DiffMesh: A Motion-aware Diffusion Framework for Human Mesh Recovery from Videos (*Supplementary Material*)

1. Overview

The supplementary material is organized into the following sections:

- Section 2: More related Work and implementation details.
- Section 3: Mathematical proof and more experiments about the number of input frames, the two-stream transformer network, initial distributions, and the auxiliary loss.
- Section 4: More human mesh visualization.
- Section 5: Broader impact and limitation

2. More Related Work and Implementation Details

2.1. Related Work

The majority of methods [4, 10, 13, 14, 31, 37–39] for HMR rely on a parametric human model, such as SMPL [21], to reconstruct the mesh by estimating pose and shape parameters. As a fundamental HMR work, SPIN [13] combines regression and optimization in a loop, where the regressed output serves as better initialization for optimization (SMPLify). METRO [20] is the first transformer-based method that models vertex-vertex and vertex-joint interaction using a transformer encoder after extracting image features with a CNN backbone. HybrIK [17] and HybrIK-X [15] present novel hybrid inverse kinematics approaches that transform 3D joints to body-part rotations via twist-and-swing decomposition. Lin et al. [19] propose a one-stage pipeline for 3D whole-body (body, hands, and face) mesh recovery. PyMAF [36] and its extension work PyMAF-X [35] capitalize on a feature pyramid to rectify predicted parameters by aligning meshes with images, extracting mesh-aligned evidence from finer-resolution features. CLIFF [18] enhances holistic feature representation by incorporating bounding box information into cropped-image features. It employs a 2D reprojection loss considering the full frame and leverages global-location aware supervision to directly predict global rotation and more accurately articulated poses. ReFit [29] proposes a feedback-update loop reminiscent of solving inverse problems via optimization, iteratively reprojections keypoints from the human model to feature maps for feedback, and utilizes a recurrent-based updater to refine the model's fit to the image. HMR2.0 [7] develops a system that can simultaneously reconstruct and track humans from video, but only reports the frame-based results for the HMR task without considering temporal information. Foo et al. [6] first introduce a diffusion-based approach for recovering human mesh from a single image. The recovered human mesh is obtained by the reverse diffusion process. However, when applied to video sequences, these image-based methods suffer from severe motion jitter due to frame-by-frame reconstruction, making them unsuitable for practical use.

Compared to image-based HMR methods, video-based methods [32–34] utilize temporal information to enhance motion smoothness from video input. In addition to the methods [3, 12, 22, 23, 30, 40] introduced in the main paper, there are several other noteworthy approaches for video-based HMR. Kanazawa et al. [11] first propose a convolutional network to learn human motion kinematics by predicting past, current, and future frames. Based on [11], Sun et al. [26] further propose a self-attention-based temporal model to improve performance. DND [16] utilizes inertial forces control as a physical constraint to reconstruct 3D human motion. GLoT [25] adopts a novel approach by decoupling the modeling of short-term and long-term dependencies using a global-to-local transformer. PMCE [32] follows a two-step process, where it first estimates 3D human pose and then regresses the mesh vertices through a co-evaluation decoder that takes into account the interactions between pose and mesh.

2.2. Datasets

3DPW [27] is a dataset that captures outdoor and in-the-wild scenes using a hand-held camera and a set of inertial measurement unit (IMU) sensors attached to body limbs. The ground-truth SMPL parameters are computed based on the returned values. This dataset includes 60 videos of varying lengths, and we use the official split to train and test the model. The split comprises 24, 12, and 24 videos for the training, validation, and test sets, respectively. The MPJPE, PA-MPJPE, MPJVE, and ACC-ERR are reported when evaluating this dataset.

Human3.6M [9] is a large-scale benchmark for the indoor 3D human pose. It includes 15 action categories and 3.6M video frames. Following [3, 12, 30], we use five subjects (S1, S5, S6, S7, S8) for the training set and two subjects (S9, S11) for the testing set. The dataset is subsampled from its original 50 fps to 25 fps for both training and evaluation purposes. When calculating MPJPE and PA-MPJPE, only 14 joints are selected for a fair comparison to the previous works.

MPI-INF-3DHP [24] is a 3D benchmark that consists of both indoor and outdoor environments. The training set includes 8 subjects, with each subject having 16 videos, resulting in a total of 1.3M video frames captured at 25 fps. The markerless motion capture system is used for providing 3D human pose annotations. The test set comprises 6 subjects performing 7 actions in both indoor and outdoor environments. Following [3, 12, 30], the MPJPE and PA-MPJPE are measured on valid frames, which include approximately every 10th frame, using 17 joints defined by MPI-INF3DHP. The ACC-ERR is computed using all frames.

InstaVariety [11] is a 2D human dataset curated by HMMR [11] , comprising videos collected from Instagram using 84 motion-related hashtags. The dataset contains 28K videos with an average length of 6 seconds, and pseudo-ground truth 2D pose annotations are acquired using OpenPose [2].

