Utilizing Task-Generic Motion Prior to Recover Full-Body Motion from Very Sparse Signals (Supplementary Material)

Myungjin Shin
Yonsei University
mjsh25@yonsei.ac.kr

Dohae Lee
Yonsei University
dlehgo1414@yonsei.ac.kr

In-Kwon Lee
Yonsei University
iklee@yonsei.ac.kr

Contents

Appendices

A. Model and Training Details

A.1. Input Derivation Details

From sparse pose signals denoted $\mathbf{x}_t$ we derive sparse motion signals denoted $\mathbf{X}_t$ as follows:

$$\mathbf{x}_t = [\mathbf{g}_t^{\mathbf{3}}, \mathbf{r}_t^{\mathbf{3}}], \tag{1}$$

$$\mathbf{X}_t = [\mathbf{g}_t^{\mathbf{3}}, \dot{\mathbf{g}}_t^{\mathbf{3}}, \mathbf{r}_t^{\mathbf{3}}, \dot{\mathbf{r}}_t^{\mathbf{3}}] \in \mathbb{R}^{54}, \tag{2}$$

where $\mathbf{g}_t^{\mathbf{3}}$ and $\mathbf{r}_t^{\mathbf{3}}$ respectively denote the 3D global positions and the global rotations in 6D form [18] of head and hands, $\dot{\mathbf{g}}_t^{\mathbf{3}}$ denoting linear velocity calculated as finite difference of $\mathbf{g}_t^{\mathbf{3}}$:

$$\dot{\mathbf{g}}_t^{\mathbf{3}} = \mathbf{g}_t^{\mathbf{3}} - \mathbf{g}_{t-1}^{\mathbf{3}}, \tag{3}$$

and $\dot{\mathbf{r}}_t^{\mathbf{3}}$ denoting angular velocity calculated as:

$$\dot{\mathbf{R}}_t^j = \mathbf{R}_{t-1}^j \mathbf{R}_t^j, \quad \forall j \in \{head, lhand, rhand\}, \tag{4}$$

where $\mathbf{R}_{t-1}^j$, $\mathbf{R}_t^j$, and $\dot{\mathbf{R}}_t^j$ denote global rotation matrices at time t and $t-1$, and angular velocity rotation matrix at time t each joint respectively. 6D angular velocity $\dot{\mathbf{r}}_t^j$ of each joint is derived by extracting the first two columns of $\dot{\mathbf{R}}_t^j$ [18].

A.2. Neural Network Architecture and Training

To train the full motion prior, we use the "paper_model" training configuration of MotionCLIP's [15] official implementation [16], namely with $\lambda_{text} = \lambda_{image} = 0.01$. AdamW [12] with learning rate of 0.0001 with batch size of 20 for 100 epochs is used for optimization. For the sparse motion prior, the configuration remains the same except we use a frozen decoder from the full motion prior as described in the main paper.

The sequence model consists of a linear layer mapping motion latent $\mathbf{M}^* \in \mathbb{R}^{512}$ predicted by the sparse motion encoder to a 64-dimensional vector $\mathbf{E}$ (motion embedding), which is concatenated with sparse motion signal $\mathbf{X}_t \in \mathbb{R}^{54}$ along the time axis, to be input to the 3-Layer LSTM [5] with hidden layer size of 128. Finally, the final hidden layer output of the LSTM passes through a linear layer to obtain 6D relative rotations of 22 SMPL joints [11]. We use the Adam optimizer [8] with initial learning rate of 0.002 decaying by half every 15 epochs and train for 200 epochs with batch size of 32.

A.3. Evaluation Configuration

The quantitative results in the main paper are computed over the entire test dataset with a sliding window

Method	Per-Joint Errors				Motion-Related Statistics	
	MPJPE	Legs MPJPE	Global MPJPE	MPJVE	Motion Distance ↓	FID ↓
Sequence Pretrained	7.43	9.28	7.75	53.9	$8.08 \cdot 10^{-3}$	$9.24 \cdot 10^{-2}$
Static Pretrained	7.68	9.62	7.67	56.2	$12.3 \cdot 10^{-3}$	$12.5 \cdot 10^{-2}$

Table 1. VAE-HMD Variations.

