al. 2020; Kaplan et al. 2020; Ramesh et al. 2021]. Here we present what we believe is the first model for autoregressive motion synthesis combining both of these desirable properties, which we expect to become crucial as motion capture datasets grow both in size and diversity. Furthermore, we use the model for music-conditioned dance generation, demonstrating that it is able to be used for tasks involving multiple modalities (music and movement in our case).

### 2.2 Other data-driven dance motion synthesis

Although the focus of this work is on purely learning-based approaches to dance synthesis, these are not the only data-driven methods to for generating dance. As we discussed in section 2.1, there is in fact a continuum between completely non-learning based and purely learning-based approaches. In general, learning-based techniques typically trade off a larger compute cost for training, for a reduced cost at inference [Holden et al. 2020], which often amortizes the training cost. There are also several dimensions along which machine learning techniques can be introduced in a dance synthesis pipeline. Here we focus on the motion synthesis part.

Several works have approached the problem of dance generation using motion graphs, where transitions between pre-made motion clips are used to create a choreography. In effect, these methods generate motion by selection, whereas deep learning can be seen as more akin to interpolation. Fan et al. [2011] and Fukayama and Goto [2015] used statistical methods for traversing the graph, but recently, deep learning techniques have been developed for traversing the motion graph [Kang et al. 2021; Ye et al. 2020]. These approaches tend to produce reliable and high quality dances, that are easier to edit. However, a lot of motion-graph-based dance synthesis works rely on datasets of key-framed animations, which may cause the generated dances to look less natural. The large mocap dance database we are introducing may therefore help enable more natural motion also for this class of techniques.

### 2.3 Dance datasets

Previous research on dance generation has been conducted with data obtained from a variety of different techniques. Alemi et al. [2017] and Zhuang et al. [2020] recorded mocap datasets of dances synchronized to music, totalling 23 min and 58 min, respectively. Li et al. [2021a] use the AIST dance dataset [Tsuchida et al. 2019] to obtain 3D dance motion using multi-view 3D pose estimation. Lee et al. [2019b] and Li et al. [2020] use monocular 3D pose estimation from YouTube videos to produce a dataset of 3D dance motion of 71 h and 50 h in total, respectively. Monocular 3D pose estimation is an area of active research [Bogo et al. 2016; Mathis et al. 2020; Rong et al. 2021], but current methods suffer from inaccurate root motion estimation [Li et al. 2021a], so these works tend to focus on the joint movements. Finally, Li et al. [2021b] introduce a 5 h animated dance dataset, created by professional animators. Overall, we find that there is a trade-off between data quality and data availability. In this work, we present a new way of collecting motion data, from remote VR user participants, which pushes the currently available Pareto frontier, with data quality approaching that of mocap equipment, and increasing availability as the number of VR users grows. We think that the democratization of motion capture with VR, brings exciting possibilities for researchers and VR users alike. In particular, the ability to crowdsource data at scale should make it possible to harness the scaling phenomena seen in many other generative-modelling tasks [Henighan et al. 2020], and also study the effects of scaling on model performance. Compared to datasets like AIST++ [Li et al. 2021a], we believe that our dataset captures a larger diversity of skill levels, as well as dance styles not present in AIST++.

## 3 DATASET

In this paper we introduce two new large scale datasets of 3D dance motion synchronized to music: the PMSD dance dataset, and the ShaderMotion VR dance dataset. We combine these datasets with two previous existing dance datasets, AIST++ [Li et al. 2021a] and GrooveNet [Alemi et al. 2017], to create our combined dataset on which we train our models. We standardize all the datasets into a common skeleton (the one used for the PMSD dataset), and will publicly release the data. We here describe the two new datasets, and compare all the sources, including existing ones, in table 1.

**Popular music and street dance (PMSD) dataset**. The PMSD dance dataset consists of synchronized audio and motion capture recordings of various dancers and dance styles. The data was recorded with an Optitrack Prime41 motion capture system (17 cameras at 120 Hz) and is divided in two parts. The first part (PMSDCasual) contains 142 minutes of casual dancing to popular music by 7 non-professional dancers. The 37 markers where solved to a skeleton with 21 joints. The second part (PMSDStreet) contains 44 minutes of street dance performed by one professional dancer. The dances are divided in three distinct styles: Hip-Hop, Popping and Krumping. The music was selected by the dancer to be appropriate for each style. In this setup we used 65 markers (45 on the body and 2 on each finger), and solved the data to a skeleton with 51 joints, including fingers and hinged toes. Compared to the casual dances, the street dances have considerably more complex choreographies with more tempo-shifts and less repetitive patterns.

**ShaderMotion VR dance dataset**. The data was recorded by participants who dance in the social VR platform VRChat[2], using a tool called ShaderMotion that encodes their avatar's motion into a video [lox9973 2021]. Their avatar follows their movement using inverse kinematics, anchored to their real body movement via a 6-point tracking system, where the head, hands, hips, and feet are being tracked, using a VR headset, and HTC Vive trackers. The videos with encoded motion can then be converted back to motion on a skeleton rig via the MotionPlayer script provided in the ShaderMotion repository. The data includes finger tracking, with an accuracy dependent on the VR equipment used. We further divide the data into four components, which have different dancers and styles (see table 1). We only included part of the ShaderMotion for

[2] https://hello.vrchat.com/

| Source | Minutes | #dancers | Styles |
|---|---|---|---|
| AIST++* | 312.1 | 30 | Break, Pop, Lock, Hip Hop, House, Waack, Krump, Street Jazz, Ballet Jazz |
| GrooveNet* | 25.0 | 1 | GrooveNet |
| PMSDCasual* | 142.1 | 1 | Casual |
| PMSDStreet* | 88.1 | 1 | Krump, Hip Hop, Pop |
| SM1* | 229.9 | 2 | Freestyle |
| SMVibe* | 40.2 | 6 | Freestyle, Ballet |
| SMJustDance | 188.3 | 1 | JustDance |
| SMDavid | 15.9 | 1 | Break, Krump, Hip Hop, Pop |
| SMKonata | 138.9 | 1 | Freestyle |
| Syrtos | 49.7 | 6 | Syrtos |
| Total | 1240.2 | 49 | - |

Table 1. **Sources of dance data in our dataset**. PMSD refers to the components of the PMSD dance dataset, and SM refers to the components of the ShaderMotion VR dance dataset. We mark with * the components which we used to train the models in this paper (we got the rest of the data recently, and are working on scaling up the models to the full dataset). Note that there is one dancer in common between SMKonata and SMVibe. Note that we classify GrooveNet and JustDance as their own styles as their dances mostly consist on repeating certain motifs. GrooveNet consists mostly of simple rhythmic motifs, while JustDance has a larger diversity of motifs that appear in the game JustDance.

training (the ShaderMotion1 and ShaderMotionVibe components), because we only obtained the rest of the data recently. We plan to release models trained on the complete data soon. Some examples of how the VR dancing in our dataset looks, for street dance style, can be seen in this URL https://www.youtube.com/playlist?list=PLmwqDOin_Zt4WCMWqoK6SdHlg0C_WeCP6

**Syrtos dataset**. The Syrtos dataset consists of synchronized audio and motion capture recordings of a specific Greek dance style – the Cretan Syrtos – from six dancers. Eleven performances are contained in the data, with all but one dancer performing twice, giving a total duration of approximately 50 min. The data was recorded with the dancers individually during ethnographic fieldwork in Crete in 2019 [Holzapfel et al. 2020], using a Xsens[3] inertia system with 17 sensors operating at 240 Hz, and post-processed through Xsens software to a skeleton with 22 joints. The audio data – performed live by musicians during the recordings – consists of single tracks containing the individual instruments. Note that we did not use the Syrtos data in this study, but release it for future research.

