

Motion-2-to-3: Leveraging 2D Motion Data to Boost 3D Motion Generation

Supplementary Material

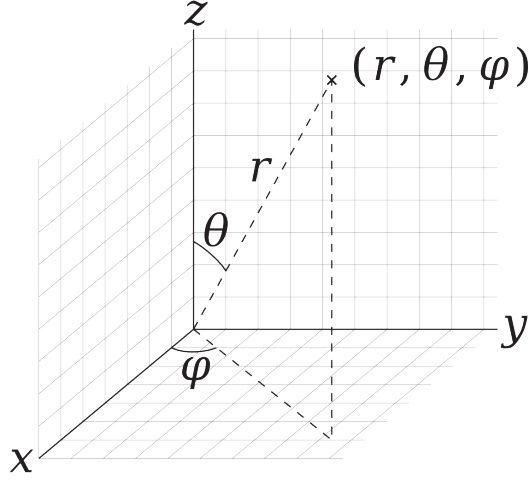


Figure 1. **Spherical coordinate system.** We use the figure from [27] to illustrate the spherical coordinate system.

1. Social impact

We propose a pipeline that leverages 2D motion extracted from video data to enhance 3D motion generation. Our approach relies on human video data to obtain 2D human motion. In this work, we strictly utilize data from open-source 2D video datasets, which helps mitigate privacy concerns typically associated with proprietary or non-public data.

Nevertheless, extending this approach to incorporate videos sourced from the internet may pose privacy risks. Although we only extract abstract 2D motion representations, human motion patterns can be inherently unique and may unintentionally expose identity-related information.

2. Implementation

2.1. Camera embedding

Following [14], we use spherical coordinates [27] to represent the camera pose. As illustrated in Figure 1, we define the human root of each frame as the origin of the coordinate system. Then we can use θ , φ , and r to represent the polar angle, azimuth angle, and the camera relative distance to the human root, respectively.

The camera embedding $C_{rel} \in V \times 4$ encodes the relative poses of each view with respect to the first view. Specifically, it includes the delta polar angle $\Delta\theta_i$, sine and cose of the delta azimuth angle $\sin(\Delta\varphi_i)$ and $\cos(\Delta\varphi_i)$, and the delta distance Δr_i , where i denotes the view index. In our setting, $V = 4$ views are used, with a randomly chosen first

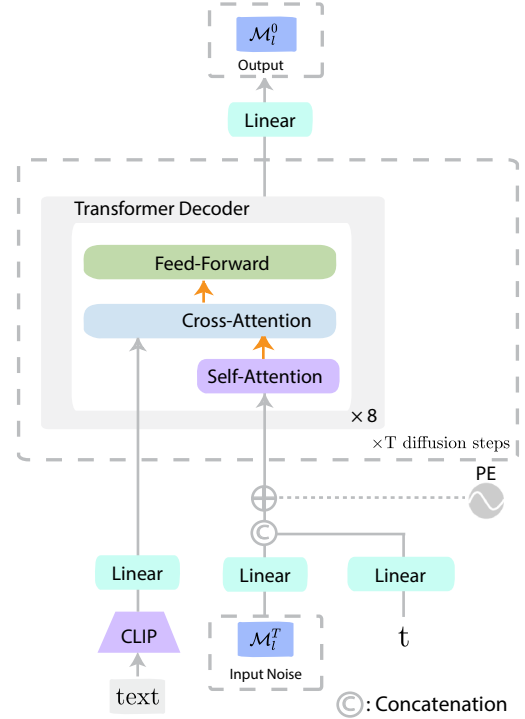


Figure 2. **2D Motion Diffusion model**

view for each sequence. The relative poses remain consistent across all frames. Additionally, we assume the identical r_i and θ for all cameras, effectively turning the camera embedding into a form of positional embedding [22].