Method	Per-Joint Errors				Motion-Related Statistics	
	MPJPE	Legs MPJPE	Global MPJPE	MPJVE	Motion Distance ↓	FID ↓
Ours (Main Paper Method)	7.17	9.22	7.33	25.5	$5.06 \cdot 10^{-3}$	$6.03 \cdot 10^{-2}$
No Velocity Loss	7.29	9.39	7.42	27.4	$5.16 \cdot 10^{-3}$	$6.55 \cdot 10^{-2}$
No GPos Normalization	7.49	9.43	7.86	26.2	$5.38 \cdot 10^{-3}$	$8.20 \cdot 10^{-2}$
Divide by STD	8.37	10.2	9.02	30.0	$6.02 \cdot 10^{-3}$	$11.7 \cdot 10^{-2}$
Normalize Vertically	10.0	13.1	10.2	29.6	$7.15 \cdot 10^{-3}$	$10.9 \cdot 10^{-2}$
ME After LSTM	7.62	9.91	7.81	26.3	$5.75 \cdot 10^{-3}$	$7.45 \cdot 10^{-2}$
With Cross-Attention	7.17	9.28	7.34	26.4	$4.85 \cdot 10^{-3}$	$6.57 \cdot 10^{-2}$
AvatarPoser	7.68	10.2	7.76	29.8	$5.36 \cdot 10^{-3}$	$7.67 \cdot 10^{-2}$
VAE-HMD	7.43	9.28	7.75	55.0	$8.08 \cdot 10^{-3}$	$9.24 \cdot 10^{-2}$

Table 2. Extra Ablation Studies.

moving by 1 frame. The extra results presented here are computed with a sliding window moving by the size of itself, so while still spanning the entire test dataset, exact figures may differ.

B. Baseline Models Details and Extra Results

B.1. 60-Frame Models

VAE-HMD [1] originally accepts sparse pose signals that contain 16-frame information, and AvatarPoser [6] 40. We require 60-frame input to compute the motion embedding, and to ensure fairness, we trained two extra baseline models that accept 60-frame input by adapting [1, 6].

For VAE-HMD, we adapt their "sequence pretrained" model with the most parameters (with 6 residual blocks) to first train the VAE [9] to have the encoder that accepts 60-frame full-body poses, and the decoder that outputs a single full-body pose. Then we train a new encoder that accepts 60-frame sparse motion signals with a frozen decoder trained in previous step. For AvatarPoser, we simply extend the window size to 60, such that the Transformer Encoder [17] accepts input of length 60 instead of 40, via their official implementation [7]. The quantitative metrics worsened after extending the window size as observed in the main paper, and we suspect the cause to be that the original models' hyperparameters including window sizes were optimized to yield the best performance during their development, just as ours were optimized for performance while developing our own model.

B.2. VAE-HMD Variations

VAE-HMD [1] presents two types of models in the original paper, namely, the "sequence pretrained" model takes 16-frame information in the encoder (this variant is compared against ours in the main paper), and the "static pretrained" model only takes a single frame information in the encoder. The quantitative results are presented in Table 1. It can be seen that the static model generally performs worse than the sequence model, reconstruction abilties of the lower body and overall motion compromised, which can also be seen in Figure 1

C. Extra Ablation Studies

We perform extra ablation studies by removing the velocity loss term, with different global position normalization methods, and with architectural variations. Results are presented in Table 2.

C.1. Train without Velocity Loss Term

We present results obtained from training our model without the velocity loss term, for our baselines [1, 6] were not optimzed to explicitly minimize velocity error. As can be seen on "No Velocity Loss" row of Table 2, MPJVE slightly increases but there is no significant difference overall. We observed from the generated motions on the other hand, that model trained with velocity loss term produced smoother motions that were overall less stiff.