## 4 METHOD

We introduce a new autoregressive probabilistic model with attention which we call Transflower. The architecture combines ideas for autoregressive cross-modal attention using transformers from Li et al. [2021a] with ideas for autoregressive probabilistic models using normalizing flows from Henter et al. [2020]. The model is designed to be applicable to any autoregressive probabilistic modelling with multiple modalities as inputs and outputs, although in this paper we focus on the application to music-conditioned motion generation. The code and trained models will be made available.

We model dance-conditioned motion autoregressively. To do this, we represent motion as a sequence of poses $\mathbf{x} = \{x_i\}_{i=1}^{i=N} \in \mathbb{R}^{N \times d_x}$ sampled at $N$ times $t_i$ at a fixed sampling rate, and music as a sequence of audio features $\{m_i\}_{i=1}^{i=N} \in \mathbb{R}^{N \times d_m}$ extracted from windows centered around the same times $t_i$. $d_x$ and $d_m$ are the number of features for the pose and music samples. For our experiments we sample motion poses and music features at 20 Hz. The autoregressive task is to predict the $i$th pose given all previous poses, and some music context. For simplicity, we restrict the prediction to depend only on the previous $k_x$ poses and the previous $k_m$ and future $l_m$ music features. The probability distribution over the entire motion can be written as a product, using the chain rule of probability:

$$p(\mathbf{x}) = \prod_{i=1}^{N} p(x_i | x_{i-k_x}, ..., x_{i-1}; m_{i-k_m}, ..., m_{i+l_m}) \tag{1}$$

where $x$ or $m$ with indices smaller than 0 are either padded, or represent the "context seed" (the initial $k_x$ poses and initial $k_m + l_m$ music features) that is fed to the model. We experimented with different motion seeds for the autoregressive generation in section 5.3.

Transflower is composed of two components, a *trans*former encoder, to encode the motion and music context, and a normalizing *flow* head, that predicts a probability distribution over future poses, given the context. We express $p(x_i | x_{i-k_x}, ..., x_{i-1}; m_{i-k_m}, ..., m_{i+l_m}) = p(x_i | \mathbf{h})$ with a normalizing flow conditioned on a latent vector $\mathbf{h}$. This vector encodes the context via a transformer encoder $\mathbf{h} = f(x_{i-k_x}, ..., x_{i-1}; m_{i-k_m}, ..., m_{i+l_m})$. For the encoder, we use the design proposed by Li et al. [2021a], with two transformers that encode the motion part of the context $\mathbf{h}_x = f_x(x_{i-k_x}, ..., x_{i-1}) \in \mathbb{R}^{k_x \times d_m}$ and the music part $\mathbf{h}_x = f_m(m_{i-k_m}, ..., m_{i+l_m}) \in \mathbb{R}^{(l_m+k_m) \times d_m}$ separately, where the output dimension $d_m$ is the same for both pose and music encoders. The outputs of these two transformers are then concatenated into a cross-modal transformer $\tilde{\mathbf{h}} = f_{cm}(\mathbf{h}_x, \mathbf{h}_x) \in \mathbb{R}^{(l_m+k_m+k_x) \times d_h}$. The latent $\mathbf{h} \in \mathbb{R}^{K \times d_h}$ corresponds to a prefix of this output $\tilde{\mathbf{h}}$ (the way $K$ is chosen will be explained later). We use the standard transformer encoder implementation in PyTorch, and use T5-style relative positional embeddings [Raffel et al. 2019] in all transformers to obtain translation invariance across time [Wennberg and Henter 2021]. While Li et al. [2021a] use the outputs of the cross-modal transformer as the deterministic prediction of the model, we interpret the first $K$ outputs of the encoder as a latent vector $\mathbf{h}$ on which we condition the normalizing flow output head.

We use a normalizing flow model based on 1x1 invertible convolutions and affine coupling layers [Henter et al. 2020; Kingma and Dhariwal 2018]. Like Ho et al. [2019], we use attention for the affine coupling layers, but unlike them, we remove the convolutional layers, and use a pure-attention affine coupling layer. The inputs to the normalizing flow correspond to $N$ predicted poses of dimension $d_x$. The affine coupling layer splits the inputs $z_i \in \mathbb{R}^{N \times d_x}$ channel-wise into $z_i', z_i'' \in \mathbb{R}^{N \times d_x/2}$ and applies an affine transformation to $z_i''$ with parameters depending on $z_i'$, i.e. $z_{i+1} = \mathbf{A}(z_i', \mathbf{h}) \odot z_i'' + \mathbf{B}(z_i', \mathbf{h})$.

[3] https://www.xsens.com/

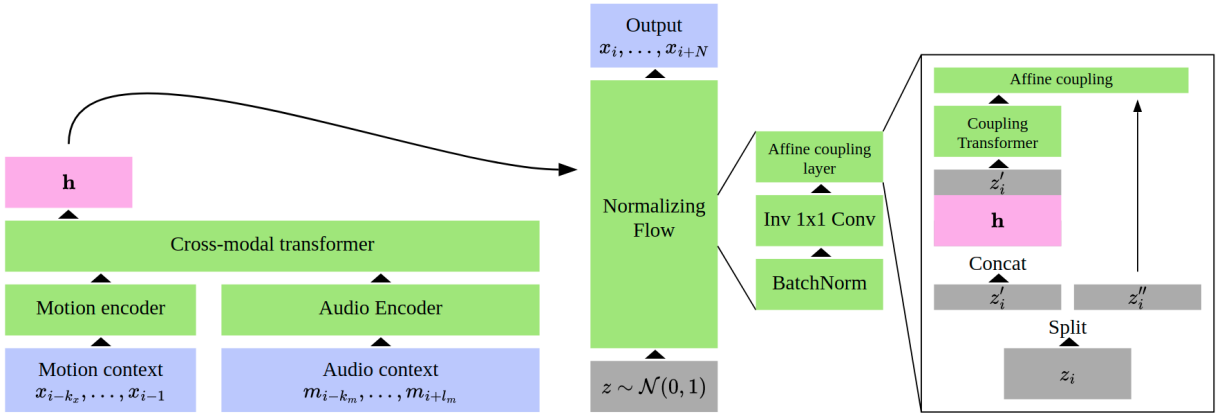



Fig. 2. **The Transflower architecture**. Green blocks represent neural network modules which take input from below and feed their output to the module above. In the affine coupling layer, split and concatenation are done channel-wise, and the 'affine coupling' is an element wise linear scaling with shift and scale parameters determined by the output of the coupling transformer, as in Henter et al. [2020]; Kingma and Dhariwal [2018]. The normalizing flow is composed of several blocks, each of containing a batch normalization, an invertible 1x1 convolution, and an affine coupling layer. The motion, audio, and cross-modal transformers are standard full-attention transformer encoders, like in Li et al. [2021a], except that we use T5-style relative positional encodings.

These affine parameters are the output of the *coupling transformer*, $(\mathbf{A}, \mathbf{B}) = f_{ct}(\tilde{x}) \in \mathbb{R}^{N \times 2d_x}$, where $\tilde{x} = (z_i', \mathbf{h}) \in \mathbb{R}^{N \times (d_x + d_h)}$. Like in MoGlow [Henter et al. 2020], we concatenate the latent vector $\mathbf{h}$ to the inputs of $f_{ct}$ along the channel dimension, to condition the normalizing flow on the context. The main architectural difference is that MoGlow primarily relies on LSTMs for propagating information over time, whereas the proposed model uses Transformers and attention mechanisms. We believe this should make it easier for the model to discern and focus on specific features in a long context window, which we think may be important for learning choreography, executing consistent dance moves, and also for being sensitive to the music. Kingma and Dhariwal [2018] used ActNorm layers motivated by the inaccuracy of batch normalization when using very small batch sizes (they used a batch size of 1). As we use larger batch sizes, we found that batch normalization sometimes produced moderately faster convergence, so we use it in our networks instead of ActNorm, unless specified otherwise. Our model uses 16 of these normalizing flow blocks.