In the proposed Multi-view Diffusion model, the camera embedding conditions the generation of multi-view 2D motion. At each diffusion step t , multi-view 2D local motion \mathcal{M}_{vl}^t and root velocities \mathcal{M}_{vr}^t are first processed by an MLP into a latent space, producing $x^t \in \mathbb{R}^{N \times V \times C}$. Then the camera embeddings are similarly projected into the latent space, resulting in $x_{rel} \in \mathbb{R}^{V \times C}$, which is expanded to match input shape $\hat{x}_{rel} \in \mathbb{R}^{N \times V \times C}$. The camera embedding is then added to the latent representation, forming $x^t + \hat{x}_{rel}$. This conditioning ensures that the generated 2D motion in each view aligns with the respective camera pose, maintaining view consistency throughout the generation process.

2.2. Training details

We use *CLIP-VIT-BASE-PATCH32* [18] as the text feature extractor, utilizing all 77 extracted tokens. The 2D Motion Diffusion model consists of 8 transformer decoder layers [22], as shown in Figure 2. Each layer has 4 attention heads



Figure 4. **2D Human Motion Data Extracted from Videos** Here we show the 2D motion data we obtained from videos. The text is the GPT3.5 finetuned version for the last row of frames. We extracted 2D human motion to pre-train the 2D model. The pipeline contains two steps: 1. Human BBOX detection, then track all bboxes using Tracker Class in EasyMocap. Then obtain the human smpl 2D poses with confidence value. 2. Smooth and merge the 2D poses: Merge the post-SmoothNet joints with the original ones as follows: For the fast-moving joints (drumming wrists, kicking legs, etc.), the original joints hold more details and don't jitter. On the other hand, when the joints are not moving, we choose the smoothed joints data.

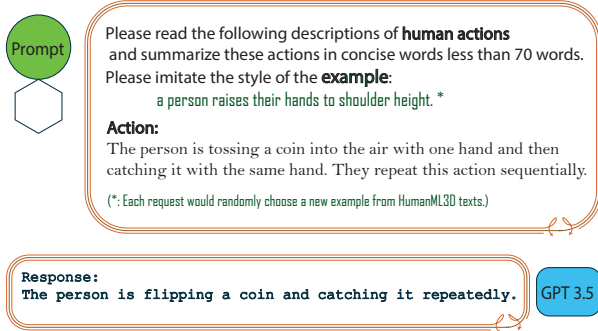


Figure 5. **GPT3.5 Text Prompt**

2.5. Triangulation details

In this section, we detail the triangulation [8, 28] process under the orthographic projection model for reconstructing 3D points from multiple 2D observations. Note that the camera settings during testing are identical to those used in multi-view 2D motion training.

Problem Formulation. Given 2D observations $\mathbf{p}_{2d}^{(v)} \in \mathbb{R}^2$ from V different views, our goal is to estimate the corresponding 3D point position $\mathbf{p}_{3d} \in \mathbb{R}^3$. Each view v has known camera extrinsic parameters, including the rotation matrix $\mathbf{R}^{(v)} \in \mathbb{R}^{3 \times 3}$ and the translation vector $\mathbf{t}^{(v)} \in \mathbb{R}^3$.

Orthographic Projection Model. Under the orthographic projection, the relationship between the 3D point \mathbf{p}_{3d} and its 2D observation $\mathbf{p}_{2d}^{(v)}$ is given by:

$$\mathbf{p}_{2d}^{(v)} = \mathbf{M} \left(\mathbf{R}^{(v)} \mathbf{p}_{3d} + \mathbf{t}^{(v)} \right), \quad (1)$$

where $\mathbf{M} \in \mathbb{R}^{2 \times 3}$ is the projection matrix that extracts the first two components:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (2)$$

Construction of the Linear System. Expanding the Equation 1, we obtain:

$$\mathbf{P}^{(v)} \mathbf{p}_{3d} = \mathbf{p}_{2d}^{(v)} - \mathbf{t}_{2d}^{(v)}, \quad (3)$$

where:

$$\mathbf{P}^{(v)} = \mathbf{M} \mathbf{R}^{(v)} \in \mathbb{R}^{2 \times 3}, \quad (4)$$

$$\mathbf{t}_{2d}^{(v)} = \mathbf{M} \mathbf{t}^{(v)} \in \mathbb{R}^2. \quad (5)$$