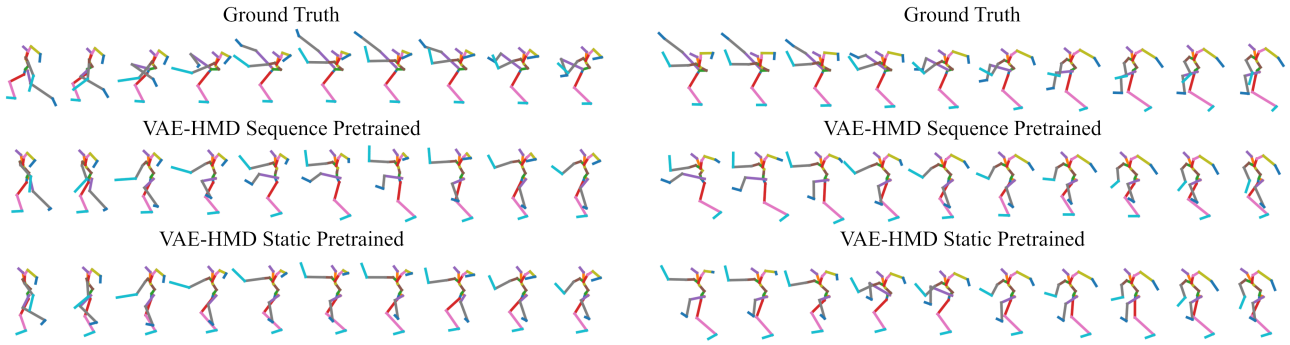

Figure 1. Qualitative Results. VAE-HMD sequence (motion) model vs. static model.

C.2. Global Position Normalization

We experimentally find normalizing global positions to improve training as explained in the main paper. "No GPos Normalization" (Table 2) shows results when no such normalization is applied. We only apply normalization horizontally for our main method, and we observe significant performance drop when normalization is also applied vertically ("Normalize Vertically"). We apply the global position normalization by subtracting the mean positions of three IMUs, and we experiment by also dividing by the STD, whose results are shown in row "Divide by STD".

C.3. Append Motion Embedding After RNN

In our main method, we concatenate the motion embedding $\mathbf{E}_t$ and the sparse motion input $\mathbf{X}_t$ before they are input to the LSTM (the sequence model). We vary the architecture by only inputting $\mathbf{X}_{t-S+1:t}$ to the LSTM and concatenating the output with $\mathbf{E}_{t-S+1:t}$, followed by a shallow MLP to recover the full-body pose. The results can be seen on the "ME After LSTM" row on Table 2.

C.4. Motion Conditioning via Cross-Attention

In an attempt to improve upon Section C.3's method to better condition the output of the LSTM on the overall motion information, we compute the cross-attention [17] between the motion embedding (as Query [17]) and the LSTM output (as Key and Value [17]), followed by a shallow MLP to compute the full-body pose. As can be seen on "With Cross-Attention" row of Table 2, only motion distance slightly improved over the main method, and we also preferred the output motions by the main method in the qualitative evaluation.

	FID ↓	FID 2 ↓
Ours	$6.03 \cdot 10^{-2}$	0.131
AvatarPoser [6]	$7.59 \cdot 10^{-2}$	0.137
AvatarPoser-60	$7.87 \cdot 10^{-2}$	0.160
VAE-HMD [1]	$9.24 \cdot 10^{-2}$	0.167
VAE-HMD-60	$9.40 \cdot 10^{-2}$	0.174

Table 3. Different FID metrics.

D. On Evaluation Methods

D.1. User Study

We show screenshots from videos containing short motion segments used for user study I (Figure 2) and user study II (Figure 3).