We also found, like in Li et al. [2021a], that training to predict the next $N$ poses improves model performance. We therefore model the normalizing flow output as a $N \times d$ tensor ($d$ being the dimension of the pose vector and $N$ the sequence dimension). With this setup, the transformers in the coupling layers $\mathbf{A}$ act along this sequence dimension, and the 1x1 convolutions act independently on each element in the sequence. The transformer encoder latent "vector" $\mathbf{h}$ is then interpreted as a $K \times d_h$ tensor where $d_h$ is the output dimension of the transformer encoder. By making $K$ and $N$ equal we can concatenate $\mathbf{h}$ with the input to $\mathbf{A}$ along the channel dimension. We show a diagram of the whole architecture in fig. 2.

**Motion features**. We retarget all the motion data to the same skeleton with 21 joints (including the root) using Autodesk Motion Builder. We represent the root (hip) motion as $(\Delta x, \Delta z, y, \theta_1, \theta_2, \theta_3, \Delta r_y)$ where $\Delta x$ and $\Delta z$ are the position changes relative to the root's ground-projected coordinate frame, i.e. the coordinate frame obtained by removing the roots rotation around the $y$ (vertical) axis, so that $\Delta x$ represents sideways movement and $\Delta z$ represents forward movement. $y$ is the vertical position of the root in the base coordinate system, that is the height from the floor. $\theta$ is a exponential map representation of 3D rotation of the root joint with respect to the root's ground-projected frame and $\Delta r_y$ is the change of 2D facing angle. For all the joints we use exponential map parametrization [Grassia 1998] of the rotation, resulting in 3 features per (non-root) joint, and a total 67 motion features.

**Audio features**. To represent the music, we combine spectrogram features with beat-related features, by concatenating:

- 80 dimensional mel-frequency logarithmic magnitude spectrum with a hop size equal to 50 ms.
- One dimensional spectral flux onset strength envelope feature as provided by Librosa[4].
- Two-dimensional output activations of the RNNDownBeatProcessor model in the Madmom toolbox[5].
- Two-dimensional beat features extracted from the two principal components of the last layer of the beat detection neural network in stage 1 of DeepSaber [Labs 2019], which is the same beat-detection architecture in Dance Dance Convolution [Donahue et al. 2017], but trained on Beat Saber levels.

All the above features were used with settings to obtain the same frame rate as for the mel-frequency spectrum (20 Hz). We included both Madmom and DeepSaber features because we observed in preliminary investigations that while the Madmom beat detector worked well for detecting the regular beats of a song, the DeepSaber features sometimes worked better as general onset detectors.

After processing the data into the above features, we individually standardize them to have a mean of 0 and standard deviation of 1 over the whole dataset used for training.

[4] https://github.com/librosa/librosa
[5] https://github.com/CPJKU/madmom

**Training**. We train both Transflower and the MoGlow baseline [Henter et al. 2020] on 4 V100 Nvidia GPUs with a batch size of 84 per GPU, for 600k steps, which took 7 days. We used a learning rate of $7 \times 10^{-5}$, which we decay by a factor of 0.1 after 200k iterations, and again after 400k iterations. The AI Choreographer baseline [Li et al. 2021a] was trained for 600k iterations on a single TPUv3 with a batch size of 128 (per TPU core) and a learning rate of $1 \times 10^{-4}$ decayed to $1 \times 10^{-5}$ and $1 \times 10^{-6}$ after 100k and 200k iterations respectively. During training, we use "teacher forcing", that is the inputs to the model come from the ground truth data, rather than autoregressively from model outputs. The architecture hyperparameters for the different models are given in table 5 in appendix A, where we also explain how these hyperparameters were chosen.

**Synthesis**. Transflower and MoGlow both run at over 20 Hz on an Nvidia V100 GPU, while the transformer from AI Choreographer runs at 100 Hz on an Nvidia V100.

**Fine-tuning**. We investigate the effect of fine-tuning Transflower on the PMSD motion dataset, the portion of our data with highest quality motion tracking. We train the model for an extra 50k iterations only on this dataset. We found that training for longer reduced the diversity, and 50k iterations produced a good trade-off between diversity and improved quality of produced motions, as we find in section 5. Note that the baselines were not fine-tuned in this manner, and should not be compared directly to this fine-tuned system.

**Motion "prompting"**. We also explored the role of the motion seed in autoregressive motion generation. We seed the different models with 5 different seeds (6 seconds long, chosen to be representative of the different styles in our dataset), and compare how the seed affects the style of dancing. We find that the seed can indeed be used to control the style of dancing, serving as a weak form of "prompting" similar to the current trend in language models [Brown et al. 2020; Reynolds and McDonell 2021]. We show more detailed results in section 5.3.

## 5 EXPERIMENTS AND EVALUATION

We compare Transflower with the deterministic transformer model from AI Choreographer [Li et al. 2021a] and with the probabilistic motion generation model MoGlow [Henter et al. 2020], which does not use attention and has not been applied to dance motion synthesis before. In our experiments, we train all the models with the same amount of past motion context, and past and future music context on the same dataset (the marked components in table 1). Comparing MoGlow with Transflower, we can find out the effect of using attention for learning the dependence on the context (Transflower), versus using a combination of recent frame concatenation and LSTMs (MoGlow). Comparing AI Choreographer with Transflower, we can discern the effect of having a probabilistic (Transflower) versus a deterministic (AI Choreographer) model.

### 5.1 Objective metrics

We look at two objective metrics that capture how realistic the motion is, and how well it matches the music. For the evaluation of the objective metrics we use a test set consisting of 33 songs and motion seeds that were not found in the training set (i.e., excluding AIST++, which had song overlap with the training data), and 27 shorter songs (from AIST++) that were found on the training set but with a different motion seed. Of the 33 non-AIST++ songs, 18 were ones randomly held out from the training set, while the other 15 were added manually. We generated samples from each of the models, for 5 different motion seeds, and evaluated the metrics on the full set of 300 generated sequences.

| Model | FPD | FMD |
|---|---|---|
| AI Choreographer | 963.9 | 2977.4 |
| MoGlow | 600.2 | 1847.6 |
| Transflower (fine-tuned) | 549.0 | 1711.9 |
| Transflower | **511.6** | **1610.5** |

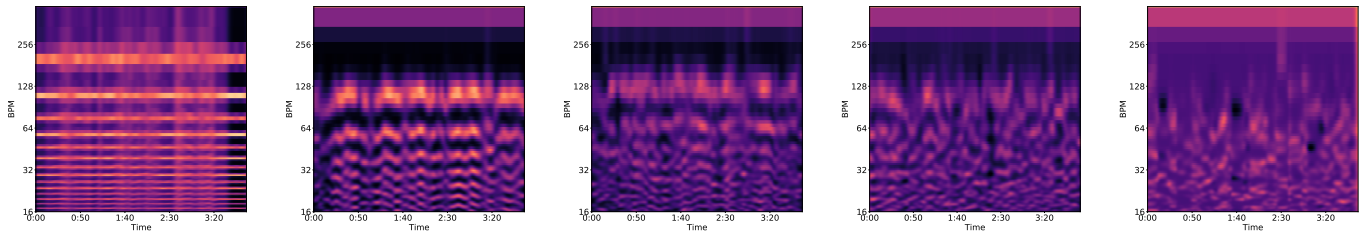
Table 2. **Realism metrics**. Fréchet pose distance (FPD) and Fréchet movement distance (FMD) for the three models we compare, as well as Transflower fine-tuned on the dance dataset.

| | TFF | TF | MG | AIC | Match | Mismatch |
|---|---|---|---|---|---|---|
| Mean | 0.25 | 0.27 | 0.34 | 0.44 | 0.20 | 0.22 |
| SD | 0.33 | 0.31 | 0.52 | 0.54 | 0.24 | 0.67 |

Table 3. **Beat alignment**. Mean and standard deviations of the time offset between musical and kinematic beats(s).