For each view, we obtain two linear equations. By stacking the equations from all views, we construct the global linear system:

$$\mathbf{A} \mathbf{p}_{3d} = \mathbf{b}. \quad (6)$$

where:

$$\mathbf{A} = \begin{bmatrix} \mathbf{P}^{(1)} \\ \mathbf{P}^{(2)} \\ \vdots \\ \mathbf{P}^{(V)} \end{bmatrix} \in \mathbb{R}^{2V \times 3}, \quad \mathbf{b} = \begin{bmatrix} \mathbf{p}_{2d}^{(1)} - \mathbf{t}_{2d}^{(1)} \\ \mathbf{p}_{2d}^{(2)} - \mathbf{t}_{2d}^{(2)} \\ \vdots \\ \mathbf{p}_{2d}^{(V)} - \mathbf{t}_{2d}^{(V)} \end{bmatrix} \in \mathbb{R}^{2V}. \quad (7)$$

Least Squares Solution. To estimate the optimal \mathbf{p}_{3d} , we solve the following least squares problem:

$$\min_{\mathbf{p}_{3d}} \|\mathbf{A}\mathbf{p}_{3d} - \mathbf{b}\|_2^2. \quad (8)$$

The closed-form solution is given by:

$$\mathbf{p}_{3d} = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{b}. \quad (9)$$

Final results. Once the 3D motion is calculated, we normalize it using the height of the neutral SMPL-X model [17], with all shape parameters set to zero. This ensures the 3D motion approximates the scale of real-world human proportions.

3. Evaluation

3.1. Evaluation protocols

For the results presented in the main paper, we directly adopt the metrics for baseline methods [2, 4, 12, 21] as reported in their original papers. For visualization, we utilize their open-sourced code to generate results. For our method, since we generate motions at 30 FPS, we downsample the results to 20 FPS to align with the evaluation protocols of other methods. For comparison with MAS [10], we use our 2D Motion Diffusion model as the motion generator and follow their inference pipeline. For comparison with MotionBERT [30], we evaluate using their released models.

3.2. User study

We invite 56 participants from various institutions to evaluate the generated motions. Each participant was presented with motions generated from 15 novel text prompts and was asked to select the best and second-best motions from each group.

For comparison, we include MDM [21] and MLD [4] as baselines. However, MAA [2] and OMG [12] are excluded due to the unavailability of their implementations. Specifically, MAA [2] has not provided any publicly available code. As for OMG [12], although the authors claim their project would be open-sourced, they have yet to release either the code or the dataset used in their experiments. Additionally, we excluded open-vocabulary methods like MotionCLIP [20] and AvatarCLIP [9] because their lack of global translation would make them easily distinguishable

Methods	R-Precision \uparrow	FID \downarrow	MM Dist \downarrow	Diversity \rightarrow
Real	0.797	0.002	2.974	9.503
w/o translation	0.697	3.001	3.898	8.176

Table 2. **Impact of global translation.**

Methods	R-Precision \uparrow	FID \downarrow	MM Dist \downarrow	Diversity \rightarrow
Real	0.797	0.002	2.974	9.503
$\theta = 70^\circ$	0.595	3.732	4.191	8.666
$\theta = 110^\circ$	0.679	0.697	3.687	8.897
Ours ($\theta = 85^\circ$)	0.702	0.328	3.547	9.072

Table 3. **Ablation study of the polar angle.** The best and second-best results are highlighted **green** and **yellow**.

by participants. To ensure fairness, we shuffled the order of results from all methods being compared.

Out of 56 responses, 49 were considered valid, with incomplete questionnaires excluded. Participants were required to complete the evaluation form shown in Figure 6.

3.3. Impact of global translation

As shown in Table 2, we observe that removing the global translation from ground-truth data degrades performance. This highlights the critical role of global translation in generating realistic 3D human motion.

Global translation captures the overall movement of a person in space, which is essential for modeling coherent and lifelike motion. Without it, the generated motions lack the natural flow and positional context necessary for realism, leading to a noticeable decline in quality.

3.4. More ablation study

Visualize model design. The visual results are shown in Figure 7. We could observe that without pretraining, the generated results are not realistic and the consistency block imposes a very strong constraint and degrades the results. We also observe that using 4 views achieves better alignment with the text prompts.

