EasyRet3D: Uncalibrated Multi-view Multi-Human 3D Reconstruction and Tracking

A. Appendix Section

This appendix is organized as follows. In Section B, we conduct ablation studies on different components of our method. In Section C we provide more qualitative examples of our method. In Section D we present details on the experiments on tracking and pose estimation. We provide details on the optimization in Section E. Finally, we discuss the limitations of our method in Section F.

B. Ablations

To validate the robustness of our proposed method, we conducted ablation studies to discern the contribution of variants in camera parameters and optimization to the aggregate accuracy of our method.

B.1. Ablation study on Camera Parameters

Variant	PA-MPJPE (mm) ↓
Average Stitching	27.4
Max Stitching	27.1
Adaptive Stitching (ours)	**25.7**
Adaptive Stitching (3 views)	26.8
Adaptive Stitching (2 views)	27.5
Single View (front)	30.1
Single View (back)	33.5

Table 1. **Effect of number of camera views** To investigate the benefit of using more camera views, we evaluated the 3D pose estimation performance of our method, in terms of PA-MPJPE, with varying numbers of views on the test split of the Human3.6M dataset. ↓ means the lower the better. We found that using more camera views yields more accurate 3D pose estimation.

We first ablated on the number of cameras on the Human3.6M dataset. As shown in Table 1, reducing the number of cameras generally increases the 3D pose estimation error (PA-MPJPE). Moreover, views that are more prone to occlusions (e.g., back views) benefit greatly from the aggregated information across different perspectives. Our Adaptive Stitching approach, which incorporates the confidence value of detected 2D keypoints from VitPose [7] as weights for the weighted average, yields the best PA-MPJPE of 25.7 (mm) under the maximum number of views ($V = 4$). Therefore, having more camera views and employing weighted average help to produce more accurate 3D reconstructions.

Adaptive 3D Human Stitching Module: After obtaining the camera poses $\hat{R}, \hat{T}$, we transform the SMPL parameters $\boldsymbol{\mathcal{P}}_t^{i,v}$ from each view's local camera coordinates to the global world coordinates. Due to occlusions or varying camera angles, different views provide incomplete observations of the scene, making it challenging to recover all the parameters for every individual. To address this, we develop an adaptive stitching algorithm that fuses the SMPL parameters from all views by computing a weighted sum. The weight assigned to each view is based on the confidence score of 2D keypoints in that view, ensuring that views with higher reliability contribute more to the final representation. The aggregated SMPL parameters provide a unified global model, improving 3D reconstruction even in cases of occlusion or missing views.

In Table 2, we investigated the impact of ground-truth (GT) camera pose information on the performance of 3D pose estimation. Our approach (Adaptive Stitching) is a calibration-free approach that utilizes an auto-calibration procedure to approximate camera pose, whereas (Adaptive Stitching w/ GT) incorporates the ground-truth camera pose data directly into the model. The PA-MPJPE results demonstrate a marginal difference between our calibration-free

Variant	PA-MPJPE (mm)↓
Adaptive Stitching (2 views)	27.5
Adaptive Stitching (3 views)	26.8
Adaptive Stitching (ours)	25.7
Adaptive Stitching w/ GT	25.2

Table 2. **Effectiveness of our auto-calibration techniques** Aiming to demonstrate the efficacy of our method in scenarios devoid of manual calibration inputs, we compared the 3D pose estimation results of our calibration-free method against our variant utilizing ground-truth camera pose data on the test split of the Human3.6M dataset.

method and the ground-truth camera pose informed version, with errors of 25.7 (mm) and 25.2 (mm) respectively. This suggests that our auto-calibration process is highly effective, yielding 3D pose estimation results comparable to the results obtained with the presence of ground-truth camera poses. Such a result underscores the robustness of our calibration-free method and its potential for practical applications where ground truth camera poses are unavailable or difficult to obtain.

B.2. Ablation study on Optimization

Our method relies on iterative optimization to jointly optimize for SMPL parameters and camera intrinsic/extrinsic after initializing multi-view people in the world. To investigate the effect of each stage of our optimization on performance, we perform ablation on the three iterative stages of our optimization process, with results shown in table 3.

We find that stage 2 optimization contributes most significantly to performance, and depending on the dataset, stage 3 optimization may further improve or degrade the performance. We hypothesize that the learned human motion prior "over-corrects" and "over-smoothens" the human poses, especially for small movements of arms and heads.

C. Further Qualitative Results

In this section, we present supplementary qualitative results on camera pose estimation and multi-human 3D tracking under occlusion.

