2023 IEEE Symposium Series on Computational Intelligence (SSCI)
Mexico City, Mexico. December 5-8, 2023

Adversarial Attention for Human Motion Synthesis

Matthew Malek-Podjaski
Computing Science
University of Glasgow
UK
matthewmalekp@gmail.com

Fani Deligianni
Computing Science
University of Glasgow
UK
fani.deligianni@glasgow.ac.uk

***Abstract*—Analysing human motions is a core topic of interest for many disciplines, from Human-Computer Interaction, to entertainment, Virtual Reality and healthcare. Deep learning has achieved impressive results in capturing human pose in real-time. Acquiring human motion datasets is highly time-consuming, challenging, and expensive. Hence, human motion synthesis is a crucial research problem within deep learning and computer vision. We present a novel method for controllable human motion synthesis by applying attention-based probabilistic deep adversarial models with end-to-end training. We show that we can generate synthetic human motion over both short- and long-time horizons through the use of adversarial attention.**

***Index Terms*—human motion synthesis, attention, classification, adversarial**

I. Introduction

Synthesising human motions is an active research problem with many cross-disciplinary applications. Popular research areas often focus on generating human motions for Human-Computer Interaction (HCI) and entertainment applications. These vary from generating gesture motions from speech [1] to creating realistic virtual avatars [2] to animating video game characters in real-time from ambiguous control signals sent from a video game controller [3]–[5]. Synthetic motions can also be used for modelling realistic human-to-human interactions to improve the realism of Virtual Reality (VR) applications [6]. Furthermore, human motions are highly sought after in more niche applications such as simulating crowd movements [7] or predicting pedestrian motions [8], which are necessary for urban planning, traffic engineering, and self-driving vehicles.

Previous techniques have applied deterministic methods, such as recurrent Long Short-Term Memory (LSTM) models [9], [10], transformers [11], phased-functioned models [3], and Mixture-of-experts models [5]. However, deterministic recurrent models often suffer from averaging poses, where the network's output eventually collapses into a mean pose. Typically, LSTM models incorporate a non-linear encoder, followed by recurrent layers (LSTM) and a non-linear decoder [12]. Therefore, this approach fails in long-term human motion prediction. On the other hand, transformers have been introduced as a way to improve recurrent neural networks by enabling attention mechanisms that are able to capture long-range dependencies [11]. Nevertheless, they require large datasets to overcome inductive biases and their ability to efficiently encode local temporal information is limited. On the other hand, phase-functioned networks can generate both short- and long-term motions without error accumulation. However, they require manually designed phase functions that make assumptions about the nature of the motion making them difficult to apply in generalised scenarios.

We propose a novel probabilistic deep adversarial architecture for learning both short- and long-term motion synthesis. We use an Attention-based Wasserstein Generative Adversarial Network with Gradient Penalty that we call the Attention WGAN-GP. Our model takes into advantage convolutional GANs and self-attention to efficiently model both local and long-range spatiotemporal dependencies and thus the underlying model is able to learn with limited human motion data. Furthermore, our model makes no assumptions about the underlying motion data or the control signals, which we show by training our model to generate various action motions. Through the use of attention and autoregression, we also show that our model can continuously generate realistic motions over both long- and short-time horizons. Our approach outperforms commonly used LSTM models in human motion synthesis that are based on an Encoder-Recurrent Layers-Decoder architecture [12].

II. Methods

To generate synthetic motions, we propose a novel online generative adversarial probabilistic model. Specifically, we exploit a Wasserstein Generative Adversarial Network (WGAN) [13] with Gradient Penalty (WGAN-GP) [14] and Self-Attention [15] modules for improved synthetic motion accuracy, in combination with a classifier network [16] for controlling the motion generation. We refer the model as the Attention WGAN-GP.

Many adversarial motion synthesis models generate data from random noise vectors. However, the generation process needs to be controllable for synthetic motions to supplement existing datasets. For controllable adversarial motion generation, we need to turn to models such as Conditional GANs [17] or ControlGANs [16]. A Conditional GAN conditions the disciminator on the real or fake label alongside a class label. Here we adapted a ControlGAN approach, which separates the prediction of each label into individual networks.

