Mocap-2-to-3: Multi-view Lifting for Monocular Motion Recovery with 2D Pretraining

Supplementary Material

1. Frequently Asked Questions

1. **Why absolute position estimation is necessary?**
   We aim to reconstruct metrically precise humans from the real physical world, which is crucial for downstream applications such as reasoning about human–human, human–object, and human–scence interactions. While previous methods can recover global trajectories after initial frame alignment, they lack correct **scale** information and are therefore limited to observing human motion in isolation. For example, when inferring different individuals, the estimated 3D height of a child might exceed that of an adult, making it impossible to accurately determine interaction states between people, their relative positioning, or their interactions with the environment. To address this, our framework is designed to recover absolute depth from monocular input while maintaining action accuracy comparable to state-of-the-art methods.
2. **Does the approach's reliance on calibrated camera parameters, while other baselines do not require camera information, raise concerns about fairness?**
   Monocular vision cannot determine absolute depth, and we need camera parameters to assist in prediction, which is the same setting as other methods for recovering absolute pose, such as [15, 20]. To ensure fair comparison, we replaced the estimated camera parameters in the baselines with ground-truth values, as indicated by the † **symbol in the table**. In addition, existing SLAM-based methods can estimate camera parameters and may serve as an alternative. However, given that our objective is to recover absolute poses in the physical world with the highest possible accuracy, and that camera calibration is not difficult in our application setting, we do not recommend introducing unnecessary errors by replacing calibration with SLAM-based estimation.
   Regarding calibration errors in real-world camera setups and their impact on the system, our test set is also calibrated using a checkerboard, which already incorporates the typical level of calibration error and is therefore equivalent to real-world applications.
3. **Does the inclusion of the ground plane limit applicability, and do the results depend on ground plane input?**
   The ground plane is calculated from camera parameters and does not require additional point cloud devices for acquisition, unlike [15] (which also estimates absolute human motion). This imposes no additional burden in practical applications, which is our advantage. Furthermore, Pointmaps is a plug-and-play module, and we demonstrate that it can assist in multi-view prediction. It can be applied to any multi-view task with similar requirements. The role of pointmaps is to help the absolute trajectory converge faster, but it does not aid the network in learning motion details (local pose). Since MPJPE and PA-MPJPE evaluate local pose performance, after removing root influence, our motion framework still maintains an advantage in learning fine-grained motion details.
4. **Is obtaining camera parameters and relying on a fixed camera setup less flexible than the methods like WHAM or GVHMR?**
   The goal of this work is not merely to enhance the quality of relative trajectory estimation, but to tackle a more fundamental task: accurately estimating both human motion and absolute positioning in long-term behavior analysis scenarios where cameras are typically fixed. While previous approaches like WHAM and GVHMR provide more flexible camera motions by estimating relative positions, their inability to recover correct human scale creates significant practical limitations. Our comparison with state-of-the-art human reconstruction methods demonstrates that our approach achieves comparable motion accuracy while properly recovering absolute positioning. While both approaches study motion estimation, their scopes differ fundamentally: tasks with moving cameras require only relative trajectory estimation, while monocular absolute position recovery applies when cameras are fixed and poses are known.
   Importantly, our framework offers two main contributions. If absolute positioning is not required, the corresponding modules (pointmaps and global movement) can be removed, eliminating the need for camera parameters as input and enabling the prediction of relative poses instead. Nevertheless, the architecture still allows 2D data to enhance 3D motion performance, addressing one of the major limitations in 3D prediction—OOD generalization. This architectural design holds significant potential for the future development of 3D motion recovery. Our flexibility lies in the ability to **leverage a wider range of data** and to accommodate more diverse keypoint formats, which are limitations in existing methods.
5. **Although data augmentation is used in the second stage, is the improvement in motion diversity still limited by the availability of 3D data?**

The second stage mainly learns multi-view consistency and global information, while diversity and priors are learned in the first stage, which is also a key innovation of our approach. Our early experiments show that diversity is highly related to the first stage—for example, even if the 3D training data only includes walking, the model can still learn to run if running data is added during 2D pretraining.

2. Related Works

2.1. Multi-view Structures

A large body of work in multi-view image generation frames 3D prediction as producing view-consistent 2D renderings across cameras. Zero-1-to-3 [11] predicts a novel view from a single input image and a specified relative pose, seeding the field with pose-conditioned novel-view synthesis. Subsequent methods train multi-view diffusion models that generate multiple views jointly to enforce cross-view consistency; representative examples include MVDream [17], SyncDreamer [12], and Wonder3D [13], with further extensions to multi-view video generation [9]. Diffusion-based architectures excel at modeling complex feature distributions and produce diverse yet coherent samples across viewpoints, offering clear advantages over deterministic regression backbones. After pretraining on large 3D object datasets [3], these models produce geometrically consistent results across viewpoints.

Inspired by these multi-view designs, we reformulate complex 3D human motion as multi-view 2D motion synthesis. Within this framework, we (1) introduce a pretraining stage to learn rich 2D priors, and (2) decouple video and pose, using only pose as input. This lets us apply arbitrary 3D-data–driven augmentations while avoiding image supervision, substantially improving data efficiency and coverage.