C.1. Camera Pose Estimation

In Figure 1, we show that our method gives an accurate estimation of ground truth camera poses.

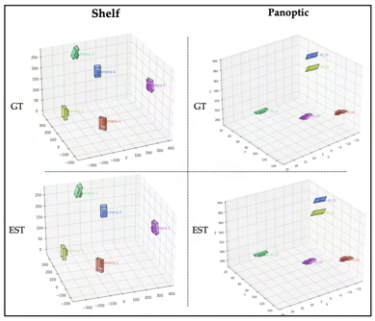

Figure 1. Qualitative results on camera pose estimation. The left column visualizes the ground truth and our estimated camera poses on the Shelf dataset, and the right column shows the ground truth and our estimated camera poses on the Panoptic dataset. Results indicate that our method can accurately estimate camera poses

C.2. Multi-human 3D Tracking

In Figure 2 and Figure 3, we showcase our method's tracking performance in severe & complete occlusion scenarios against our backbones.

D. Additional Experiments Details

D.1. Paralleling Motion Prior Optimization

Our intentional use of non-autoregressive motion prior model allows to take advantage of parallel optimization, which significantly improves the effiency of our global iterative optimization module. We divide a motion sequence into overlapping segments, each with 120 frames, and include 24 overlapping frames to reduce discontinuities between segments. This approach of using overlapping windows also enables the prior's latent representation to model a consistent length of motion. Because our motion prior operates non-autoregressively, allowing us to optimize all segments simultaneously. We provide a run time comparison between the state-of-the-art monocular view optimization based method SLAHMR and our method in table 4. We observe a significant improvement compared to SLAHMR optimization time: EasyRet3D takes 10 minutes (1.667 fps) to optimize 1000 frames, while SLAHMR [8] takes 260 minutes (0.06 fps), an 26x improvement in optimization speed.

D.2. Pose estimation evaluation details

D.2.1 PHALP track matching

For each frame within the evaluation subset, since the number of detected individuals n_{detected} might be larger than the ground-truth number of individuals n_{gt} (i.e., $n_{\text{detected}} > n_{\text{gt}}$), it is crucial that we iteratively compared the PHALP [5] 2D keypoints against the ground-truth 2D keypoints to establish the best match for each track ID, based on their spatial correspondence. The keypoints matrices are structured as as $(N, T, J, 3)$, representing N individuals, T timestamps, J joints, and 3 coordinates $(x, y, \text{confidence})$. For each individual's ground-truth keypoints we computed their encompassing bounding box and for each PHALP track's keypoints, we filtered out any points with a zero confidence value and obtained respective bounding boxes. To derive the best matches, we calculated the Intersection over Union (IoU) between each track's bounding box and the ground-truth bounding box. In this sense, we selected the track with the highest IoU for each frame as the best match and recorded its index. The output of this matching procedure is a matrix with shape (N_{track}, T) , where N_{track} is the number of tracks and T is the number of frames. Each element in this matrix represents the index of the ground-truth individual that best matches each track ID across the sequence of frames.

Variant	Shelf					Human3.6M
	A1 ↑	A2 ↑	A3 ↑	Avg. ↑	PA-MPJPE (mm) ↓	PA- MPJPE (mm) ↓
Stage 1	99.5	95.0	97.5	97.3	44.8	26.2
Stage 1,2	99.7	95.6	97.6	97.6	42.4	**25.7**
Stage 1,2,3	**99.8**	**95.9**	**97.8**	**97.8**	**41.7**	26.9

Table 3. **Effect of Iterative Optimization** Ablation three iterative stages of our optimization process. We report the 3D pose estimation metric PA-MPJPE on the Shelf and Human3.6M datasets. ↑ means the larger the better, ↓ means the lower the better. The most critical component for performance is the second stage of the optimization. We observe that the dataset impacts the performance of the optimization process: notably, our full-system optimization (Stages 1, 2, and 3) did not yield an improvement in PA-MPJPE on the Human3.6M dataset.

