Incorporating Test-Time Optimization into Training with Dual Networks for Human Mesh Recovery

Yongwei Nie[1], **Mingxian Fan**[1], **Chengjiang Long**[2], **Qing Zhang**[3], **Jian Zhu**[4], **Xuemiao Xu**[1*]
[1]South China University of Technology, China
[2]Meta Reality Labs, USA
[3]Sun Yat-sen University, China
[4]Guangdong University of Technology, China
{nieyongwei, xuemx}@scut.edu.cn, fanmingxian123@gmail.com
cjfykx@gmail.com, zhangq93@mail.sysu.edu.cn, rockeyzhu@163.com

Abstract

Human Mesh Recovery (HMR) is the task of estimating a parameterized 3D human mesh from an image. There is a kind of methods first training a regression model for this problem, then further optimizing the pretrained regression model for any specific sample individually at test time. However, the pretrained model may not provide an ideal optimization starting point for the test-time optimization. Inspired by meta-learning, we incorporate the test-time optimization into training, performing a step of test-time optimization for each sample in the training batch before really conducting the training optimization over all the training samples. In this way, we obtain a meta-model, the meta-parameter of which is friendly to the test-time optimization. At test time, after several test-time optimization steps starting from the meta-parameter, we obtain much higher HMR accuracy than the test-time optimization starting from the simply pretrained regression model. Furthermore, we find test-time HMR objectives are different from training-time objectives, which reduces the effectiveness of the learning of the meta-model. To solve this problem, we propose a dual-network architecture that unifies the training-time and test-time objectives. Our method, armed with meta-learning and the dual networks, outperforms state-of-the-art regression-based and optimization-based HMR approaches, as validated by the extensive experiments. The codes are available at https://github.com/fmx789/Meta-HMR.

1 Introduction

Human mesh recovery (HMR) from a single image is of great importance to human-related applications, such as action capture without MoCap device, action transfer with vision-based system, and VR/AR entertainments, etc. This topic has received extensive research during past years, for which most of previous approaches represent a 3D human mesh by the parametric human model SMPL [39] with parameters $\Theta = (\theta, \beta)$, where θ encodes the pose of the mesh and β describes the body shape. The aim is thus to estimate Θ of a human in a given image.

Originally, the problem is solved by optimizing a standard human mesh so that its 2D projection matches the 2D joints of the target human (e.g., SMPLify [5]). Later, works of [23, 33, 58, 37, 8, 67, 73, 66] propose end-to-end networks trained on large datasets to directly output a 3D SMPL mesh given an input image. In SPIN [28], the regression-based approach [23] and optimization-based approach SMPLify [5] are combined together by interleaved training, where the regression method

*Corresponding author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

 https://doi.org/10.52202/079017-0371

provides an initial solution for optimization and then the optimization method provides supervision for regression. Different from SMPLify [5] that directly optimizes a 3D mesh, works of EFT [22] and BOA [17] finetune a pretrained regression model on each test sample, which indirectly optimize the target human mesh (i.e., the the outcome of the regression models) .

In this paper, we are particularly interested in the test-time optimization-based approaches of EFT [22] and BOA [17]. Since BOA is developed for video leveraging consistency properties between frames, we mainly discuss and compare with EFT that performs HMR for images, like ours. Our observation is that the test-time optimization is analogous to one-shot learning. That is, it finetunes a pretrained model on a specific sample before really applying the model to solve the human mesh recovery task for that sample. However, the pretrained model, which is not specially tailored for the one-shot learning problem, may not be so effective for the test-time adaptation as desired.

Based on the above analysis and inspired by Kim et al. [24], we incorporate the test-time optimization into the training process, re-formulating the test-time optimization from the perspective of learning to learn, i.e., meta learning [14]. Specifically, given a batch of training samples (or saying a set of tasks), our method first performs test-time optimization on each sample to update the regression network parameters temporally for that sample. Then, based on all pieces of regression parameters after test-time optimizations, we further optimize the training-time objectives over the whole batch of training samples. By performing training-time optimization after test-time optimization, we imagine that the training optimization works as a faithful supervision to correct the wrong optimization directions of the test-time optimizations. After the training, the obtained parameters of the regression network can be viewed as meta-parameters which will be instantiated to parameters actually used for human mesh recovery through several test-time optimization steps.

We find that the test-time objectives for the human mesh recovery task are different from training-time objectives, because we sometimes have ground-truth human meshes at training time but forever not at test time. This may produce an obstacle in the meta-learning process, since the optimization directions of the test-time and training optimizations are not identical. To alleviate this problem, we design a dual-network structure to implement our method, which owns a main regression network and an auxiliary network. The auxiliary network provides the main network with pseudo ground-truth SMPL meshes, by which we unify the training and test-time objectives elegantly.

We demonstrate through extensive experiments that our method equipped with meta-learning and the dual networks greatly outperforms state-of-the art approaches. To summarize, our main contributions are three-fold: (1) We propose a novel dual-network HMR framework with test-time optimization involved into the training procedure, which improves the effectiveness of the test-time optimizations. (2) We ensure the test-time objectives identical to the training objectives, further facilitating the joint-training of the test-time and training-time optimizations. (3) Extensive experiments validate that our results outperform those of previous approaches both quantitatively and qualitatively.

2 Related Work

Regression-based HMR methods typically employ neural networks to regress the human body mesh representation from images. Methods of [26, 48, 45, 72, 23, 42, 63, 68, 56, 71, 13, 36, 65, 35, 34] choose to regress parametric human body model, i.e., SMPL [39]. HMR [23] was the first employing CNN [19] to extract features and MLP layers to output 3D mesh parameters. Later, sophisticated networks were proposed for improving the reasoning accuracy. For example, PyMAF [72, 71] extracted features in a pyramid structure and iteratively aligned 3D vertices with human body in the image. Xue et al. [66] used a learnable mask to automatically identify the most discriminative features related to 3D mesh recovery. Works of [27, 33, 58, 44] observed that prior works overlooked the importance of camera parameters. Among them, CLIFF [33] innovatively considered using the cropping bounding boxes as input to reduce the ambiguity of reprojection loss. Zolly [58] considered the camera distortion produced by perspective projection. Recently, Nie et al. [44] proposed a RoI-aware feature extraction and fusion network, guided by camera consistency and contrastive loss functions tailored to the multi-RoI setting.

There are also non-parameterized methods directly regressing mesh vertices. For example, METRO [37] utilized Transformer to model the global relationship between human keypoints and mesh vertices. FastMETRO [8] separated backbone features from the features corresponding to keypoints

and vertices. Recently, [67] combined pyramid structure in PyMAF [72, 71] with the Transformer-based HMR regression method, further improving the regression accuracy.

The regression model is a key component in our method. In this paper, We test HMR [23] and CLIFF [33] as the regression network in our method.

Optimization-based HMR methods [5, 47, 30, 16, 18, 74, 3, 22, 52] usually attempt to estimate a 3D body mesh consistent with 2D image cues. Bogo et al. [5] proposed an approach called SMPLify, which iteratively adjusts SMPL parameters to fit detected 2D keypoints. SPIN [28] combined regression-based methods with SMPLify in an interleaved training strategy. CycleAdapt [43] alternately trained a HMR network and a motion denoising network to enhance each other. Unlike SMPLify, EFT [22] and BOA [17] fine-tuned a pretrained regression network via 2D reprojection loss or temporal consistency loss at test phase, updating the SMPL parameters indirectly. Some approaches proposed learning stronger 3D priors [29, 47, 12, 43] or utilizing trainable neural networks to update parameters in lieu of gradient updates [69, 9, 53]. Inverse kinematics (IK) has also been explored. These methods address IK problems by decomposing relative rotations [32], designing networks that integrate forward and inverse kinematics [31], or incorporating UV position maps [51]. Different from all the above, our method integrates test-time optimization into the training process, obtaining a meta-model and meta-parameters.

Meta Learning Our method is most related to the model-agnostic meta-learning (MAML, or more precisely FOMAML) [14]. MAML first samples a number of tasks, then performs local optimization on each task, and finally conducts a global optimization to update the parameters of the original network. Similarly, our method first executes test-time optimization on each training sample and then performs training optimization on a batch of training samples. Although our method is inspired by MAML and its extensions [2, 49, 50], our goal is fundamentally different from theirs. The goal of MAML is usually for few-shot learning or domain adaptation, which assumes there is ground-truth labeled data in the target domain. In contrast, our goal is to adapt the network to a single test sample which is free of ground-truth human mesh. We have noted MAML has been proven effective in various domains, such as talking head generation [70], SVBRDF recovery [75, 15], image super resolution [46], etc. As far as we know, the work of Kim et al. [24] is the first that applies meta learning to HMR. The key difference is that that Kim et al. [24] used the 2D reprojection loss only (please see Eq. 1 in their paper) in both inner and outer loops of meta learning, while we incorporate the ground-truth 3D SMPLs into the outer loop of meta learning and additionally generate pseudo SMPLs and incorporate them into the inner loop of the metal learning. The utilization of the GT and pseudo SMPLs greatly improves the results of our method upon the method of [24].

