Co-Evolution of Pose and Mesh for 3D Human Body Estimation from Video –Supplemental Material–

Yingxuan You[1] Hong Liu[1 ✉] Ti Wang[1] Wenhao Li[1] Runwei Ding[1] Xia Li[2]
[1]Key Laboratory of Machine Perception, Shenzhen Graduate School, Peking University
[2]Department of Computer Science, ETH Zürich
{youyx,tiwang}@stu.pku.edu.cn, {hongliu,wenhaoli,dingrunwei}@pku.edu.cn, xia.li@inf.ethz.ch

This supplemental material contains the following parts:

(A) The architecture of 3D pose estimation stream.
(B) Additional quantitative results.
(C) Additional ablation study.
(D) Details about loss functions.
(E) Additional visualization results.

A. Architecture of 3D Pose Estimation Stream

Figure A shows the detailed architecture of the 3D pose estimation stream. Firstly, the normalized 2D pose sequence is projected to high-dimensional joint features by a linear embedding layer. Secondly, we project and expand the static image features, which are added to their corresponding joint features in the same frame. Then we add the spatial and temporal embeddings to joint features and feed joint features to the spatial-temporal Transformer, which consists of cascaded spatial and temporal parts. In the spatial part, the spatial MSA calculates the similarities between joint tokens in the same frame. In the temporal part, the joint features are reshaped from $(T \times J \times C_1)$ to $(J \times T \times C_1)$, and thus the temporal MSA can calculate the similarities between frame tokens of the same joint. Finally, the joint features are regressed from C_1 to 3 and fused from T frames to one frame to get the mid-frame 3D pose.

B. Additional Quantitative Results

Comparison with Single RGB-Based Methods. Table A compares our PMCE with single RGB-based methods on the 3DPW dataset. All methods use ResNet as the backbone. We evaluate the models trained with and without 3DPW training set for fair comparisons. Single RGB-based methods focus on per-frame accuracy and propose advanced networks to extract image features [1, 5, 10, 13] and generate human mesh, which shows high performance. In contrast, our PMCE takes pre-trained backbone [6] to extract feature vectors following previous video-based methods [2, 4, 12]. Compared to the single RGB-based methods, our PMCE achieves competitive performance in PA-MPJPE and outperforms the state-of-the-art method in the metrics of MPJPE, PVE, and ACCEL. The results demonstrate the superiority and effectiveness of our pose and mesh co-evolution design in terms of both per-frame accuracy and temporal consistency for 3D human motion estimation.

Table A: Comparison with single RGB-based methods. All methods use ResNet as the backbone. '†' represents training w/o 3DPW training dataset. '∗' represents training with 3DPW training set. The top two best results are highlighted in bold and underlined, respectively.

	Method	3DPW			
		MPJPE ↓	PA-MPJPE ↓	PVE ↓	ACCEL ↓
RGB-based	HMR† [3]	130	76.7	-	37.4
	GraphCMR† [7]	-	70.2	-	-
	SPIN† [6]	96.9	59.2	116.4	29.8
	I2L-MeshNet† [10]	93.2	57.7	110.1	30.9
	PyMAF† [13]	92.8	58.9	110.1	-
	PARE† [5]	82.9	52.3	99.7	-
	ROMP∗ [11]	79.7	49.7	94.7	-
	METRO∗ [1]	77.1	47.9	88.2	-
	CLIFF∗ [8]	72.0	**45.7**	85.3	24.7
	PMCE (Ours)†	81.6	52.3	99.5	6.8
	PMCE (Ours)∗	**69.5**	46.7	**84.8**	**6.5**

Table B: Generalization evaluation in unseen views on Human3.6M. The test view is View 4.

Training views	Only-pose model		PMCE		Improvements	
	MPJPE ↓	PVE ↓	MPJPE ↓	PVE ↓	MPJPE	PVE
1	161.7	165.3	82.9	89.4	78.8	75.9
1, 2	100.2	112.7	59.2	69.9	40.9	42.8
1, 2, 3	85.8	96.0	58.4	67.1	27.4	28.9

Generalization in Unseen Views. Our method decouples 2D poses and image features from image sequences, which can not only provide complementary pose and shape information for better mesh estimation but also improve the generalization. To verify the latter, we compare our PMCE with

✉ Corresponding author.