D.2. FID

FID (Fretchet Inception Distance [4]) measures similarty between two distributions, in our experiments between ground truth motions and motions generated from predictions from our model or one of the baselines. While methods that are meant to generate arbitrary motions make use of this metric [2, 13, 10, 3], our distribution of generated motions derives from estimations of ground truth motions, resulting in very low FID compared to other works. To mitigate this effect, we devise a second FID metric, where we randomly split the underlying motions into two, and use one set of underlying motions for ground truth motion distribution and the other for predicted motion distribution. Because the splitting is random, there can be a large variation depending on the split, which is why apply the splitting process 50 times and compute the average value. While the result of the second FID metric is not used in the main paper, we present the results in Table 3.

E. BABEL Action Types

We present qualitative results of our model and baselines performing actions categorized by BABEL [14] on the test dataset (Figures 4, 5, 6, 7, 8, 9, 10).

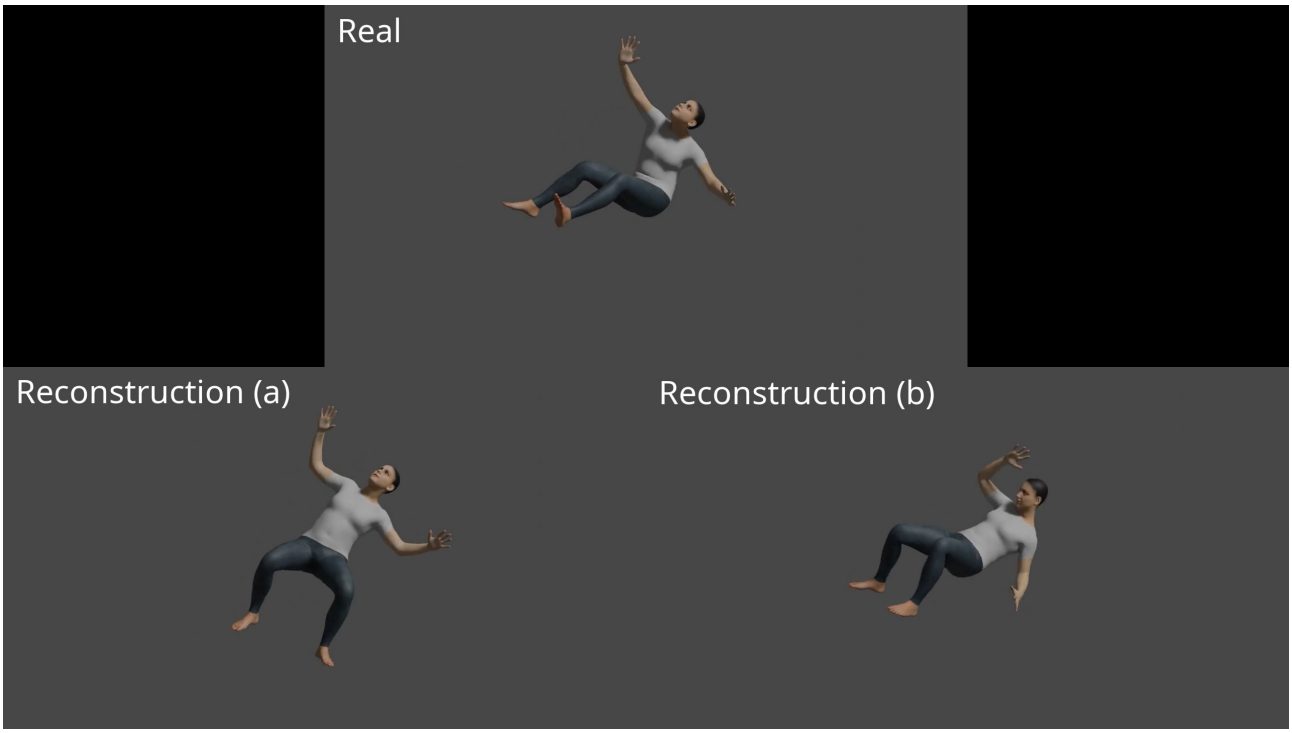

Figure 2. User Study I Example Video. Participants were told that the bottom two animations are reconstructions of the top animation from partial information.