3 Our Method

Our goal is to estimate a 3D SMPL human mesh parameterized by $\Theta = (\theta, \beta)$ together with a camera π from a given image $\mathbf{I}$ of a person, where $\theta \in \mathbb{R}^{24\times 3}$ and $\beta \in \mathbb{R}^{10}$ are pose and shape parameters of the SMPL human model [39], respectively. Let $\{\mathbf{I}_{i,j}, \hat{\Theta}_{i,j}, \hat{\mathbf{J}}_{i,j}\}_{i=1,j=1}^{B,M}$ be a training dataset, where $\mathbf{I}_{i,j}$ is a training image, $\hat{\Theta}_{i,j}$ is the ground-truth (GT) human mesh, $\hat{\mathbf{J}}_{i,j}$ is the 2D GT joints (or joints detected by such as OpenPose [7]) of the human in the input image, B is the number of batches, and M is the batchsize. End-to-end HMR regression approaches usually train a neural network $f_{\mathbf{w}} : \mathbf{I}_{i,j} \rightarrow (\Theta_{i,j}, \pi_{i,j})$ by minimizing the following training-time loss function $\mathcal{L}_{train}$:

$$\mathbf{w}_{pre} = \arg\min_{\mathbf{w}} \sum_{i=1}^{B} \sum_{j=1}^{M} \mathcal{L}_{train}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\Theta}_{i,j}, \hat{\mathbf{J}}_{i,j}), \tag{1}$$

where

$$\mathcal{L}_{train}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\Theta}_{i,j}, \hat{\mathbf{J}}_{i,j}) = \mathcal{L}_{2D}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\mathbf{J}}_{i,j}) + \mathcal{L}_{3D}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\Theta}_{i,j}), \tag{2}$$

with

$$\mathcal{L}_{2D}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\mathbf{J}}_{i,j}) = \|\pi(\Theta_{i,j}) - \hat{\mathbf{J}}_{i,j}\|_2^2, \quad \mathcal{L}_{3D}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\Theta}_{i,j}) = \|X(\Theta_{i,j}) - X(\hat{\Theta}_{i,j})\|_2^2. \tag{3}$$

As seen, $\mathcal{L}_{train}$ is composed of a 2D reprojection loss $\mathcal{L}_{2D}$ and a 3D loss $\mathcal{L}_{3D}$. The 2D loss first projects mesh $\Theta_{i,j}$ to the 2D plane by the camera π and then computes difference between the projected 2D joints and the given 2D joints $\hat{\mathbf{J}}_{i,j}$. The 3D loss computes difference between 3D meshes, where X can be an identity transformation or transformations computing 3D human joints or mesh vertices from the mesh parameters.

https://doi.org/10.52202/079017-0371

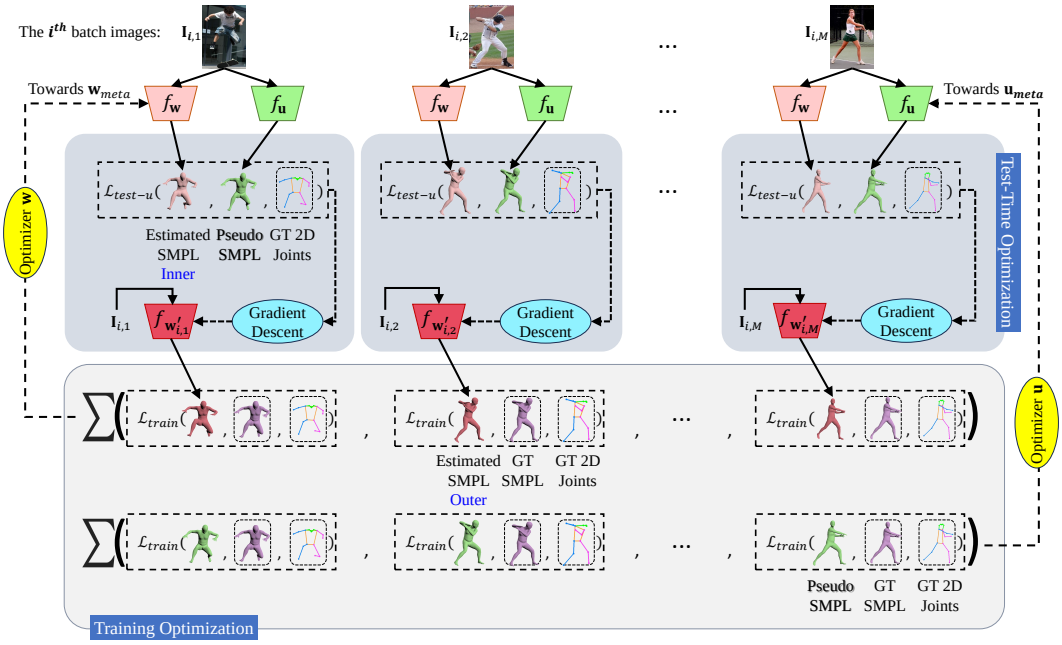


Figure 1: **Overview of the dual-network meta-learning HMR method,** composed of a main HMR regression network $f_{\mathbf{w}}$ and an auxiliary network $f_{\mathbf{u}}$. Both networks have the same architecture but different parameters. Given i^{th} batch of images, test-time optimization is first executed for each training image $\mathbf{I}_{i,j}$ in the batch individually, updating $f_{\mathbf{w}}$ to $f_{\mathbf{w}'_{i,j}}$ by performing a gradient descent step w.r.t. the test-time loss function $\mathcal{L}_{test-u}$. Then based on $\{f_{\mathbf{w}'_{i,j}}|j \in [1, M]\}$ (M is the batch size), the training optimization is executed to update the parameters of both main and auxiliary networks by $\mathcal{L}_{train}$ with different arguments respectively. $\mathbf{w}_{meta}$ and $\mathbf{u}_{meta}$ are the finally generated meta-parameters. $f_{\mathbf{u}}$ generates "Pseudo SMPLs" that are used in the test-time loss to supervise the learning of the "Estimated SMPL Inner". GT SMPLs are used in the training loss to supervise the learning of "Estimated SMPL Outer" and the Pseudo SMPLs.

3.1 Test-time Optimization

Exemplar-Fine-Tuning (EFT) [22] was the first work proposing test-time optimization for HMR. In particular, the pretrained network $\mathbf{w}_{pre}$ is further finetuned by performing the following test-time optimization loss function on a specific test sample $\mathbf{I}_{i,j}$:

$$\mathbf{w}^*_{i,j} = \arg\min_{\mathbf{w}_{i,j}} \mathcal{L}_{test}(f_{\mathbf{w}_{i,j}}(\mathbf{I}_{i,j}), \hat{\mathbf{J}}_{i,j}), \text{ initially } \mathbf{w}_{i,j} = \mathbf{w}_{pre}, \tag{4}$$

with

$$\mathcal{L}_{test}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\mathbf{J}}_{i,j}) = \mathcal{L}_{2D}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\mathbf{J}}_{i,j}). \tag{5}$$

The test-time optimization starts from the initial solution $\mathbf{w}_{pre}$ provided by the pretrained model. It resembles one-shot learning but actually does not, because the pretrained model is obtained using normal supervised learning techniques while not introducing any strategy for guaranteeing the properties of one-shot learning. The parameters of the pretrained model may be not ideal as the starting point for the test-time optimization.

3.2 Incorporating Test-time Optimization into Training

To solve the above problem, we propose to integrate the test-time optimization into the training procedure as shown in Figure 1, inspired by optimization-based meta-learning [14]. Specifically, for each sample in a batch, we first perform test-time optimization on that sample to update the parameters of the regression network temporally corresponding to the sample. After that, we perform training optimization over all M training samples, based on the M temporally updated regression

Algorithm 1 Meta learning of Dual networks for 3D Human Recovery

• **The stage of training**

Require: Training dataset $\{\mathbf{I}_{i,j}, \hat{\Theta}_{i,j}, \hat{\mathbf{J}}_{i,j}\}_{i=1,j=1}^{B,M}$
Require: $f_{\mathbf{w}}, f_{\mathbf{u}}$: main and auxiliary networks, randomly initialized
Require: α, β: step size hyperparameters

while not done **do**
  Sample a batch of images $\{\mathbf{I}_{i,j} \mid j \in [1, M]\}, i \sim \mathcal{U}(1, B)$ ▷ $\mathcal{U}$ is uniform distribution
  for all $\mathbf{I}_{i,j}$ **do**
    Compute SMPL meshes by $f_{\mathbf{w}}(\mathbf{I}_{i,j})$ and $f_{\mathbf{u}}(\mathbf{I}_{i,j})$, respectively
    Compute $L_{\text{test-u}}$ in Eq. 9, evaluate $\nabla_{\mathbf{w}} L_{\text{test-u}}$, and update $\mathbf{w}'_{i,j} \leftarrow \mathbf{w} - \alpha \nabla_{\mathbf{w}} L_{\text{test-u}}$
  end for
  Compute L^1_{train} in Eq. 8, evaluate $\nabla_{\mathbf{w}} L^1_{\text{train}}$, and update $\mathbf{w} \leftarrow \mathbf{w} - \beta \sum_j \nabla_{\mathbf{w}_{i,j}} L^1_{\text{train}}$
    ▷ $\mathbf{w}'_{i,j}$ is used in L^1_{train}
  Compute L^2_{train} in Eq. 8, evaluate $\nabla_{\mathbf{u}} L^2_{\text{train}}$, and update $\mathbf{u} \leftarrow \mathbf{u} - \beta \sum_j \nabla_{\mathbf{u}} L^2_{\text{train}}$
end while
$\mathbf{w}_{\text{meta}} \leftarrow \mathbf{w}, \quad \mathbf{u}_{\text{meta}} \leftarrow \mathbf{u}$