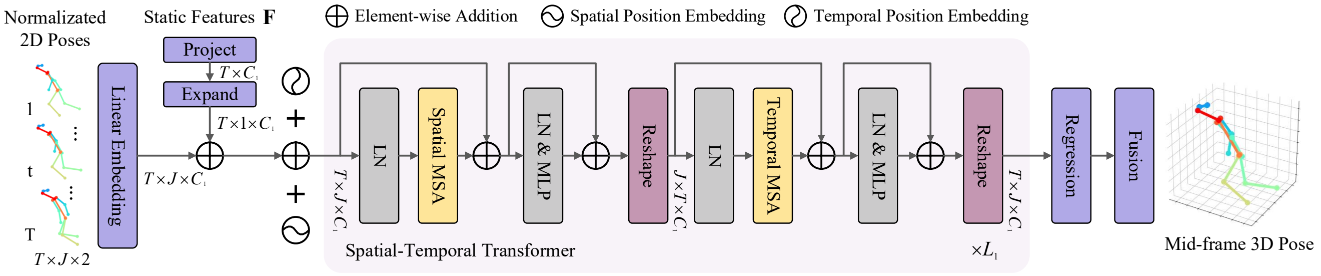

Figure A: Architecture of 3D Pose Estimation Stream.

Table C: Performance comparison between different initializations of mesh vertices on 3DPW.

Mesh initialization	MPJPE ↓	PA-MPJPE ↓	PVE ↓
Zeros	72.3	48.5	88.5
Template	71.4	47.6	86.2
Nearest joints (Ours)	**69.5**	**46.7**	**84.8**

the only-pose model (PMCE without using the image features) on the Human3.6M dataset. Specifically, based on the four camera views of Human3.6M, we train the networks on View 1, View 2, and View 3, then test them on the unseen View 4 to evaluate their generalization in unseen views. As shown in Table B, the only-pose model suffers from the domain gap between training and testing views, especially when training with few view data (top line). In contrast, our PMCE has better generalization ability and improves performance by a large margin. The results indicate that our method, complementing the pose information and image features, is effective for a robust mesh estimation in unseen views.

C. Additional Ablation Study

Impact of Mesh Initializations. Mesh initialization serves as a human body prior in our method. Table C examines the impact of different mesh initializations, including setting mesh vertices to zeros, using T-shape template mesh from SMPL [9] or setting the position of per mesh vertex as that of its nearest joint in estimated 3D pose P_0 (the distances between vertices and joints are pre-calculated from the template mesh and pose provided by SMPL). Compared with template mesh, setting vertices to their nearest joints makes the initialized mesh closer to the final mesh, which can provide a more precise human body prior and contribute to the final mesh performance.

D. Loss Functions

For the 3D pose estimation stream, we use the L1 joint loss to supervise the intermediate 3D pose P_0, which is defined as follows:

$$\mathcal{L}_{joint}^{int} = \frac{1}{J}\sum_{i=1}^{J} \left\| P_{gt} - P_0 \right\|_1 . \tag{A}$$

After training the 3D pose estimation stream, we train the whole network using the following four loss functions.

Mesh Loss. We use the L1 loss between the ground truth 3D mesh vertices $M_{gt} \in \mathbb{R}^{V\times 3}$ and the predicted 3D mesh vertices $M \in \mathbb{R}^{V\times 3}$. The mesh vertex loss is calculated as:

$$\mathcal{L}_{mesh} = \frac{1}{V}\sum_{i=1}^{V} \left\| M_{gt} - M \right\|_1 . \tag{B}$$

Joint Loss. We multiply the predicted 3D mesh M by a predefined matrix $\mathcal{J} \in \mathbb{R}^{J\times V}$ to obtain the regressed 3D joints and calculate the joint loss with ground truth 3D joints P_{gt}:

$$\mathcal{L}_{joint} = \frac{1}{J}\sum_{i=1}^{J} \left\| P_{gt} - \mathcal{J}M \right\|_1 . \tag{C}$$

Surface Normal Loss. This loss is used to improve surface smoothness and local details. It is calculated by the normal vectors of the ground truth mesh and the predicted mesh:

$$\mathcal{L}_{normal} = \sum_{f} \sum_{\{i,j\}\subset f} \left| \left\langle \frac{m_i - m_j}{\left\| m_i - m_j \right\|_2}, n_{gt} \right\rangle \right| , \tag{D}$$

where f denotes a triangle face in the mesh, m_i and m_j denote the i_{th} and j_{th} mesh vertices of the triangle face respectively. And n_{gt} is the unit normal vector of the triangle face f in the ground truth mesh.