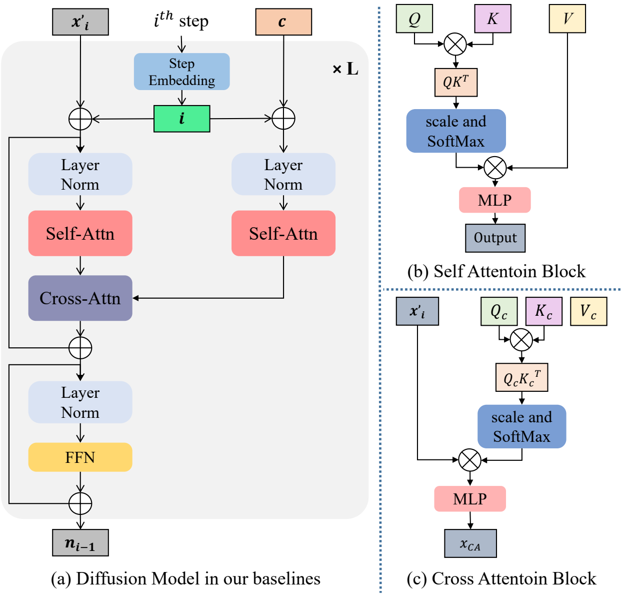
PoseTrack [1] is a 2D benchmark designed for multi-person pose estimation and tracking in videos. This dataset comprises 1.3K videos and 46K annotated frames, captured at varying fps around 25 fps. There are 792 videos used for the official train set, which includes 2D pose annotations for 30 frames located in the middle of each video.

2.3. Loss Function

Our DiffMesh relies on the SMPL model [21] to reconstruct the human mesh. The SMPL model can generate the body mesh $M \in \mathbb{R}^{N \times 3}$ with $N = 6890$ vertices by taking in the predicted pose parameters θ and the shape parameters β as inputs, which can be expressed as $M = SMPL(\theta, \beta)$. Once the body mesh M is obtained, the body joints J can be estimated by applying the predefined joint regression matrix W, i.e., $J \in \mathbb{R}^{k \times 3} = W \cdot M$, where k represents the number of joints. We adopt the same loss function as previous methods TCMR [3].

$$\mathcal{L}_{HMR} = w_1 \|\beta - \beta^*\| + w_2 \|\theta - \theta^*\| + w_3 \|J - J^*\| \tag{1}$$

where * denote the ground-truth value, $w_1 = 0.5$, $w_2 = 10$, and $w_3 = 1000$.



(b) Self Attentoin Block

(a) Diffusion Model in our baselines

(c) Cross Attentoin Block

Figure 1. The diffusion model in our baselines

Besides this general loss for mesh recovery, we add additional auxiliary loss as mentioned in Section 3.4 of the main paper. Our designed transformer-based diffusion model can predict the previous conditional feature $\hat{c}_{i-1}$ given the current conditional feature input c_i. A MSE loss is applied between the ground truth c_{i-1} and predicted $\hat{c}_{i-1}$:

$$\mathcal{L}_{aux} = \|c_{i-1} - \hat{c}_{i-1}\|_2^2 \tag{2}$$

This auxiliary loss contributes to the refinement of our transformer-based diffusion model during the training process. Thus, the overall loss for our DiffMesh is the sum of the $\mathcal{L}_{HMR}$ and $\mathcal{L}_{aux}$:

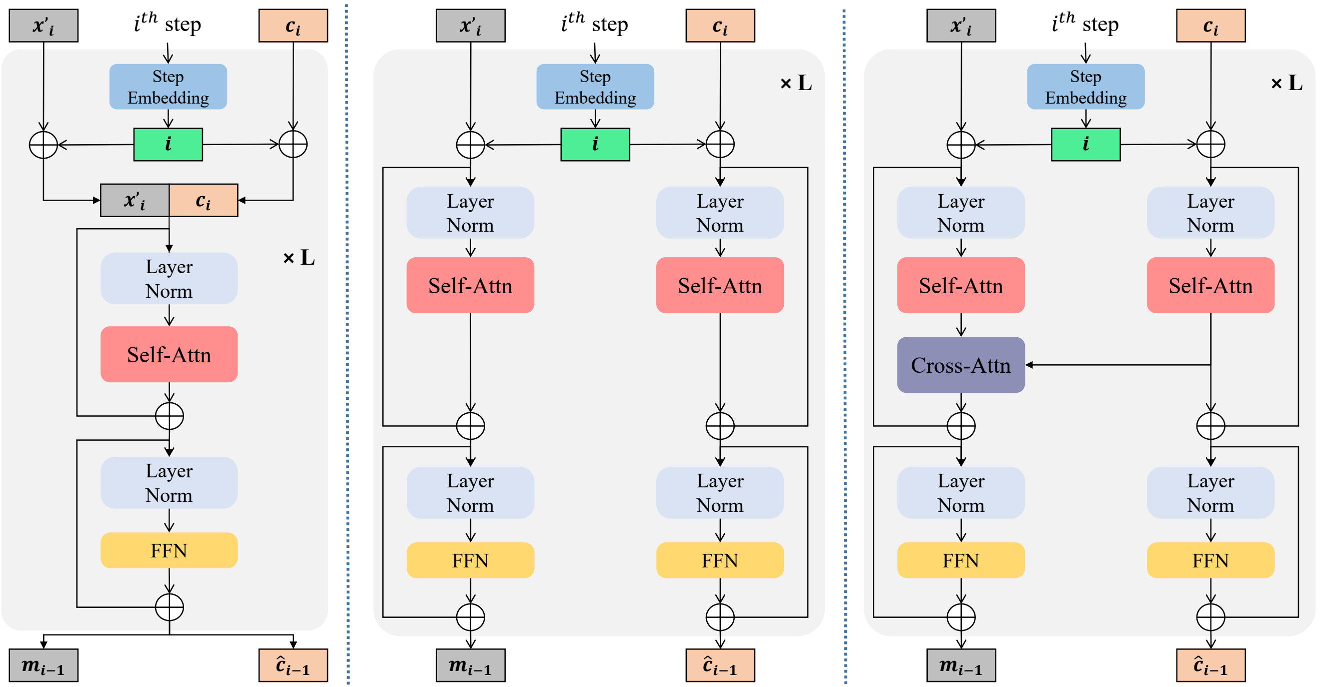
$$\mathcal{L}_{overall} = \mathcal{L}_{HMR} + w_4 \mathcal{L}_{aux} \tag{3}$$