• **The stage of testing**

Require: Input image $\mathbf{I}$
Require: Main and auxiliary networks with meta parameters $\mathbf{w}_{\text{meta}}$ and $\mathbf{u}_{\text{meta}}$

$\mathbf{w} = \mathbf{w}_{\text{meta}}$
for $i = 1$ to m **do** ▷ Iterate test-time optimization m times
  Pseudo GT mesh $\leftarrow f_{\mathbf{u}_{\text{meta}}}(\mathbf{I})$ ▷ Using frozen auxiliary network
  Compute $L_{\text{test-u}}$ using Eq. 9 ▷ Given input $\mathbf{I}$, the pseudo GT mesh, and 2D joints
  Evaluate $\nabla_{\mathbf{w}} L_{\text{test-u}}$
  Update $\mathbf{w} \leftarrow \mathbf{w} - \alpha \nabla_{\mathbf{w}} L_{\text{test-u}}$
end for
$\mathbf{w}_{\text{final}} \leftarrow \mathbf{w}$
Compute SMPL mesh by $f_{\mathbf{w}_{\text{final}}}(\mathbf{I})$

networks. This process is formulated as:

$$\mathbf{w}_{meta} = \arg\min_{\mathbf{w}} \sum_{i=1}^{B} \sum_{j=1}^{M} \mathcal{L}_{train}(f_{\mathbf{w}'_{ij}}(\mathbf{I}_{ij}), \hat{\Theta}_{ij}, \hat{\mathbf{J}}_{ij}), \tag{6}$$

where,

$$\mathbf{w}'_{ij} = \mathbf{w} - \alpha \nabla_{\mathbf{w}} \mathcal{L}_{test}(f_{\mathbf{w}}(\mathbf{I}_{ij}), \hat{\mathbf{J}}_{ij}). \tag{7}$$

The difference between Eq. 6 and Eq. 1 is in the parameters of f to be optimized. Instead of directly optimizing the current parameters $\mathbf{w}$ of f using the training objective $\mathcal{L}_{train}$, we first perform a step of test-time optimization using Eq. 7 on each sample $\mathbf{I}_{ij}$ to obtain network parameters $\mathbf{w}'_{ij}$ specific to that sample. Then, $\{\mathbf{w}'_{ij} | j \in [1, M]\}$ over all training samples in a batch are in turn used in Eq. 6 to evaluate the training objective. In Eq. 7, α is the learning rate of the test-time optimization.

By performing test-time optimization before training optimization in Eq. 6 and 7, we take test-time optimization into consideration in the training procedure. That means, the "test-time optimization" is trained on the training dataset, thus having better generalization ability to test samples. The proposed method resembles the optimization-based meta-learning [14] and we call the obtained parameters $\mathbf{w}_{meta}$ meta-parameters.

3.3 Unifying Training and Test-time Optimization Objectives with Dual Networks

There are ground-truth human meshes at training time while not at test time, causing the difference between $\mathcal{L}_{train}$ (see Eq. 2)) and $\mathcal{L}_{test}$ (see Eq. 5). Since both the test-time and training optimizations update parameters of the same network, the difference between the two optimization objectives yields different gradient descent directions, causing potential conflicts that reduce the effectiveness of the training (see ablation studies in Section 4.4).

https://doi.org/10.52202/079017-0371

To make the test-time optimization more compatible with the training objective, we propose a method that unifies the training and test-time optimization objectives by introducing an auxiliary regression network $f_{\mathbf{u}}$ parameterized by $\mathbf{u}$ which is trained together with the main network:

$$\mathbf{w}_{meta}, \mathbf{u}_{meta} = \arg\min_{\mathbf{w},\mathbf{u}} \sum_{i=1}^{B} \sum_{j=1}^{M} (\mathcal{L}^1_{train}(f_{\mathbf{w}'_{i,j}}(\mathbf{I}_{i,j}), \hat{\Theta}_{i,j}, \hat{\mathbf{J}}_{i,j}) + \mathcal{L}^2_{train}(f_{\mathbf{u}}(\mathbf{I}_{i,j}), \hat{\Theta}_{i,j}, \hat{\mathbf{J}}_{i,j})). \quad (8)$$

The above equation is a combination of Eq. 1 and Eq. 6 (superscript 1 and 2 are used to denote the first and second term respectively), with Eq. 1 applied to the auxiliary network $f_{\mathbf{u}}$, and Eq. 6 applied to $f_{\mathbf{w}}$. We use $f_{\mathbf{u}}$ to generate a pseudo GT mesh $\hat{\Theta}^u_{i,j}$ for a training image $\mathbf{I}_{i,j}$, i.e., $\hat{\Theta}^u_{i,j} = f_{\mathbf{u}}(\mathbf{I}_{i,j})$, and use the pseudo label to supervise the gradient descent in the test-time optimization:

$$\mathbf{w}'_{i,j} = \mathbf{w} - \alpha \nabla_{\mathbf{w}} \mathcal{L}_{test-u}(f_{\mathbf{w}}(\mathbf{I}_{i,j}), \hat{\Theta}^u_{i,j}, \hat{\mathbf{J}}_{i,j}). \quad (9)$$

Please compare between Eq. 9 and Eq. 7. The difference is that there is an additional input $\hat{\Theta}^u_{i,j}$ to $\mathcal{L}_{test-u}$ in Eq. 9, and note that this new form of $\mathcal{L}_{test-u}$ is identical to the form of $\mathcal{L}_{train}$.

3.4 Inference with Dual Networks

Both the training and testing pseudo codes of our method are given in Algorithm 1. The training process is fully elaborated in the above sections. Now we introduce how to perform inference at test time. For each test sample $\mathbf{I}$, at our hand are two networks $f_{\mathbf{w}}$ and $f_{\mathbf{u}}$ with $\mathbf{w} = \mathbf{w}_{meta}$ and $\mathbf{u} = \mathbf{u}_{meta}$, respectively. We freeze the parameters of the auxiliary network, and use it to compute the pseudo GT human mesh for the test image $\mathbf{I}$. Then, under the supervision of the pseudo mesh, we compute $\mathcal{L}_{test-u}$ and use Eq. 9 to iteratively update the parameters $\mathbf{w}$ of f from $\mathbf{w}_{meta}$ to $\mathbf{w}_{final}$. We run at most $m = 14$ iterations, and automatically stop the iteration if losses of two consecutive iterations are close enough. We finally use $f_{\mathbf{w}_{final}}$ to estimate the human mesh for image $\mathbf{I}$.

3.5 Implementation Details

We implement the main network $f_{\mathbf{w}}$ and auxiliary network $f_{\mathbf{u}}$ with the same network architecture but different parameters. Specifically, we use HMR [23] or CLIFF [33] as f due to their simplicity. The two methods and many other approaches [37, 8, 66, 72, 71, 26, 67] adopt ResNet-50 [19] or HRNet-W48 [54] to extract features from the input image, and estimate human mesh based on the features. We provide results of both kinds of backbones.

We implement our method in PyTorch using the Adam optimizer [25] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The batchsize for ResNet backbone is 40, and for HRNet backbone is 30. The number of training epochs for ResNet backbone is 65, and for HRNet backbone is 25. The learning rate α used in the test-time optimization is 1e-5, and the learning rate β (see Algorithm 1) for the training optimization is 1e-4. Our method takes about 3 days training on a single NVIDIA RTX3090 GPU.

4 Experiments

4.1 Datasets

Following previous work [33, 26, 73], we employ the following datasets in our experiments: (1) **Human3.6M** [20], an indoor dataset with precise GT human mesh and 2D joints captured through MoCap devices. (2) **MPI-INF-3DHP** [41], another widely used indoor dataset whose GT human meshes are obtained through multi-view reconstruction. (3) **COCO** [38] and (4) **MPII** [1], two in-the-wild outdoor datasets with human annotated 2D joints for which we use the pseudo GT mesh provided by [33]. (5) **3DPW** [57], a challenging in-the-wild dataset providing accurate human mesh fitted from IMU sensor data.