**Realism metric**. For assessing realism, we use the Fréchet distance between the distribution of poses $p_i$ and the distribution of concatenations of three consecutive poses $(p_{i-1}, p_i, p_{i+1})$, which captures information about pose, joint velocity, and joint acceleration. We call these measures the Fréchet pose distance (FPD) and the Fréchet movement distance (FMD), respectively. The measures were computed on the "raw" pose features, without mean and variance normalization. The results are shown in table 2. We can see that AI Choreographer struggles to faithfully capture the variety (of both styles and tracking methods) in our dataset. We observe it often produces chaotic movement or freezes into a mean pose. MoGlow does better, while Transflower (both fine-tuned and non-fine-tuned) capture the distribution of real movements best (with the non-fine-tuned slightly better, as expected).

**Music-matching metric**. In addition to the realism measure above, we objectively assess rhythmical aspects of the music and dance based on metrics calculated from audio and motion beats. These are relatively easy to track, but are by far are not the only aspect that matters in good dancing. To extract the audio beats, we employed the beat-tracking algorithms of Böck and Schedl [2011]; Krebs et al. [2015], included in the Madmom library. To detect dance beats, we rely on a large body of studies on the relation between musical meter and periodicity of dance movements [Burger et al. 2013; Haugen 2014; Misgeld et al. 2019; Naveda and Leman 2011; Toiviainen et al. 2010]. In specific, we draw on observations from Toiviainen et al. [2010], showing that audio beat locations closely match the points of maximal downward velocity of the body (which is straightforward to measure at the hip), a finding consistent with observations from Haugen [2014]; Misgeld et al. [2019] that the center of gravity of the body provides stable beat information. Hence, we extracted the kinematic beat locations as the local minima of the $y$-velocity of the Hips-joint. A visualization of the relation between




(a) Music (b) TFF (c) TF (d) MG (e) AIC

Fig. 3. Music and kinematic tempograms for one complete dance. From left to right: Music, Transflower fine-tuned (TFF), Transflower (TF), MoGlow (MG) and AI Choreographer (AIC). The vertical axis corresponds to frequency, measured in beats per minute (bpm).

the tempo processes emerging from dance beats of the various systems and the audio beats is shown in Figure 3, where we depict audio and visual tempograms [Davis and Agrawala 2018] for one complete dance. For the visual histograms, we used the local Hips velocity at the kinematic beat locations as magnitude information.

The visual tempograms in Figure 3 show that the music has a very strong and regular rhythmic structure, visible as horizontal bands. Among the different models, it is apparent that the dance generated by Transflower (especially after fine-tuning) is characterized by the strongest relation to the audio beat, with a pronounced periodicity at half the estimated audio beat (i.e., around 120 bpm). This periodicity becomes more and more washed out for the other models as we move to the right through the subfigures, indicating less rhythmically consistent motion. To quantify the alignment between the audio and kinematic beats, we calculate the absolute time offset between each music beat and its closest kinematic beat over the complete set of generated seeds and motions. The mean and standard deviation of the offsets for the evaluated systems are reported in table 3 together with the corresponding values for the complete (matched) training data as well as randomly paired (mismatched) music and motion. It is apparent that the proposed model leads to an improved alignment as compared to the baseline systems.

## 5.2 User study

Qualitative user studies are important to evaluate generative models, as perceived quality of different metrics tends to be the most relevant metric for many downstream applications. We thus performed a user study to evaluate our model and the baselines along three axis: naturalness of the motion, appropriateness to the music, and diversity of the dance movements.

To perform the user study we used 17 songs that were not present in the training set. Among these, we include 10 songs obtained "in-the-wild" from YouTube, that may not necessarily match the genres found in our datasets, and for which no ground truth dance is available. We use a fixed motion seed from the PMSDCasual dataset, as it consists of a generic T-pose. However, for the AI Choreographer baseline, this seed often caused the model to degenerate into a frozen pose, probably because the PMSDCasual T-pose has particularly high uncertainty over which motion will follow, making the deterministic model more likely to regress into a mean pose. To arrive to the 17 songs used for evaluation, we thus used a seed from ShaderMotion for AI Choreographer to alleviate this problem, and removed from the evaluation the remaining sequences where AI Choreographer still regressed to a mean pose.

We rendered 15-second animation clips with a stylized skinned character for each of the 17 songs and the four models, yielding a total of 68 stimuli. We performed three separate perception experiments, detailed below. All experiments were carried out online and participants were recruited via a crowd-sourcing platform (Prolific). The presentation order of the stimuli was randomized between the participants. In each of the experiments, 25 participants rated each video clip on a five point Likert-type scale. Participants were between 22 and 66 years of age (median 34), 39% male, 61% female.

- **Naturalness**: We asked participants to rate videos of dances generated by the different models according to the question *On a scale from 1 to 5: how natural is the dancing motion? I.e. to what extent the movements look like they could be carried out by a dancing human being?* where 1 is *very unnatural* and 5 is *very natural*. We removed the audio so that participants would only be able to judge the naturalness of the motion.
- **Appropriateness**: We asked participants to rate videos of dances generated by the different models according to the question *On a scale from 1 to 5: To what extent is the character dancing to the music? I.e. how well do the movements match the audio?* where 1 is *not at all* and 5 is *very well*.
- **Within-dance diversity**: Models can exhibit many types of diversity. They can show diverse motions when changing the motion seed, the song, or at different points within the same song. Probabilistic models can furthermore generate different motions even for the same seed and song. For this study, we decided to focus on the diversity of motions within the same song, for a fixed seed, as this scenario is representative of how dance generation models may be used in practice. We thus selected two disjoint pieces of generated dance from within the same song, for each model, and presented the videos side by side. We asked participants to rate the generated motions (without audio) according to the question *On a scale from 1 to 5: How different are the two dances from each other?* where 1 is *very similar* and 5 is *very different*.

**Results**. The *fine-tuned Transflower* model was rated highest both in terms of naturalness and appropriateness, followed by the standard *Transflower* model. Figure 4 shows the mean ratings for all four models across the three experiments. A one-way ANOVA and a post-hoc Tukey multiple comparison test was performed, in order

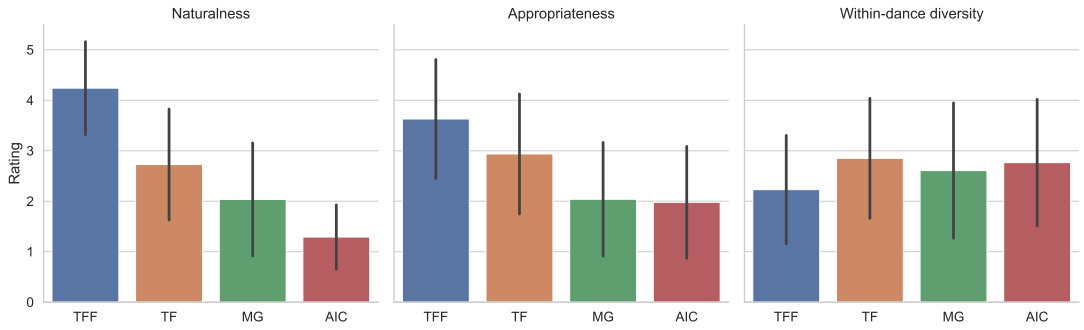



Fig. 4. Results of user study, mean ratings with standard deviation bars. From left to right: naturalness, appropriateness and diversity, for the four models in the study: Transflower fine-tuned (TFF), Transflower (TF), MoGlow (MG) and AI Choreographer (AIC). 95% confidence intervals for the mean ratings are not shown in the figure, but equate to ±0.13 or less for all models in the three experiments, which is much narrower than the plotted standard deviations.

to identify significant differences. For *naturalness*, all differences between models were significant ($p < 0.001$). For *appropriateness*, all differences except between *MoGlow* and *AI Choreographer* were significant ($p < 0.001$). For diversity, *fine-tuned Transflower* was rated less diverse than the other models ($p < 0.001$), and *Transflower* was more diverse than *MoGlow* $p = 0.02$). However, we again emphasize that only the non-fine-tuned *Transflower* model is directly comparable to the two baselines, since the latter were not fine-tuned on the higher-quality components of the data.