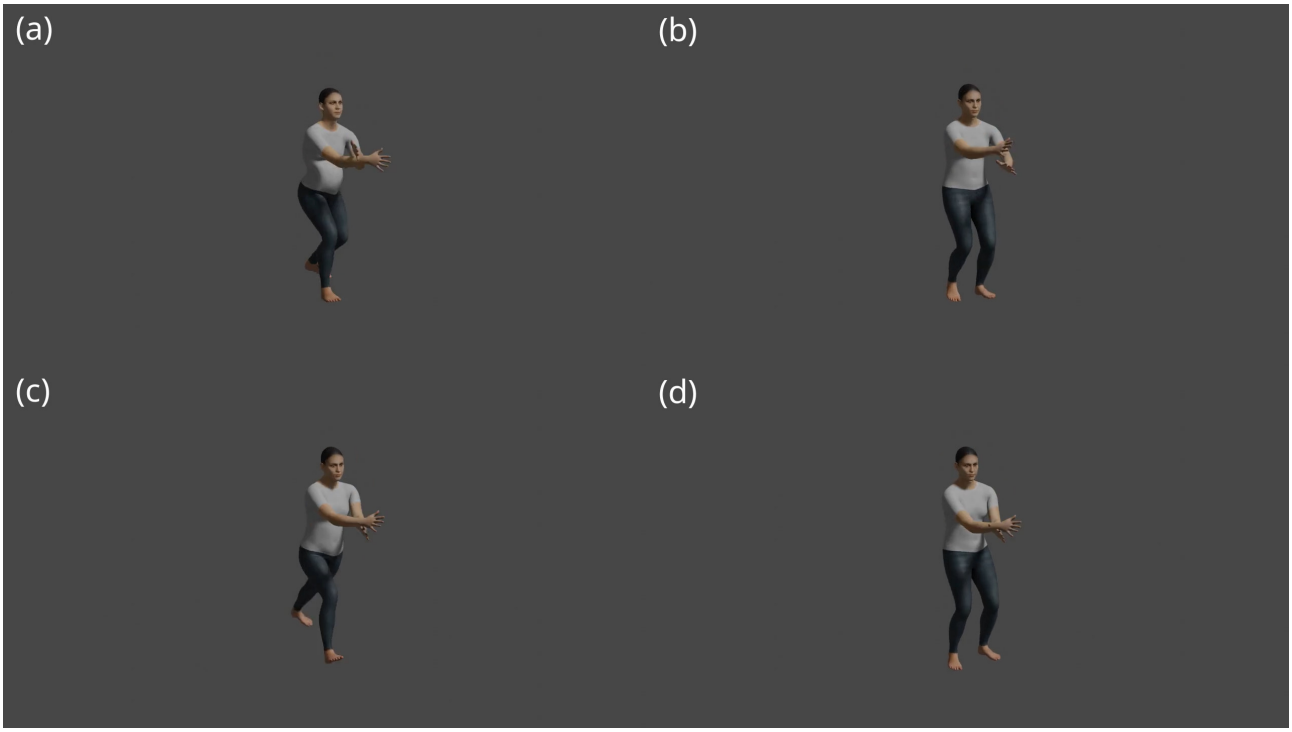

Figure 3. User Study II Example Video. Participants were asked to score each animation for naturalness on the scale of 1 to 7.