4.2 Training, Testing and Metrics

Following prior arts [37, 66, 8, 4], we first train our method on a mixture of four datasets, including Human3.6M [20], MPI-INF-3DHP [41], COCO [38], and MPII [1], and then test our method on

The authors Yongwei Nie and Mingxian Fan signed the license and produced all the experimental results in this paper. Meta did not have access to the datasets.

the test dataset of Human3.6M [20]. After that, we further fine-tune our model for 5 epochs by introducing the training dataset of 3DPW [57], and then evaluate our method on the test dataset of 3DPW [57]. We use **MPJPE** (Mean Per Joint Position Error), **PA-MPJPE** (Procrustes-aligned MPJPE), **PVE** (Mean Per-vertex Error) as the metrics to evaluate our method.

Table 1: **Quantitative comparison with state-of-the-art methods** on 3DPW [57] and Human3.6M [20]. "†": using 2D joints detected by OpenPose [7], "$*$": using 2D joints detected by RSN [6].

	Method	Backbone	3DPW			Human3.6M	
			MPJPE↓	PA-MPJPE↓	PVE↓	MPJPE↓	PA-MPJPE↓
Regression-based	HMR [23]'18	Res-50	130.0	81.3	-	88.0	56.8
	PARE [26]'21	HR-W32	74.5	46.5	88.6	-	-
	ROMP [55]'21	HR-W32	76.7	47.3	93.4	-	-
	PyMAF [72]'21	HR-W48	74.2	45.3	87.0	54.2	37.2
	METRO [37]'21	HR-W64	77.1	47.9	88.2	54.0	36.7
	FastMETRO [8]'22	HR-W64	73.5	44.6	84.1	52.2	33.7
	CLIFF [33]'22	HR-W48	69.0	43.0	81.2	47.1	32.7
	LearnSample [66]'22	HR-W32	70.5	43.3	82.7	45.9	33.5
	ProPose [13]'23	HR-W48	68.3	40.6	79.4	45.7	**29.1**
	POTTER [73]'23	ViT	75.0	44.8	87.4	56.5	35.1
	DeFormer [67]'23	HR-W48	72.9	44.3	82.6	44.8	31.6
Optimization-based	LearnedGD [53]'20	-	-	55.9	-	-	56.4
	HUND [69]'21	Res-50	81.4	57.5	-	69.5	52.6
	SPIN [28]'21	Res-50	96.9	59.2	116.4	62.5	41.1
	EFT [22]'21	Res-50	85.1	52.2	98.7	63.2	43.8
	HybrIK [32]'21	Res-34	74.1	45.0	86.5	55.4	33.6
	NIKI [31]'23	HR-W48	71.3	40.6	86.6	-	-
	ReFit [59]'23	HR-W48	65.8	41.0	-	48.4	32.2
	PLIKS [51]'23	HR-W48	66.9	42.8	82.6	49.3	34.7
	Ours$_{\mathrm{CLIFF}}$†	HR-W48	62.9	39.7	80.1	43.9	30.3
	Ours$_{\mathrm{CLIFF}}*$	HR-W48	**62.4**	**39.5**	**78.1**	**42.0**	**29.1**

4.3 Comparison with Previous Approaches

Quantitative results. We present accuracy comparison with SOTA methods in Table 1, including regression-based approaches [23, 26, 55, 37, 33, 13, 73, 67, 72, 8, 66] and optimization-based approaches [28, 22, 32, 59, 31, 51, 53, 69]. For all the compared approaches, we report the best results their papers provide. For our method, we adopt CLIFF [33] as f and HRNet-W48 as the backbone network. Since our method needs 2D joints for test-time optimization, we report results using joints estimated by OpenPose [7] (denoted by †) and RSN [6] (denoted by $*$).

Please compare "Ours$_{\mathrm{CLIFF}}$† (HR-W48)" in Table 1 with SOTA approaches that also use HRNet-W48 as backbone. Our method outperforms most of previous approaches. Compared with [33], we improve it from 69.0 to 62.9 taking MPJPE of 3DPW as an example, which is a large margin. If using RSN joints for the test-time optimization, our method can further improve the metrics.

Qualitative results. We show qualitative comparison with CLIFF [33] and Refit [59] in Figure 2. Our method estimates faithful human poses and meshes which are better than those of the compared approaches. Comparisons with HybrIK [32], NIKI [31], ProPose [13] and EFT$_{\mathrm{CLIFF}}$ can be found in the supplementary material.

4.4 Ablation study

Influence of Regression Model. The adopted regression model f influences the effectiveness of our method. In Table 2, we show the results of using HMR [23] or CLIFF [33] as the regression model. Since CLIFF is a stronger baseline than HMR, our method based on CLIFF performs better than that based on HMR.

Influence of Accuracy of 2D Joints. Since our method needs 2D joints as the supervision at test time, it is interesting to see how the quality of the 2D joints affects the effectiveness of the method. We have already repported results on 2D joints detected by OpenPose [7] and RSN [6]. In Table 2, we further test GT 2D joints. As seen, our method with GT joints outperforms our method using detected joints by OpenPose and RSN. This indicates our method will become more effective as 2D pose detectors continue to develop.

https://doi.org/10.52202/079017-0371

Figure 2: **Qualitative comparison with SOTA methods.** We show results produced by CLIFF [33], ReFit [59], and our method (†: OpenPose, ∗: RSN). All the three methods use HRNet-W48 as the backbone. In the novel views, green represents the ground truth, orange represents CLIFF, purple represents ReFit, pink and blue represent the two variants of our method, respectively.

Table 2: **Ablation study on regression model and 2D joints** on 3DPW [57] and Human3.6M [20]. "†": using 2D joints detected by OpenPose [7], "◇": using GT 2D joints.

Method	Backbone	3DPW			Human3.6M	
		MPJPE↓	PA-MPJPE↓	PVE↓	MPJPE↓	PA-MPJPE↓
Ours_{HMR}†	Res-50	73.3	44.3	90.3	55.8	36.4
Ours_{HMR}◇	Res-50	68.9	39.6	85.5	53.7	33.8
$\text{Ours}_{\text{CLIFF}}$†	HR-W48	62.9	39.7	80.1	43.9	30.3
$\text{Ours}_{\text{CLIFF}}$◇	HR-W48	57.8	35.3	74.4	39.4	27.5

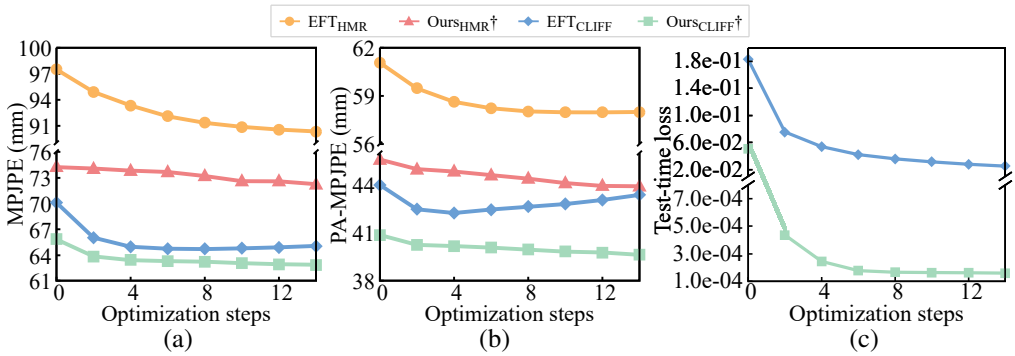


Figure 3: **Influence of optimization steps during inference.** Our method outperforms EFT when using the same regression model. As optimization proceeds, our results continuously become better, while those of EFT become better at first and then become worse (see (a) and (b)). (c) shows that our method achieves faster convergence compared to EFT.

Influence of Optimization Steps at Inference Time. At inference time, we perform at most m test-time optimization steps. In Figure 3 (a) and (b), we show how the evaluation metrics become as the number of optimization steps increases. As seen, our results consistently become better in terms of both MPJPE and PA-MPJPE. We also show the results of EFT_{HMR} and $\text{EFT}_{\text{CLIFF}}$. With the same regression model, our method is better than EFT [22]. The results of EFT become better at the first few optimization steps, but become worse as more optimization steps execute. This is probably because EFT is only finetuned with 2D reprojection loss and is more sensitive to the errors in the 2D joints. At the first several optimization steps, the estimated 3D SMPL approaches the 2D joints from a relatively distant initialization, therefore the result gets better gradually. With more optimization steps, the SMPL may overfit the 2D joints whose annotation-errors then distort the SMPLs, thus yielding worse evaluation metrics. In contrast, our method is guided by both 3D and 2D

https://doi.org/10.52202/079017-0371

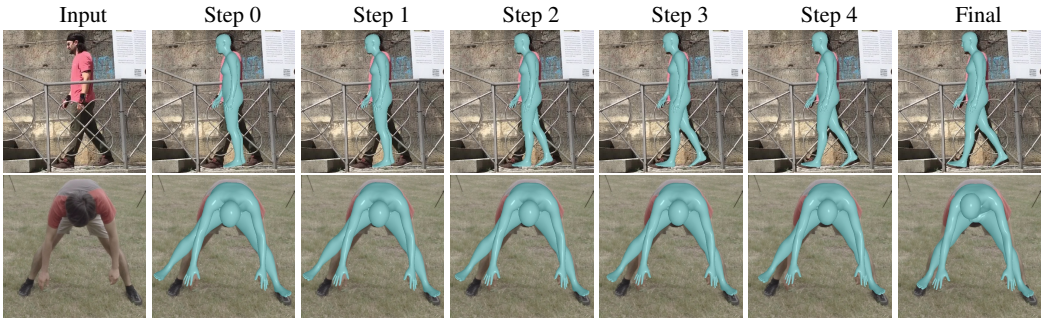


Figure 4: **Stepwise visualization.** From left to right, we showcase results after different steps of test-time optimization during testing.