### 5.3 Motion prompting

Autoregressive models require an initial context seed to initiate the generation. We experimented with feeding the model different motion seeds, and observed that the seed has a significant effect on the style of the generated dance. To make this observation more quantitative, we measured the FMD between the distribution of motions that Transflower produces when seeded with a motion seed of different styles (and over the different songs in the test set) and the ground truth data for those styles. Results are shown in table 4, where we see that the seed changes the FMD distribution (and thus the style of the dance), and tends to make the style closer to the style represented by the seed, in the sense that the smallest FMD is found on the diagonal (matched seed and style) in three of five cases. This appears to be a weak version of the effect of "prompting" observed in language models [Brown et al. 2020; Reynolds and McDonell 2021]. We conjecture that this prompting effect will allow more controlability of motion and dance generation models, as we make the models more powerful, by increasing dataset and model sizes.

We also observe in table 4 how the FMD is a lot higher for some styles than others. Freestyle (with data from SM1) seems like the most challenging for the model to capture, seeing that the FMD is very high for all seeds. This is expected as this dataset is both highly diverse and has a lower motion-tracking quality. If we discount this style, the smallest FMD is on the diagonal in three out of four cases.

| | | Motion seed | | | | |
|---|---|---|---|---|---|---|
| | | FS | CA | HH | BR | GN |
| Ground truth | FS | 5615.7 | 5649.1 | 5492.8 | 5495.6 | 6054.4 |
| | CA | 352.3 | **7.4** | 192.9 | **155.2** | 242.9 |
| | HH | **92.1** | 712.0 | **187.9** | 238.4 | 1619.4 |
| | BR | 487.7 | 109.7 | 286.1 | 238.0 | 254.0 |
| | GN | 1326.3 | 340.1 | 979.8 | 881.2 | **51.3** |

Table 4. **Effect of different motion seeds on dance style**. We compare the FMD between the distribution of motions for Transflower seeded with a motion seed of different styles (and different songs), and the ground truth data for those styles. FS: freestyle, CA: casual, HH: hip hop, BR: break dance, GN: GrooveNet.

## 6 DISCUSSION

In this work, we have described the first model for dance synthesis combining powerful probabilistic modelling based on normalizing flows, and an attention-based encoder for encoding the motion and music context. We have shown that both of these properties are important, by comparing our model with two previously proposed models: MoGlow, a general motion synthesis model not previously applied to dance generation, which is based on autoregressive normalizing flows and an LSTM context encoder [Henter et al. 2020], and AI Choregographer, a deterministic dance generation model based on a cross-modal attention encoder [Li et al. 2021a].

In section 5, we found that Transflower matches the ground truth distribution of poses and movements better than MoGlow and AI Choreographer (table 2), and is also ranked higher in naturalness and appropriateness to the music by human subjects (section 5.2). We observe the same trend in the kinematic tempograms in fig. 3 which give a more objective view on how well the motion matches the music. Comparing the two baselines, we further see that MoGlow (which is probabilistic, but lacks transformers) achieved a substantially better naturalness rating than the deterministic, transformer-based AI Choreographer. We take this as evidence that a probabilistic approach was particularly important for good results in the present evaluation. In preliminary experiments, we found that AI Choreographer produced more natural motion when trained on the AIST++ data only than when trained on our full training set. In previous works where AI Choreographer tended to reach relatively high scores in naturalness, the model was only trained on a single data source at a time [Li et al. 2021b,a]. Taken together, this suggests that the high diversity and heterogeneity of our dataset significantly degrades the performance of a deterministic model, while the

probabilistic models are better able to adapt to the heterogeneity. That said, random sampling from Transflower trained on on AIST++ alone also exhibited improved motion quality. Unlike results from natural language [Henighan et al. 2020], we thus did not see any evidence of favourable scaling behaviour when trading increased dataset size for greater data diversity and potentially reduced quality, except perhaps in the diversity of generated dances. However, part of this might be due to a difference in output-generation methods, where our strongest language models [Brown et al. 2020] benefit from sampling only among the most probable outcomes of the learned distribution [Holtzman et al. 2020] (often called "reducing the temperature"), whereas our experiments sampled directly from the distribution learned by the flow without temperature reduction.

We also evaluated the models in terms of the diversity of movements found at different points in time for a single dance. In fig. 4 we see that although Transflower scored higher than MoGlow, the difference with AI Choreographer was not significant. However, considering the low score of AI Choreographer for naturalness, the high diversity of the movements of AI Choreographer may not correspond to meaningful diversity in terms of dance. We observed AI Choreographer tended to produce much more chaotic movements, and also was more likely to regress to a frozen pose (the latter stimuli were not included in the user study). Therefore we argue that Transflower achieved more meaningful diversity than the baselines.

To explore the effect of fine-tuning the model on higher quality data, we trained Transflower only on the PMSD dance data, for an extra 50k iterations. We found that resulted in significantly more natural motion, as well as motion that was judged more appropriate to the music by the participants in our user study (fig. 4), compared to without fine-tuning. However, this improvement came at the expense of reduced (within-dance) diversity in the dance, which probably explains the increased Fréchet distribution distances for the fine-tuned model (table 2).

In section 5.3, we studied the effect of the autoregressive motion seed on the generated dance. We found that the seed had a big effect on the style of the dance, with a tendency to make the dance more similar to the style from which the 6s motion seed was taken. We call this effect "motion prompting" in analogy to the effect of prompting in language models [Reynolds and McDonell 2021]. Our results suggest that this may be a new way to achieve stylistic control over autoregressive dance and motion models.

In order to drive research into better learning-based dance synthesis, that approach more human-like dance, an important piece is the availability of large dance datasets. In this work, we introduce the largest dataset of 3D dance movements obtained with motion capture technologies. This can benefit not only learning-based approaches, but data-driven dance synthesis in general (see section 2.2). We think that the growing use of VR could offer novel opportunities for research, like the one we explored here. Finally, we think that investigating the most effective ways to scale models to bigger datasets is an interesting direction for future work.

## 7 LIMITATIONS

Learning-based methods for dance synthesis have certain limitations. As discussed above, they require large amounts of data, and may produce less reliable and controllable results than approaches that impose more structure. On the upside, they are more flexible (require fewer changes to apply to other tasks), and tend to produce more natural results when enough data is available.

More specific to our model, normalizing flows show certain advantages and limitations relative to other generative modelling methods [Bond-Taylor et al. 2021]. They allow exact likelihood maximization, which leads to stable training, large diversity in samples, and good quality results for large enough data and expressive enough models. However, they are less parameter efficient and slower to train than other approaches such as generative adversarial networks or variational autoencoders, at least for images [Bond-Taylor et al. 2021]. Furthermore, the full-attention transformer encoder which we use is slower to train than purely decoder-based models like GPT, due to being less parallelizable [Vaswani et al. 2017]. We think that exploring architectures that overcome these limitations while preserving performance, is an important area for future work.

Finally, we think that further work is needed in evaluation. This includes comparing state-of-the-art learning-based methods like Transflower with current motion-graph-based methods like ChoreoMaster [Kang et al. 2021], in terms of naturalness, appropriateness, and diversity of the generated dance, as well as how these depend on the data available. Models that include physics constraints, like those discussed in section 2.1, also have shown promise in terms of naturalness of the motion, and we think should be compared with less constrained models like Transflower in future work. We note that the Transflower model we propose could be used to parametrize the policy for physics-based IL algorithms, as it allows for exact computation of the log probability over the output. Even more so, however, it encompasses work towards understanding what are the best metrics by which to evaluate dance synthesis models, in terms of reliability and relevance for downstream tasks. In particular, although we looked at beat alignment as one of our objective metrics, this is primarily due to a lack of other well-developed objective evaluation measures. Good dancing is not a beat-matching task, and previous dance synthesis works have found that the quality of the dance is a big factor in determining how humans judge how well the dance matches the music. For example, in Li et al. [2021a], ground truth dances paired with a random piece of music from the data set was ranked significantly higher than any of the models, with regards to how well it matched the music. This may be linked to a previously observed effect where people tend to find good dance moves with a diversity of nested rhythms to match almost any music piece they are played with [Avirgan 2013; Miller et al. 2013]. This diversity of nested rhythms may also go some ways toward explaining the relatively strong results of mismatched motion in table 3.