The authors would like to acknowledge funding from The Royal Society (RGS/R2/212199) and EPSRC (EP/W01212X/1).

978-0-7381-4408-5/23/$31.00 ©2023 IEEE

A. Self-Attention

Attention mechanisms are applied in GANs through a self-attention module [15] to guide the model to relate distant data sections, such as distant poses in a motion. Although attention has been recently explored within motion synthesis applications [18], [19], to our knowledge, it has not yet been used within an adversarial scenario. Since the task of generating motions inherently relies on the spatio-temporal relation of past and future motions, we can leverage attention mechanisms to reduce error accumulation in autoregressive models [19] for more accurate short- and long-term motion prediction.

Self-attention works by extending the convolutional layer of the GAN with an additional term, the attention map, which is added to the output. The attention map acts as a mask that determines the contribution that a particular section of the data sample has on the generation of another section. Initially, the model will explore only the local area of each section as it performs the convolution, just as a typical convolutional GAN. However, over time, it can learn how different output regions relate to each other and condition the generated output on related areas. Self-attention is applied to both the generator and critic networks, allowing both to learn distant spatial relationships in the data, resulting in more realistic motion synthesis.

According to the original definition for the Self-Attention GAN module [15], the features from the previous layer $\boldsymbol{x} \in \mathbb{R}^{C \times N}$ (C being the number of channels and N the number of feature locations) are transformed into two feature spaces $\boldsymbol{f}, \boldsymbol{g}$ with corresponding weight matrices W to calculate the attention, where $f(x) = W_f x, g(x) = \boldsymbol{W_g x}$

$$\beta_{j,i} = \frac{\exp\left(s_{ij}\right)}{\sum_{i=1}^{N} \exp\left(s_{ij}\right)}, \text{ where } s_{ij} = \boldsymbol{f}\left(\boldsymbol{x_i}\right)^T \boldsymbol{g}\left(\boldsymbol{x_j}\right) \tag{1}$$

The attention map $\beta_{j,i}$ creates a mapping between the i^{th} and j^{th} regions, which is the amount the model attributes to the i^{th} region, when generating j^{th} region. Thus, the output of the attention layer is $\boldsymbol{o} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o_j}, \ldots, \boldsymbol{o}_N) \in \mathbb{R}^{C \times N}$, where $\boldsymbol{o_j}$ is defined as follows:

$$\boldsymbol{o_j} = \boldsymbol{v}\left(\sum_{i=1}^{N} \beta_{j,i} \boldsymbol{h}\left(\boldsymbol{x_i}\right)\right), \boldsymbol{h}\left(\boldsymbol{x_i}\right) = \boldsymbol{W_h x_i}, \boldsymbol{v}\left(\boldsymbol{x_i}\right) = \boldsymbol{W_v x_i} \tag{2}$$

and $\boldsymbol{W}_g \in \mathbb{R}^{\bar{C} \times C}, \boldsymbol{W}_f \in \mathbb{R}^{\bar{C} \times C}$, $\boldsymbol{W_h} \in \mathbb{R}^{\bar{C} \times C}$, $\boldsymbol{W_v} \in \mathbb{R}^{C \times \bar{C}}$ are the weight matrices learned through 1x1 convolutions as seen in Figure **??**. Finally, the attention map is then added back onto the input feature map multiplied by the scalar parameter γ, giving the final output of the self-attention module $\boldsymbol{y_i}$.

$$\boldsymbol{y}_i = \gamma \boldsymbol{o}_i + \boldsymbol{x}_i \tag{3}$$

The scalar parameter γ is initialised to 0 and is used to control the impact of the attention map over time. It allows the mode to begin learning through the exploration of features within the local space, just as a traditional convolutional GAN, and gradually over time learn to give more weight to the non-local features.

B. Autoregression

A core component of our training architecture is the iterative autoregressive training of the generator to enable it to perform long-term motion generation. The generative model was trained to generate motion data for one second time intervals. In contrast, for the motion classification task, we need full-length motions that can be used to supplement our existing motion dataset (detailed in section III-A). Furthermore, since the motions in our dataset are 3-4 seconds long, the model needs to accurately predict more than just the next second of motion without compromising its online ability to generalise to varying lengths of motion. Hence, we introduce the multiple critic iterations into the training algorithm.

