Exploring Human Motion Synthesis with Latent-space GANs

Avinash Amballa
aamballa@umass.edu

Gayathri Akkinapalli
gakkinapalli@umass.edu

Vinitra Muralikrishnan
vmuralikrish@umass.edu

Abstract

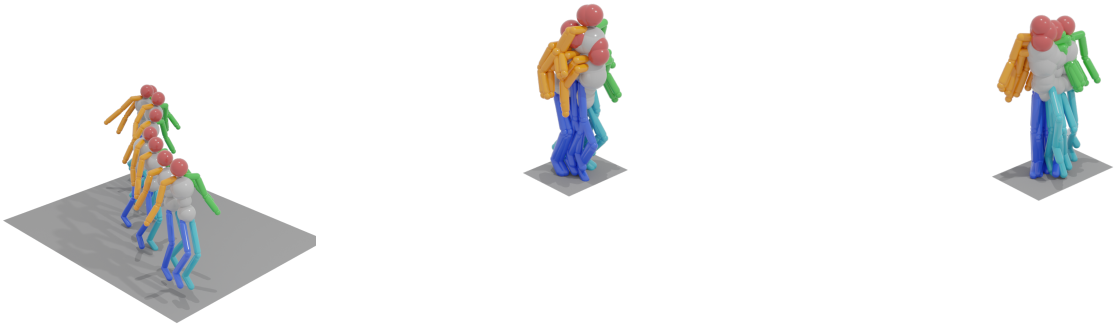
Human motion synthesis conditioned on textual inputs has gained significant attention in recent years due to its potential applications in various domains such as gaming, film production, and virtual reality. Conditioned Motion synthesis takes a text input and output a 3D motion corresponding to the text. While previous works have explored motion synthesis using raw motion data and latent space representations with diffusion models, these approaches often suffer from high training and inference times. In this paper, we introduce a novel framework that utilizes Generative Adversarial Networks (GANs) in the latent space to enable faster training and inference while achieving comparable results to state-of-the-art methods. We conduct experiments on the HumanML3D benchmark and demonstrate that a simple GAN architecture with three linear layers in the latent space achieves an FID of 2.39 and a diversity score of 8.92. Our work opens up new possibilities for efficient and high-quality human motion synthesis using latent space GANs. Code available at https://github.com/AmballaAvinash/motion-latent-diffusion

1. Introduction

Human motion synthesis has recently seen rapid advancements in a multi-modal generative fashion, fueled by various conditional inputs such as music, control signals, action categories, and notably, natural language descriptions. This field significantly enhances industries like gaming, film production, and virtual/augmented reality, with text-based conditioning standing out for its convenience and interpretability. As text-based conditioning emerges as a dominant factor due to its natural and intuitive interface for human-computer interaction, this project delves into exploring conditional-based human motion synthesis by integrating cutting-edge techniques from generative modeling and 3D deep learning. However, the direct mapping of textual descriptors to raw motion sequences often leads to misalignments and high computational demands due to stark differences in distributions between language descriptors and motion sequences, making the task of probabilistic mapping complex.

The Motion Latent Diffusion [3] model previously addressed these issues by encoding motion into a latent space using a Variational Autoencoder (VAE) [14]. However, it relied on computationally intensive diffusion processes that are less efficient, especially during the training and inference phases. However, it relied on computationally intensive diffusion processes that are less efficient, especially during the training and inference phases. In this project, we propose replacing the diffusion model [10] with a Generative Adversarial Network (GAN) [5] to capitalize on its efficient adversarial training dynamics. Recognizing the effectiveness of Generative Adversarial Networks (GANs) in learning complex representations across diverse modalities, and their efficiency in training and inference compared to diffusion models, we opt to utilize GANs within this latent space. By adopting GANs, we aim to accelerate the mapping between text embeddings and latent space, thus producing higher-quality motion sequences more efficiently.