where $w_4 = 0.01$.

2.4. More Details about the Architecture

Diffusion model in our baselines: The architecture of the diffusion model employed in our baselines is illustrated in Fig. 1. It shares similarities with the architecture within our DiffMesh, featuring two self-attention blocks designed to capture global dependencies and one cross-attention block focused on integrating information between the denoising input x_i and the constant conditional feature c. In the baseline approach, as the conditional feature c remains the same throughout the denoising process, there is no need to estimate the conditional feature for each subsequent denoising step. Thus, it only return the estimated noise term n_{i-1}.

Conditional features generation block: Our chosen backbone to extract features for both our proposed method and the baselines is ResNet-50 [8] or DSTformer [40].



(a) One Self-Attn Block

(b) Two Self-Attn Blocks

(c) Two Self-Attn Blocks and one Cross-Attn Block

Figure 2. Different design choices of the transformer architecture: (a) Only one Self-Attn Block. (b) Two Self-Attn Blocks. (c) Two Self-Attn Blocks and one Cross-Attn Block.

After extracting features from each frame b_i, where $i \in \{1, \dots f\}$, using the backbone, our goal is to generate N conditional features to be utilized during the reverse process. To achieve this, we pad additional $N - f$ zero features, $b_{f+1}, \cdots, b_N$. Then, we combine them with the existing features, creating new features $b \in \mathbb{R}^{N \times D}$, where D represents the embedded dimension. Subsequently, we apply a transformer block [5] to model these features and return the required conditional features denoted as $c \in \mathbb{R}^{N \times D}$.

3. Mathematical Proof and More Experiments

3.1. Mathematical Proof of modeling human motion as noise in diffusion model

Our approach draws an analogy between human motion and noise, treating the motion between adjacent frames as a structured form of noise. By operating in a high-dimensional latent space, we capture the complexity of human motion, where small perturbations (or ”noise”) in this space can be modeled as Gaussian. This allows us to align the problem with the core principles of diffusion models.

Following Equation 6 in the main paper, we have

$$\mathbb{E}_{x_0 \sim q}\left[-\log p(x_0)\right] = \mathbb{E}_{x_0 \sim q}\left[\log \mathbb{E}_{x_{1:T}, y_{0:T} \sim q} \frac{q(x_{1:T}, y_{0:T} | x_0)}{p(x_{0:T}, y_{0:T})}\right] \quad (4)$$

Using Jensen’s Inequality, we can bound Eq 4 by moving the expectation inside the logarithm:

$$\mathbb{E}_{x_0 \sim q}\left[-\log p(x_0)\right] \leq \mathbb{E}_{x_0, x_{1:T}, y_{0:T} \sim q}\left[\log \frac{q(x_{1:T}, y_{0:T} | x_0)}{p(x_{0:T}, y_{0:T})}\right] \quad (5)$$

The forward process in DiffMesh is designed to model the human motion between adjacent frames as structured noise. This noise is Gaussian in the latent space, where the human motion patterns are simplified. We define the forward process for the latent motion as:

$$q(m_{t+1} | m_t) = \mathcal{N}(m_{t+1}; \sqrt{1 - \beta_t} m_t, \beta_t I) \quad (6)$$

Here m_t represents the motion noise at time step t, and β_t controls the level of noise added to the motion between consecutive frames. This equation models the motion between frames as Gaussian perturbations in latent space.