Table 3: **Ablation study on meta-learning and auxiliary network.** Models are trained on COCO [38] and tested on 3DPW [57]. "Test. Opt." is "test-time optimization".

Model	Integrating Test. Opt. (meta-learning)	Auxiliary Net	MPJPE ↓	PA-MPJPE ↓
$\text{EFT}_{\text{CLIFF}}$	✗	✗	84.6	54.2
$\text{Ours}_{\text{CLIFF}}$	✓	✗	78.5	49.9
$\text{Ours}_{\text{CLIFF}}$	✓	✓	**76.7**	**49.5**

Table 4: **Quantitative comparison** with $\text{EFT}_{\text{CLIFF}}$ on the LSP-Extended dataset [21].

Method	2D Loss
$\text{EFT}_{\text{CLIFF}}$	8.3e-3
$\text{Ours}_{\text{CLIFF}}$	6.1e-3

Table 5: **Quantitative comparison** with $\text{EFT}_{\text{CLIFF}}$ on the Human3.6M dataset [20].

Method	MPJPE↓	PA-MPJPE↓
$\text{EFT}_{\text{CLIFF}}$	85.5	51.0
$\text{Ours}_{\text{CLIFF}}$	83.8	48.6

supervisions. The 3D pseudo SMPLs plays the role of regularization that mitigates the influences of errors in 2D joints (see more explanations in the supplemental material). In Figure 3 (c), we present the loss curve of the test-time optimization of EFT and our method. Our method converges in about 6 steps, demonstrating a faster convergence compared with EFT.

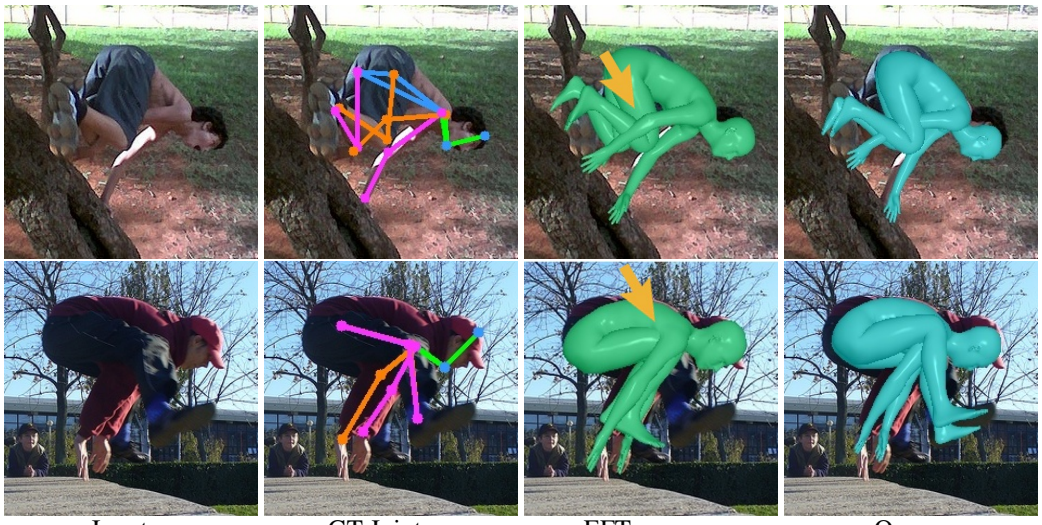
Figure 4 shows our meshes after different optimization steps. Initially, the mesh does not fit with the target human. After more steps, the mesh progressively deforms itself to achieve perfect fitting.

Influence of Meta-Learning and Dual Networks. We propose meta-learning to improve the performance of test-time optimization. We also introduce an auxiliary network to unify the formulation of test-time and training optimizations, and hope this can reduce the conflict in training and improve the estimation accuracy. We conduct an ablation study to validate the two components as shown in Table 3. We train all the models in the ablation study on COCO [38] and test them on 3DPW [57]. The first row in Table 3 shows results of training with no test-time (Test.) optimization (Opt.) and no auxiliary network, *i.e.*, CLIFF [33]. The second row shows our method without auxiliary network. The third row shows our full method. As shown, the introduction of the meta-learning and auxiliary network strategies both improve the evaluation results.

Out-of-Domain Adaptation. Test-time optimization performs post-processing on each test image and has the ability of adaptation to out-of-domain data. To comprehensively validate our method's effectiveness in out-of-distribution (OOD) scenarios, we first conduct a quantitative comparison with $\text{EFT}_{\text{CLIFF}}$ on the LSP-Extended dataset, as shown in Table 4 (Training dataset: COCO, MPII, MPI-INF-3DHP, Human3.6M, 3DPW, Backbone: HR-W48). Since LSP provides ground truth 2D joints but not GT SMPLs, the comparison is based on 2D loss relative to the GT joints. Our method achieves a lower 2D loss than $\text{EFT}_{\text{CLIFF}}$, with results of 6.1e-3 (our) versus 8.3e-3 ($\text{EFT}_{\text{CLIFF}}$), indicating that our method is more accurate in approaching the GT joints.

To further evaluate the OOD performance, we train both our method and $\text{EFT}_{\text{CLIFF}}$ on the COCO dataset (an outdoor dataset) and test them on the Human3.6M dataset (an indoor dataset). With ground truth SMPLs available, we report results using MPJPE and PA-MPJPE metrics. As shown

https://doi.org/10.52202/079017-0371



Input GT Joints $\text{EFT}_{\text{CLIFF}}$ Ours

Figure 5: **Out-of-domain adaptation.** Please see the depth order of the arms where $\text{EFT}_{\text{CLIFF}}$ fails to infer. Our method correctly identify the correct configuration.

in Table 5, our method demonstrates superior performance over $\text{EFT}_{\text{CLIFF}}$, further validating its effectiveness in OOD scenarios.

Figure 5 shows two examples from the LSP-Extended dataset. The persons in the two images take complex actions. The shadows in the images and the similar color of the black pants and shoes make it difficult even for humans to identify the configuration of the 3D meshes. Our method successfully estimates correct meshes, while the arms in the results of $\text{EFT}_{\text{CLIFF}}$ exhibit wrong depth orders.

5 Conclusion

To conclude, this paper presents a new training paradigm towards better test-time optimization performance at test time. We mainly propose two strategies. First, we integrate the test-time optimization into the training procedure, which performs test-time optimization before running the typical training in each training iteration. Second, we propose a dual-network architecture to implement the proposed novel training paradigm, aiming at unifying the space of the test-time and training optimization problems. Experiments and comparisons prove that the proposed training scheme improves the effectiveness of the test-time optimization during testing, demonstrating that it successfully learns meta-parameters that benefit the test-time optimization for specific samples. Our method can perform even better with stronger regressor baseline or better 2D joints, and can adapt to out-of-domain challenging test cases.

Acknowledgments

The work was supported by Guangdong Provincial Natural Science Foundation for Outstanding Youth Team Project (No. 2024B1515040010), the Fundamental Research Funds for the Central Universities (No. 2024ZYGXZR021), China National Key R&D Program (Grant No. 2023YFE0202700), Key-Area Research and Development Program of Guangzhou City (No.2023B01J0022), National Natural Science Foundation of China for Key Program (No. 62237001), Natural Science Foundation of China (No. 62072191).