3. Model Architectures

3.1. Definition of world coordinate system

In the training set, the world coordinate system is defined either with the origin at the person's position in the first frame or at a fixed location at the center of the area (Fig.1 (a)). To establish a unified representation of the world coordinate system, we transform it to the ground plane of the current inference viewpoint $\mathcal{V}_0$. The world coordinate system follows the right-hand rule, as shown in Fig.1 (b), where green represents the y-axis, blue the x-axis, and red the z-axis. We rotate the y-axis (by angle θ) r° such that the z-axis points toward the optical center of camera V_0, with the origin positioned below the camera's focal point. The x-z-axes remain parallel to the ground, and $y = 0$ denotes the ground plane. We denote this transform as $T_w^{\mathcal{V}_0}$. This transformation is

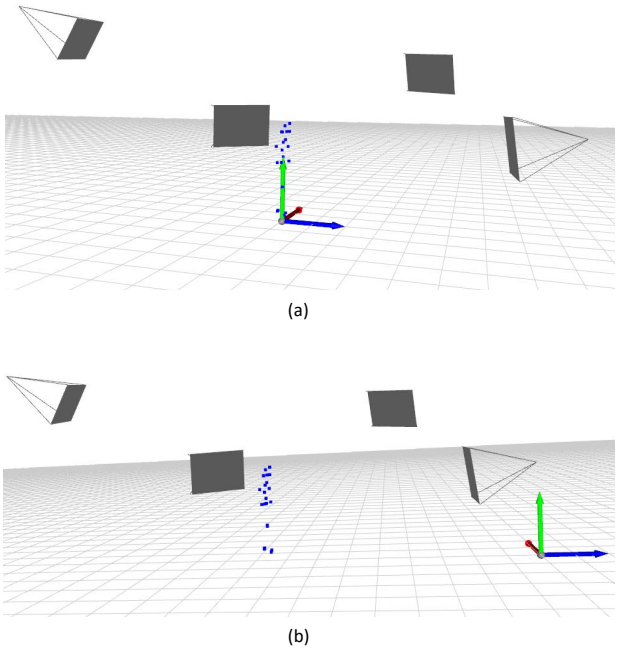

Figure 1. (a) Original world coordinate system. (b) The transformed world coordinate system is positioned beneath the primary camera $\mathcal{V}_0$.

then applied to each camera and motion sequence.

$$
\begin{aligned}
R &= F(r, 0, 0), \\
t &= (t_x, 0, t_z), \\
T_w^{\mathcal{V}_0} &= \{R, t\}, \\
\mathcal{V}_i &= \mathcal{V}_i^o \cdot {T_w^{\mathcal{V}_0}}^{-1}, \\
\mathcal{W}_{3d} &= \mathcal{W}_{3d} \cdot {T_w^{\mathcal{V}_0}}^{-1}.
\end{aligned} \tag{1}
$$

Let $F(\cdot)$ denote the transformation from Euler angles to a rotation matrix, $\mathcal{V}_i^o$ represent the original pose of the i-th camera, and $\mathcal{W}_{3d}^o$ odenote the original 3D coordinates of the sequence. The transformed coordinate system determines the observation viewpoint directly from camera parameters during training and inference, facilitating a standardized unified representation.

3.2. Virtual Multi-view System

Our benchmark comprises K cameras $\mathcal{V}_{0:K}$. During the pretraining phase, for 3D projection data, we randomly sample one camera from the K available cameras for projection at each batch, while the 2D data incorporates samples from all K cameras. In the fine-tuning phase, our multi-view system consists of 4 views: one primary camera $\mathcal{V}_0$ and three virtual cameras (sampled from the K cam-

eras). This ensures both the observation camera and virtual cameras leverage the motion priors learned during pretraining, with fine-tuning focusing exclusively on learning multi-view geometric consistency.

For instance, in industrial applications, we deploy K monitoring sites, each equipped with a single camera. Our core idea is to effectively utilize monocular cameras from different sites to form a virtual multi-view system, rather than deploying multiple cameras in a single site, which is resource-intensive. In this setup, each site's monocular camera serves as the primary camera, while cameras from other sites act as virtual auxiliaries. Since each site can collects 2D data, every camera inherently learns a prior mapping from image space to 2D pose during pretraining. During fine-tuning, cameras from other sites can form virtual multi-view systems with the current observation site's camera. When significant scale differences between sites prevent overlapping observation ranges, we manually configure virtual cameras that participate in training during both phases. This design enables the model to focus on learning motion-to-camera mapping in the first stage. In the second stage, where each view already possesses motion priors, it only needs to additionally learn cross-view geometric consistency and standard human skeletal proportions. This framework allows motion generalization to be achieved by merely collecting 2D data from each viewpoint.