Figure 4. Qualitative Results. BABEL action label: “knee movement”

Figure 5. Qualitative Results. BABEL action label: “crouch”

Figure 6. Qualitative Results. BABEL action label: “squat”

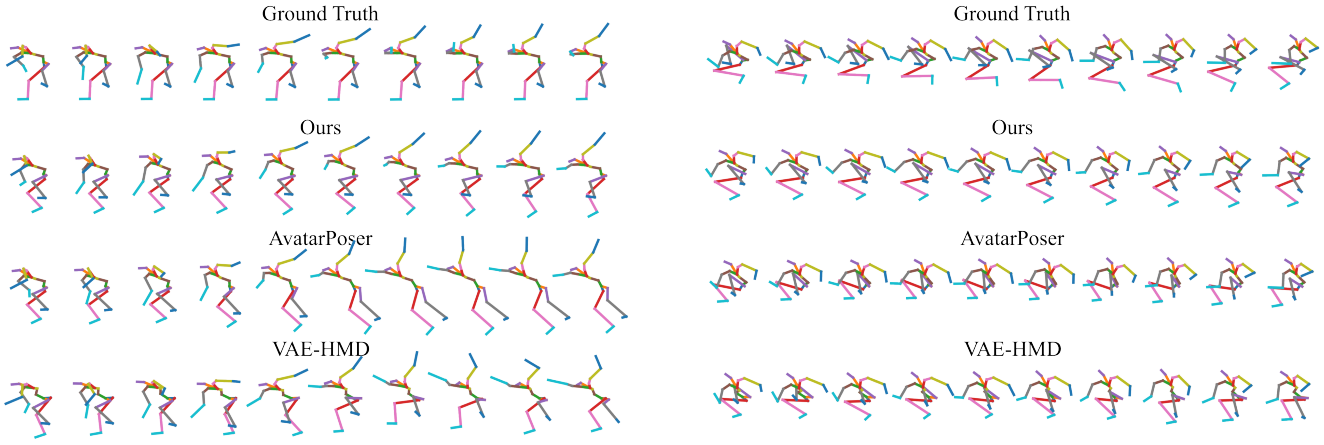

Figure 7. Qualitative Results. BABEL action label: “bend”

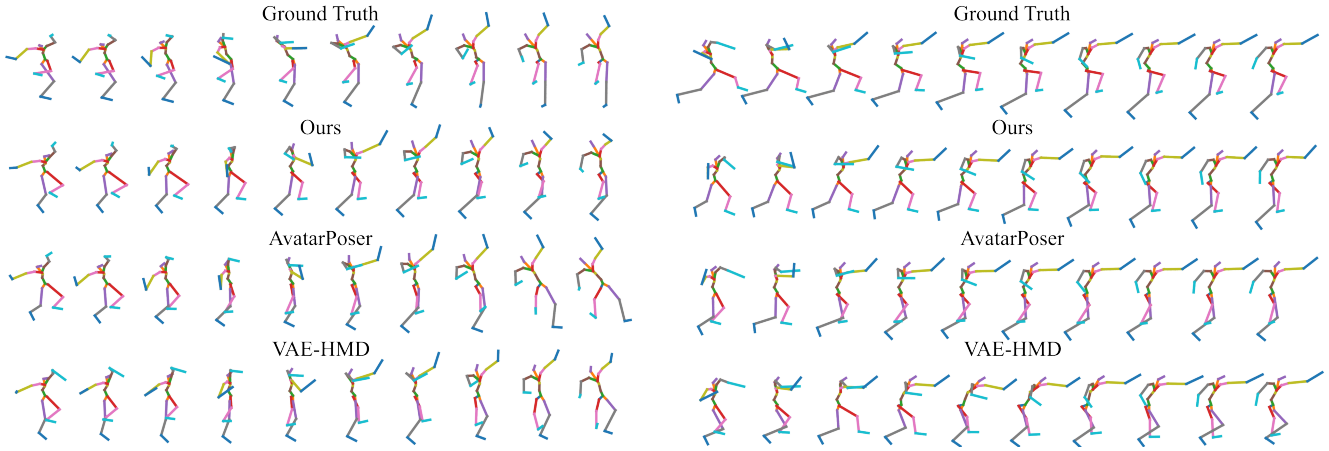

Figure 8. Qualitative Results. BABEL action label: “throw”