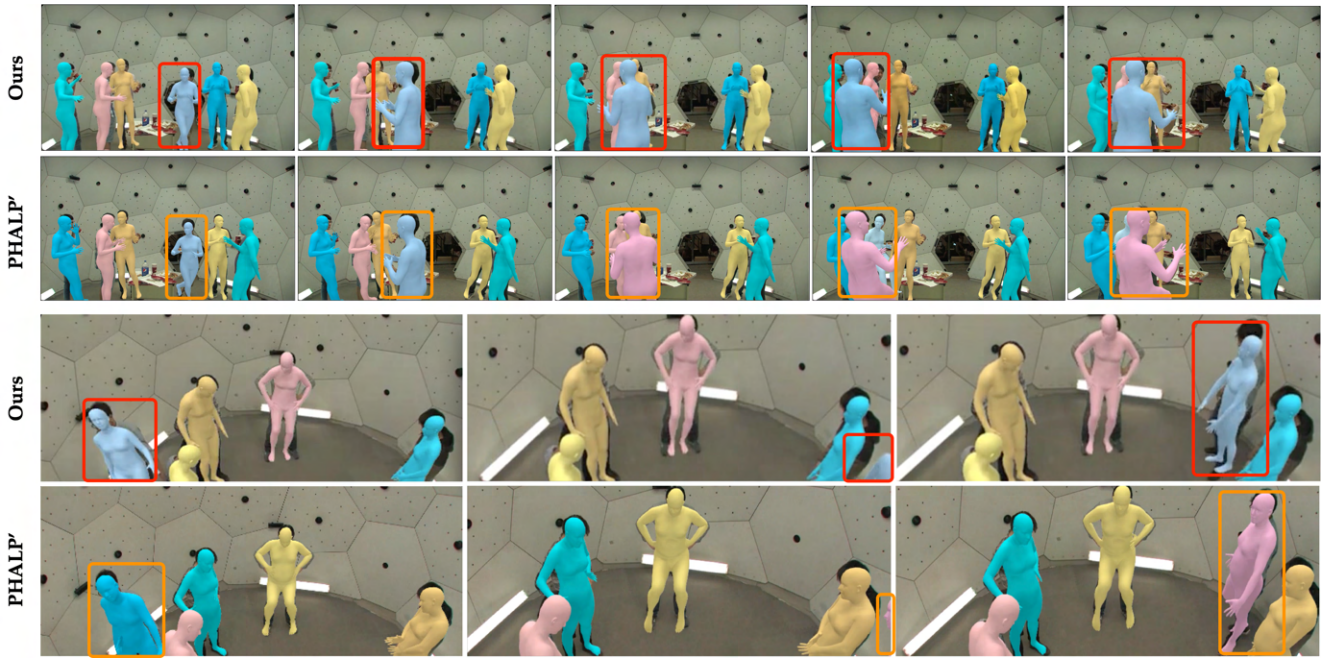

Figure 2. **Side-by-side qualitative evaluations of 3D pose tracking performance on the CMU Panoptic dataset [3].** The superior consistency of our method is illustrated in the top rows of each scene, where the individual encased in the red bounding box consistently retains their color designation, showcasing our method's robust identity tracking over time. In contrast, the bottom rows, representing the PHALP' results, reveal the system's vulnerability to occlusions, as highlighted by the individual in the orange bounding box receiving varied color identities, indicating identity switches.

Methods	Runtime per 1000 frames (fps) ↑
SLAHMR [8]	260 minutes (0.064)
EasyRet3D	10 minutes (1.667)

Table 4. **Effect of number of camera views** Total optimization time (running time) of our methhod optimization module and SLAHMR global optimization. We present this both as total runtime per 1000 frames (minutes) and frames per second (fps).

D.2.2 Keypoint ordering conversion

The default 3D keypoint ordering of our method is in the OpenPose-25 format, which needs to be converted to the format compatible with the ground-truth keypoint ordering in Table 5. As they are not specified in OpenPose-25, we estimated the Bottom Head and Top Head using the following steps:

$$\text{Mid Shoulder (MS)} = \frac{\text{LShoulder} + \text{RShoulder}}{2}$$

$$\text{Mid Ear (ME)} = \frac{\text{LEar} + \text{REar}}{2}$$

$$\text{Center Head (CH)} = \frac{\text{Nose} + \text{ME}}{2}$$

$$\text{Bottom Head (BH)} = \frac{\text{MS} + \text{CH}}{2}$$

$$\text{Top Head} = \text{BH} + 2 \times (\text{CH} - \text{BH})$$

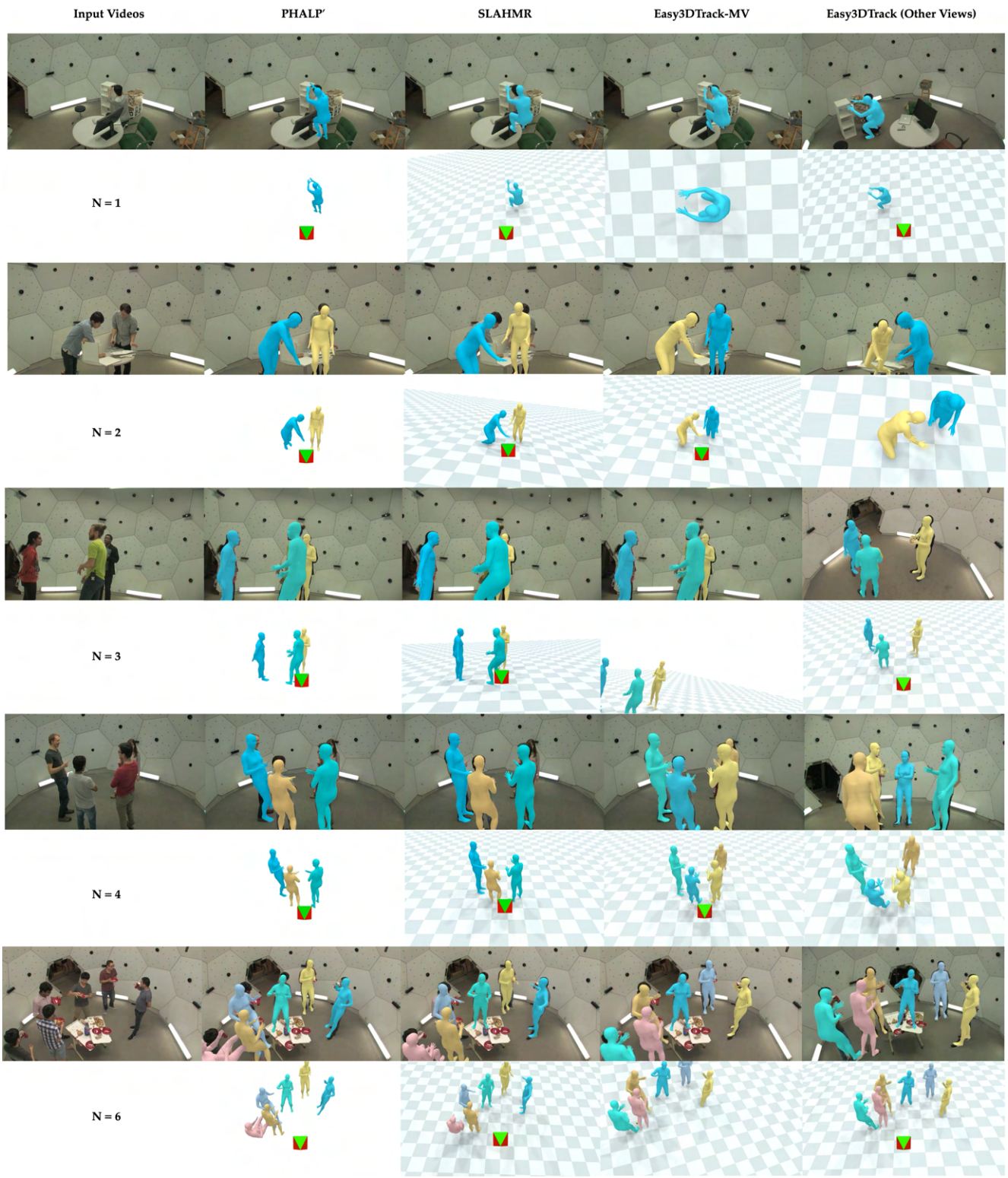

Figure 3. **Additional Qualitative results of the proposed approach.** All input videos are from the panoptic dataset [3]. The columns compare results from three methods: PHALP' [2], SLAHMR [8], and EasyRet3D. The examples include different ranges of people (from $N = 1$ to $N = 6$) in a given scene. The examples include unusual poses, unusual viewpoints, people in close interaction, and severe occlusions. For each example we show the input image, the reconstruction overlay, a front view and a additional view. For our method EasyRet3D, we also show the reconstruction overlay to another view.