C. Model Formulation

Given a motion M^{2T}, of temporal length $2T$, we can define the motion as a sequence of poses $M^{2T} = m_{0:2T} = [m_0, ..., m_{2T}]$, where each pose $m_t \in \mathbb{R}^{3J}$, is a skeleton with J being the number of joints, each specified by $(x_c\hat{i}, y_c\hat{j}, z_c\hat{k})$ cartesian coordinates, and can be defined as a probability distribution conditioned on the τ previous poses:

$$p(m_{0:2T}) = \prod_{t=\tau}^{2T} p(m_t | m_{t-\tau:t-1}) \tag{4}$$

To make the sampling of poses controllable we also condition the pose probability distribution on an additional parameter Y that acts as a control signal to confine to set of possible poses at each time frame $y_t \in \mathbb{R}^Y$, similar to other probabilistic motion models [20]. This gives us a probability distribution for each pose conditioned on the set of previous poses and a control vector y:

$$p(m|y) = \prod_{t=\tau}^{2T} p(m_t | m_{t-\tau:t-1}, y_{t-\tau:t}) \tag{5}$$

Splitting the motions into two sequences of length T we can define a set of prior motions P (also referred to as seed motions) and a set of future poses F, where $x = m_{0:T-1}$, $z = m_{T:2T}$ and $(x \in P)$, $(z \in F)$.

Our aim to create an architecture where we can model the underlying probability distribution that maps a sequence of prior poses to a sequence of future poses. Towards this end a generator G is trained to generate synthetic sequences of future poses from sequences of prior poses, $\tilde{z} = G(x)$.

By recursively feeding the generated poses back into the generator network we can continuously generate motions autoregressively, given we have an initial seed motion.

D. Loss Functions

To train our model, we use five different objective functions. A critic loss function based on the gradient penalty Wasserstein critic loss [14] (L_{critic}), a cross-entropy classifier loss (L_{class}), and a combination of three loss functions for the generator (L_{gen}). L_{gen} consists of a generator Wasserstein loss [13], a skeleton loss to constrain the size of the skeleton (L_{skel}) and a blending loss (L_{blend}) to improve continuity between prior and future motions.

1) Critic Gradient Penalty Loss: The critic uses the improved gradient penalty Wasserstein loss [21]. The Wasserstein loss requires a Lipschitz constraint on the critic, and the original loss achieves this by applying weight clipping on the critic [13]. We found in our training that enforcing a compact space on the weights of the critic network results in a generator network that is unable to output a human pose. Applying a penalty on the gradient norm rather than clipping the weights solves this issue.

Given a real prior motion x, a corresponding real future motion z, and a generated future motion $\tilde{\boldsymbol{z}} = G(\boldsymbol{x})$. We can define a real motion as $r = (x, z)$, and a fake motion as $f = (x, \tilde{z})$. We define the critic loss as:

$$L_{critic} = \mathbb{E}[D(\boldsymbol{r})] - \mathbb{E}[D(\boldsymbol{f})] + \lambda L_{gp}. \tag{6}$$

To calculate the gradient penalty we need to interpolate between the real and fake motions. As such, with random number sampled from a uniform distribution $\epsilon \sim U[0, 1]$, the interpolated motion is defined as $\hat{\boldsymbol{m}} = \epsilon \boldsymbol{r} + (1 - \epsilon)\tilde{\boldsymbol{f}}$. Using the interpolated motion, the gradient penalty is given by:

$$L_{gp} = \mathbb{E}\left[\left(\|\nabla_{\hat{\boldsymbol{m}}} D(\hat{\boldsymbol{m}})\|_2 - 1\right)^2\right]. \tag{7}$$

2) Generator Loss: For the generator loss we make use of the original generator Wasserstein loss function [13], which is the negative of the critic output for the fake generated motions.

$$L_{gen} = -\mathbb{E}[D(\boldsymbol{f}))] + L_{skel} + L_{blend} + L_{class} \tag{8}$$

We also extend the generator loss with a skeleton loss, a blend loss and a classification loss.