Surface Edge Loss. This loss is used to improve the smoothness of the areas with dense vertices, *e.g.*, hands and feet. The edge length consistency loss is calculated by the ground truth edges and the predicted edges as:

$$\mathcal{L}_{edge} = \sum_{f} \sum_{\{i,j\}\subset f} \left| \left\| m_{gt_i} - m_{gt_j} \right\|_2 - \left\| m_i - m_j \right\|_2 \right| . \tag{E}$$

Given the four loss functions, the final loss is calculated as the weighted sum:

$$\mathcal{L} = \lambda_m \mathcal{L}_{mesh} + \lambda_j \mathcal{L}_{joint} + \lambda_n \mathcal{L}_{normal} + \lambda_e \mathcal{L}_{edge}, \tag{F}$$

where λ_m=1, λ_j=1, λ_n=0.1, λ_e=20 in the experiments.

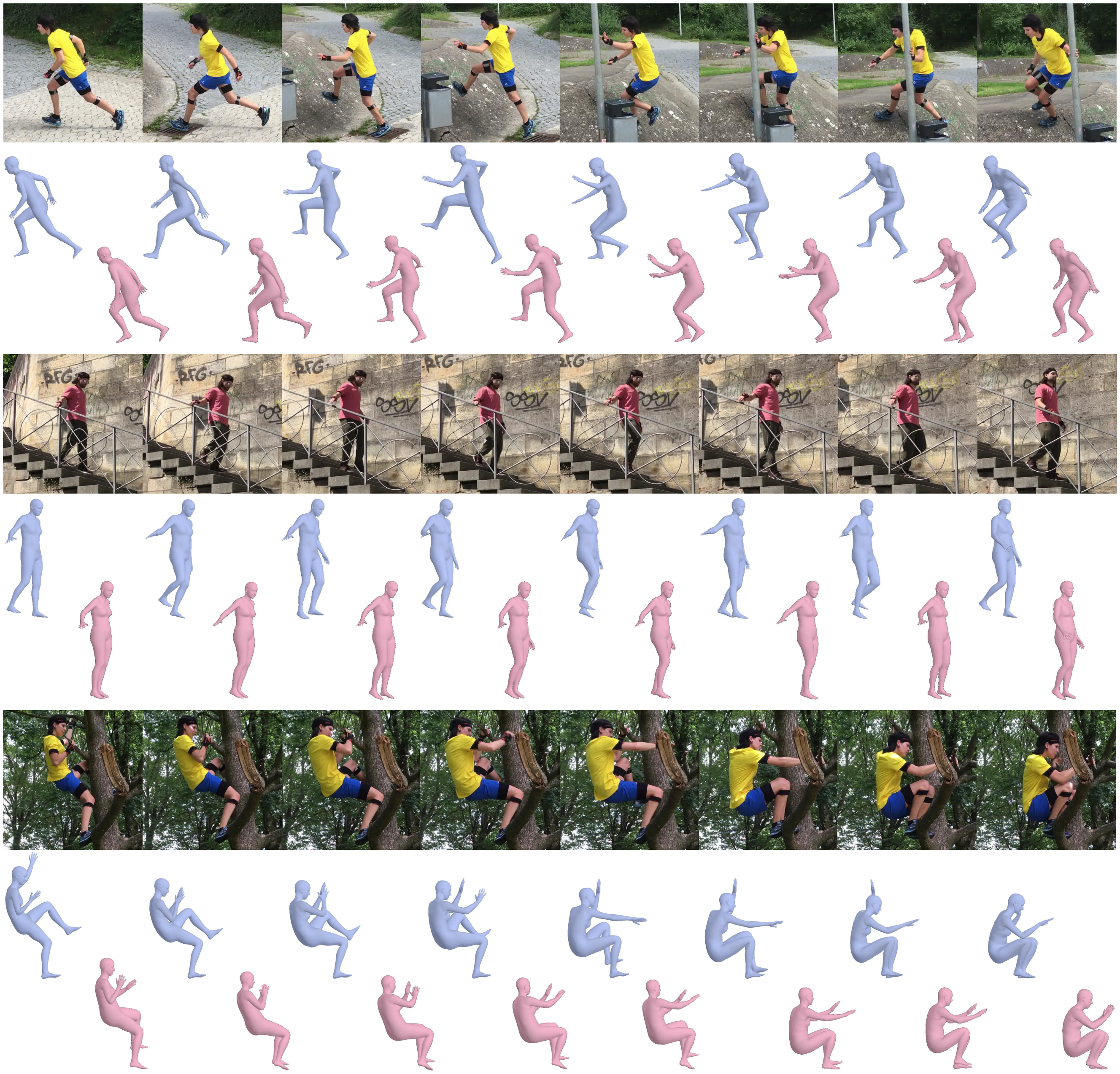

Figure B: Qualitative comparison between MPS-Net [12] and our PMCE. For each video sequence, the top rows show the video frames, the middle rows show the predicted mesh results from our PMCE (blue), and the bottom rows show the mesh results from MPS-Net (pink). Our method can produce more accurate and smooth 3D human motion in fast motions (first sequence), occlusions (second sequence), and slight pose changes (last sequence).