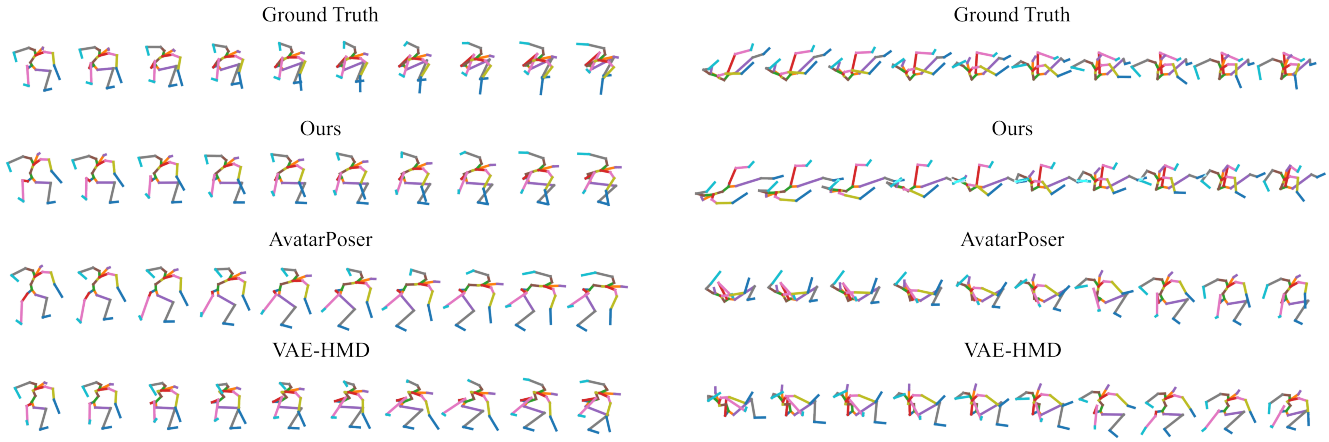

Figure 9. Qualitative Results. BABEL action label: “touch ground”

Figure 10. Qualitative Results. BABEL action label: “place something”

References

[1] Andrea Dittadi, Sebastian Dziadzio, Darren Cosker, Ben Lundell, Thomas J Cashman, and Jamie Shotton. Full-body motion from a single head-mounted device: generating smpl poses from partial observations. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11687–11697, 2021.

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

[3] Chengan He, Jun Saito, James Zachary, Holly Rushmeier, and Yi Zhou. Nemf: Neural motion fields for kinematic animation. In NeurIPS, 2022.

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

[5] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

[6] Jiaxi Jiang, Paul Streli, Huajian Qiu, Andreas Fender, Larissa Laich, Patrick Snape, and Christian Holz. Avatarposer: Articulated full-body pose tracking from sparse motion sensing. In Proceedings of European Conference on Computer Vision. Springer, 2022.

[7] Jiaxi Jiang, Paul Streli, Huajian Qiu, Andreas Fender, Larissa Laich, Patrick Snape, and Christian Holz. Avatarposer: Articulated full-body pose tracking from sparse motion sensing. https://github.com/eth-siplab/AvatarPoser, 2022. [Online; accessed 1-March-2023].

[8] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[9] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[10] Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. Character controllers using motion vaes. ACM Trans. Graph., 39(4), aug 2020.

[11] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. ACM Trans. Graphics (Proc. SIGGRAPH Asia), 34(6):248:1–248:16, Oct. 2015.

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2017.

[13] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In International Conference on Computer Vision (ICCV), 2021.

[14] Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black. BABEL: Bodies, action and behavior with english labels. In Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), pages 722–731, June 2021.

[15] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII, pages 358–374. Springer, 2022.

[16] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. https://github.com/GuyTevet/MotionCLIP, 2022. [Online; accessed 1-March-2023].

[17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, Advances in Neural Information Processing Systems, volume 30. Curran Associates, Inc., 2017.

[18] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.