The phenomenon of rhythms at multiple levels extends to many aspects of human and animal communication [Pouw et al. 2021]. A similar effect as for mismatched dance has been observed in the related field of speech-driven gesture generation, where a recent international challenge [Kucherenko et al. 2021] found that mismatched ground-truth motion clips, unrelated to the speech, were rated as more appropriate for the speech than the best-rated synthetic system. For gestures, the disparity between matched and mismatched motion becomes more obvious to raters if the data contains periods of both speech and silence (during which no gesture

activity is expected), as observed in Yoon et al. [2019]. Bringing this back to dance, this might correspond to several songs with silence in between, or music that exhibits extended dramatic pauses. More broadly, these findings point to the quality of dance movements being a more important factor for downstream applications, but also suggest a need for further research on how to evaluate dance and dance synthesis and disentangling aspects of motion quality from rhythmic and stylistic appropriateness. For instance, one could evaluate on songs with extensive pausing, to make the difference between appropriate and less appropriate dancing more pronounced. Our current models, never having seen data on silence or standing still during training, generate dance motion even for silent audio input, mirroring results in locomotion generation, where many models find it difficult to stand still [Henter et al. 2020].

## 8 CONCLUSION

In conclusion, we introduced a new model for probabilistic autoregressive modelling of high-dimensional continuous signals, conditioned on a multimodal context, which we applied to the problem of music-conditioned dance generation. We created the currently largest 3D dance motion dataset, and used it to evaluate our model versus two representative baselines. Our results show that our model improves on previous state of the art along several benchmarks, and the two main features of our model, the probabilistic modelling of the output, and the attention-based encoding of the inputs, are both necessary to produce realistic and diverse dance that is appropriate for the music.

## ACKNOWLEDGMENTS

We thank the dancers: Mario Perez Amigo, Konata, Max and others. We thank Esther Ericsson for dance and motion capture, and lox9973 for the VR dance recording. We thank the Syrtos dance collaborators Stella Pashalidou, Michael Hagleitner, and Rainer and Ronja Polak. We are also grateful to the reviewers for their thoughtful feedback.

This work benefited from access to the HPC resources of IDRIS under the allocation 2020-[A0091011996] made by GENCI, using the Jean Zay supercomputer. This research was partially supported by the Google TRC program, the Swedish Research Council projects 2018-05409 and 2019-03694, the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, and Marianne and Marcus Wallenberg Foundation MMW 2020.0102. Guillermo Valle-Pérez benefited from funding of project "DeepCuriosity" from Région Aquitaine and Inria, France.

## REFERENCES

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

Simon Alexanderson, Gustav Eje Henter, Taras Kucherenko, and Jonas Beskow. 2020. Style-controllable speech-driven gesture synthesis using normalising flows. *Comput. Graph. Forum* 39, 2 (2020), 487–496. https://doi.org/10.1111/cgf.13946

Okan Arikan and David A. Forsyth. 2002. Interactive motion generation from examples. *ACM Trans. Graph.* 21, 3 (2002), 483–490. https://doi.org/10.1145/566570.566606

Jody Avirgan. 2013. Why Spiderman is Such a Good Dancer. https://www.wnycstudios.org/podcasts/radiolab/articles/299399-why-spiderman-such-good-dancer

Bettina Bläsing, Beatriz Calvo-Merino, Emily S Cross, Corinne Jola, Juliane Honisch, and Catherine J Stevens. 2012. Neurocognitive control in dance perception and performance. *Acta psychologica* 139, 2 (2012), 300–308.

Sebastian Böck and Markus Schedl. 2011. Enhanced beat tracking with context-aware neural networks. In *Proc. Int. Conf. Digital Audio Effects*. 135–139.

Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. 2016. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *European Conference on Computer Vision*. Springer, 561–578.

Sam Bond-Taylor, Adam Leach, Yang Long, and Chris G Willcocks. 2021. Deep Generative Modelling: A Comparative Review of VAEs, GANs, Normalizing Flows, Energy-Based and Autoregressive Models. *arXiv preprint arXiv:2103.04922* (2021).

Andrew Brock, Jeff Donahue, and Karen Simonyan. 2018. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In *International Conference on Learning Representations*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Proc. NeurIPS*, Vol. 33. 1877–1901. https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

Birgitta Burger, Marc R Thompson, Geoff Luck, Suvi Saarikallio, and Petri Toiviainen. 2013. Influences of rhythm-and timbre-related musical features on characteristics of music-induced movement. *Frontiers in Psychology* 4, Article 183 (2013), 10 pages. https://doi.org/10.3389/fpsyg.2013.00183

Judith Bütepage, Michael J. Black, Danica Kragic, and Hedvig Kjellström. 2017. Deep representation learning for human motion prediction and classification. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'17)*. IEEE Computer Society, Los Alamitos, CA, USA, 1591–1599. https://doi.org/10.1109/CVPR.2017.173

Shih-Pin Chao, Chih-Yi Chiu, Jui-Hsiang Chao, Shi-Nine Yang, and T-K Lin. 2004. Motion retrieval and its application to motion synthesis. In *24th International Conference on Distributed Computing Systems Workshops*. IEEE, 254–259.

CMU Graphics Lab. 2003. Carnegie Mellon University motion capture database. http://mocap.cs.cmu.edu/

Luka Crnkovic-Friis and Louise Crnkovic-Friis. 2016. Generative choreography using deep learning. *arXiv preprint arXiv:1605.06921* (2016).

Abe Davis and Maneesh Agrawala. 2018. Visual rhythm and beat. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2532–2535.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. 2020. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341* (2020).

Chris Donahue, Zachary C Lipton, and Julian McAuley. 2017. Dance dance convolution. In *International conference on machine learning*. PMLR, 1039–1048.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* (2020).

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

Ylva Ferstl and Rachel McDonnell. 2018. IVA: Investigating the use of recurrent motion modelling for speech gesture generation. In *IVA '18 Proceedings of the 18th International Conference on Intelligent Virtual Agents*. https://trinityspeechgesture.scss.tcd.ie

Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. 2015. Recurrent network models for human dynamics. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV'15)*. IEEE Computer Society, Los Alamitos, CA, USA, 4346–4354. https://doi.org/10.1109/ICCV.2015.494

Satoru Fukayama and Masataka Goto. 2015. Music content driven automated choreography with beat-wise motion connectivity constraints. *Proceedings of SMC* (2015), 177–183.

F. Sebastian Grassia. 1998. Practical parameterization of rotations using the exponential map. *J. Graph. Tools* 3, 3 (1998), 29–48. https://doi.org/10.1080/10867651.1998.10487493

Keith Grochow, Steven L. Martin, Aaron Hertzmann, and Zoran Popović. 2004. Style-based inverse kinematics. *ACM Trans. Graph.* 23, 3 (2004), 522–531. https://doi.org/10.1145/1015706.1015755

Ikhansul Habibie, Daniel Holden, Jonathan Schwarz, Joe Yearsley, and Taku Komura. 2017. A recurrent variational autoencoder for human motion synthesis. In *Proceedings of the British Machine Vision Conference (BMVC'17)*. BMVA Press, Durham, UK, Article 119, 12 pages. https://doi.org/10.5244/C.31.119

Ikhsanul Habibie, Weipeng Xu, Dushyant Mehta, Lingjie Liu, Hans-Peter Seidel, Gerard Pons-Moll, Mohamed Elgharib, and Christian Theobalt. 2021. Learning Speech-driven 3D Conversational Gestures from Video. *arXiv preprint arXiv:2102.06837* (2021).

