FastHMR: Accelerating Human Mesh Recovery via Token and Layer Merging with Diffusion Decoding

Supplementary Material

A. Datasets

3DPW [11] is an outdoor in-the-wild dataset containing challenging scenarios, such as walking in the city and going upstairs, with ground-truth SMPL annotations. We use the test split of 3DPW for evaluation of our model.

EMDB [6] is a recently-captured dataset that recoded 10 participants in 81 indoor and outdoor environments using body-worn electromagnetic (EM) sensors and provides ground-truth SMPL parameters. It has two test splits, EMDB 1 for evaluation of 3D pose and shape in camera coordinates, and EMDB 2 for global trajectory estimation. We evaluate our model on EMDB 1 test split.

BEDLAM [1] is a large-scale synthetic dataset containing 1 million video frames with ground-truth SMPL/SMPL-X parameters. We use the train split of BEDLAM to train our network.

Human3.6M [5] is a large-scale motion capture dataset captured in indoor environment. it contains 3.6 million video frames of 11 subjects performing 15 distinct actions recorded using a single motion-capture system and 4 calibrated video cameras. Our network is trained using motion capture data from five subjects (S1, S5, S6, S7, and S8), with the data downsampled to 25 fps.

MPI-INF-3DHP[10] is a markerless dataset that spans both indoor and outdoor environments, featuring a variety of camera viewpoints, clothing styles, and human poses, along with ground-truth 3D keypoint annotations. The training dataset consists of 8 subjects, each captured in 16 videos.

AMASS[9] is a large motion-capture dataset that unifies 15 different optical marker-based mocap datasets and represent them all using SMPL [8] parametrization. It includes over 40 hours of motion data, covering more than 300 subjects and over 11,000 distinct motions.

B. Additional Ablation Studies

Effect of swing-twist decomposition. HybrIK [7] introduces an inverse kinematics approach to decompose SMPL pose parameters into joint locations and twist rotations. Tab. 1 shows that this decomposition reduces MPJPE by 33 mm, demonstrating a notable improvement in model performance. This enhancement can be attributed to two key factors. First, representing the pose parameters in the SO(3) space makes it challenging for the diffusion denoiser to denoise effectively. In SMPL's hierarchical structure, each joint rotation is defined relative to its parent joint. Consequently, when denoising a joint rotation like the wrist, the denoiser may also adjust the parent joint, such as the elbow, to reduce error. However, this can unintentionally alter the elbow's position from the true location. Second, since the pose parameters are defined in the SO(3) space, attributes like bone length depend on the shape parameters, which we estimate using HMR 2.0 [4] and may contain errors. Bone length, however, is critical for accurately determining joint positions. By decomposing the pose parameter into joint location and wrist rotation, the model reduces its reliance on the shape parameter, allowing it to be used for other attributes, such as body mass.

Data representation	PA-MPJPE	MPJPE	MVE
w/o HybrIK	58.1	95.1	109.2
w/ HybrIK	37.0	62.1	72.6

Table 1. Ablation study on the effect decomposing pose parameter into joint location and twist rotation using HybrIK [7] method. All the evaluations are with a single-step diffusion model on 3DPW dataset. Errors are in mm.

Method	**MPJPE (3DPW)**	**MPJPE (EMDB)**
w/ merging	62.2	71.6
w/o merging	60.3	72.6

Table 2. Impact of merging (Mask-ToMe + ECLM) on model performance after introducing the diffusion decoder. The MPJPE errors are reported in mm.

Using diffusion alone. We propose diffusion decoder as a replacement of the naive MLP decoder used in transformer-based HMR models to enhance the robustness of model against merging methods proposed to enhance the throughput. Tab. 2 shows that removing the merging methods from FastHMR and using only the diffusion decoder reduces MPJPE by 1.9 mm on the 3DPW dataset (used for hyperparameter tuning), but increases it by 1 mm on the EMDB benchmark (an external evaluation set). This indicates that while Mask-ToMe alone slightly worsens performance likely due to the MLP being less robust to token changes. the diffusion decoder can recover this loss. Moreover, Mask-ToMe supports the diffusion decoder by merging background tokens, which improves the model's generalizability.

Effect of Segmentor. Tab. 3 shows the effect of using different segmentors in Mask-ToMe on throughput and estimation error. Although YOLO11x-seg slightly reduces the es-

Segmentor	Throughput (fps)	MPJPE (mm)
YOLO11n-seg	1500	62.2
YOLO11x-seg	300	61.7

Table 3. Performance comparison when using different segmentors.

n_l	MPJPE (3DPW)	MPJPE (EMDB)
Baseline	62.1	73.3
10	74.8	85.3
20	74.7	85.0
40	73.6	84.0
60	73.1	83.8
80	74.2	85.3
100	80.2	93.5

Table 4. The effect of merging ratio n_l in Mask-ToMe, evaluated on CameraHMR model.