3) Skeleton Loss: A skeleton loss enforces a physics constraint on the shape of the skeleton to avoid it changing shape during the generated motion. Without constraining the size of the skeleton, the bone lengths in the skeleton would often vary throughout the motion, something that is not physically possible [10], [22], [23]. To constrain the bone-length of the skeleton and keep it consistent between prior and future motions, we take the first frame of the prior motion and use its pose as a reference skeleton. Then the skeleton loss is defined as the squared distance between the reference skeleton and the current time frame for a set of joint pairs that define the skeletal bones.

Given a set of joint pair definitions S that describe a given skeleton. A bone is defined as $s = \{i, j\}$, $(s \in S)$, where i and j are joints defined by (x, y, z), then $m_t^s = m_t^j - m_t^i$ describes a bone vector of pose m at time t. Using the first pose of the prior motion x_0 as a reference skeleton, we define the skeleton loss of the generated future motion $\hat{z}$ as:

$$L_{skel} = \frac{1}{T} \sum_{\mathrm{s} \in S} \sum_{t=0}^{T} \|$$

$\boldsymbol{x}_0^s - \hat{\boldsymbol{z}}_t^s$(9)

4) Blending Loss: Another issue observed with only using the Wasserstein loss functions is that the resulting future motions would not continue from past seed motions. As a result, when combining past and future motions, the person would appear to teleport mid-motion. Furthermore, this discontinuity would also results in positional error accumulation in long-term motion generation. Although combining the past and future frames before passing them to the critic alleviated some of these problems, it did not solve them as we had anticipated. We hoped that by giving the critic information from both the past and future motions, it would be able to discern fake motions based on the lack of continuity; however, in practice, this did not seem to be the case.

We found that it was necessary to enforce an additional constraint on the generation to ensure continuity in the motions, the so called blending loss. We calculate the loss to be the distance between the last input motion frame and the resulting first generated motion frame to ensure the generated motions blend with the input motion. This constraint results in synthetic motion that correctly preserve the continuity between past and future motions.

Given a prior motion ($x \in P^T$) and a generated future motion ($\hat{z} \in F^T$), of length T, we define the blend loss as the mean square distance between the last prior pose and the first future pose:

$$L_{blend} = \mathbb{E}\left[\left(\|\boldsymbol{x}_T - \hat{\boldsymbol{z}}_0\|\right)^2\right] \tag{10}$$

5) Classification Loss: For the classifier in our model we use a cross-entropy loss function. We encode the control input as a one-hot encoded vector and use it as the target label for the classifier network.

$$L_{class} = -\frac{1}{m} \sum_{i=1}^{m} y_i \cdot \log\left(\hat{y}_i\right) \tag{11}$$

The classifier is only ever updated based on the real data during its update. However, during the generator update the classifier weights are frozen and the classification loss is calculated on the fake generated motion. The classification loss is used as part of the generator loss to encourage the generator to produce motions that respect the control signal.

E. Training

Much like a typical WGAN [13], the training of our model is split into multiple stages. Furthermore, we introduce a classifier update stage in addition to the critic and generator update stages. Firstly for the critic update, the generator is given a 'fake' seed motion from which it generates a fake future motion. Similarly, a 'real' seed motion and the ground truth future motions are concatenated to generate a real motion. A seed motion in this scenario results from the concatenation of a prior motion with the corresponding control vector. The critic loss is then calculated based on how well it scored motions. Furthermore, the critic scores motions randomly interpolated between the real and fake data. The score of which is then used to calculate the gradient penalty (L_{gp}), thus giving us the critic loss from Equation 6.

The generator loss (Eq. 8), the skeleton (Eq. 9), blending (Eq. 10), and classification (Eq. 11) losses are calculated on the generated future motion, and this process is repeated for $n_{generator}$ iterations. It is important to note that although all the models are involved in calculating the generator loss, the model weights are only ever updated in their respective update steps. In every other step, the weights of the networks are frozen.