Our approach leverages the foundational work laid out in previous studies and extends it by utilizing the capabilities of GANs to produce more plausible and diverse human motion sequences efficiently. Specifically, the paper undertakes the task of text-to-motion synthesis using conditional Generative Adversarial Networks in latent space, as depicted in the accompanying figure 2. We employ a Variational Autoencoder (VAE) to transition from motion space to latent space and utilize pretrained CLIP models from [3] to enhance our understanding of textual inputs. We experiment with various GAN architectures, including simple GAN, Dense GAN, MLP GAN [12] and Wasserstein GAN - GP [6], applying loss functions such as cross-entropy and Wasserstein to optimize performance and fidelity in the generated motion sequences. Our dataset benchmark includes HumanML3D [7] which covers wide range of human actions including regular human activities. This strategic shift not only addresses the computational inefficiencies associated with previous diffusion-based models but also leverages the rapid generative capabilities of GANs to enhance the quality and diversity of motion synthesis, suitable for real-time applications.



(a) A person walks backward slowly. (Dense GAN 100 epoch)

(b) A person is skipping rope. (Simple GAN 600 epoch)

(c) A man kicks with something or someone with his left leg. (Dense GAN 100 epoch)

Figure 1. Human Motion synthesis with Latent-space GAN.

2. Related work.

Motion synthesis is broadly categorized into conditional and unconditional motion synthesis. Unconditional motion synthesis, which models the entire motion space without requiring specific annotations, is discussed by Raab et al. [21] in an unsupervised setting using unstructured and unlabeled datasets. Conditional motion synthesis, on the other hand, employs inputs from various modalities such as music [16] and text [13] to generate motion sequences. Text-to-motion synthesis, in particular, has become a dominant area of research due to the user-friendly nature of natural language interfaces. Studies like those by Rombach et al. [23] and Chen et al. [3] have shown success in using latent space for image and motion synthesis tasks, providing a foundation for our work in this paper. The use of GAN networks for motion synthesis has been done in Ganimator [15] but uses an additional motion sequence as conditional input.

The Motion Latent Diffusion (MLD) model [3] utilizes a Variational Autoencoder (VAE) to encode human motion sequences into a low-dimensional latent space and decode them back to motion sequences. This model minimizes Mean Squared Error (MSE) and Kullback-Leibler (KL) losses during training. The MLD model then employs diffusion processes in this latent space, inspired by other latent diffusion models [23]. To condition the motion sequences on specific inputs like text or actions, the model utilizes CLIP encodings [22], demonstrating robust performance on tasks such as text-to-motion and action-to-motion. Building on this, our project aims to replace the diffusion process with a Generative Adversarial Network (GAN) in the latent space, while maintaining the use of the pretrained VAE and CLIP model for enhancing text-to-motion synthesis.

Additional recent advancements in the field include the development of joint-latent models like TEMOS [18] and conditional diffusion models [13, 26, 29], which have led to significant progress. TEMOS, uses a VAE architecture to create a shared latent space for motion and text based on a Gaussian distribution. However, aligning the complex and distinct distributions of natural language and human motion often presents challenges, as simple Gaussian distributions do not adequately represent the intricacies involved [24]. Moreover, approaches like MotionGPT [11] integrates language modeling for both motion and text, treating human motion as a distinct language to construct a generalized model capable of executing various motion tasks through VQ-VAE [27]. These developments underscore the evolving landscape of motion synthesis, guiding our approach to leveraging GANs in latent space for efficient and high-quality motion generation.

Generative Adversarial Networks (GANs) [5] function through a unique adversarial process where a generative model (G) learns to mimic data distribution while a discriminative model (D) distinguishes between real and generated data, effectively playing a minimax game. This setup allows GANs to produce high-quality, realistic outputs efficiently using only backpropagation, without needing Markov chains or complex inference networks, making them particularly effective for generating diverse and realistic samples. StyleGAN [12] uses style transfer techniques in its architecture to automatically distinguish high-level attributes and stochastic variations in generated images, improving control, interpolation quality, and the disentanglement of latent factors.