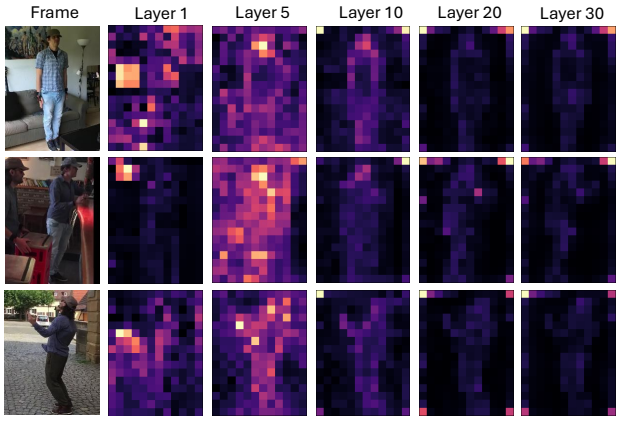

Figure 1. Attention map visualization of CameraHMR.

timation error, it requires significantly more computational resources. As a result, we use YOLO11n-seg during inference in the proposed FastHMR framework.

Effect of masking ratio. In the Mask-ToMe method, we merge n_l background tokens at each layer. As shown in Tab. 4, the choice of n_l directly affects MPJPE on the 3DPW and EMDB benchmarks. In general, larger merging ratios increase the error. For example, $n_l = 100$ produces the worst results. Interestingly, even smaller ratios such as 10 or 20 also lead to higher errors than expected.

To understand the underlying reason, we analyzed the attention maps (Fig. 1). They reveal that corner background tokens, which carry little useful information, are repurposed by the model as register tokens [3]. As the layers progress, the model gradually shifts important information into these tokens. When the merging ratio is too low, we risk merging them after they begin storing meaningful information, which could potentially reduce the accuracy of the final estimation.

Setting	PA-MPJPE	MPJPE	MVE
w/o person mask	67.2	98.1	112.9
+ diffusion decoder	51.0	75.0	84.2
w/ person mask	54.8	84.0	96.7
+ diffusion decoder	46.7	71.6	82.4

Table 5. Ablation of token merging with and without person mask, and the added benefit of the diffusion decoder.

Method	Parameters (M)	MACs (G)
HMR2.0		
Baseline	670.2	122.6
Baseline + ECLM	591.5	107.5
Baseline + ECLM + Mask-ToMe	591.5	52.8
CameraHMR		
Baseline	737.1	144.0
Baseline + ECLM	619.0	121.3
Baseline + ECLM + Mask-ToMe	619.0	70.7
Diffusion Decoder	29.5	6.8

Table 6. Effect of proposed methods on model parameters (M) and MACs (G).

Excluding person mask in Mask-ToMe We use a person mask to merge only the background tokens while preserving the person tokens for human mesh recovery. An alternative method, proposed in ToMe [2], merges tokens based on similarity. Table 5 shows that removing the mask increases the MPJPE by 14.1 mm. Moreover, replacing the MLP decoder with a diffusion decoder does not recover this loss. These results indicate that excluding the person mask risks merging person tokens, which removes critical information for mesh recovery. Once that information is lost, even a strong decoder cannot fully compensate.

Parameters and MACs. Table 6 compares the impact of ECLM, Mask-ToMe, and the diffusion decoder on model complexity. For both HMR2.0 and CameraHMR, incorporating ECLM reduces parameters and multiply–accumulate operations (MACs), and combining it with Mask-ToMe yields substantial computational savings. The diffusion decoder, shown separately, is a lightweight module with relatively few parameters and MACs, highlighting its efficiency compared to the backbone models.

In-the-wild Evaluation. Fig. 2 shows qualitative results of FastHMR on challenging in-the-wild examples, including dynamic actions (jumping, weightlifting, climbing) and complex body articulations. Across diverse settings, our method reconstructs plausible and temporally consistent human meshes, even under fast motion, self-occlusion, and