III. Evaluation Methodology and Results

A. Dataset

To train and test our models, we use a publicly available motion capture library [24]. The dataset contains a set of motion captured movements of 27 non-professional subjects (13 male, 14 female, mean age 22, ranging from 17 to 29 years) performing various motions such as walking or throwing, labelled based on their identity, gender and emotion. The motions were captured using retroreflective markers and a state-of-the-art motion capture system.

B. Preprocessing

For preprocessing the data, we centre-mean unit-variance normalise the dataset and remove the effects of global displacement and rotation from the data, similarly to our previous work [25]. This improves the convergence of the model during training. Since the original motion data involves the subjects moving around a room as they perform the actions, the effects of global displacement way out-weight any motion effects. Especially after centre-mean and unit-variance normalisation, the magnitude of the joint movements caused by the action being performed ends up far out-weighed by the extent of the global displacement in the (x,y,z) axes. This makes it challenging for classification models to predict the required motion labels. Therefore, we found much better convergence when training the models by normalising the global displacement.

C. Results

We train our model to generate various types of motions by using the action labels as control inputs. For the evaluation, we want to determine the quality of the generated motions. We compare a baseline LSTM model with our Attention WGAN-GP model, using the same input and output data. The baseline LSTM model is inspired by previous work [26] and it consists from just one single layer to avoid overfitting. It is trained by directly minimising the mean square error (MSE) between the real future motions and the generated fake future motions. For each motion in the training dataset we generate a corresponding synthetic motion, thus doubling the training data. We present both qualitative results of continuous gait motion as well as quantitative results based on the synthetic motions generated.

Qualitative results are shown in Figure 1 with sequences of generated poses over 10 and 30 second intervals, respectively. We evaluate the contribution of each feature of the Attention WGAN-GP model by quantifying the quality of the synthetic motions, comparing them to the ground truth data within angular space. We evaluate motions generated by the WGAN-GP model without attention but with both the skeleton loss (L_{skel}) and the blending loss (L_{blend}). We also evaluate WGAN-GP models with attention but without the skeleton and blending losses. Finally, we evaluate the quality of the generated motions of the Attention WGAN-GP model against a baseline LSTM model. The results show clearly that only the proposed Attention WGAN-GP model is capable of realistic synthetic motion generation over long time horizons. We also note how the quality of the motions degrades over time as different components such as attention or the blend/skeleton losses are removed from the model.

These results are also supported by estimating the angle-space representation, which is a common approach in evaluating synthetic motions [27], [28]. In this case, motions are expressed in a scale and rotation invariant notation by calculating joint angles. This allows a rough quantitative comparison of different approaches with relation to the ground truth. Figure 2 shows the mean angle in the motion data over time for just over 2 seconds of generated motion (motions are recorded at 30 frames per second). We see a significant improvement in the accuracy of the generated motions over time with the addition of attention to the WGAN-GP model. We observe that although the LSTM starts very close to the ground truth data, it quickly collapses into a mean pose, and the mean angles decrease over time.

Similarly to Figure 1, the mean angular estimation in Figure 2 shows the impact of the blend loss (L_{blend}) and the skeleton loss $L_{skeleton}$ in the Attention WGAN-GP model. We observe that with the skeleton loss and without the blend loss, the motion begins close to the ground truth, but it quickly diverges. We have observed that without the blend loss the model does not respect the continuity between the prior and future motions. For example, this may result in the model producing motions that are out of sync by a few frames, and at other times it may jump from walking forward with the left leg to walking forward with the right leg instead. Hence over time, the motions generated from the model without the blend loss quickly deviate from the ground truth motions. On the other hand, without the skeleton loss, the model does not have the intrinsic correlation between different joints moving together, and as a result, the accuracy of the generated motions dramatically suffers.