3. Our proposed Method

Our work focuses on exploring how the latent space can be used by other architectures for motion synthesis and attempting to improve their performance. While diffusion models have shown tremendous promise and exhibit state-of-the-art performance they are expensive to train, requiring a huge corpus of data. The use of latent space in MLD opens up avenues for other architectures to also leverage it and our efforts focus on attempting to improve the perfor-

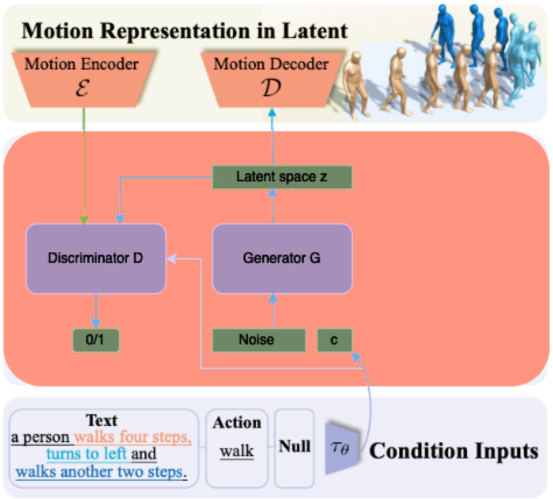


Figure 2. Our proposed Conditonal GAN based architecture for text to Motion synthesis in latent space

mance of simpler models. We start with some simpler architectures for the generator and discriminator and expand it to include residual connections. We also explore architectures like StyleGAN and borrow parts of its architecture to fit the needs of this problem i.e predicting latent space representation of the motion sequences.

3.1. VAE and CLIP

Our VAE architecture is borrowed from the MLD [3], which uses transformer model as Encoder and Decoder with skip connections. The VAE is trained in a similar fashion as MLD with the reconstruction MSE loss and KL divergence. Once the VAE is trained, it is frozen.

We use Pretrained CLIP-ViT-L-14 as text encoder. Intructions to download this model can be found in the readme section of the code.

3.2. Latent space GAN

Previous works explored diffusion models in the latent space. In addition to MLD, Diffusion GAN [28] integrate the GAN with the diffusion process. To the best of our knowledge, we are first to introduce GAN in the latent space. We chose GAN models for 3 reasons - (1) Their effectiveness in generational capabilities from latent space representations. (2) The flexibility of implementing any architecture for the generator and discriminator and the potential adversarial training offers. (3) reduced training and inference time compared to Diffusion models.

We follow the architecture from the conditional GAN [17], where the generator receives the noise and conditioned input and generates the motion latent space, whereas the discriminator receives the real motion latent space from the VAE encode along with the conditioned text embeddings. Refer to figure 2 for the model pipeline.

3.3. GAN architectures

We explored different GAN architectures in the latent space setting.

3.3.1 Simple GAN

We start with a simple Generator with 3 Linear layers that maps the [noise, conditioned text embeddings] to a motion latent space, and the discriminator with 4 linear layers followed by a sigmoid activation maps this [motion latent space, conditioned text embedding] to binary value.

3.3.2 Dense GAN

We add two residual blocks (with 1D convolutions) in between two linear layers into the Generator and discriminator architectures. Residual connection [9] helps to train deeper networks by overcoming the vanishing gradient problem.

3.3.3 MLP GAN

We borrowed some components from Style GAN [12], and adapted it the generate the motion latent space. We borrowed the mapping network and also initially experimented with a 3-layer Adaptive Instance Normalization and Noise layers. The introduction of noise in a latent space prior seemed to disrupt the predictions, giving a high FID metric of 94.56. Elimination of this block and using just the mapping network gave significantly better quantitative metrics, which ultimately reduced the generator architecture to