3.3. Pointmap calculation

For any dataset where the world coordinate system is grounded on the ground plane and the camera parameters are known, we compute the ground plane equation. This equation is then converted into a more intuitive pointmap representation. Pointmaps represent the (x_w, y_w, z_w) values in the world coordinate system corresponding to each pixel (u, v) on the image I from any given viewpoint $\mathcal{V}$. The ground plane is defined as $y = 0$ in the world frame and can be transformed into any camera view as $\pi : ax + by + cz + d = 0$, where a, b, c, d are the plane coefficients. Following [8], for any pixel (u, v), we compute the projected depth z_c on the plane using π and camera intrinsics:

$$\begin{cases} ax + by + cz + d = 0, \\ f_x \dfrac{x}{z} + c_x = u, \\ f_y \dfrac{y}{z} + c_y = v. \end{cases} \tag{2}$$

Where f_x and f_y are the focal lengths of the camera, and c_x and c_y are the principal point offsets. The depth z_c can be calculated as:

$$z_c = \frac{-d}{\frac{a(u-c_x)}{f_x} + \frac{b(v-c_y)}{f_y} + c}. \tag{3}$$

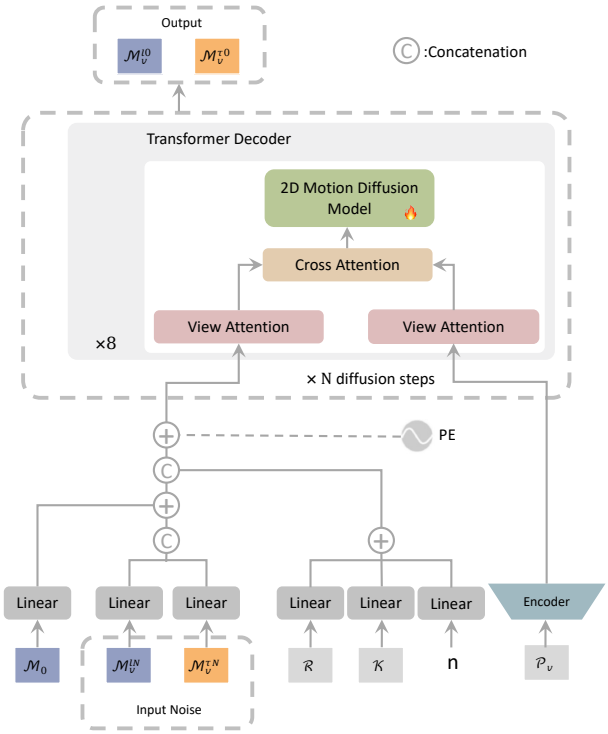

Figure 2. **Conditional Multi-view Diffusion model.** The model takes $\mathcal{V}_0$ 2D motion sequence as conditional input, where pointmaps and motion sequences separately undergo View Attention to learn multi-view relationships, followed by Cross Attention to guide motion generation. The 2D Motion Diffusion Model is initialized with weights pretrained on 2D data.

Given (u, v) and z_c, the corresponding coordinates in the camera coordinate system (x_c, y_c) can be computed for any pixel:

$$\begin{aligned} x_c &= \frac{(u - c_x) \cdot z_c}{f_x}, \\ y_c &= \frac{(v - c_y) \cdot z_c}{f_y}. \end{aligned} \tag{4}$$

The coordinates (x_c, y_c, z_c) are transformed into the world coordinate system (x_w, y_w, z_w) using the camera extrinsics. In this way, we obtain the corresponding world coordinate (x_w, y_w, z_w) for each pixel (u, v) in the image I, and all such points together form the pointmap $\mathcal{P}$.

Note that the computed ground is virtual and makes no assumption that the person is standing on it; this formulation ensures that motions on non-flat ground remain unaffected. We refer the reader to our video demo for detailed visualizations.

3.4. Conditional Diffusion Model

The diffusion model is defined as predicting motion sequences across all viewpoints under given conditions $p_\theta(\mathcal{M}_{0:4}|\mathcal{C}, n)$,where θ is the parameter learned by the network, n is the time step. The condition $\mathcal{C} = [\epsilon, \mathcal{M}_0, \mathcal{K}, \mathcal{RT}, \mathcal{P}]$, includes the 2D motion sequence $\mathcal{M}_0$ from viewpoint V_0, camera extrinsic parameters $\mathcal{RT}$, intrinsic matrix $\mathcal{K}$ and pointmaps $\mathcal{P}$.

The camera extrinsic $\mathcal{RT} = \{\psi, \phi, t_y\}$, where ψ denotes the pitch angle (rotation about the x-axis), ϕ represents the roll angle (rotation about the z-axis) and t_y represents camera height.The camera intrinsics $\mathcal{K} = \{f_x, f_y, c_x, c_y\}$, where $f.$ denotes the camera focal length and $c.$ represents the principal point offset. The pointmaps are first downsampled from the original image to 224×224 resolution, with $\mathcal{P} \in \mathbb{R}^{v\times 224\times 224\times 3}$, then processed through a ResNet for feature extraction $\tilde{\mathcal{P}} = Res(\mathcal{P})$. The View Attention layer performs self-attention across different views at the same (t, j) location, ensuring cross-view consistency. The structure of the Conditional Multi-view Diffusion model is shown in Fig.2.