References

[1] Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: Proceedings of the IEEE Conference on computer Vision and Pattern Recognition. pp. 3686–3693 (2014)

[2] Antoniou, A., Edwards, H., Storkey, A.: How to train your maml. In: International Conference on Learning Representations (2019)

[3] Bălan, A.O., Black, M.J.: The naked truth: Estimating body shape under clothing. In: Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part II 10. pp. 15–29. Springer (2008)

[4] Black, M.J., Patel, P., Tesch, J., Yang, J.: Bedlam: A synthetic dataset of bodies exhibiting detailed lifelike animated motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8726–8737 (2023)

[5] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it smpl: 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. pp. 561–578. Springer (2016)

[6] Cai, Y., Wang, Z., Luo, Z., Yin, B., Du, A., Wang, H., Zhang, X., Zhou, X., Zhou, E., Sun, J.: Learning delicate local representations for multi-person pose estimation. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 455–472. Springer (2020)

[7] Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7291–7299 (2017)

[8] Cho, J., Youwang, K., Oh, T.H.: Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In: European Conference on Computer Vision. pp. 342–359. Springer (2022)

[9] Choutas, V., Bogo, F., Shen, J., Valentin, J.: Learning to fit morphable models. In: European Conference on Computer Vision. pp. 160–179. Springer (2022)

[10] Dang, L., Nie, Y., Long, C., Zhang, Q., Li, G.: Msr-gcn: Multi-scale residual graph convolution networks for human motion prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11467–11476 (2021)

[11] Dang, L., Nie, Y., Long, C., Zhang, Q., Li, G.: Diverse human motion prediction via gumbel-softmax sampling from an auxiliary space. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 5162–5171 (2022)

[12] Davydov, A., Remizova, A., Constantin, V., Honari, S., Salzmann, M., Fua, P.: Adversarial parametric pose prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10997–11005 (2022)

[13] Fang, Q., Chen, K., Fan, Y., Shuai, Q., Li, J., Zhang, W.: Learning analytical posterior probability for human mesh recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8781–8791 (2023)

[14] Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: International conference on machine learning. pp. 1126–1135. PMLR (2017)

[15] Fischer, M., Ritschel, T.: Metappearance: Meta-learning for visual appearance reproduction. ACM Transactions on Graphics (TOG) **41**(6), 1–13 (2022)

[16] Guan, P., Weiss, A., Balan, A.O., Black, M.J.: Estimating human shape and pose from a single image. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 1381–1388. IEEE (2009)

[17] Guan, S., Xu, J., Wang, Y., Ni, B., Yang, X.: Bilevel online adaptation for out-of-domain human mesh reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10472–10481 (2021)

[18] Hasler, N., Ackermann, H., Rosenhahn, B., Thormählen, T., Seidel, H.P.: Multilinear pose and body shape estimation of dressed subjects from image sets. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 1823–1830. IEEE (2010)

 https://doi.org/10.52202/079017-0371

[19] He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 630–645. Springer (2016)

[20] Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: 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)

[21] Johnson, S., Everingham, M.: Learning effective human pose estimation from inaccurate annotation. In: CVPR 2011. pp. 1465–1472. IEEE (2011)

[22] Joo, H., Neverova, N., Vedaldi, A.: Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation. In: 2021 International Conference on 3D Vision (3DV). pp. 42–52. IEEE (2021)

[23] Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7122–7131 (2018)

[24] Kim, M., Min, Y., Kim, J., Kim, S.: Meta-learned initialization for 3d human recovery. In: 2022 IEEE International Conference on Image Processing (ICIP). pp. 4238–4242. IEEE (2022)

[25] Kinga, D., Adam, J.B., et al.: A method for stochastic optimization. In: International conference on learning representations (ICLR). vol. 5, p. 6. San Diego, California; (2015)

[26] Kocabas, M., Huang, C.H.P., Hilliges, O., Black, M.J.: Pare: Part attention regressor for 3d human body estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11127–11137 (2021)

[27] Kocabas, M., Huang, C.H.P., Tesch, J., Müller, L., Hilliges, O., Black, M.J.: Spec: Seeing people in the wild with an estimated camera. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11035–11045 (2021)

[28] Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: 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. pp. 2252–2261 (2019)

[29] Kolotouros, N., Pavlakos, G., Jayaraman, D., Daniilidis, K.: Probabilistic modeling for human mesh recovery. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11605–11614 (2021)

[30] Lassner, C., Romero, J., Kiefel, M., Bogo, F., Black, M.J., Gehler, P.V.: Unite the people: Closing the loop between 3d and 2d human representations. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6050–6059 (2017)

[31] Li, J., Bian, S., Liu, Q., Tang, J., Wang, F., Lu, C.: 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. pp. 12933–12942 (2023)

[32] Li, J., Xu, C., Chen, Z., Bian, S., Yang, L., Lu, C.: 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. pp. 3383–3393 (2021)

[33] Li, Z., Liu, J., Zhang, Z., Xu, S., Yan, Y.: Cliff: Carrying location information in full frames into human pose and shape estimation. In: European Conference on Computer Vision. pp. 590–606. Springer (2022)

[34] Liao, H.R., Lin, J.C., Lee, C.Y.: Progressive hypothesis transformer for 3d human mesh recovery. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 6323–6332 (2024)

[35] Liao, X., Zhang, C., Xu, J., Su, W., Chen, Z., Tao, W.: Instahmr: Instance-aware one-stage multi-person human mesh recovery. IEEE Transactions on Visualization and Computer Graphics (2024)

[36] Lin, K., Lin, C.C., Liang, L., Liu, Z., Wang, L.: Mpt: mesh pre-training with transformers for human pose and mesh reconstruction. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 3415–3425 (2024)

[37] Lin, K., Wang, L., Liu, Z.: End-to-end human pose and mesh reconstruction with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1954–1963 (2021)

[38] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)

[39] Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. ACM Transactions on Graphics **34**(6) (2015)

[40] Ma, T., Nie, Y., Long, C., Zhang, Q., Li, G.: Progressively generating better initial guesses towards next stages for high-quality human motion prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6437–6446 (2022)

[41] Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3d human pose estimation in the wild using improved cnn supervision. In: 2017 international conference on 3D vision (3DV). pp. 506–516. IEEE (2017)

[42] Moon, G., Choi, H., Lee, K.M.: Accurate 3d hand pose estimation for whole-body 3d human mesh estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2308–2317 (2022)

[43] Nam, H., Jung, D.S., Oh, Y., Lee, K.M.: Cyclic test-time adaptation on monocular video for 3d human mesh reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14829–14839 (2023)

[44] Nie, Y., Liu, C., Long, C., Zhang, Q., Li, G., Cai, H.: Multiple-crop human mesh recovery with contrastive learning and camera consistency in a single image. In: Proceedings of the European Conference on Computer Vision (ECCV) (2024)

[45] Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., Schiele, B.: Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In: 2018 international conference on 3D vision (3DV). pp. 484–494. IEEE (2018)

[46] Park, S., Yoo, J., Cho, D., Kim, J., Kim, T.H.: Fast adaptation to super-resolution networks via meta-learning. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16. pp. 754–769. Springer (2020)

[47] Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10975–10985 (2019)

[48] Pavlakos, G., Zhu, L., Zhou, X., Daniilidis, K.: Learning to estimate 3d human pose and shape from a single color image. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 459–468 (2018)

[49] Raghu, A., Raghu, M., Bengio, S., Vinyals, O.: Rapid learning or feature reuse? towards understanding the effectiveness of maml. In: International Conference on Learning Representations (2020)

[50] Rajeswaran, A., Finn, C., Kakade, S.M., Levine, S.: Meta-learning with implicit gradients. Advances in neural information processing systems **32** (2019)

[51] Shetty, K., Birkhold, A., Jaganathan, S., Strobel, N., Kowarschik, M., Maier, A., Egger, B.: Pliks: A pseudo-linear inverse kinematic solver for 3d human body estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 574–584 (2023)

 https://doi.org/10.52202/079017-0371

[52] Sigal, L., Balan, A., Black, M.: Combined discriminative and generative articulated pose and non-rigid shape estimation. Advances in neural information processing systems **20** (2007)

[53] Song, J., Chen, X., Hilliges, O.: Human body model fitting by learned gradient descent. In: European Conference on Computer Vision. pp. 744–760. Springer (2020)

[54] Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5693–5703 (2019)

[55] Sun, Y., Bao, Q., Liu, W., Fu, Y., Black, M.J., Mei, T.: Monocular, one-stage, regression of multiple 3d people. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 11179–11188 (2021)

[56] Tan, V., Budvytis, I., Cipolla, R.: Indirect deep structured learning for 3d human body shape and pose prediction. In: BMVC. vol. 3, p. 6 (2017)

[57] Von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3d human pose in the wild using imus and a moving camera. In: Proceedings of the European conference on computer vision (ECCV). pp. 601–617 (2018)

[58] Wang, W., Ge, Y., Mei, H., Cai, Z., Sun, Q., Wang, Y., Shen, C., Yang, L., Komura, T.: Zolly: Zoom focal length correctly for perspective-distorted human mesh reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3925–3935 (October 2023)

[59] Wang, Y., Daniilidis, K.: Refit: Recurrent fitting network for 3d human recovery. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 14644–14654 (October 2023)

[60] Xiao, P., Xie, Y., Xu, X., Chen, W., Zhang, H.: Multi-person pose forecasting with individual interaction perceptron and prior learning. In: European Conference on Computer Vision. pp. 402–419. Springer (2025)

[61] Xiao, W., Xu, C., Zhang, H., Xu, X.: Spatial-aware gan for instance-guided cross-spectral face hallucination. In: CAAI International Conference on Artificial Intelligence. pp. 93–105. Springer (2022)

[62] Xu, C., Chen, Z., Mai, J., Xu, X., He, S.: Pose-and attribute-consistent person image synthesis. ACM Transactions on Multimedia Computing, Communications and Applications **19**(2s), 1–21 (2023)

[63] Xu, X., Chen, H., Moreno-Noguer, F., Jeni, L.A., De la Torre, F.: 3d human shape and pose from a single low-resolution image with self-supervised learning. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16. pp. 284–300. Springer (2020)