Different polar angle. As shown in Table 3, we conduct an ablation study to explore the effect of different polar angles θ for synthetic cameras in the multi-view setting. Our results indicate that $\theta = 85^\circ$ yields the best performance. We believe the reason is that, in real-world scenarios, cameras are typically positioned slightly above the human subject, and $\theta = 85^\circ$ closely mimics this common setup.

Without real 2D motion data. In our training, we incorporate both synthetic 2D data sampled from the HumanML3D dataset [7] and real 2D motion data from EgoExo4D [6] and Motion-X++ [13]. As shown in Table 4, removing real 2D motion data results in worse FID scores.

This suggests that real 2D motion data provides crucial information for the model, enabling it to learn more robust

User Study for Text-to-Motion Task

Here we show you 15 groups of Text-to-Motion Generation results.
Each contains 3 human motions, generated by 3 methods individually. They are randomly arrayed.

There are 15 sections in total and for each section you only need to do 2 multiple choice:

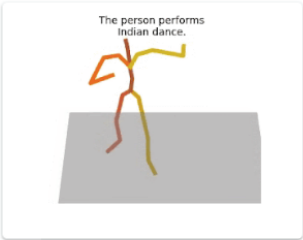
The first question is attached with the videos of motions, and you should choose the best motion within the options. The following question asked you to pick out the second best motion from the same group.

If you are not sure about the "good motion" standard. You can start from two perspectives:
(1) How realistic it looks? Is it natural and human-like? (2) How well is the motion matched with the text prompt?

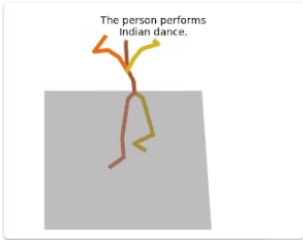
1: The person performs Indian dance

Choose the best motion, then the second best one.

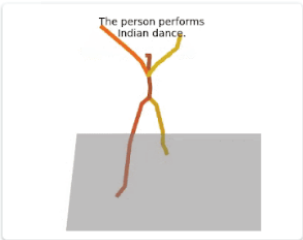
1-1: Which is the best one



☐ 1



☐ 2



☐ 3

1-2: Which is the second best one

☐ 1

☐ 2

☐ 3

Back
Next
Page 2 of 16
Clear form

Figure 6. **User Study Google Form.** We use Google Forms [5] to collect user study results on text-to-motion generation. Each generated result is presented as a continuously looping video.

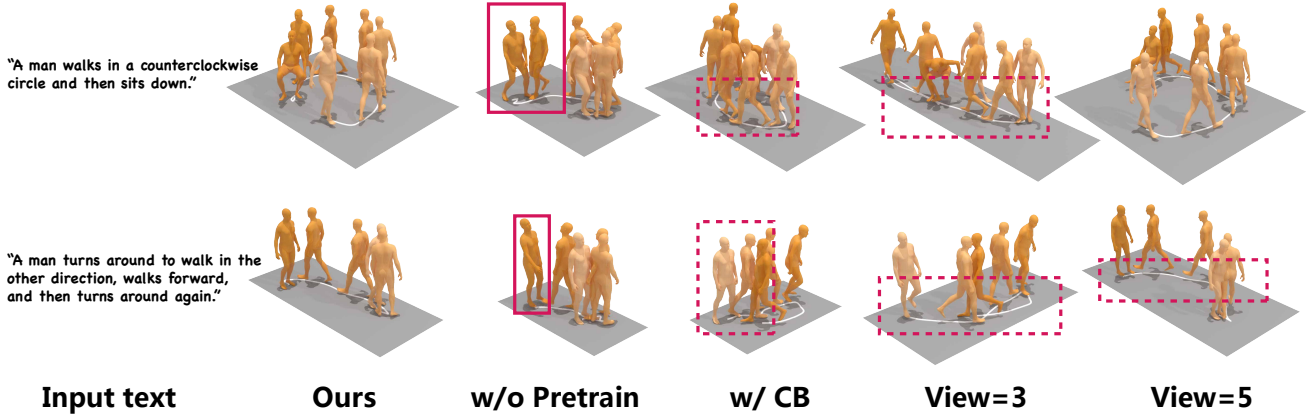


Figure 7. **Qualitative results of ablation study.** Our full model generates more natural motion than the ablations. The unnatural poses are highlighted in the red boxes. The semantics misalignment is highlighted in the dashed boxes.