3.5. Inference details

The denoising process contains N (in this paper $N = 100$) iterative steps, where the initial noise is produced through the projection of randomly sampled 3D noise ϵ_{3d} onto the observation space.

This ensures geometric consistency in the initialization. At each timestep n, we transform the predicted local poses $\mathcal{M}_v^{ln}$ of each view into global joint coordinates $\mathcal{M}_v^{gn}$ using Eq. (5).

$$\begin{aligned} \mathcal{M}^g_{v,\{1:J\}} &= \mathcal{M}^l_v \cdot s_v + \tau_v, \\ \mathcal{M}^g_v &= [\tau_v\ , \mathcal{M}^g_{v,\{1:J\}}]. \end{aligned} \tag{5}$$

The 3D motion $\mathcal{M}^n_{3d}$ in world coordinates is obtained through triangulation, which is then projected to all camera views. To ensure multi-view geometric consistency at every timestep, we subsequently recompute the local pose $\tilde{\mathcal{M}}_v^{ln}$ and global movement $\tilde{\mathcal{M}}_v^{\tau n}$. Finally, this information is used to update step $n-1$.

$$\begin{aligned} \widehat{\mathcal{M}}_v^{gn} &= proj(\mathcal{W}^n_{3d}), \\ \tilde{\mathcal{M}}_v^{ln} &= norm(\widehat{\mathcal{M}}_v^{gn}), \\ \tilde{\mathcal{M}}_v^{\tau n} &= bbox(\widehat{\mathcal{M}}_v^{gn}), \\ \tilde{\mathcal{M}}_v &= \left\{\tilde{\mathcal{M}}_v^{ln}, \tilde{\mathcal{M}}_v^{\tau n}\right\}. \end{aligned} \tag{6}$$

Where $proj(\cdot)$ denotes the projection function, $norm(\cdot)$ computes normalized coordinates within bounding boxes, and $bbox(\cdot)$ calculates both the root joint's global coordinates and bounding box scale. Crucially, at each timestep, we update per-view local pose and global movement using the world-coordinate 3D motion $\mathcal{W}^n_{3d}$, thereby strictly enforcing multi-view consistency. The overall multi-view diffusion denoising inference process is illustrated by the following pseudocode.

Algorithm 1 Inference for Conditional Multi-view Diffusion

Require: Timesteps N; condition $\mathcal{C} = [\epsilon, \mathcal{M}_0, \mathcal{K}, \mathcal{RT}, \mathcal{P}]$
Ensure: World-coordinate 3D motion $\mathcal{W}^0_{3D} \in \mathbb{R}^{T\times J\times 3}$

$\tilde{\mathcal{P}} \leftarrow \text{ResNet-18}(\mathcal{P})$
Sample ϵ_{3D} and set $\mathcal{M}^N \leftarrow \text{repro}(\epsilon_{3D})$
for $n = N-1, \ldots, 0$ **do** ▷ reverse denoising
  $\{\mathcal{M}_v^{\ell n},\ \mathcal{M}_v^{\tau n}\}_{v=0}^{V-1} \leftarrow \mathcal{D}_{mv}(\epsilon, \mathcal{M}_0, \mathcal{K}, \mathcal{RT}, \tilde{\mathcal{P}}, n)$
  $\mathcal{M}_v^{gn} \leftarrow g(\mathcal{M}_v^{\ell n}, \mathcal{M}_v^{\tau n})$ by Eq. (5), $\quad \forall v$
  $\mathcal{W}^n_{3D} \leftarrow \text{triangulate}(\{\mathcal{M}_v^{gn}\}_v, \mathcal{K}, \mathcal{RT})$
  for each view $\sqsubseteq$ **do**
    $\widehat{\mathcal{M}}_v^{gn} \leftarrow \text{proj}(\mathcal{W}^n_{3D}, \mathcal{K}_v, \mathcal{RT}_v)$
    $\tilde{\mathcal{M}}_v^{\ell n} \leftarrow \text{norm}(\widehat{\mathcal{M}}_v^{gn})$
    $\tilde{\mathcal{M}}_v^{\tau n} \leftarrow \text{bbox}(\widehat{\mathcal{M}}_v^{gn})$
  end for
  $\mathcal{M}^{n-1} \leftarrow \{\tilde{\mathcal{M}}_v^{\ell n}, \tilde{\mathcal{M}}_v^{\tau n}\}_v$ ▷ inputs for next step
end for
return $\mathcal{W}^0_{3D}$
(optional) If SMPL needed: $(\boldsymbol{\beta}, \boldsymbol{\theta}) \leftarrow \text{SMPLify}(\mathcal{W}^0_{3D})$

4. Experiment Details

4.1. Implementation

Our diffusion model is constructed with 8 transformer decoder layers, each layer having 4 heads and 512 hidden units. We trained the model using eight NVIDIA V100 GPUs. In pretraining, HumanML3D [5] is projected onto random viewpoints to speed up multi-view model convergence, and in-distribution 2D human data like RICH [6] training set can be added. This phase uses a learning rate of 1e-4, a batch size of 64, and runs for 2k epochs. Subsequently, we fine-tuning the multi-view diffusion model using HumanML3D [5], BEDLAM [1], and Human3.6M [7], where HumanML3D [5] includes HumanAct12 [4] and AMASS [14]. The multi-view training is configured with 4 views (V = 4), a reduced learning rate of 1e-5, a batch size of 39, and extends over 3k epochs. Throughout both stages, we employ the Adam optimizer for parameter updates. The model supports a maximum sequence length of L = 300.