[64] Xu, X., Li, K., Xu, C., He, S.: Gdface: Gated deformation for multi-view face image synthesis. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 12532–12540 (2020)

[65] Xuan, H., Zhang, J., Lai, Y.K., Li, K.: Mh-hmr: Human mesh recovery from monocular images via multi-hypothesis learning. CAAI Transactions on Intelligence Technology (2024)

[66] Xue, Y., Chen, J., Zhang, Y., Yu, C., Ma, H., Ma, H.: 3d human mesh reconstruction by learning to sample joint adaptive tokens for transformers. In: Proceedings of the 30th ACM International Conference on Multimedia. pp. 6765–6773 (2022)

[67] Yoshiyasu, Y.: Deformable mesh transformer for 3d human mesh recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17006–17015 (2023)

[68] Zanfir, A., Bazavan, E.G., Xu, H., Freeman, W.T., Sukthankar, R., Sminchisescu, C.: Weakly supervised 3d human pose and shape reconstruction with normalizing flows. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16. pp. 465–481. Springer (2020)

[69] Zanfir, A., Bazavan, E.G., Zanfir, M., Freeman, W.T., Sukthankar, R., Sminchisescu, C.: Neural descent for visual 3d human pose and shape. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14484–14493 (2021)

[70] Zhang, B., Qi, C., Zhang, P., Zhang, B., Wu, H., Chen, D., Chen, Q., Wang, Y., Wen, F.: Metaportrait: Identity-preserving talking head generation with fast personalized adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22096–22105 (2023)

[71] Zhang, H., Tian, Y., Zhang, Y., Li, M., An, L., Sun, Z., Liu, Y.: Pymaf-x: Towards well-aligned full-body model regression from monocular images. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023)

[72] Zhang, H., Tian, Y., Zhou, X., Ouyang, W., Liu, Y., Wang, L., Sun, Z.: Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11446–11456 (2021)

[73] Zheng, C., Liu, X., Qi, G.J., Chen, C.: Potter: Pooling attention transformer for efficient human mesh recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1611–1620 (2023)

[74] Zhou, S., Fu, H., Liu, L., Cohen-Or, D., Han, X.: Parametric reshaping of human bodies in images. ACM transactions on graphics (TOG) **29**(4), 1–10 (2010)

[75] Zhou, X., Kalantari, N.K.: Look-ahead training with learned reflectance loss for single-image svbrdf estimation. ACM Transactions on Graphics (TOG) **41**(6), 1–12 (2022)

 https://doi.org/10.52202/079017-0371

A Appendix

A.1 Inference Speed Comparison with Previous Methods

In Table 6, we provide a comparison with ReFit [59], NIKI [31], and PLIKS [51] in terms of inference speed. Except for NIKI using ResNet-34, all others employ HRNet-W48, and all tested on a single NVIDIA RTX3090 GPU. Ours$_{\mathrm{CLIFF}}$ takes 0.072s for an iteration, and about 1.1s for 14 iterations in default. ReFit, NIKI, and PLIKS take 0.043s, 0.068s, 0.041s, respectively, which are faster. When our method performs single-step optimization, the time required is comparable to the above three methods. When we conduct additional iterations of optimization, our method consumes more time. As a reward, the additional optimizations improve human mesh recovery accuracy upon regression approaches.

Table 6: **Inference speed comparison** with ReFit [59], NIKI [31], PLIKS [51].

Method	Inference Speed (per sample)
ReFit	0.043s
NIKI	0.068s
PLIKS	0.041s
Ours$_{\mathrm{CLIFF}}$	0.072s (1 iteration)

A.2 Ablation on Learning Rate

In Table 7, we report the experimental results when using different learning rates in our method. We adjusted the learning rates for both the test-time optimization and the ordinary training optimization. Among the feasible learning rates, we observe that utilizing 1×10^{-5} for the test-time optimization and 1×10^{-4} for the training optimization is the most suitable configuration. We find that our training process is unstable under some combinations of the two learning rates, e.g., when the two learning rates are very different from each other (1e-6 for the test-time optimization and 1e-4 for the training optimization), or using too large learning rates (e.g., 1e-3).

Table 7: **Ablation study of different learning rate settings**, with COCO [38] as the training dataset and 3DPW [57] as the testing dataset. "-" means the training is not stable, and no result is obtained. Gray row is the default setting.

Test-time_lr	Training_lr	MPJPE ↓	PA-MPJPE ↓	PVE ↓
1e-6	1e-4	-	-	-
1e-5	1e-4	**76.8**	**49.5**	90.1
1e-4	1e-4	77.1	49.6	**89.1**
1e-3	1e-4	-	-	-
1e-4	1e-3	-	-	-
1e-4	1e-5	78.8	50.8	91.0
1e-4	1e-6	121.6	74.0	134.5

A.3 Ablation on the Number of Test-time Optimization Steps at Training

In the main paper, we perform just one step of test-time optimization in each training iteration. Here, we explore using more test-time optimization steps during training, and show its impact on the model performance. The results are shown in Table 8. It can be observed that with the increment in the number of steps, there is a slight improvement in model performance. However, this comes at the cost of increased memory usage and longer training time. To save training time, we opted for a single test-time optimization step at the training stage.

A.4 Per-Joint Error Analysis

As shown in Figure 6, we explore the performance gains of Ours$_{\mathrm{CLIFF}}$ over EFT$_{\mathrm{CLIFF}}$ at each joint. Specifically, we compute the per-joint error of MPJPE and PA-MPJPE, then we subtract the result of EFT$_{\mathrm{CLIFF}}$ from our result. Darker red means our method is more better. It can be observed that the joints on feet achieve larger performance gains, primarily due to the higher motion frequency in foot joints. This phenomenon was also observed in the study [11, 40, 10, 60, 62, 61, 64]. The advantages

Table 8: **Ablation study on using different test-time optimization (Opt.) steps at training stage**, with COCO [38] as the training dataset and 3DPW [57] as the testing dataset. Gray row is the default setting. Bold values are the best.

Test-time Opt. Steps at Training	MPJPE ↓	PA-MPJPE ↓	PVE ↓	Training Time (mins/epoch)
1	76.6	49.5	90.1	**5.8mins**
2	76.7	49.0	90.0	6.8mins
3	**76.3**	**48.5**	**89.5**	7.4mins

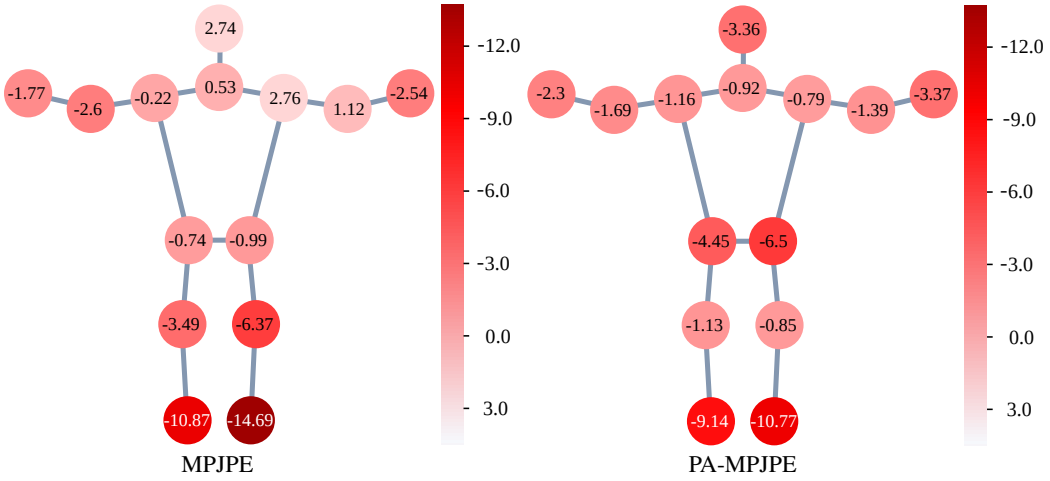


Figure 6: **Per-joint error analysis** between $\text{Our}_{\text{CLIFF}}$ and $\text{EFT}_{\text{CLIFF}}$. The testing dataset is 3DPW [57].

Table 9: **Ablation study on 3D Pseudo SMPLs.** Models are trained on full training datasets and tested on 3DPW [57].

Method	MPJPE↓	PA-MPJPE↓	PVE↓
Ours $_{\text{CLIFF}}$† (Res-50) w/o pseudo	68.7	44.7	85.9
$\text{Ours}_{\text{CLIFF}}$† (Res-50) w/ pseudo	66.0	42.1	83.6

of our method are more obvious on PA-MPJPE, *i.e.*, our method outperforms $\text{EFT}_{\text{CLIFF}}$ in terms of PA-MPJPE for all joints, demonstrating that our method can better capture the pose and shape of the 2D human than $\text{EFT}_{\text{CLIFF}}$.