We now need to decompose the joint probabilities $q(x_{1:T}, y_{0:T} | x_0)$ and $p(x_{0:T}, y_{0:T})$ into their respective transition probabilities over time t:

The forward process $q(x_{1:T}, y_{0:T} | x_0)$ can be written as:

$$q(x_{1:T}, y_{0:T} | x_0) = q(x_T | x_0) \prod_{t=2}^{T} q(x_{t-1} | x_t, x_0) \prod_{t=0}^{T} q(y_t | x_t) \quad (7)$$

Similarly, the reverse process $p(x_{0:T}, y_{0:T})$ is:

$$p(x_{0:T}, y_{0:T}) = p(x_T) \prod_{t=T}^{1} p(x_{t-1} | x_t) \prod_{t=T}^{0} p(y_t | x_t) \quad (8)$$

Next, we focus on deriving the bound based on KL divergence. Substituting the decomposed expressions, we obtain two main terms: one for the states x and another for the observations y

$$\mathbb{E}_{x_0,x_{1:T},y_{0:T}\sim q}\left[\log \frac{q(x_T|x_0)\prod_{t=2}^{T} q(x_{t-1}|x_t,x_0)}{p(x_T)\prod_{t=T}^{1} p(x_{t-1}|x_t)}\right] + \mathbb{E}_{x_0,x_{1:T},y_{0:T}\sim q}\left[\log \frac{\prod_{t=0}^{T} q(y_t|x_t)}{\prod_{t=T}^{0} p(y_t|x_t)}\right] \tag{9}$$

Each of these terms corresponds to the forward process (adding Gaussian motion noise) and the reverse process (denoising to recover the original human motion).

The final step is to express the result as a sum of KL divergences. For the forward and reverse processes of both the states x and the observations y, we can represent this as:

$$= D_{\mathrm{KL}}(q(x_T|x_0)||p(x_T)) + \mathbb{E}_q\left[-\log p(x_0|x_1)\right] + \sum_{t=2}^{T} D_{\mathrm{KL}}(q(x_{t-1}|x_t,x_0)||p(x_{t-1}|x_t)) + \sum_{t=0}^{T} D_{\mathrm{KL}}(q(y_t|x_t)||p(y_t|x_t)) \tag{10}$$

Here $D_{\mathrm{KL}}(q(x_T|x_0)||p(x_T))$ measures the divergence between the final denoised human mesh and the actual motion, and $D_{\mathrm{KL}}(q(x_{t-1}|x_t,x_0)||p(x_{t-1}|x_t))$ measures the divergence at each intermediate step in recovering the motion, helping to maintain temporal consistency. Thus, we can derive Equation 7 in the main paper.

3.2. Performance on MPI-INF-3DHP dataset

	MPI-INF-3DHP		
Methods	MPJPE ↓	PA-MPJPE ↓	ACC-ERR ↓
VIBE [12]	103.9	68.9	27.3
TCMR [3]	97.6	63.5	8.5
MAED [28]	83.6	56.2	-
MPS-Net [30]	96.7	62.8	9.6
GLoT [25]	93.9	61.5	7.9
DiffMesh (ours)	**78.9**	**54.4**	**7.0**

Table 1. Performance comparison with state-of-the-art methods on MPI-INF-3DHP dataset. All methods use pre-trained ResNet-50 [8] (fixed weights) to extract features except MAED.

To conduct experiments on the MPI-INF-3DHP [24] dataset, we follow the same setting as VIBE [12], TCMR [3], and MPS-Net [30] . The input features of each frame are extracted from ResNet-50 [8] without fine-tuning for fair comparisons. The results are shown in Fig. 1. Our DiffMesh consistently outperforms previous methods with significant improvement (more than 5.9 mm ↓ of MPJPE, 1.8 ↓ of PA-MPJPE, and 0.7 ↓ of ACC-ERR). This showcases the remarkable performance enhancement achieved by our approach, highlighting its potential as a state-of-the-art solution for video-based human mesh recovery across various datasets and real-world applications.

3.3. Effectiveness of the number of input frames and additional steps

Following the same setting as previous video-based methods such as VIBE [12], TCMR [3], and MPS-Net [30], the number of input frames f is set to be 16. To further investigate the impact of the number of input frames, we conduct experiments on the 3DPW dataset given the different number of input frames. The results are shown in Table. 2.

In general, the performance can be improved (lower MPJPE, PA-MPJPE, MPVPE, and ACC-ERR) when the number of input frames f is increased. Specifically, when maintaining the total number of steps N at 30 and varying f from 8 to 16 to 24, the improvements are notable. In our ablation study, the lowest MPVE, MPJPE, and ACC-ERR are achieved when $f = 32$ with total steps of 40.

To strike an optimal balance between efficiency and performance, it's crucial to seek improved results with a reduced total number of steps N. For instance, when $f = 16$, the optimal N is determined to be 30, demonstrating comparable results to $N = 40$ at a faster processing speed. Similarly, for $f = 24$, the optimal N is identified as 30 based on the results.

input frames	steps for output sequence	additional steps	Total steps	MPVE ↓	MPJPE ↓	ACC-ERR ↓
8	7	0	7	89.8	77.9	6.9
16	15	0	15	88.5	77.4	6.5
24	23	0	23	87.6	75.2	6.2
8	7	13	20	88.6	76.9	6.7
16	15	5	20	88.0	77.1	6.5
8	7	23	30	87.4	76.5	6.5
16	15	15	30	86.4	75.7	6.1
24	23	7	30	86.2	74.7	5.9
16	15	25	40	87.1	75.6	6.2
24	23	17	40	86.5	74.7	6.1
32	31	8	40	86.0	74.9	5.8

Table 2. Performance of the different number of input frames and the number of additional steps on the 3DPW dataset.