4.2. Evaluation datasets

RICH [6] comprises multi-view outdoor and indoor video sequences at 4K resolution, providing accurate 3D global motion labels. It includes 52 test scenes, with 3-4 test viewpoints per scene, totaling 191 viewpoints. Compared to

other human 3D pose datasets that are mostly collected in laboratory settings, the RICH [6] dataset is derived from real-life scenarios, encompassing a wide range of human actions and interactions with the environment, such as walking, sitting, grasping, and more.

AIST++ [10] is currently the largest and most diverse 3D human keypoint annotated database. It includes 1,408 dance clips across 10 dance genres, comprising 10,108,015 frames of human images captured from 9 different camera angles. Building upon the original multi-view videos, the dataset has been annotated with 3D skeletal data.

4.3. Qualitative results on RICH

Fig.3 presents additional qualitative results after first-frame alignment on RICH. Within this coordinate system, we observe the standardization of motion details. Although the regression+optimization baseline mainly serves to align inputs for fair comparison, its two-stage nature is less practical than our end-to-end approach. Our method demonstrates body heights closer to ground truth in challenging squatting and prone postures, proving its robust generalization capability in OOD scenarios.

Fig.4 displays absolute global trajectories in the world coordinate system. Although SA-HMR [15] incorporates complete environmental point clouds, it still exhibits inaccuracies at human-environment contact points, suggesting environmental information alone cannot effectively improve generalization. Notably, SA-HMR [15] produces some anomalous outliers that would significantly impact downstream action recognition quality. In contrast, our temporally-optimized model achieves superior performance in both motion quality and absolute trajectory accuracy.

4.4. Lifting SMPL keypoints with detector

While robust 2D detectors for SMPL-format poses are still limited compared to ViTPose [19], we further evaluate our framework using SMPL-format 2D inputs. Specifically, (1) **Ours***: noisy 2D inputs with keypoint-wise bias from dataset statistics, and (2) **Ours**$^+$: 2D keypoints projected from GVHMR [16]. All other baselines use ViTPose [19] detections. Quantitative results are shown in Tab. 1.

Despite the domain gap of SMPL-format detectors, our model maintains competitive pose accuracy and achieves clear improvements in all global-trajectory metrics, highlighting its robustness and accurate world-scale estimation. In particular, Ours* outperforms all baselines even under noisy inputs, and Ours$^+$ achieves comparable PA-MPJPE to GVHMR [16] while significantly surpassing it in MPJPE and absolute-position metrics. Although slight performance drops are observed under detector domain bias, the overall robustness of our method remains consistent, which is reasonable given the cross-domain discrepancy.

These results demonstrate that our framework generalizes well across different 2D keypoint sources, while exhibiting clear superiority in estimating global motion and absolute positioning, and progressively refining motion geometry and global accuracy throughout the denoising process. In future work, we aim to improve the 2D SMPL-format motion prediction model and explore incorporating detection confidence during training to further enhance robustness.

Methods	PA-MPJPE↓	MPJPE↓	W-MPJPE↓	WA-MPJPE	Abs-MPJPE↓
SMPLify [2]	98.5	187.4	417.0	218.2	473.2
SA-HMR [15]	51.1	93.2	–	–	268.3
WHAM [18]	44.3	80.0	198.5	116.6	–
GVHMR [16]	39.8	66.1	126.3	78.8	–
Ours*	**26.7**	**40.5**	**83.4**	**50.7**	**160.3**
Ours$^+$	40.6	62.1	106.6	67.8	194.3

Table 1. **Quantitative results on RICH.** The symbols * denotes noisy 2D input, and $^+$ indicates 2D keypoints projected from GVHMR [16] as input.

It should be emphasized that although we compare with WHAM [18] and GVHMR [16], we address fundamentally different problems: our primary objective is to recover **absolute human poses** in world coordinates from monocular input, rather than estimating root aligned global trajectories, as this enables wider applications in interactive scenarios. We include comparisons with WHAM [18] and GVHMR [16] to demonstrate that our method achieves comparable motion performance while additionally recovering absolute positioning.

4.5. Qualitative results on AIST++

Fig.5 demonstrates our qualitative results using COCO-format skeletons. In OOD scenarios (e.g., squatting kicks), our method produces more standardized motions. While both our approach and GVHMR [16] employ temporal optimization, our method avoids extreme outliers (such as unrealistic "floating" poses) and generates more physically plausible movement trajectories. This further validates our framework's extensibility to arbitrary human keypoint formats, including custom-defined skeletons, offering greater flexibility than alternative methods.