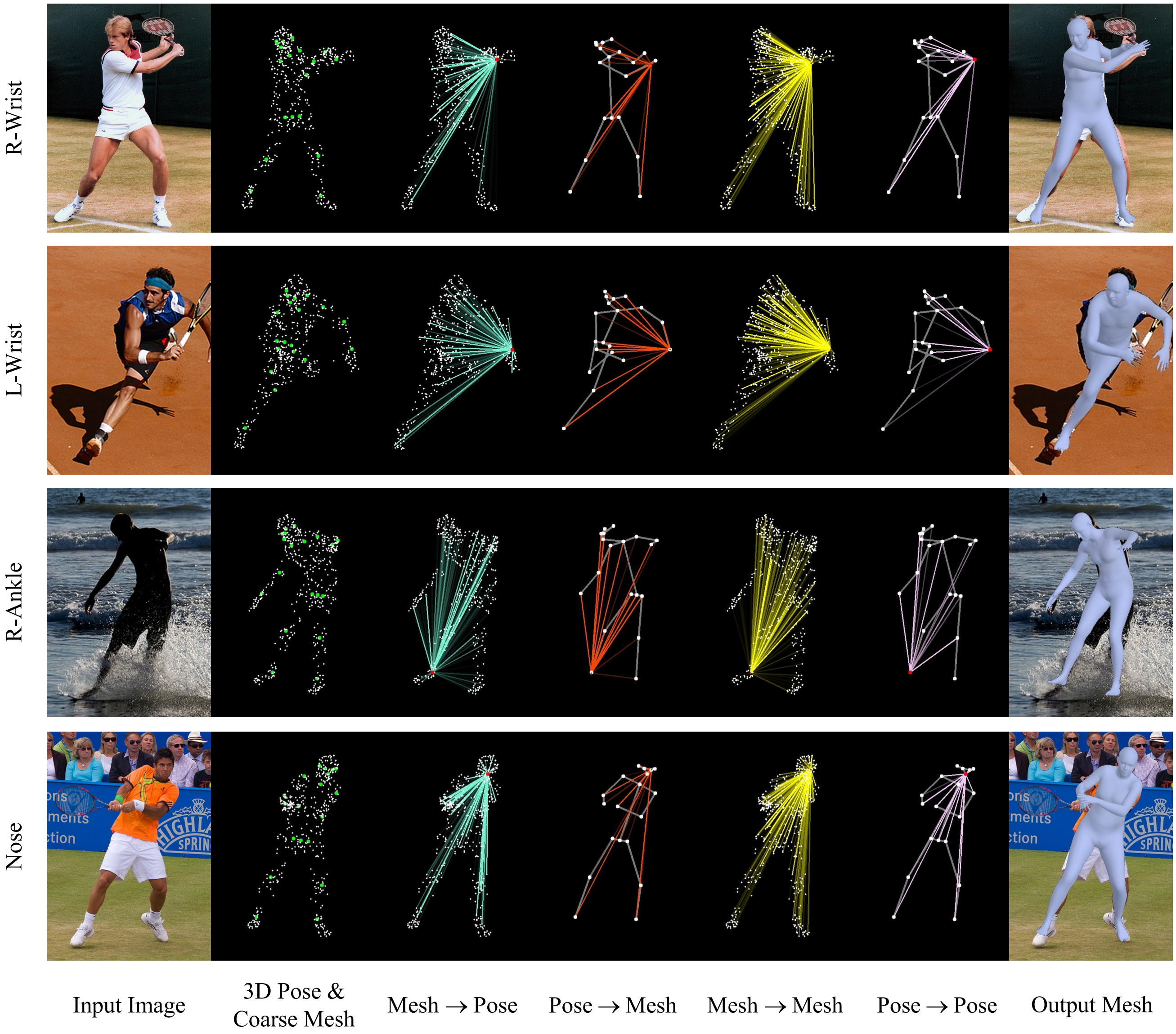

Figure C: Visualization of attention maps. From left to right: input image, the generated 3D pose and coarse mesh, four kinds of interactions, and the output mesh. '→' denotes the direction of information flow. The brighter color indicates higher attention. And the color of lines in each attention map is normalized with the corresponding maximum. In Col. 3 'Mesh → Pose' interaction, the joint learns human body shape information from vertices. In Col. 4 'Pose → Mesh' interaction, the vertex can be guided by joints to perform mesh deformation.

E. Additional Visualization Results

Qualitative Comparison. Figure B shows the qualitative comparison between the previous state-of-the-art video-based method MPS-Net [12] and our PMCE on the challenging video sequences. It shows that our method can produce more accurate and temporally consistent mesh results, especially in fast motions, occlusions, and delicate body deformations.

Visualization of Attention Maps. We further study the interactions of pose and mesh in the proposed co-evolution decoder, including Mesh → Pose, Pose → Mesh, Mesh → Mesh, and Pose → Pose interactions. We obtain the above four kinds of attention maps from the last layer of the co-evolution decoder by averaging the attention values of all attention heads in their corresponding attention blocks. Figure C shows the visualization of attention maps taking different reference nodes. In 'Mesh →

Pose' interaction (Col. 3), each joint can obtain the global shape information from vertices which is not available in its original pose representations. In 'Pose → Mesh' interaction (Col. 4), each mesh vertex aggregates pose information that can guide the mesh deformation. And in 'Mesh → Mesh' (Col. 5) and 'Pose → Pose' (Col. 6) interactions, vertices and joints perform internal adjustments, respectively.