3.4. Different design choices of our transformer-based diffusion model

As introduced in Section 3.3 of the main paper, our proposed transformer-based diffusion model consists of two self-attn blocks with one cross-attn block (also depicted in Fig. 2 (c)). Given the input feature x_i' and corresponding conditional feature c_i, the transformer-based diffusion model produces the predicted noise m_{i-1} and the predicted previous conditional feature $\hat{c}_{i-1}$. We apply two self-attention blocks for x_i' and c_i separately, then a cross-attention block is adopted to fuse the conditional features with mesh features. To validate the effectiveness, we compare this design with (a): a self-attention block applied for the concatenated features; and (b) two two self-attention blocks for x_i' and c_i separately without cross-attention block. The results are shown in Table 3. Clearly, our design (c) in DiffMesh outperforms (a) and (b) for all evaluation metrics on the 3DPW dataset due to enhanced information integration using two-stream and cross-attention fusion design.

	3DPW			
	MPVE ↓	MPJPE ↓	PA-MPJPE ↓	ACC ↓
(a) one self-attn	86.9	76.9	47.5	6.3
(b) two self-attn	87.4	76.4	45.9	6.2
(c) self-attn and cross attn	**86.4**	**75.7**	**45.6**	**6.1**

Table 3. Ablation study of transformer block design on 3DPW dataset.

3.5. Effectiveness of the auxiliary loss:

To validate the effectiveness of our proposed auxiliary loss, we compare the results as shown in Table 4, which demonstrated that our proposed auxiliary loss can help to improve the reconstruction performance (MPJPE, PA-MPJPE, and MPJVE) and the motion smoothness (ACC-ERR).

	3DPW			
loss	MPVPE↓	MPJPE↓	PA-MPJPE↓	Accel↓
Without $\mathcal{L}_{aux}$	86.8	76.0	47.1	6.2
With $\mathcal{L}_{aux}$	**86.4**	**75.7**	**45.6**	**6.1**

Table 4. Evaluation of the combinations of loss functions on the 3DPW dataset.

3.6. Inference Time Analysis:

Methods like MPS-Net [30] and GLoT [25] only estimate the human mesh of the center frame given 16 frames as their input. Considering these methods can extract all features by their backbone once and then utilize batch processing to accelerate the inference speed, we provide a more thorough inference time comparison in Table 5.

In this experiment, the video input comprises a total of 64 frames. Upon feature extraction from the backbone (with the shape of $[64, 2048]$), MPS-Net and GLoT require the creation of 64 batch input tubes $[64, 16, 2048]$ through padding and sliding window. Since their models only return the output mesh of the center frame, the output would be $[64, 1, 6890, 3]$, indicating output mesh vertices $[6890, 3]$ across 64 frames. In contrast, our DiffMesh just needs to reshape the input $[64, 2048]$ into 4 batches, resulting in the shape of $[4, 16, 2048]$. Consequently, the output of DiffMesh is $[4, 16, 6890, 3]$, which is then reshaped back to $[64, 6890, 3]$. Based on the total processing time, our DiffMesh is more efficient than MPS-Net [30] and GLoT [25] since DiffMesh can output human meshes of all input frames.

4. Human Mesh Visualization

We first visualize the qualitative comparison on the 3DPW [27] dataset in Fig. 3. The circle areas highlight locations where our DiffMesh performs better than GLoT [25].

In our experimental setup, we utilize 16 input frames, and the total number of steps is set to 30. In the reverse motion process, DiffMesh outputs $[y_1, y_2 \cdots, y_{30}]$ over 30 steps. For the output mesh sequence of 16 frames, we use $[y_1, y_2 \cdots, y_{16}]$. Additionally, we generate the mesh from $[y_{16}, y_{17} \cdots, y_{30}]$, as visually depicted in Fig 4. This visualization illustrates the trend of the generated human mesh gradually decoding toward the desired human mesh of each input frame.

Furthermore, we show the qualitative results of DiffMesh on **in-the-wild videos** in Fig. 5. We observe that DiffMesh demonstrates remarkable performance in reconstructing more reliable human mesh sequences with temporal consistency compared to previous methods. Please refer to our **video demo** for the more reconstructed mesh sequence results.

5. Broader impact and limitation

DiffMesh establishes an innovative connection between diffusion models and human motion, facilitating the generation of accurate and temporal smoothness output mesh sequences by integrating human motion into both the forward and reverse processes of the diffusion model. By enabling direct 3D human mesh reconstruction from 2D video sequences, DiffMesh eliminates the dependency on additional motion sensors and equipment, thereby streamlining the process and reducing costs.