Mari Romarheim Haugen. 2014. Studying rhythmical structures in Norwegian folk music and dance using motion capture technology: A case study of Norwegian telespringar. *Musikk og Tradisjon* 28 (2014), 27–52.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. 2020. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701* (2020).

Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. 2020. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–14.

Jonathan Ho, Xi Chen, Aravind Srinivas, Yan Duan, and Pieter Abbeel. 2019. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In *International Conference on Machine Learning*. PMLR, 2722–2730.

Daniel Holden, Oussama Kanoun, Maksym Perepichka, and Tiberiu Popa. 2020. Learned motion matching. *ACM Trans. Graph.* 39, 4 (2020), 53–1.

Daniel Holden, Taku Komura, and Jun Saito. 2017. Phase-functioned neural networks for character control. *ACM Trans. Graph.* 36, 4, Article 42 (2017), 13 pages. https://doi.org/10.1145/3072959.3073663

Daniel Holden, Jun Saito, and Taku Komura. 2016. A deep learning framework for character motion synthesis and editing. *ACM Trans. Graph.* 35, 4, Article 138 (2016), 11 pages. https://doi.org/10.1145/2897824.2925975

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of Neural Text Degeneration. In *International Conference on Learning Representations*.

Andre Holzapfel, Michael Hagleitner, and Stella Pashalidou. 2020. Diversity of Traditional Dance Expression in Crete: Data Collection, Research Questions, and Method Development. In *Proceedings of the ICTM Study Group on Sound, Movement, and the Sciences Symposium*.

Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Ian Simon, Curtis Hawthorne, Noam Shazeer, Andrew M Dai, Matthew D Hoffman, Monica Dinculescu, and Douglas Eck. 2018. Music Transformer: Generating Music with Long-Term Structure. In *International Conference on Learning Representations*.

Chen Kang, Zhipeng Tan, Jin Lei, Song-Hai Zhang, Yuan-Chen Guo, Weidong Zhang, and Shi-Min Hu. 2021. ChoreoMaster: Choreography-oriented Music-driven Dance Synthesis. (2021). https://www.youtube.com/watch?v=V8MlYa_yhF0 accepted for publication at SIGGRAPH 2021.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361* (2020).

Tero Karras, Samuli Laine, and Timo Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4401–4410.

Diederik P. Kingma and Prafulla Dhariwal. 2018. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems (NeurIPS'18)*. Curran Associates, Inc., Red Hook, NY, USA, 10236–10245. http://papers.nips.cc/paper/8224-glow-generative-flow-with-invertible-1x1-con

Lucas Kovar and Michael Gleicher. 2004. Automated extraction and parameterization of motions in large data sets. *ACM Trans. Graph.* 23, 3 (2004), 559–568. https://doi.org/10.1145/1015706.1015760

Lucas Kovar, Michael Gleicher, and Frédéric Pighin. 2002. Motion graphs. *ACM Trans. Graph.* 21, 3 (2002), 473–482. https://doi.org/10.1145/566654.566605

Florian Krebs, Sebastian Böck, and Gerhard Widmer. 2015. An Efficient State-Space Model for Joint Tempo and Meter Tracking.. In *ISMIR*. 72–78.

Taras Kucherenko, Patrik Jonell, Youngwoo Yoon, Pieter Wolfert, and Gustav Eje Henter. 2021. A Large, Crowdsourced Evaluation of Gesture Generation Systems on Common Data: The GENEA Challenge 2020. In *26th International Conference on Intelligent User Interfaces* (College Station, TX, USA) *(IUI '21)*. ACM, New York, NY, USA, 11–21. https://doi.org/10.1145/3397481.3450692

OxAI Labs. 2019. DeepSaber. https://github.com/oxai/deepsaber/.

Kimerer LaMothe. 2019. The dancing species: how moving together in time helps make us human. *Aeon* (June 2019). https://aeon.co/ideas/the-dancing-species-how-moving-together-in-time-helps-make-us-human

Ben Lang. 2021. The Future is Now: Live Breakdance Battles in VR Are Connecting People Across the Globe. https://www.roadtovr.com/vr-dance-battle-vrchat-breakdance/

Gilwoo Lee, Zhiwei Deng, Shugao Ma, Takaaki Shiratori, Siddhartha S Srinivasa, and Yaser Sheikh. 2019a. Talking with hands 16.2 m: A large-scale dataset of synchronized body-finger motion and audio for conversational motion analysis and synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 763–772.

Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. 2019b. Dancing to music. *arXiv preprint arXiv:1911.02001* (2019).

Jehee Lee, Jinxiang Chai, Paul S. A. Reitsma, Jessica K. Hodgins, and Nancy S. Pollard. 2002. Interactive control of avatars animated with human motion data. *ACM Trans. Graph.* 21, 3 (2002), 491–500. https://doi.org/10.1145/566654.566607

Sergey Levine, Jack M. Wang, Alexis Haraux, Zoran Popović, and Vladlen Koltun. 2012. Continuous character control with low-dimensional embeddings. *ACM Trans. Graph.* 31, 4, Article 28 (2012), 10 pages. https://doi.org/10.1145/2185520.2185524

Buyu Li, Yongchi Zhao, and Lu Sheng. 2021b. DanceNet3D: Music Based Dance Generation with Parametric Motion Transformer. *arXiv preprint arXiv:2103.10206* (2021).

Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. 2020. Learning to Generate Diverse Dance Motions with Transformer. *arXiv preprint arXiv:2008.08171* (2020).

Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. 2021a. Learn to Dance with AIST++: Music Conditioned 3D Dance Generation. *arXiv preprint arXiv:2101.08779* (2021).

Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel van de Panne. 2020. Character controllers using motion VAEs. *ACM Trans. Graph.* 39, 4, Article 40 (2020), 12 pages. https://doi.org/10.1145/3386569.3392422

lox9973. 2021. ShaderMotion. https://gitlab.com/lox9973/ShaderMotion.

Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. 2019. AMASS: Archive of Motion Capture as Surface Shapes. In *International Conference on Computer Vision*. 5442–5451.

Christian Mandery, Ömer Terlemez, Martin Do, Nikolaus Vahrenkamp, and Tamim Asfour. 2015. The KIT whole-body human motion database. In *2015 International Conference on Advanced Robotics (ICAR)*. IEEE, 329–336.

Alexander Mathis, Steffen Schneider, Jessy Lauer, and Mackenzie Weygandt Mathis. 2020. A primer on motion capture with deep learning: principles, pitfalls, and perspectives. *Neuron* 108, 1 (2020), 44–65.

Josh Merel, Yuval Tassa, Sriram Srinivasan, Jay Lemmon, Ziyu Wang, Greg Wayne, and Nicolas Heess. 2017. Learning human behaviors from motion capture by adversarial imitation. *arXiv preprint arXiv:1707.02201* (2017).

Jared E Miller, Laura A Carlson, and J Devin McAuley. 2013. When what you hear influences when you see: listening to an auditory rhythm influences the temporal allocation of visual attention. *Psychological science* 24, 1 (2013), 11–18.

Olof Misgeld, Andre Holzapfel, and Sven Ahlbäck. 2019. Dancing Dots – Investigating the Link between Dancer and Musician in Swedish Folk Dance. In *Sound & Music Computing Conference*.

Luiz Naveda and Marc Leman. 2011. Hypotheses on the choreographic roots of the musical meter: a case study on Afro-Brazilian dance and music. *Debates actuales en evolución, desarrollo y cognición e implicancias socio-culturales* (2011), 477–495.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. 2021. Normalizing Flows for Probabilistic Modeling and Inference. *Journal of Machine Learning Research* 22, 57 (2021), 1–64. http://jmlr.org/papers/v22/19-1028.html

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. 2019. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2337–2346.