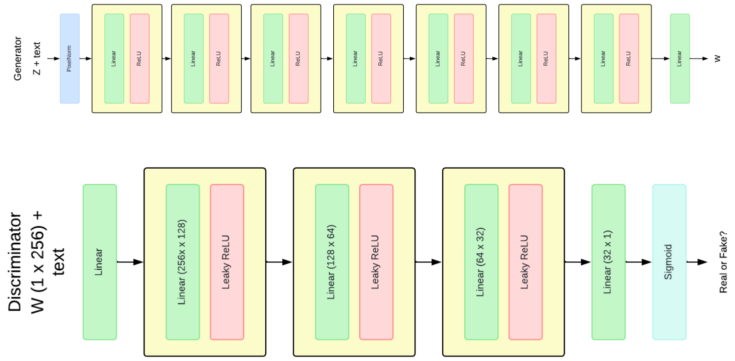


Figure 3. Our proposed Conditional MLP-GAN architecture for text to Motion synthesis in latent space

the mapping network i.e the MLP. The discriminator was a simple 3-layer network that reduced the latent space to a single scalar predicting whether or not the generated space represents a real or fake motion sequence. The final architecture we adopted is shown in figure 3

3.3.4 WGAN-GP

To improve the qualitative and quantitative metrics, we implemented the Wasserstein GAN - GP architecture [6]. WGAN-GP by design tries to reduce the Wasserstein distance between the distribution of real and fake data. Models based on this architecture have been shown better generational capabilities and do not suffer from the same training instabilities that the traditional GAN does. We have followed all the training methods suggested in [6] on a simple 4-block (Linear-ReLU) GAN network consisting of 1 residual block.

In terms of training, this has been easier to fine-tune. Although experiments performed so far have been on simple GAN architectures, we were able to achieve the same performance as the BCE loss with less time spent on hyperparameter fine-tuning. Our experiment results show the flexibility of adapting latent space and getting good qualitative performance on even simpler architectures. Given our resource and time constraints, we couldn't take the training further than 100 epochs for both the models but WGAN-GP adapted on deeper networks shows greater potential in the effective use of latent space.

3.4. Loss functions

3.4.1 Binary Cross entropy:

We have followed the Condition GAN architecture and experimented with the standard BCE Loss function, which predicts the probability of latent spacing representing real or fake motion sequences.

Metric	VAE 250 epoch	VAE 1250 epoch
APE_root/mean ↓	$0.0897^{\pm 0.0002}$	$\mathbf{0.0756^{\pm 0.0002}}$
APE_traj/mean ↓	$0.0857^{\pm 0.0002}$	$\mathbf{0.0723^{\pm 0.0002}}$
APE_mean_pose/mean ↓	$0.0379^{\pm 0.0000}$	$\mathbf{0.0312^{\pm 0.0000}}$
APE_mean_joints/mean ↓	$0.1008^{\pm 0.0002}$	$\mathbf{0.0845^{\pm 0.0002}}$
AVE_root/mean ↓	$0.0221^{\pm 0.0001}$	$\mathbf{0.0201^{\pm 0.0001}}$
AVE_traj/mean ↓	$0.0220^{\pm 0.0001}$	$\mathbf{0.0200^{\pm 0.0001}}$
AVE_mean_pose/mean ↓	$0.0021^{\pm 0.0000}$	$\mathbf{0.0015^{\pm 0.0000}}$
AVE_mean_joints/mean ↓	$0.0241^{\pm 0.0001}$	$\mathbf{0.0216^{\pm 0.0001}}$
R_precision_top_1 ↑	$0.4422^{\pm 0.0030}$	$\mathbf{0.4891^{\pm 0.0020}}$
R_precision_top_2 ↑	$0.6337^{\pm 0.0020}$	$\mathbf{0.6803^{\pm 0.0023}}$
R_precision_top_3 ↑	$0.7379^{\pm 0.0025}$	$\mathbf{0.7787^{\pm 0.0021}}$
FID ↓	$1.1754^{\pm 0.0030}$	$\mathbf{0.2661^{\pm 0.0010}}$
Diversity →	$9.3856^{\pm 0.0843}$	$\mathbf{9.6901^{\pm 0.0990}}$
MultiModality ↑	$\mathbf{0.2056^{\pm 0.0095}}$	$0.1237^{\pm 0.0058}$

Table 1. Comparison of metrics between VAE 250 epoch and VAE 1250 epoch. The right arrow → means the closer to real motion the better. **Bold** indicate the best result.