4.6. Root trajectories comparison

To validate our method's ability to recover motion over long horizons and along complex trajectories, Fig.6 visualizes the human motion trajectories projected onto the $y = 0$ plane of the world coordinate system. All methods are aligned to the first frame for fair comparison. As shown in the figure, due to monocular depth ambiguity, methods such as GVHMR [16] tend to exhibit trajectory drift over time in long sequences. In contrast, our method is able to recover the absolute position without suffering from error accumulation over time, resulting in a global trajectory that remains consistently close to the ground truth.

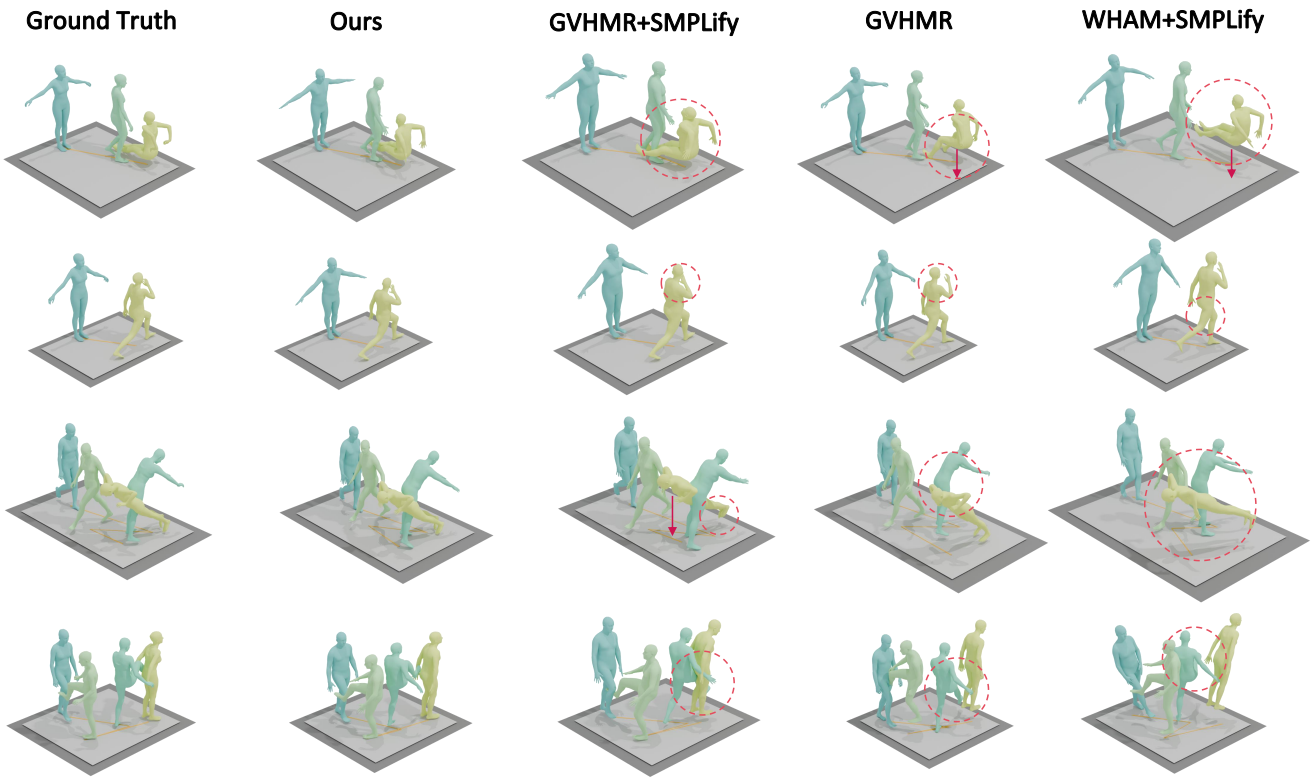

Figure 3. **Qualitative comparison on RICH.** Compare global motions after first-frame alignment in world coordinates. Our method eliminates floating artifacts present in baseline results.