References

[1] Junhyeong Cho, Kim Youwang, and Tae-Hyun Oh. Cross-attention of disentangled modalities for 3D human mesh recovery with transformers. In *ECCV*, pages 342–359, 2022. 1

[2] 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 *CVPR*, pages 1964–1973, 2021. 1

[3] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pages 7122–7131, 2018. 1

[4] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. VIBE: Video inference for human body pose and shape estimation. In *CVPR*, pages 5253–5263, 2020. 1

[5] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. PARE: Part attention regressor for 3D human body estimation. In *ICCV*, pages 11127–11137, 2021. 1

[6] 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 *ICCV*, pages 2252–2261, 2019. 1

[7] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, pages 4501–4510, 2019. 1

[8] 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 *ECCV*, pages 590–606, 2022. 1

[9] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. SMPL: A skinned multi-person linear model. *ACM TOG*, 34(6):1–16, 2015. 2

[10] Gyeongsik Moon and Kyoung Mu Lee. I2L-MeshNet: Image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image. In *ECCV*, pages 752–768, 2020. 1

[11] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3D people. In *ICCV*, pages 11179–11188, 2021. 1

[12] 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 *CVPR*, pages 13211–13220, 2022. 1, 3, 4

[13] 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 *ICCV*, pages 11446–11456, 2021. 1