Methods	R-Precision \uparrow	FID \downarrow	MM Dist \downarrow	Diversity \rightarrow
Real	0.797	0.002	2.974	9.503
w/o 2D motion from video data	0.620	1.448	4.084	9.494
Ours	0.702	0.328	3.547	9.072

Table 4. **Ablation study of 2D motion extracted from video data.** The best and second-best results are highlighted **green** and **yellow**.

Methods	FID \downarrow	Parameters (M) \downarrow	Speed (seconds) \downarrow
MDM ($T = 1000$)	0.544	17.05	3.82
MAS ($T = 100$)	11.893	24.86	1.15
Ours ($T = 100$)	0.328	34.89	0.54

Table 5. **Parameters and inference speed.** T denotes the diffusion steps. The best and second-best results are highlighted **green** and **yellow**.

and accurate representations.

3.5. Parameters and speed

We present the model parameters and inference speed in Table 5. The evaluation is conducted on a system with an Intel(R) Core(TM) i9-14900K CPU and a single NVIDIA GeForce RTX 4090 GPU with 24 GB of memory. Inference speed is measured using a batch size of 1, generating 150 frames of motion.

4. Discussion

4.1. Limitations

Noise in extracted 2D motion. 2D motion extracted from videos may contain noise and jitter. Although we filter out joint motions with low confidence, some high-confidence joints can still contain errors. These errors are not always

purely high variance; they may also introduce biases that could mislead the model into learning incorrect motion patterns. Such biases could negatively impact the quality and generalizability of the generated 3D motion. Addressing this limitation would require more advanced filtering techniques or robust learning strategies to mitigate the influence of noisy or biased data.

Motion space. We demonstrate that our method is capable of generating motions not included in the HumanML3D [7] dataset, such as juggling and layup shots. This indicates that our model can learn and generate a broader range of motions by leveraging 2D motion data. However, the generated motions are limited to those that exist within the 2D motion data seen during training. For motions entirely absent from both 2D and 3D datasets, the model lacks the necessary priors and, therefore, cannot generate such unseen motions.

Explicit control. Although our multi-view representation effectively leverages 2D data, it poses challenges for precise control over the generated motions. For instance, providing specific joint trajectories or other detailed 3D conditions is difficult to implement within this representation, as the multi-view 2D approach lacks a direct mechanism to incorporate precise 3D constraints. This limitation suggests that while our method excels at generating diverse and realistic motions, integrating precise 3D control remains an area for future improvement.

Physical plausibility. Similar to other kinematics-based methods [21], our approach may still produce physically implausible motions, such as floating or penetration issues. The floating mainly comes from our root velocity predictions, where the model is unaware of the real floor. This issue is also observed in other velocity representation meth-

ods like [19]. The penetration stems from the lack of explicit physical constraints during motion generation. To address this, one potential solution is to apply physics-based methods [29], such as motion tracking on the generated results to ensure physical plausibility. This post-processing step could refine the motions and make them more suitable for applications requiring strict adherence to physical laws.

Maximum length. In this work, we follow the previous setting and limit the maximum length of generated motions to 10 seconds. While this constraint aligns with the experimental setup, generating infinite-length motions remains an open challenge. Addressing this would require exploring strategies such as autoregressive generation, temporal stitching, or recurrent models that can seamlessly extend motion sequences without compromising consistency or quality.

4.2. Future work

More 2D motion. In this work, we utilized 97.63 hours of 2D motion data, which is significantly more than the 3D data used in our experiments. However, this amount is still relatively small compared to the scale of data commonly used in video generation tasks [3]. Incorporating larger and more diverse 2D motion datasets could further enhance our method’s ability to generate a wider variety of motions, potentially expanding its applicability to more complex and nuanced motion types.