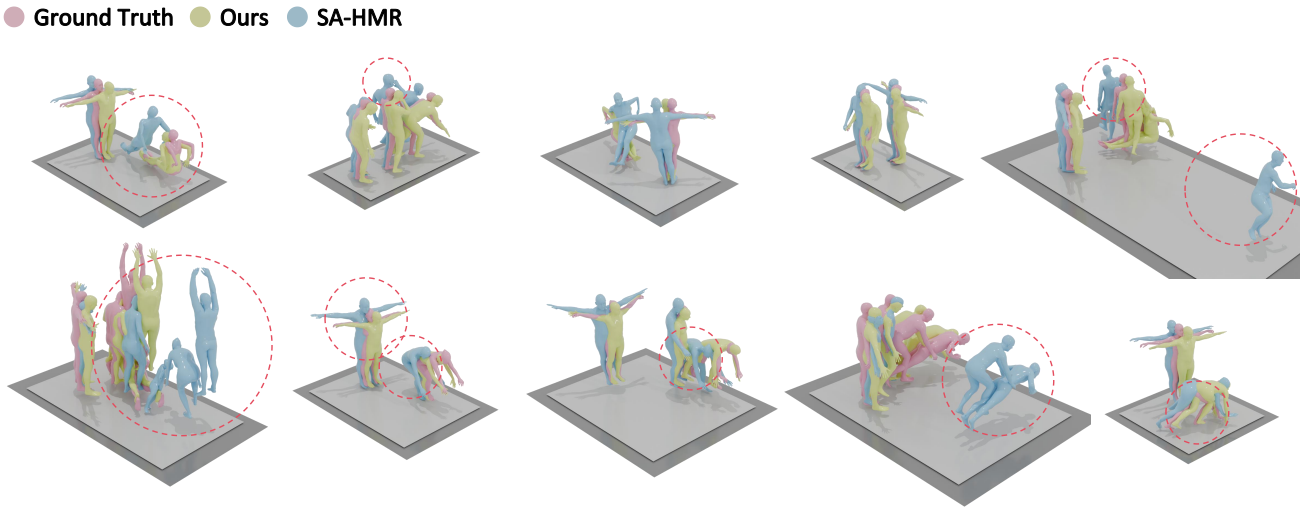

Figure 4. **Qualitative comparison on RICH.** Comparison of unaligned absolute poses in shared world coordinates. Our method achieves more accurate absolute position estimation.

4.7. Qualitative results of the ablation study

Fig.7 compares the results with and without incorporating homologous 2D data during pretraining. The visual comparison demonstrates noticeable quality improvements in motion details, particularly for out-of-distribution actions, when 2D data is included. This indicates that our architecture can effectively learn motion priors and diversity during pretraining, and mocap quality can be enhanced by adding

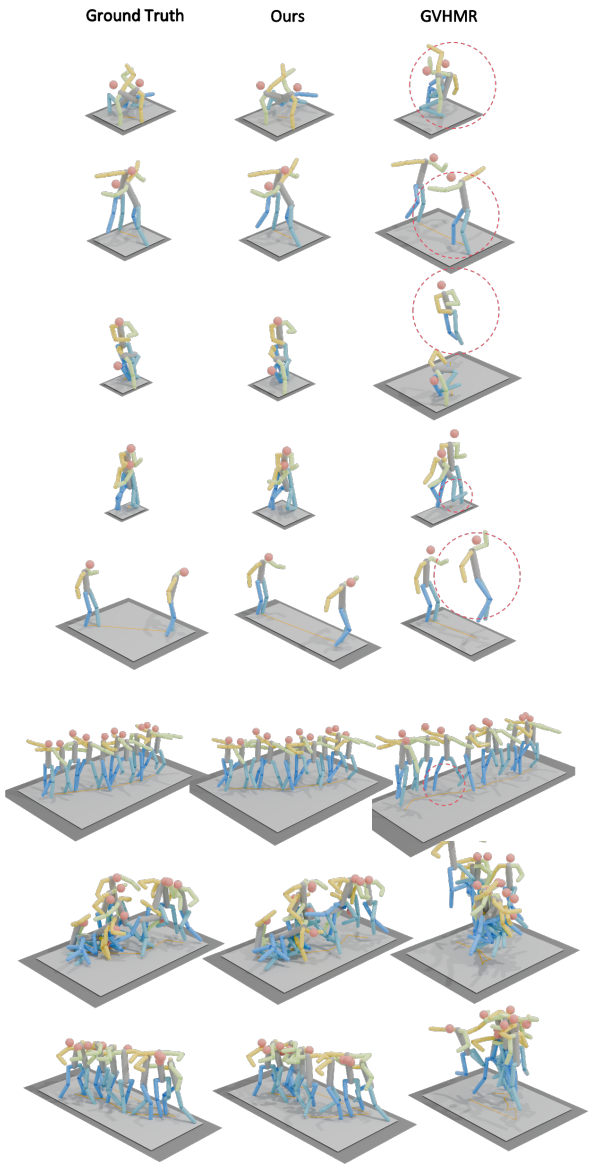

Figure 5. Qualitative evaluation on AIST++ (COCO keypoints): Our method achieves better OOD motion performance and is free from spatial outliers.

2D data.

Fig.8 illustrates the impact of adding pointmaps as additional input on network convergence speed. The training curves clearly show faster convergence when pointmaps are utilized.