Dario Pavllo, David Grangier, and Michael Auli. 2018. QuaterNet: A quaternion-based recurrent model for human motion. In *Proceedings of the British Machine Vision Conference (BMVC'18)*. BMVA Press, Durham, UK, 14 pages. http://www.bmva.org/bmvc/2018/contents/papers/0675.pdf

Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. 2018a. DeepMimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Trans. Graph.* 37, 4 (2018), 1–14.

Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. 2018b. Sfv: Reinforcement learning of physical skills from videos. *ACM Transactions On Graphics (TOG)* 37, 6 (2018), 1–14.

Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. 2021. AMP: Adversarial Motion Priors for Stylized Physics-Based Character Control. *ACM Trans. Graph.* 40, 4, Article 1 (July 2021), 15 pages. https://doi.org/10.1145/3450626.3459670

Mathis Petrovich, Michael J Black, and Gül Varol. 2021. Action-Conditioned 3D Human Motion Synthesis with Transformer VAE. *arXiv preprint arXiv:2104.05670* (2021).

Wim Pouw, Shannon Proksch, Linda Drijvers, Marco Gamba, Judith Holler, Christopher Kello, Rebecca S. Schaefer, and Geraint A. Wiggins. 2021. Multilevel rhythms in multimodal communication. *Philosophical Transactions of the Royal Society B* 376, 1835 (2021), 20200334. https://doi.org/10.1098/rstb.2020.0334

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. 2019. WaveGlow: A flow-based generative network for speech synthesis. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'19)*. IEEE Signal Processing Society, Piscataway, NJ, USA, 3617–3621. https://doi.org/10.1109/ICASSP.2019.8683143

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683* (2019).

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092* (2021).

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. *arXiv preprint arXiv:2102.07350* (2021).

Yu Rong, Takaaki Shiratori, and Hanbyul Joo. 2021. FrankMocap: A Monocular 3D Whole-Body Pose Estimation System via Regression and Integration. In *IEEE International Conference on Computer Vision Workshops*.

Alla Safonova and Jessica K. Hodgins. 2007. Construction and Optimal Search of Interpolated Motion Graphs. *ACM Trans. Graph.* 26, 3 (July 2007), 106–es. https://doi.org/10.1145/1276377.1276510

Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu. 2018. Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'18)*. IEEE Signal Processing Society, Piscataway, NJ, USA, 4799–4783. https://doi.org/10.1109/ICASSP.2018.8461368

Sebastian Starke, Yiwei Zhao, Taku Komura, and Kazi Zaman. 2020. Local motion phases for learning multi-contact character movements. *ACM Trans. Graph.* 39, 4, Article 54 (2020), 14 pages. https://doi.org/10.1145/3386569.3392450

Statista. 2020. Augmented reality (AR) and virtual reality (VR) headset shipments worldwide from 2020 to 2025. https://www.statista.com/statistics/653390/worldwide-virtual-and-augmented-reality-headset-shipments/

Wataru Takano, Katsu Yamane, and Yoshihiro Nakamura. 2010. Retrieval and Generation of Human Motions Based on Associative Model between Motion Symbols and Motion Labels. *Proceedings of Journal of the Robotics Society of Japan* 28, 6 (2010), 723–734.

Taoran Tang, Jia Jia, and Hanyang Mao. 2018. Dance with melody: An LSTM-autoencoder approach to music-oriented dance synthesis. In *Proceedings of the 26th ACM International Conference on Multimedia*. 1598–1606.

Petri Toiviainen, Geoff Luck, and Marc R Thompson. 2010. Embodied meter: hierarchical eigenmodes in music-induced movement. *Music Perception* 28, 1 (2010), 59–70.

Nikolaus F Troje. 2002. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision* 2, 5 (2002), 2–2.

Shuhei Tsuchida, Satoru Fukayama, Masahiro Hamasaki, and Masataka Goto. 2019. 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*. Delft, Netherlands, 501–510.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS'17)*. Curran Associates, Inc., Red Hook, NY, USA, 5998–6008. https://papers.nips.cc/paper/7181-attention-is-all-you-need

Jack M. Wang, David J. Fleet, and Aaron Hertzmann. 2008. Gaussian process dynamical models for human motion. *IEEE T. Pattern Anal.* 30, 2 (2008), 283–298. https://doi.org/10.1109/TPAMI.2007.1167

Ulme Wennberg and Gustav Eje Henter. 2021. The Case for Translation-Invariant Self-Attention in Transformer-Based Language Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers) (ACL '21)*. ACL, 130–140. https://doi.org/10.18653/v1/2021.acl-short.18

Chenfei Wu, Lun Huang, Qianxi Zhang, Binyang Li, Lei Ji, Fan Yang, Guillermo Sapiro, and Nan Duan. 2021. GODIVA: Generating Open-DomaIn Videos from nAtural Descriptions. *arXiv preprint arXiv:2104.14806* (2021).

Zijie Ye, Haozhe Wu, Jia Jia, Yaohua Bu, Wei Chen, Fanbo Meng, and Yanfeng Wang. 2020. ChoreoNet: Towards Music to Dance Synthesis with Choreographic Action Unit. In *Proceedings of the 28th ACM International Conference on Multimedia*. 744–752.

Youngwoo Yoon, Woo-Ri Ko, Minsu Jang, Jaeyeon Lee, Jaehong Kim, and Geehyuk Lee. 2019. Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'19)*. IEEE Robotics and Automation Society, Piscataway, NJ, USA, 4303–4309. https://doi.org/10.1109/ICRA.2019.8793720

Yi Zhou, Zimo Li, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. 2018. Auto-conditioned recurrent networks for extended complex human motion synthesis. In *Proceedings of the International Conference on Learning Representations (ICLR'18)*. 13 pages. https://openreview.net/forum?id=r11Q2SlRW

Wenlin Zhuang, Congyi Wang, Siyu Xia, Jinxiang Chai, and Yangang Wang. 2020. Music2Dance: DanceNet for Music-driven Dance Generation. *arXiv e-prints* (2020), arXiv–2002.

## A MODEL DETAILS

We give the main architecture hyperparameter details in table 5. The hyperparameters for the AI Choreographer architecture were chosen to match those in the original AI Choreographer [Li et al. 2021a], with the only difference that we used T5-style relative positional embeddings, and that we used positional embeddings in the cross-modal transformer, as we found these gave slightly better convergence. The hyperparameters for the transformer encoders and cross-modal transformer in Transflower are identical to those of AI Choreographer, while the normalizing flow parameters are the same as in MoGlow, except for the affine coupling layers for which we use 2-layer transformers. The hyperparameters for MoGlow were chosen to be similar to the original implementation in Henter et al. [2020], except that we increased the concatenated context length fed at each time step, from 10 frames in the original MoGlow, to 40 frames of motion and 50 frames of music (40 in the past, 10 in the future). This makes the concatenated input to the LSTM have a dimension of $85 \times 50 + 67 \times 40 = 6930$ which accounts for the significant parameter increase of the model. We did this change because we found that an increased concatenated context length for MoGlow was necessary for it to produce good results.

| Model | #Params | $L_{mo}$ | $L_{mu}$ | $L_{cm}$ | $d_{model}$ | $K$ | $L_{ac}$ | $L_{lstm}$ | $k_x$ | $k_m$ | $l_m$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| AI Choreographer | 64M | 2 | 2 | 12 | 800 | N/A | N/A | N/A | 120 | 120 | 20 |
| MoGlow | 281M | N/A | N/A | N/A | N/A | 16 | N/A | 2 | 120 | 120 | 20 |
| Transflower | 122M | 2 | 2 | 12 | 800 | 16 | 2 | N/A | 120 | 120 | 20 |

Table 5. **Basic architecture hyperparameters for the different models**. $L_{mo}$, $L_{mu}$, $L_{cm}$, $L_{ac}$, $L_{lstm}$ are the number of layers in the motion encoder transformer, the music encoder transformer, the cross-modal transformer, the affine coupling layer, and the LSTM respectively. $K$ is the number of blocks in the normalizing flow, and $d_{model}$ is the latent dimension of the encoder transformers.