3.4.2 Wasserstein loss:

The Wasserstein loss seeks to reduce the distance between the distribution observed in the training data and the generated samples. To achieve this the discriminator was modified to predict a scalar value (not bounded between 0 and 1) scoring the generated samples so it can better guide the generator to produce better samples in WGAN-based methods.

4. Datasets

We use HumanML3D [7] benchmark. HumanML3D is a 3D human motion-language dataset which covers a wide range of human actions including human activities like walking, jumping, swimming, playing golf etc. It contains 14,616 motion sequences from AMASS and annotates 44,970 sequence-level textual descriptions. In this paper,

Methods	R Precision ↑			FID↓	MM Dist↓	Diversity→	MModality↑
	Top 1	Top 2	Top 3				
Real	$0.511^{\pm.003}$	$0.703^{\pm.003}$	$0.797^{\pm.002}$	$0.002^{\pm.000}$	$2.974^{\pm.008}$	$9.503^{\pm.065}$	-
Seq2Seq [20]	$0.180^{\pm.002}$	$0.300^{\pm.002}$	$0.396^{\pm.002}$	$11.75^{\pm.035}$	$5.529^{\pm.007}$	$6.223^{\pm.061}$	-
LJ2P [1]	$0.246^{\pm.001}$	$0.387^{\pm.002}$	$0.486^{\pm.002}$	$11.02^{\pm.046}$	$5.296^{\pm.008}$	$7.676^{\pm.058}$	-
T2G [2]	$0.165^{\pm.001}$	$0.267^{\pm.002}$	$0.345^{\pm.002}$	$7.664^{\pm.030}$	$6.030^{\pm.008}$	$6.409^{\pm.071}$	-
Hier [4]	$0.301^{\pm.002}$	$0.425^{\pm.002}$	$0.552^{\pm.004}$	$6.532^{\pm.024}$	$5.012^{\pm.018}$	$8.332^{\pm.042}$	-
TEMOS [19]	$0.424^{\pm.002}$	$0.612^{\pm.002}$	$0.722^{\pm.002}$	$3.734^{\pm.028}$	$3.703^{\pm.008}$	$8.973^{\pm.071}$	$0.368^{\pm.018}$
T2M [8]	$0.457^{\pm.002}$	$0.639^{\pm.003}$	$0.740^{\pm.003}$	$1.067^{\pm.002}$	$3.340^{\pm.008}$	$9.188^{\pm.002}$	$2.090^{\pm.083}$
MDM [25]	$0.320^{\pm.005}$	$0.498^{\pm.004}$	$0.611^{\pm.007}$	$\underline{0.544}^{\pm.044}$	$5.566^{\pm.027}$	$\mathbf{9.559}^{\pm.086}$	$\mathbf{2.799}^{\pm.072}$
MotionDiffuse [29]	$\mathbf{0.491}^{\pm.001}$	$\mathbf{0.681}^{\pm.001}$	$\mathbf{0.782}^{\pm.001}$	$0.630^{\pm.001}$	$\mathbf{3.113}^{\pm.001}$	$\underline{9.410}^{\pm.049}$	$1.553^{\pm.042}$
MLD [3]	$\underline{0.481}^{\pm.003}$	$\underline{0.673}^{\pm.003}$	$\underline{0.772}^{\pm.002}$	$\mathbf{0.473}^{\pm.013}$	$\underline{3.196}^{\pm.010}$	$9.724^{\pm.082}$	$\underline{2.413}^{\pm.079}$
Simple GAN epoch = 150 (Ours)	$0.286^{\pm.003}$	$\mathbf{0.452^{\pm.003}}$	$\mathbf{0.568^{\pm.002}}$	$7.645^{\pm.047}$	-	$8.628^{\pm.094}$	$0.623^{\pm.033}$
Simple GAN epoch = 600 (Ours)	$\mathbf{0.288^{\pm.003}}$	$0.443^{\pm.003}$	$0.552^{\pm.002}$	$\mathbf{2.397^{\pm.029}}$	-	$\mathbf{8.921^{\pm.083}}$	$0.457^{\pm.022}$
Dense GAN epoch = 100 (Ours)	$0.244^{\pm.002}$	$0.391^{\pm.002}$	$0.498^{\pm.002}$	$7.507^{\pm.052}$	-	$7.573^{\pm.069}$	$0.874^{\pm.045}$
Dense GAN epoch = 250 (Ours)	$0.180^{\pm.002}$	$0.300^{\pm.002}$	$0.394^{\pm.003}$	$18.058^{\pm.069}$	-	$6.637^{\pm.045}$	$0.880^{\pm.041}$
MLP GAN epoch = 100 (Ours)	$0.072^{\pm.001}$	$0.137^{\pm.002}$	$0.196^{\pm.002}$	$21.660^{\pm.041}$	-	$5.776^{\pm.075}$	$\mathbf{2.633^{\pm.086}}$
Simple WGAN epoch = 150 (Ours)	$0.062^{\pm.001}$	$0.118^{\pm.001}$	$0.170^{\pm.002}$	$34.012^{\pm.098}$	-	$5.472^{\pm.080}$	$1.246^{\pm.066}$
Dense WGAN epoch = 150 (Ours)	$0.058^{\pm.002}$	$0.109^{\pm.002}$	$0.156^{\pm.002}$	$11.073^{\pm.058}$	-	$7.668^{\pm.092}$	$1.737^{\pm.105}$
Simple WGANGP epoch = 30 (Ours)	$0.0379^{\pm.001}$	$0.0778^{\pm.002}$	$0.111^{\pm.001}$	$32.428^{\pm.001}$	-	$2.811^{\pm.001}$	$0.616^{\pm.002}$