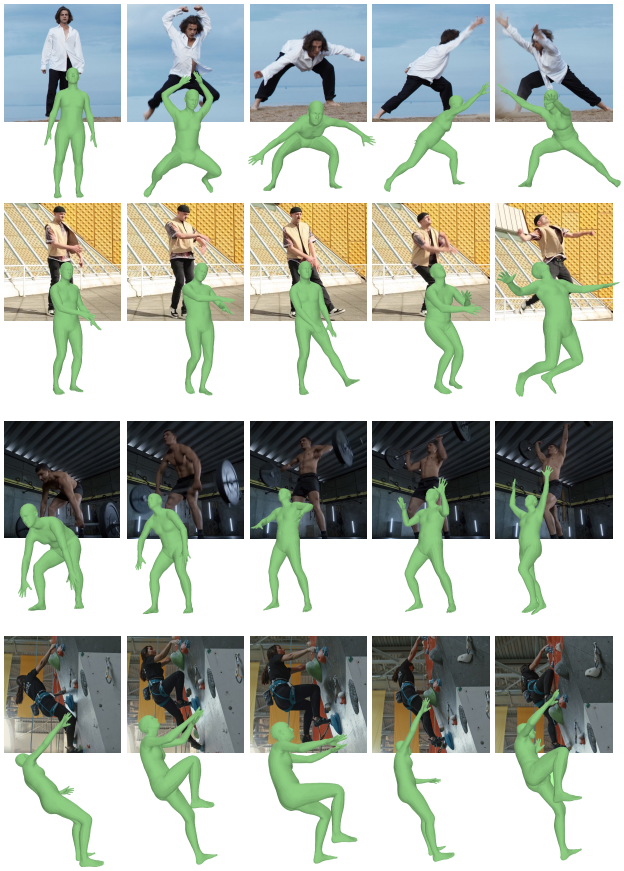

Figure 2. In-the-wild video evaluation of FastHMR

cluttered backgrounds, highlighting its robustness beyond standard benchmarks.

C. Additional Hyperparameters

Tables 7, 8, and 9 denote the hyperparameters used during diffusion training, denoiser configuration, and VAE pretraining, respectively.

D. Additional Qualitative Comparison

Fig. 3 compares CameraHMR and FastHMR-CameraHMR across four different frames. Although the results may appear similar depending on the camera viewpoint, CameraHMR is more likely to produce erroneous estimates for occluded body parts, whereas FastHMR demonstrates greater robustness due to its temporal awareness.

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. 1

[2] Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. *arXiv preprint arXiv:2210.09461*, 2022. 2

[3] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv*

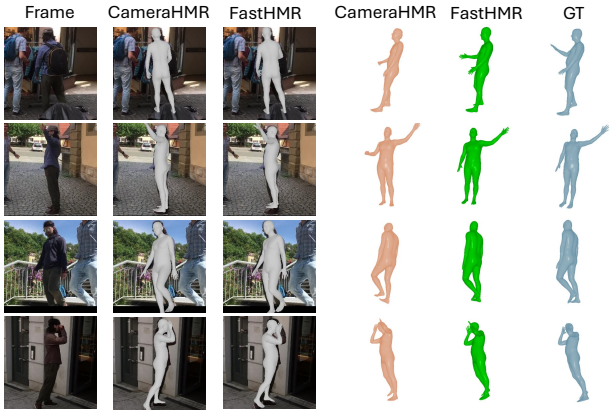

Figure 3. Qualitative comparison between CameraHMR and FastHMR-CameraHMR.

Hyper-parameter	Value
# steps	1000
β_{start}	0.00085
β_{end}	0.012
sheduler	scaled linear
clip sample	False
variance type	fixed small

Table 7. Diffusion Hyperparameters.

Hyper-parameter	Value
Type	Transformer Encoder
condition dim	1024
embedding dim	512
flip sin to cos	True
# frames	243
# encoded framed	27
frequency shift	0
# heads	4
Feedforward dim	1024
Dropout	0.001
Activation	GELU
Normalize before	False
# Layers	5

Table 8. Denoiser Hyperparameters.

Hyper-parameter	Value
input dim	243×133
latent dim	27×512
feedforward dim	1024
# layers	9
# heads	4
dropout	0.1
activation	GELU
positional embedding	learned

Table 9. VAE Hyperparameters.

preprint arXiv:2309.16588, 2023. 2

[4] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4d: Reconstructing and tracking humans with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14783–14794, 2023. 1

[5] 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. 1

[6] Manuel Kaufmann, Jie Song, Chen Guo, Kaiyue Shen, Tianjian Jiang, Chengcheng Tang, Juan José Zárate, and Otmar Hilliges. EMDB: The electromagnetic database of global 3D human pose and shape in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14632–14643, 2023. 1

[7] Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. HybrIK: A hybrid analytical-neural inverse kinematics solution for 3D human pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3383–3393, 2021. 1

[8] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. SMPL: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 1

[9] 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. 1

[10] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 1

[11] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using imus and a moving camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 601–617, 2018. 1