IV. Discussion and Conclusions

We have presented a novel autoregressive probabilistic and adversarial deep learning model based on end-to-end-training capable of both short- and long-term motion predic-

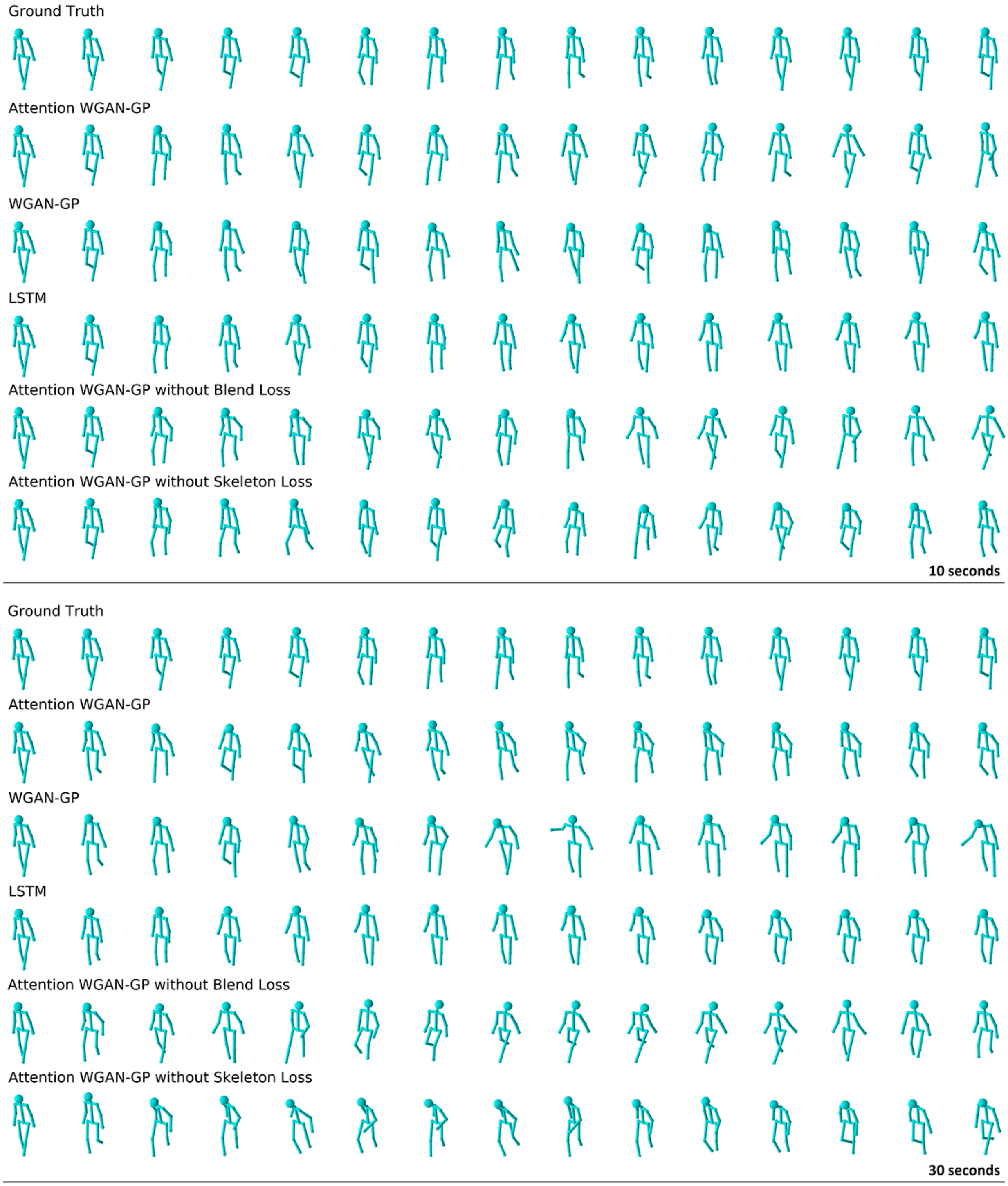
Ground Truth
Attention WGAN-GP
WGAN-GP
LSTM
Attention WGAN-GP without Blend Loss
Attention WGAN-GP without Skeleton Loss
10 seconds
Ground Truth
Attention WGAN-GP
WGAN-GP
LSTM
Attention WGAN-GP without Blend Loss
Attention WGAN-GP without Skeleton Loss
30 seconds

Fig. 1. A sequence of poses from a 10 and 30 seconds generated motion, respectively. Ground truth motion is compared tothe proposed architecture (Attention WGAN-GP) as well as WGAN-GP and LSTM. Furthermore, we demonstrate the effectiveness of incorporating a blend loss and a skeleton loss. The Attention WGAN-GP model is the only model capable of generating realistic motions over long time sequences.

tion through the use of attention. We have shown that this model outperforms existing recurrent LSTM commonly used models in the task of human motion synthesis, especially in generating plausible motions over long time horizons.