However, despite its advancements, DiffMesh is not without limitations. Similar to previous methods, DiffMesh may face challenges in scenarios with substantial occlusions, resulting in the production of unrealistic mesh outputs. To address this issue, further exploration into spatial-temporal interactions within the human body is warranted, serving as a focal point for our future research. Additionally, DiffMesh may encounter difficulties in rare and complex pose scenarios due to the constraints of limited training data, highlighting the necessity for ongoing development and refinement efforts.

Video-based Methods	total frames for video input	Backbone	features after backbone are reshaped for model processing	Output shape	processing time (without backbone time)
MPS-Net [30]	64	ResNet50 [8]	[64,2048] to [64,16,2048]	[64,1,6890,3] to [64,6890,3]	1.04 s
GLoT [25]	64	ResNet50 [8]	[64,2048] to [64,16,2048]	[64,1,6890,3] to [64,6890,3]	1.17 s
DiffMesh (ours)	64	ResNet50 [8]	[64,2048] to [4,16,2048]	[4,16,6890,3] to [64,6890,3]	0.34 s

Table 5. Inference time comparison on 3DPW dataset between our DiffMesh and previous video-based HMR methods with the same hardware platform (single NVIDIA A5000 GPU is used).

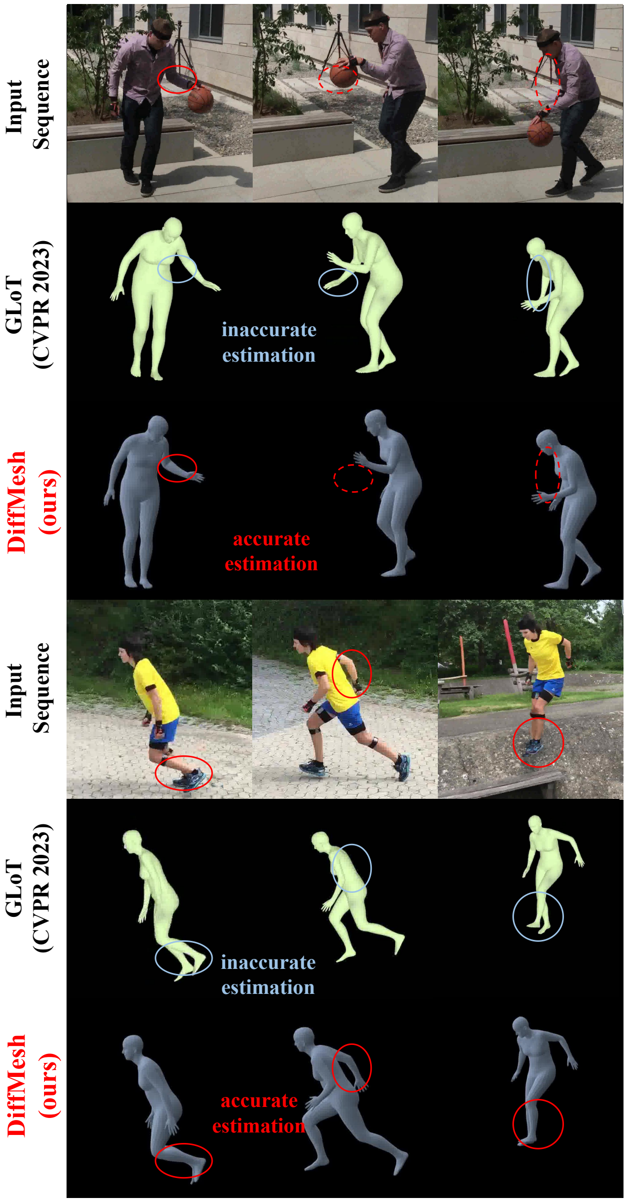


Figure 3. Qualitative comparison on the 3DPW dataset

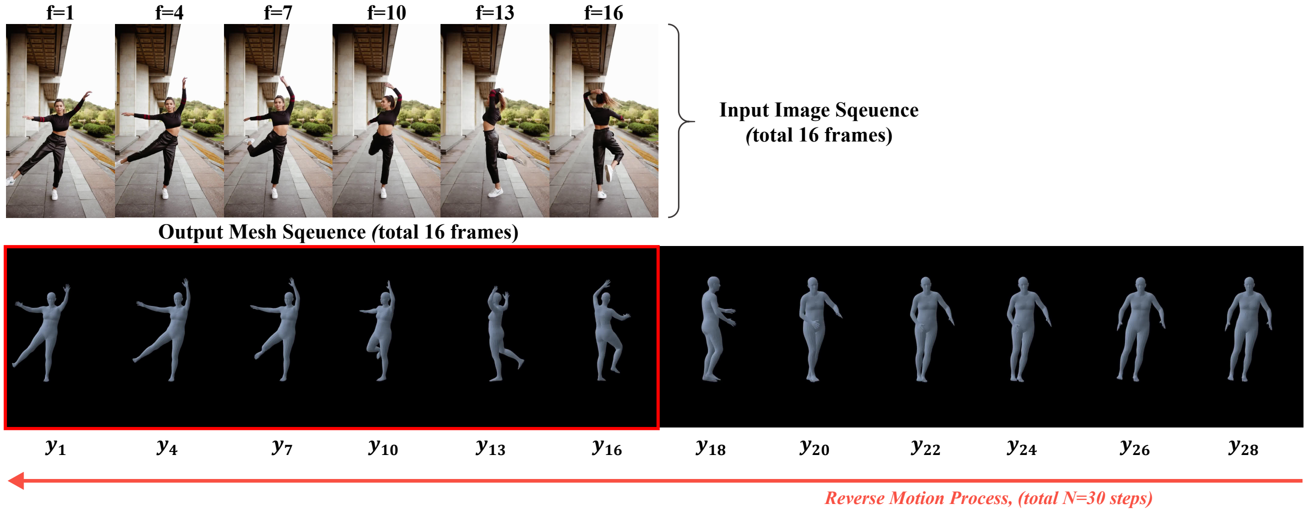


Figure 4. Visualization of decoding steps during the reverse motion process.

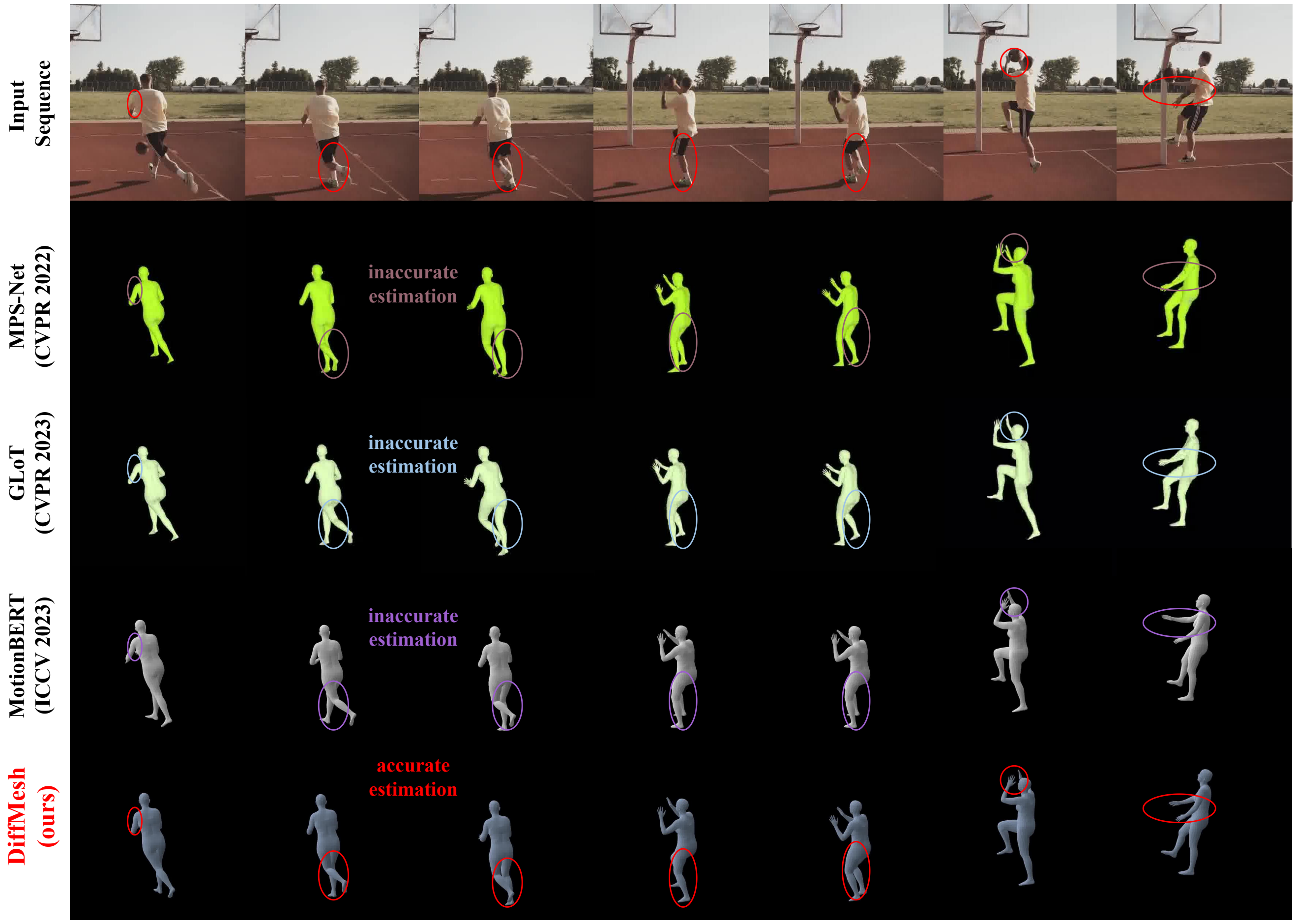


Figure 5. Other qualitative results of our DiffMesh on in-the-wild videos. Please refer to our **video demo** for the more reconstructed mesh sequences results.