Hand motion. Generating realistic hand motion poses significant challenges [1], as it requires precise motion capture equipment to record intricate finger movements accurately [16, 23]. This makes collecting high-quality hand motion data particularly difficult. Incorporating hand motion into our framework could greatly enhance its applicability, especially for tasks requiring detailed finger movements. Exploring methods to seamlessly integrate hand motion, such as by stitching finger actions onto existing body motions, would be a meaningful direction for future work.

Object interaction. Capturing the motion of humans interacting with objects is inherently challenging due to the complexity of such interactions and the difficulty of obtaining high-quality motion capture data. However, object interaction is an essential aspect of human daily activities, making it a critical area for further exploration. Incorporating object interaction into motion generation frameworks could significantly enhance the realism and applicability of generated motions, paving the way for more comprehensive and context-aware human motion synthesis.

References

- [1] Easymocap - make human motion capture easier. Github, 2021. 7
- [2] Samaneh Azadi, Akbar Shah, Thomas Hayes, Devi Parikh, and Sonal Gupta. Make-an-animation: Large-scale text-conditional 3d human motion generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15039–15048, 2023. 4
- [3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 7
- [4] 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. *arXiv e-prints*, art. arXiv:2212.04048, 2022. 4
- [5] Google. Google forms. <https://forms.google.com>, 2024. 5
- [6] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400, 2024. 2, 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 *CVPR*, 2022. 2, 4, 6
- [8] Richard I. Hartley and Peter Sturm. Triangulation. *Computer Vision and Image Understanding*, 68(2):146 – 157, 1997. 3
- [9] Fangzhou Hong, Mingyuan Zhang, Liang Pan, Zhongang Cai, Lei Yang, and Ziwei Liu. Avatarclip: zero-shot text-driven generation and animation of 3d avatars. *ACM Trans. Graph.*, 41(4), 2022. 4
- [10] Roy Kapon, Guy Tevet, Daniel Cohen-Or, and Amit H Bermano. Mas: Multi-view ancestral sampling for 3d motion generation using 2d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1965–1974, 2024. 2, 4
- [11] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, 2014. 2
- [12] Han Liang, Jiacheng Bao, Ruichi Zhang, Sihan Ren, Yuecheng Xu, Sibe Yang, Xin Chen, Jingyi Yu, and Lan Xu. Omg: Towards open-vocabulary motion generation via mixture of controllers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 482–493, 2024. 4
- [13] Jing Lin, Ailing Zeng, Shunlin Lu, Yuanhao Cai, Ruimao Zhang, Haoqian Wang, and Lei Zhang. Motion-x: A large-scale 3d expressive whole-body human motion dataset. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 4
- [14] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023. 1
- [15] OpenAI. Openai: Introducing chatgpt. <https://openai.com/blog/chatgpt>, 2022. 2
- [16] OptiTrack. Optitrack. <https://www.optitrack.com/>. 7
- [17] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, 2019. 4
- [18] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1
- [19] Soyong Shin, Juyong Kim, Eni Halilaj, and Michael J Black. Wham: Reconstructing world-grounded humans with accurate 3d motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2070–2080, 2024. 7
- [20] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *ECCV*, 2022. 4
- [21] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *ICLR*, 2023. 4, 6
- [22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 1
- [23] Vicon. Vicon. <https://www.vicon.com/>. 7
- [24] Wikipedia. Orthographic projection — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Orthographic_projection, 2024. 2
- [25] Wikipedia. Perspective (graphical) — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/wiki/Perspective_\(graphical\)](https://en.wikipedia.org/wiki/Perspective_(graphical)), 2024. 2
- [26] Wikipedia. Pinhole camera — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Pinhole_camera, 2024. 2
- [27] Wikipedia. Spherical coordinate system — Wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Spherical_coordinate_system, 2024. 1
- [28] Wikipedia. Triangulation (computer vision) — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/wiki/Triangulation_\(computer_vision\)](https://en.wikipedia.org/wiki/Triangulation_(computer_vision)), 2024. 3
- [29] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. Physdiff: Physics-guided human motion diffusion model. In *ICCV*, 2023. 7
- [30] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15085–15099, 2023. 4