This model can be also extended to incorporate a multi-modal control input. Although we use the control input reasonably simply for conditioning the motion class, we adopted an architecture design to allow incorporating more complex

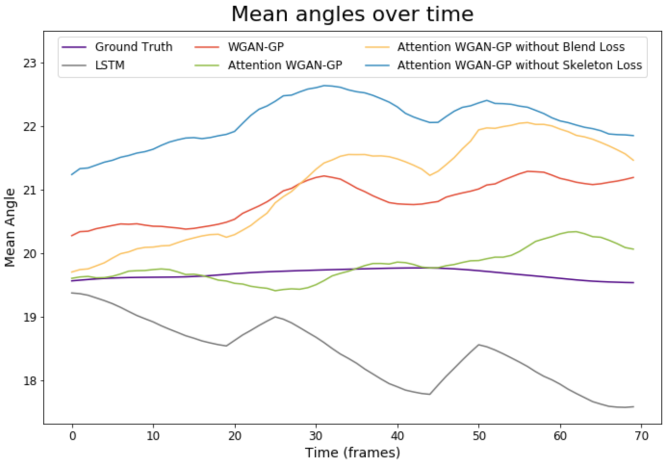

Fig. 2. Mean angle error over the generation of 70 frames (2.33 seconds) of action motions. The motions generated through our proposed method Attention WGAN-GP most closely resemble the average mean angles in the ground truth data, and have the lowest deviation from the ground truth over time.

signals, such as sound and music. For example, this would allow generating complex motions, such as dancing driven by musical context. Nevertheless, to accomplish this would require expanding the architecture with additional encoding modules such that the model can incorporate information from the different modalities. Furthermore, it would require to adopt the attention layer accordingly.

References

[1] S. Alexanderson, G. E. Henter, T. Kucherenko, and J. Beskow, “Style-Controllable Speech-Driven Gesture Synthesis Using Normalising Flows,” *Computer Graphics Forum*, 2020.

[2] S. Narang, A. Best, and D. Manocha, “Simulating movement interactions between avatars and agents in virtual worlds using human motion constraints,” in *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, 2018, pp. 9–16.

[3] D. Holden, T. Komura, and J. Saito, “Phase-functioned neural networks for character control,” *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–13, Jul. 2017. [Online]. Available: https://dl.acm.org/doi/10.1145/3072959.3073663

[4] H. Zhang, S. Starke, T. Komura, and J. Saito, “Mode-adaptive neural networks for quadruped motion control,” *ACM Transactions on Graphics*, vol. 37, no. 4, pp. 1–11, Aug. 2018. [Online]. Available: https://dl.acm.org/doi/10.1145/3197517.3201366

[5] S. Starke, Y. Zhao, T. Komura, and K. Zaman, “Local motion phases for learning multi-contact character movements,” *ACM Transactions on Graphics*, vol. 39, Jul. 2020.

[6] Q. Men, H. P. H. Shum, E. S. L. Ho, and H. Leung, “GAN-based Reactive Motion Synthesis with Class-aware Discriminators for Human-human Interaction,” *arXiv:2110.00380 [cs]*, Oct. 2021.

[7] S. J. Guy, J. Chhugani, S. Curtis, P. Dubey, M. Lin, and D. Manocha, “PLEdestrians: A Least-Effort Approach to Crowd Simulation,” *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation*, p. 10 pages, 2010.

