

# Lifting Motion to the 3D World via 2D Diffusion

## Supplementary Material

The supplementary material includes details on the implementation, evaluation, data preparation, and limitations. We also encourage readers to watch our supplementary video and [project page](#) for more qualitative results.

### 1. Implementation Details

**Model Architectures.** Our denoising network in Stage 1 (line-conditioned 2D motion diffusion) employs a transformer-based architecture with four self-attention blocks. Each block consists of a multi-head attention layer and a position-wise feed-forward layer. The self-attention layer uses four attention heads, and the dimension of the key, query, and value is 256. The model is trained on motion data with a window size of 120 at 30 fps.

For the denoising network in Stage 4 (multi-view 2D motion diffusion), we adopt the same model architecture, with an added cross-view attention layer in each block. Similar to the positional embeddings in the transformer model, we use sinusoidal encoding to generate view embeddings, which are added before each cross-view attention layer to distinguish different views.

**Details of Multi-View 2D Sequence Optimization.** In Stage 2, we use the Adam optimizer with a learning rate of 0.001 to optimize 2D pose sequences across five different views. The optimization process requires 5000 steps. At each step, we randomly sample a noise level from  $[1, 999]$  for optimization using Score Distillation Sampling (SDS).

**Data Processing of 2D Pose Sequences.** A 2D pose sequence represents a sequence of 2D keypoint positions. In Stages 1 and 4, we conduct a data normalization for 2D pose sequences. We normalize each 2D sequence through two steps: (1) translating the entire sequence based on the root joint position of the first frame, and (2) computing a scale factor that adjusts the first pose’s bounding box size to fall within a target range  $[s_{min}, s_{max}]$ , then applying this same scale factor to all frames in the sequence. For the 2D pose input, we center the sequence on the first frame’s root joint and detect invisible keypoints to create a visibility mask that is consistently applied across all camera views. With our virtual cameras positioned strategically around the centered root, the majority of pose keypoints typically remain visible from all views throughout our standard 120-frame motion sequence window.

**Details of Multi-View 2D Motion Diffusion Model.** The “fixe” camera setup in our approach refers to the predefined relative angles between our predicted auxiliary views ( $90^\circ$ ,  $180^\circ$ ,  $270^\circ$  relative to the input view), rather than constraining the absolute camera viewpoint in world space. This

design ensures that MVLift can effectively process 2D pose sequences captured from any arbitrary viewpoint, eliminating restrictions on the initial camera positioning during capture.

**Optimization Efficiency in Stage 2.** Our Stage 2 optimization takes 10 minutes for a batch (32 sequences) to generate 5 views. Sequences can be optimized in parallel, enabling scalable processing for generating a synthetic 3D motion dataset in Stage 3.

### 2. Evaluation Details

**Baseline Details.** MAS [2] was originally designed for the unconditional 3D motion synthesis task without relying on 3D ground truth motion data. To adapt MAS to our setting, where 3D motion is predicted from a 2D pose sequence, we modified the optimization objective for a single camera view and replaced the 2D pose sequence in the selected view. Additionally, we increased the optimization weight for the input view to encourage better alignment with the input 2D sequence, as suggested by the authors of MAS.

**FID Evaluation for 2D Pose Sequences.** For a generated 3D motion sequence to be realistic, the reprojected 2D pose sequences in different views should appear natural and realistic. Therefore, for evaluation datasets without available 3D ground truth motion (Steezy, NicoleMove, CatPlay), we introduce FID evaluation for 2D pose sequences reprojected from the generated 3D motion of each approach. Prior work on text-to-motion synthesis tasks [1] proposed training a motion feature extractor using an autoencoder. Similarly, we train an autoencoder for 2D pose sequences to represent a window of motion as latent vectors. This autoencoder consists of an encoder and a decoder, both based on a temporal convolution model architecture. During training, we use a reconstruction loss and a latent vector sparsity loss, following previous work.

### 3. Dataset Details

**AIST++.** We used a subset of AIST++ [5]. While the original dataset provides 10 camera views per sequence, we randomly selected one view per sequence for our experiments. Following the original paper’s data split, we used 611 sequences for training and 360 sequences for testing.

**Steezy.** We utilized dance videos from prior work [3]. The dataset was randomly split into 726 videos for training and 81 videos for testing. Since Steezy videos typically feature multiple dancers, we manually verified the 2D pose

sequences extracted by ViTPose for the testing set. We discarded sequences showing non-dancing audience members, resulting in 33 high-quality test sequences with durations ranging from 20 seconds to 2 minutes.

**NicoleMove.** We collected videos from the YouTube channel “Move With Nicole” [6]. Each video was clipped into 10-second segments. The dataset was randomly split into 13,212 sequences for training and 1,468 sequences for testing. We used ViTPose [7] to obtain 2D keypoints and generate 2D pose visualizations for each sequence. For the testing set, we manually verified and filtered the sequences, discarding those with noisy predictions or redundant similar motions. This curation process resulted in a representative testing set of 106 sequences.

**CatPlay.** We captured monocular videos of 2 subjects playing with a cat teaser, with each video featuring a single cat. This dataset was collected in an indoor environment. We captured 20 videos, each lasting 10 to 20 minutes. For each video, we first clipped them into 5-second segments. Then we used an advanced keypoint estimation approach [8] to generate 2D keypoints and obtain 2D pose visualizations for each clip. We manually verified each 2D sequence visualization and discarded those where the subject was out of the camera view or had very noisy predictions. Finally, we obtained 343 sequences for training and 82 sequences for testing.

**OMOMO.** OMOMO [4] is a human-object interaction dataset containing motions with 15 different objects. For our experiments, we focused on interactions with a *largebox* object. The dataset provides 5 marker positions captured during motion capture sessions. Following the original data split, we used sequences from 15 subjects for training (896 sequences) and 2 subjects for testing (119 sequences).

## 4. Limitations

While our approach demonstrates superior results across various datasets, there are some limitations. First, our current problem formulation requires the 2D pose sequence to be captured in a static camera setting. Second, we do not currently apply any physics-based constraints, which means artifacts such as feet-floor penetration and feet floating cannot be entirely prevented. Addressing these limitations presents interesting and promising directions for future work.

## References

- [1] 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. 1
- [2] Roy Kapon, Guy Tevet, Daniel Cohen-Or, and Amit H Bermano. Mas: Multi-view ancestral sampling for 3d motion generation using 2d diffusion. In *CVPR*, 2024. 1
- [3] Jiaman Li, Yihang Yin, Hang Chu, Yi Zhou, Tingwu Wang, Sanja Fidler, and Hao Li. Learning to generate diverse dance motions with transformer. *arXiv preprint arXiv:2008.08171*, 2020. 1
- [4] Jiaman Li, Jiajun Wu, and C Karen Liu. Object motion guided human motion synthesis. *ACM Trans. Graph.*, 42(6), 2023. 2
- [5] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++. In *ICCV*, 2021. 1
- [6] Nicole Pearce. Move with nicole, 2024. YouTube Channel, accessed March 2024. 2
- [7] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vit-pose: Simple vision transformer baselines for human pose estimation. *NeurIPS*, 2022. 2
- [8] Jie Yang, Ailing Zeng, Ruimao Zhang, and Lei Zhang. Unipose: Detecting any keypoints. *arXiv preprint arXiv:2310.08530*, 2023. 2