Table 2. Comparison of text-conditional motion synthesis on HumanML3D dataset. These metrics are evaluated by the motion encoder from [8]. Empty MModality indicates the non-diverse generation methods. We employ real motion as a reference and sort all methods by descending FIDs. The right arrow → means the closer to real motion the better. **Bold** and underline indicate the best (in our work aswell)and the second best result.

the main dataset used in our experiments is HumanML3d.

5. Evaluation metrics

We assess the performance of our models utilizing various metrics:

1. FID: Fréchet Inception Distance is used to evaluate the feature distributions between the generated and real motions by feature extractor.

2. R-precision: Motion-retrieval precision (R Precision) calculates the text and motion Top 1/2/3 matching accuracy.

3. Diversity: Diversity (DIV) calculates variance through features. It assesses the variance among different generated outputs from the same input, indicating the model's ability to produce varied and distinct results.

4. Multi-modality: MultiModality (MM) measures the model's ability to effectively generate outputs that match multiple modal types of data, such as generating motion from textual descriptions in text-to-motion synthesis. It measures the generation diversity within the same text input.

5. MM Dist (Multimodal Distance): Multi-modal Distance (MM Dist) calculates the distance between motions and texts. Evaluates how well the generated outputs from a model cover the diversity and accuracy of all potential correct answers or outputs, especially in multimodal contexts.

6. APE: Average position error (APE) measures the average Euclidean distance between the predicted or reconstructed motion and the ground truth motion in terms of joint positions.

7. AVE: Average variance error (AVE) is a metric that measures the average difference in the variance of joint positions between the predicted or reconstructed motion and the ground truth motion.

6. Training details and Results

We train all our models on A100 GPU cluster with 1 GPU. VAE took around 24-30 hours to reach 1250 checkpoint. Below are test metrics for 20 trials and calculated mean and std. Given the time constraints we faced limitations in training any model beyond 600 epochs. We have recorded the best results we got and the epoch where it happened.