[8] K. Mangalam, E. Adeli, K.-H. Lee, A. Gaidon, and J. C. Niebles, “Disentangling Human Dynamics for Pedestrian Locomotion Forecasting with Noisy Supervision,” in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*. Snowmass Village, CO, USA: IEEE, Mar. 2020, pp. 2773–2782.

[9] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, “Structural-RNN: Deep Learning on Spatio-Temporal Graphs,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, Jun. 2016.

[10] S. Li, Y. Zhou, H. Zhu, W. Xie, Y. Zhao, and X. Liu, “Bidirectional recurrent autoencoder for 3D skeleton motion data refinement,” *Computers & Graphics*, vol. 81, Jun. 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0097849319300330

[11] S. Lu and A. Feng, “The deepmotion entry to the GENEA challenge 2022,” in *GENEA: Generation and Evaluation of Non-verbal Behaviour for Embodied Agents Challenge & Workshop 2022*, 2022. [Online]. Available: https://openreview.net/forum?id=zEqdFwAPhhO

[12] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, “Recurrent Network Models for Human Dynamics,” in *2015 IEEE International Conference on Computer Vision (ICCV)*. Santiago, Chile: IEEE, Dec. 2015.

[13] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ser. ICML’17. JMLR.org, 2017, p. 214–223.

[14] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved Training of Wasserstein GANs,” *arXiv:1704.00028 [cs, stat]*, Dec. 2017.

[15] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-Attention Generative Adversarial Networks,” *arXiv:1805.08318 [cs, stat]*, Jun. 2019.

[16] M. Lee and J. Seok, “Controllable Generative Adversarial Network,” *arXiv:1708.00598 [cs, stat]*, Mar. 2019.

[17] M. Mirza and S. Osindero, “Conditional Generative Adversarial Nets,” *arXiv:1411.1784 [cs, stat]*, Nov. 2014.

[18] G. Valle-Pérez, G. E. Henter, J. Beskow, A. Holzapfel, P.-Y. Oudeyer, and S. Alexanderson, “Transflower: probabilistic autoregressive dance generation with multimodal attention,” *arXiv:2106.13871 [cs, eess]*, Oct. 2021.

[19] E. Aksan, M. Kaufmann, P. Cao, and O. Hilliges, “A Spatio-temporal Transformer for 3D Human Motion Prediction,” in *2021 International Conference on 3D Vision (3DV)*. London, United Kingdom: IEEE, Dec. 2021.

[20] G. E. Henter, S. Alexanderson, and J. Beskow, “MoGlow: Probabilistic and controllable motion synthesis using normalising flows,” *ACM Transactions on Graphics*, vol. 39, no. 6, pp. 1–14, Nov. 2020. [Online]. Available: http://arxiv.org/abs/1905.06598

[21] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, “Improved training of wasserstein gans,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS’17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 5769–5779.

[22] S.-J. Li, H.-S. Zhu, L.-P. Zheng, and L. Li, “A Perceptual-Based Noise-Agnostic 3D Skeleton Motion Data Refinement Network,” *IEEE Access*, vol. 8, 2020.

[23] E. Barsoum, J. Kender, and Z. Liu, “HP-GAN: Probabilistic 3D Human Motion Prediction via GAN,” in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. Salt Lake City, UT, USA: IEEE, Jun. 2018.

[24] Y. Ma, H. M. Paterson, and F. E. Pollick, “A motion capture library for the study of identity, gender, and emotion perception from biological motion,” *Behavior Research Methods*, vol. 38, no. 1, pp. 134–141, Feb. 2006. [Online]. Available: http://link.springer.com/10.3758/BF03192758

[25] M. Malek–Podjaski and F. Deligianni, “Towards explainable, privacy-preserved human-motion affect recognition,” in *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2021, pp. 01–09.

[26] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, “Recurrent network models for human dynamics,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[27] J. Martinez, M. J. Black, and J. Romero, “On Human Motion Prediction Using Recurrent Neural Networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, Jul. 2017, pp. 4674–4683.

[28] X. Shu, L. Zhang, G.-J. Qi, W. Liu, and J. Tang, “Spatiotemporal Co-attention Recurrent Neural Networks for Human-Skeleton Motion Prediction,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.