References

[1] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5167–5176, 2018. 2

[2] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 2

[3] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3d human pose and shape from a video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1964–1973, 2021. 1, 2, 4

[4] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *ECCV*, 2020. 1

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 3

[6] Lin Geng Foo, Jia Gong, Hossein Rahmani, and Jun Liu. Distribution-aligned diffusion for human mesh recovery. *arXiv preprint arXiv:2308.13369*, 2023. 1

[7] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa*, and Jitendra Malik*. Humans in 4D: Reconstructing and tracking humans with transformers. In *International Conference on Computer Vision (ICCV)*, 2023. 1

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 4, 6

[9] 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, jul 2014. 2

[10] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 1

[11] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5614–5623, 2019. 1, 2

[12] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, 2020. 1, 2, 4

[13] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2252–2261, 2019. 1

[14] Jiefeng Li, Siyuan Bian, Qi Liu, Jiasheng Tang, Fan Wang, and Cewu Lu. Niki: Neural inverse kinematics with invertible neural networks for 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12933–12942, 2023. 1

[15] Jiefeng Li, Siyuan Bian, Chao Xu, Zhicun Chen, Lixin Yang, and Cewu Lu. Hybrik-x: Hybrid analytical-neural inverse kinematics for whole-body mesh recovery. *arXiv preprint arXiv:2304.05690*, 2023. 1

[16] Jiefeng Li, Siyuan Bian, Chao Xu, Gang Liu, Gang Yu, and Cewu Lu. D&d: Learning human dynamics from dynamic camera. In *European Conference on Computer Vision*, 2022. 1

[17] 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

[18] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. Cliff: Carrying location information in full frames into human pose and shape estimation. In *European Conference on Computer Vision*, pages 590–606. Springer, 2022. 1

[19] Jing Lin, Ailing Zeng, Haoqian Wang, Lei Zhang, and Yu Li. One-stage 3d whole-body mesh recovery with component aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21159–21168, 2023. 1

[20] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1954–1963, 2021. 1

[21] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM TOG*, 2015. 1, 2

[22] Zhengyi Luo, S Alireza Golestaneh, and Kris M Kitani. 3d human motion estimation via motion compression and refinement. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1

[23] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes. In *ICCV*, 2019. 1

[24] 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 *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017. 2, 4

[25] Xiaolong Shen, Zongxin Yang, Xiaohan Wang, Jianxin Ma, Chang Zhou, and Yi Yang. Global-to-local modeling for video-based 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8887–8896, 2023. 1, 4, 5, 6

[26] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, Yili Fu, and Tao Mei. Human mesh recovery from monocular images via

a skeleton-disentangled representation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5349–5358, 2019. 1

[27] Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018. 1, 5

[28] Ziniu Wan, Zhengjia Li, Maoqing Tian, Jianbo Liu, Shuai Yi, and Hongsheng Li. Encoder-decoder with multi-level attention for 3d human shape and pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13033–13042, 2021. 4

[29] Yufu Wang and Kostas Daniilidis. Refit: Recurrent fitting network for 3d human recovery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14644–14654, 2023. 1

[30] Wen-Li Wei, Jen-Chun Lin, Tyng-Luh Liu, and Hong-Yuan Mark Liao. Capturing humans in motion: temporal-attentive 3d human pose and shape estimation from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13211–13220, 2022. 1, 2, 4, 5, 6

[31] Yingxuan You, Hong Liu, Xia Li, Wenhao Li, Ti Wang, and Runwei Ding. Gator: Graph-aware transformer with motion-disentangled regression for human mesh recovery from a 2d pose. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023. 1

[32] Yingxuan You, Hong Liu, Ti Wang, Wenhao Li, Runwei Ding, and Xia Li. Co-evolution of pose and mesh for 3d human body estimation from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1

[33] Ailing Zeng, Xuan Ju, Lei Yang, Ruiyuan Gao, Xizhou Zhu, Bo Dai, and Qiang Xu. Deciwatch: A simple baseline for 10× efficient 2d and 3d pose estimation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*, pages 607–624. Springer, 2022. 1

[34] Ailing Zeng, Lei Yang, Xuan Ju, Jiefeng Li, Jianyi Wang, and Qiang Xu. Smoothnet: a plug-and-play network for refining human poses in videos. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*, pages 625–642. Springer, 2022. 1

[35] Hongwen Zhang, Yating Tian, Yuxiang Zhang, Mengcheng Li, Liang An, Zhenan Sun, and Yebin Liu. Pymaf-x: Towards well-aligned full-body model regression from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[36] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *Proceedings of the IEEE International Conference on Computer Vision*, 2021. 1

[37] Ce Zheng, Xianpeng Liu, Guo-Jun Qi, and Chen Chen. Potter: Pooling attention transformer for efficient human mesh recovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1611–1620, 2023. 1

[38] Ce Zheng, Matias Mendieta, Pu Wang, Aidong Lu, and Chen Chen. A lightweight graph transformer network for human mesh reconstruction from 2d human pose. *arXiv preprint arXiv:2111.12696*, 2021. 1

[39] Ce Zheng, Matias Mendieta, Taojiannan Yang, Guo-Jun Qi, and Chen Chen. Feater: An efficient network for human reconstruction via feature map-based transformer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[40] 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*, 2023. 1, 2