Table 5. Comparison of OpenPose-25 and Shelf Dataset Keypoint Ordering

Index	OpenPose-25	Shelf Dataset
0	Nose	RAnkle
1	Neck	RKnee
2	RShoulder	RHip
3	RElbow	LHip
4	RWrist	LKnee
5	LShoulder	LAnkle
6	LElbow	RWrist
7	LWrist	RElbow
8	MidHip	RShoulder
9	RHip	LShoulder
10	RKnee	LElbow
11	RAnkle	LWrist
12	LHip	Bottom Head
13	LKnee	Top Head
14	LAnkle	-
15	REye	-
16	LEye	-
17	REar	-
18	LEar	-
19	LBigToe	-
20	LSmallToe	-
21	LHeel	-
22	RBigToe	-
23	RSmallToe	-
24	RHeel	-

D.2.3 PA-MPJPE

We calculated PA-MPJPE for both Shelf and Human3.6M datasets. With the matching matrices, we aligned the matched predicted joints and ground-truth joints by performing Procrustes Analysis to find the optimal scaling factor s_{PA}, rotation matrix R_{PA}, and translation matrix T_{PA}. After stacking the aligned point sets, we computed the per joint position errors and calculated the mean to obtain PA-MPJPE.

$$e_i = ||P_i - G_i|| \tag{1}$$

$$\text{PA-MPJPE} = \frac{1}{n}\sum_{i=1}^{n} e_i \tag{2}$$

D.2.4 PCP3D

As the Shelf dataset contains complex scenes where interactions between multiple individuals create severe occlusions, we computed the PCP3D on this dataset using the matched predicted joints and ground-truth joints. Let P_{start} and P_{end} be the start and end points of the predicted limb, and let G_{start} and G_{end} be the start and end points of the ground truth limb. The limb is considered correctly estimated if the following condition is met:

$$\frac{\|P_{\text{start}} - G_{\text{start}}\|_2 + \|P_{\text{end}} - G_{\text{end}}\|_2}{2} \leq \alpha \cdot \|G_{\text{start}} - G_{\text{end}}\|_2, \tag{3}$$

where:

- $\|\cdot\|_2$ denotes the Euclidean (L2) norm.
- α is the threshold ratio, typically set to 0.5.

This condition is checked for each limb to compute the PCP3D score. We followed the bone group selection proposed by [1]. Table 6 shows the specific PCP results for each bone group, complementing Section 4.3.

Table 6. PCP scores for each bone group across actors

Bone Group	Actor 1	Actor 2	Actor 3	Average
Head	100.0	100.0	99.8	99.9
Torso	100.0	100.0	100.0	100.0
Upper arms	99.6	99.5	100.0	99.7
Lower arms	99.3	83.8	87.6	90.2
Upper legs	100.0	100.0	100.0	100.0
Lower legs	100.0	100.0	100.0	100.0
Total	99.8	97.2	97.9	98.3

E. Additional Details on Optimization

In Section 3.5 of the main manuscript, we present an iterative optimization to jointly solve for SMPL parameters $\{{}^{w}\boldsymbol{\mathcal{P}}_t^i\}$ for a person i at timestamp t and camera extrinsic and intrinsic $\{R_{\boldsymbol{W}}^{\boldsymbol{C}_v}, T^{\boldsymbol{C}_v}, f^v\}$ for each view v. Here, we provide more details about the individual loss term in the second stage ($\mathcal{L}_{\text{stage2}}$) and third stage ($\mathcal{L}_{\text{stage3}}$) of our optimization. Recall the second stage of our loss function:

$$\mathcal{L}_{\text{stage2}} = \lambda_{\text{J2D}}\mathcal{L}_{\text{J2D}} + \lambda_\beta\mathcal{L}_\beta + \lambda_\Theta\mathcal{L}_\Theta + \lambda_{\text{smooth}}\mathcal{L}_{\text{smooth}}. \tag{4}$$

The $\mathcal{L}_{\text{smooth}}$ is a simple loss based on minimal kinetic motion:

$$\mathcal{L}_{\text{smooth}} = \sum_{i=1}^{N}\sum_{t=1}^{T}\left\|{}^{w}J_t^i - {}^{w}J_{t+1}^i\right\|^2. \tag{5}$$

Recall the third stage of our loss function:

$$\mathcal{L}_{\text{stage3}} = \lambda_{\text{J2D}}\mathcal{L}_{\text{J2d}} + \lambda_\beta\mathcal{L}_\beta + \lambda_\Theta\mathcal{L}_\Theta + \lambda_{\text{motion}}\mathcal{L}_{\text{motion}} + \lambda_{\text{env}}\mathcal{L}_{\text{env}}. \tag{6}$$

The loss term $\mathcal{L}_{\text{motion}}$ is introduced to better capture the possible distribution of human motions. It's made up of two

loss terms $\{\mathcal{L}_{\text{cvae}}, \mathcal{L}_{\text{stab}}\}$, which we draw from transition-based motion prior HuMoR [6].