Table 1 shows the various metrics for a VAE trained for 250 epochs and VAE trained for 1250 epochs on the HumanML dataset. We observe that VAE 1250 epoch model has lower APE and AVE, lower FID, higher diversity and higher R precision. Hence, we took the VAE 1250 checkpoint and froze this for the GAN training.

Table 2 shows that the Simple and dense architectures give the best empirical metrics and qualitative results when used with BCE loss. When used with WGAN, metrics are higher but they are more stabilized and potentially could be trained further to achieve better results, especially with deeper networks. It is recommended to implement deeper, more complex architectures with WGAN as it provides methods to exert more control over adversarial training.

A simple 3 layered GAN with 600 epochs, beats Seq2Seq [20], LJ2P [1], T2G [2] , Hier [4], TEMOS [19]. These results demonstrate that a simple GAN in latent space can achieve impressive results with minimal compute, training and inference time compared to the complex model such as Diffusion. One can integrate quantized VAE and explore other GAN training procedure to achieve comparable or even more performance compared to the state-of-the-art MLD [3] or MotionDiffuse [29] .

7. Conclusion

Latent space GANs are unexplored in the filed of Computer vision. In this paper, we introduced a novel approach for text-to-motion synthesis using Generative Adversarial Networks (GANs) in the latent space. By leveraging the power of GANs and the compact representation of motion sequences in the latent space, our framework achieves faster training and inference times compared to previous methods while maintaining high-quality motion synthesis results. We explore different GAN architectures with 2 loss functions. Results demonstrate that a simple GAN in latent space is comparable to complex models. This work will open a new direction in exploring latent space GANs that can have faster training and inference time compared to latent space diffusion.

References

[1] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *2019 International Conference on 3D Vision (3DV)*, pages 719–728. IEEE, 2019. 5, 6

[2] Uttaran Bhattacharya, Nicholas Rewkowski, Abhishek Banerjee, Pooja Guhan, Aniket Bera, and Dinesh Manocha. Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pages 1–10. IEEE, 2021. 5, 6

[3] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, Jingyi Yu, and Gang Yu. Executing your commands via motion diffusion in latent space, 2023. 1, 2, 3, 5, 6

[4] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1396–1406, 2021. 5, 6

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 1, 2

[6] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans, 2017. 1, 4

[7] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5152–5161, 2022. 1, 4

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

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 3

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 1

[11] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language, 2023. 2

[12] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks, 2019. 1, 2, 3

[13] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. Flame: Free-form language-based motion synthesis editing, 2023. 2

[14] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. 1

[15] Peizhuo Li, Kfir Aberman, Zihan Zhang, Rana Hanocka, and Olga Sorkine-Hornung. Ganimator: neural motion synthesis from a single sequence. *ACM Transactions on Graphics*, 41(4):1–12, July 2022. 2

[16] Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++, 2021. 2

[17] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets, 2014. 3

[18] Mathis Petrovich, Michael J. Black, and Gül Varol. Temos: Generating diverse human motions from textual descriptions, 2022. 2

[19] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, 2022. 5, 6

[20] Matthias Plappert, Christian Mandery, and Tamim Asfour. Learning a bidirectional mapping between human whole-body motion and natural language using deep recurrent neural networks. *Robotics and Autonomous Systems*, 109:13–26, 2018. 5, 6

[21] Sigal Raab, Inbal Leibovitch, Peizhuo Li, Kfir Aberman, Olga Sorkine-Hornung, and Daniel Cohen-Or. Modi: Unconditional motion synthesis from diverse data, 2022. 2

[22] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 2

[23] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 2

[24] Guy Tevet, Brian Gordon, Amir Hertz, Amit H. Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space, 2022. 2

[25] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 5

[26] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H. Bermano. Human motion diffusion model, 2022. 2

[27] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning, 2018. 2

[28] Zhendong Wang, Huangjie Zheng, Pengcheng He, Weizhu Chen, and Mingyuan Zhou. Diffusion-gan: Training gans with diffusion, 2023. 3

[29] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. 2, 5, 6