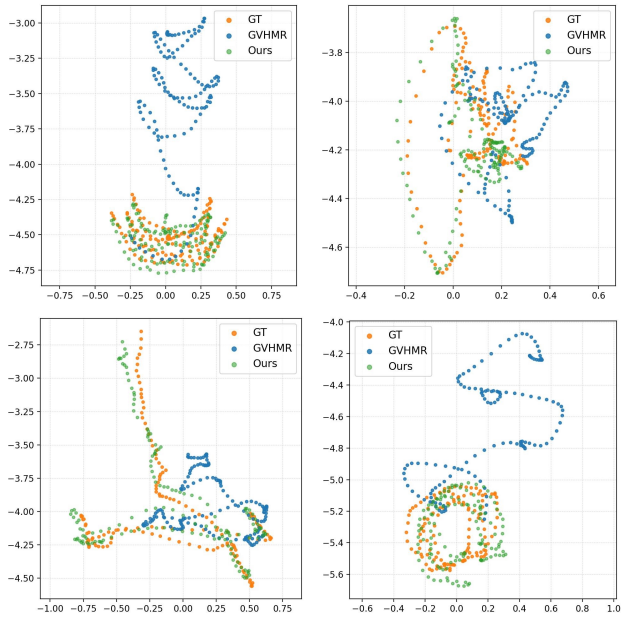

Figure 6. Complex root trajectories comparison on AIST++.

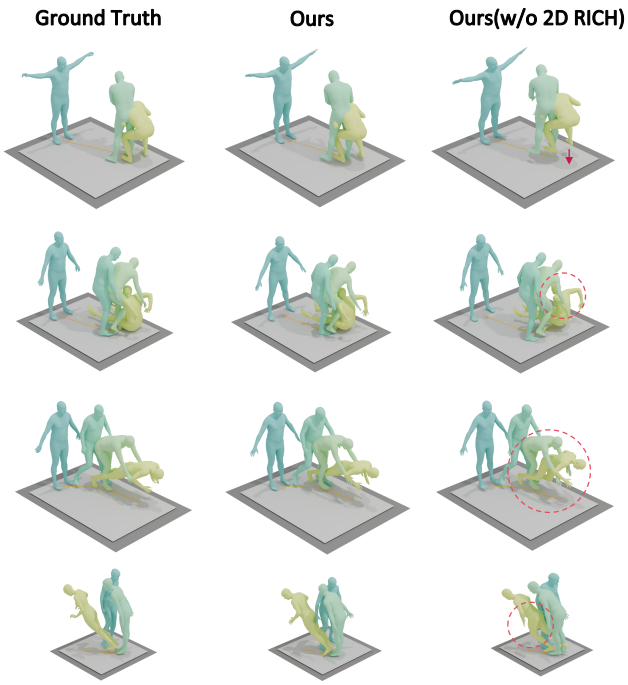

Figure 7. Performance comparison on RICH: pretraining with vs. without homologous 2D data. The model with homologous 2D pretraining achieves finer motion details closer to ground truth.

References

[1] Michael J Black, Priyanka Patel, Joachim Tesch, and Jinlong Yang. Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8726–8737, 2023. 4

[2] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl:

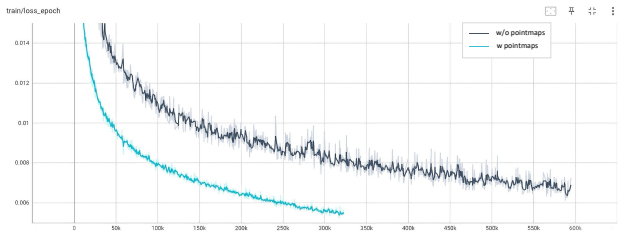

Figure 8. Comparison of training loss with and without pointmaps. The incorporation of pointmaps accelerates convergence.

Automatic estimation of 3d human pose and shape from a single image. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 561–578. Springer, 2016. 5

[3] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023. 2

[4] 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. 4

[5] 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. 4

[6] Chun-Hao P Huang, Hongwei Yi, Markus Höschle, Matvey Safroshkin, Tsvetelina Alexiadis, Senya Polikovsky, Daniel Scharstein, and Michael J Black. Capturing and inferring dense full-body human-scene contact. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13274–13285, 2022. 4, 5

[7] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 4

[8] Jia Jinrang, Zhenjia Li, and Yifeng Shi. Monouni: A unified vehicle and infrastructure-side monocular 3d object detection network with sufficient depth clues. *Advances in Neural Information Processing Systems*, 36:11703–11715, 2023. 3

[9] Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas J Guibas, and Gordon Wetzstein. Collaborative video diffusion: Consistent multi-video generation with camera control. *Advances in Neural Information Processing Systems*, 37:16240–16271, 2024. 2

[10] Ruilong Li, Shan Yang, David A Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13401–13412, 2021. 5

[11] 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. 2

[12] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2

[13] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9970–9980, 2024. 2

[14] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5442–5451, 2019. 4

[15] Zehong Shen, Zhi Cen, Sida Peng, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Learning human mesh recovery in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17038–17047, 2023. 1, 5

[16] Zehong Shen, Huaijin Pi, Yan Xia, Zhi Cen, Sida Peng, Zechen Hu, Hujun Bao, Ruizhen Hu, and Xiaowei Zhou. World-grounded human motion recovery via gravity-view coordinates. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024. 5

[17] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2

[18] 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. 5

[19] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 5

[20] Yu Zhan, Fenghai Li, Renliang Weng, and Wongun Choi. Ray3d: ray-based 3d human pose estimation for monocular absolute 3d localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13116–13125, 2022. 1