Through HuMoR, the probability of a trajectory sequence $\{s_0, \ldots, s_T\}$ is decomposed into the probabilities of transitions between consecutive states, $p_\theta(s_t|s_{t-1})$, with s_t being an enriched state representation. This state encompasses the SMPL pose parameters ${}^w\boldsymbol{\mathcal{P}}$, in addition to extra velocity and joint location predictions.

The transition likelihood $p_\theta(s_t|s_{t-1})$, captured within a conditional variational autoencoder (cVAE) framework, is given by:

$$p_\theta(s_t|s_{t-1}) = \int_{z_t} p_\theta(z_t|s_{t-1}) p_\theta(s_t|z_t, s_{t-1}), \quad (7)$$

where $z_t \in \mathbb{R}^{48}$ is a latent variable that encodes the transition from s_{t-1} to s_t. This latent variable is anchored by a conditional prior $p_\theta(z_t|s_{t-1})$, which is modeled as a Gaussian distribution with a mean and variance that are functions of s_{t-1}. Incorporating this prior, a loss term based on z_t is utilized:

$$\mathcal{L}_{\text{cvae}} = -\sum_{i=1}^{N}\sum_{t=1}^{T} \log \mathcal{N}(z_t; \mu_\theta(s_{t-1}), \sigma_\theta(s_{t-1})). \quad (8)$$

Additional stabilization losses $\mathcal{L}_{\text{stab}}$ are employed to ensure the physical validity and consistency of the predicted velocity and joint location aspects of s_t with its pose parameters. For further details, consult rempe2021humor, ye2023decoupling.

The loss term $\mathcal{L}_{\text{motion}}$ is introduced to optimize the ground plane $g \in \mathbb{R}^3$ in the scene. Likewise, it's also made up of two loss terms $\{\mathcal{L}_{\text{skate}}, \mathcal{L}_{\text{con}}\}$. We use the HuMoR decoder to obtain the ground contact probability $c(j) \in [0, 1]$ for the joints ${}^wJ$. A zero velocity prior is enforced on joints likely to be in ground contact g, mitigating the foot-skate effect:

$$\mathcal{L}_{\text{skate}} = \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{j=1}^{J} c_t^i(j) \left\| {}^wJ_t^i(j) - {}^wJ_{t+1}^i(j) \right\|. \quad (9)$$

Simultaneously, a constraint is applied to keep their distance from the ground under a predetermined threshold δ:

$$\mathcal{L}_{\text{con}} = \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{j=1}^{J} c_t^i(j) \max(d({}^wJ_t^i(j), g) - \delta, 0), \quad (10)$$

where $d(\mathbf{p}, g)$ is the distance between ground plane g and points $\mathbf{p} \in \mathbb{R}^3$.

We provide the hyperparameter settings used in our optimization in table 7. We implemented our optimization in pytorch [4] with a lr of 1.

	Stage	Hyperparameter	Value
Optimization	1	λ_{J2D}	0.004
	2	λ_β	0.05
		λ_θ	0.04
		λ_{smooth}	5
	3	λ_{cvae}	0.075
		λ_{skate}	100
		λ_{stab}	0.075
		λ_{con}	10

Table 7. Hyperparameter configuration for different optimization stages of Easy3DTrack-MV

F. Limitation

Our approach is by design modular. For instance, it utilizes the advanced cross-view matching algorithm by xu2022multi for automatic camera calibration; it uses the single-view tracking and SMPL pose output from PHALP' [2] for our stitching algorithm. However, it should be noted that if any of these individual methods were to fail, it could potentially propagate its failure to our final optimization process.

References

[1] Long Chen, Haizhou Ai, Rui Chen, Zijie Zhuang, and Shuang Liu. Cross-view tracking for multi-human 3d pose estimation at over 100 fps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3279–3288, 2020. 5

[2] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. *arXiv preprint arXiv:2305.20091*, 2023. 4, 6

[3] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Scott Godisart, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social interaction capture. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 3, 4

[4] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 6

[5] Jathushan Rajasegaran, Georgios Pavlakos, Angjoo Kanazawa, and Jitendra Malik. Tracking people by predicting 3d appearance, location and pose. In *CVPR*, pages 2740–2749, 2022. 2

[6] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. Humor: 3d human motion model for robust pose estimation. In *International Conference on Computer Vision (ICCV)*, 2021. 6

[7] 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. 1

[8] Vickie Ye, Georgios Pavlakos, Jitendra Malik, and Angjoo Kanazawa. Decoupling human and camera motion from videos in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21222–21232, 2023. 2, 3, 4