A.5 More Explanations about Unifying the Training and Testing Objectives with Dual Networks

There is a concern that even with the introduction of the auxiliary network, we still can not fully achieve the goal of matching training and testing objectives, i.e., there still remains a discrepancy between the training loss (using ground truth labels) and the testing-time loss (using pseudo labels). We argue that, since the auxiliary network is trained simultaneously with the main network, the auxiliary network learns the pseudo 3D meshes that are most suitable for optimizing the network during test-time optimization. In other words, we use the pseudo 3D labels generated by the auxiliary network to help mitigate the gap between training and testing losses, compared with using the 2D joints only as supervision at the test stage.

To verify the impact of 3D pseudo SMPLs, we have already conducted an ablation study, as shown in Rows 2 and 3 of Table 3. In Row 2, we discard the auxiliary network, meaning the pseudo SMPLs generated by the auxiliary network are not used in the test-time optimization function. The comparison reveals that the inclusion of pseudo SMPLs improves the model's performance, demonstrating their contribution to the optimization process.

Since the above ablation was conducted using a smaller dataset (COCO), we further validate the results by repeating the experiment on the full training datasets. As shown in Table 9, these additional

https://doi.org/10.52202/079017-0371

Table 10: **Ablation Study about Ensembling Effects.** Models are trained on COCO [38] and tested on 3DPW [57].

Method	MPJPE↓	PA-MPJPE↓
$\text{EFT}_{\text{CLIFF}}$	84.6	54.2
$\text{EFT}_{\text{2CLIFFs}}$	82.7	53.6
$\text{Ours}_{\text{CLIFF}}$	76.7	49.5

Table 11: **Quantitative Comparison with Kim et al. [24].**

Method	Backbone	Training Dataset	Testing Dataset	PA-MPJPE↓
Kim et al. [24]	SPIN [28]	COCO, MPII, Human3.6M, MPI-INF-3DHP, 3DPW, LSP	3DPW	57.88
Ours	HMR [23]	COCO, MPII, Human3.6M, MPI-INF-3DHP, 3DPW	3DPW	44.3

findings reinforce the effectiveness of the generated pseudo 3D SMPLs, confirming their positive influence on model performance.

One may concern that using dual networks introduces an ensembling effect, with the second term of Eq. 9 serving as an adjustment towards an intermediate estimation between the two networks.

To evaluate the ensembling effect introduced by our dual-network setup, we conducted an experiment using two CLIFF models to generate SMPLs and computed their average SMPL, on which EFT optimization was performed. The final results, including the averaged output, are shown in the Table 10 as $\text{EFT}_{\text{2CLIFFs}}$, demonstrating the impact of using two CLIFFs within EFT.

The results indicate that incorporating two CLIFFs does indeed improve EFT performance, highlighting an ensembling effect. However, even with these improvements, $\text{EFT}_{\text{2CLIFFs}}$ underperforms compared to our method. This suggests that the ensembling effect achieved through meta-learning in our approach is more effective than simply using two CLIFF networks.

A.6 Comparison with Kim et al. [24] and Analysis

To provide a quantitative comparison, we collect results from Kim et al. [24]. Kim et al. [24] use the SPIN backbone—a stronger model than our HMR backbone, and include the LSP dataset in their training data (which we exclude). Our method achieves a better PA-MPJPE score (44.3 vs. 57.88), as shown in Table 11. This result indicates that our approach surpasses Kim et al.'s performance despite using a relatively simpler backbone and fewer training resources.

We elaborate the key difference between Kim et al. [24] and our method in detail.

The key difference is that that Kim et al. [24] use the 2D reprojection loss only (please see Eq. 1 in their paper) in both inner and outer loops of meta learning, while our method uses 2D and 3D losses. This is a small difference in formulation, but a large difference in contextualization.

EFT [22] is performed under the guidance of 2D reprojection loss. Kim et al. [24] extended EFT [22] to both the inner and outer loops of meta learning. In this sense, the method of Kim et al. [24] is a direct extension of EFT [22] to meta learning.

In contrast, our method considers meta learning from the perspective of the complete HMR model trained with 2D and 3D losses. In other words, we use the complete HMR model in both inner and outer loops of meta learning. This is more reasonable, as our aim is to personalize the HMR model on each test sample, rather than a model trained with only 2D reprojection loss.

The above differences enable us to design a dual-network architecture that is very different from the network used in Kim et al. [24]. Overall, we find that incorporating 3D SMPLs into the meta learning is very helpful. It is the utilization of the GT SMPLs that greatly improves the results of our method. Besides utilizing GT 3D SMPLs in the outer loop of meta learning, we additionally generate pseudo SMPLs and incorporate them into the inner loop of the metal learning.

A.7 More Intermediate Results

In Figure 7, we show more stepwise optimization results during testing. It can be seen from these examples that the mesh deviates more or less from the target human after step 0, and the deviation is progressively repaired after several optimization steps. For example, please check the results in row 1. The left arm does not match with the evidence in the image at the very beginning, and this error is corrected after about 4 optimization steps. We show more examples from row 2 to 5 where the arms are wrong at first and corrected afterwards. In row 6 and 7, please pay attention to the legs. In the last three rows, please pay attention to the whole bodies.

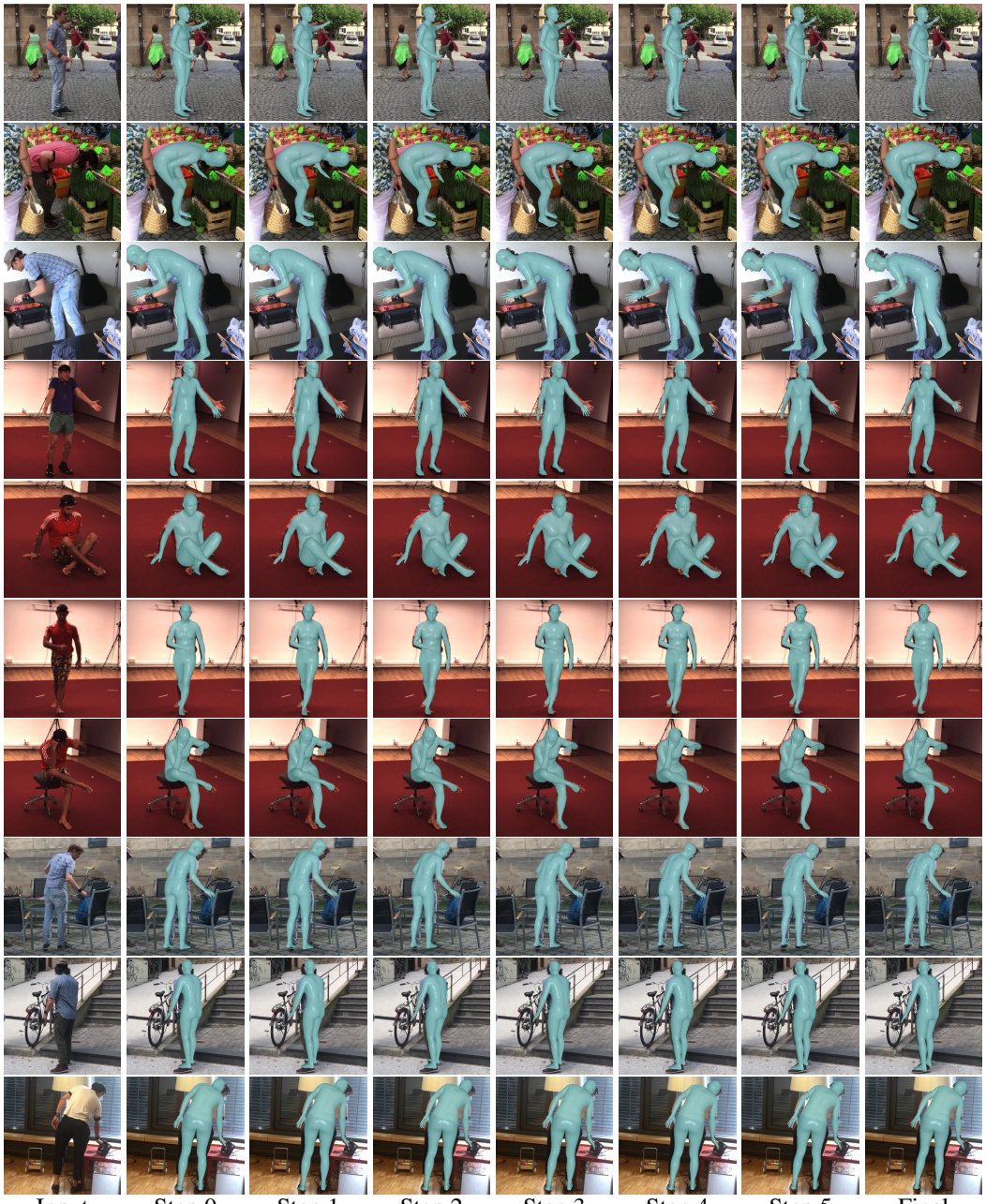


Figure 7: **Stepwise visualization.** From left to right, we showcase results after different steps of test-time optimization during testing.

https://doi.org/10.52202/079017-0371

A.8 More Qualitative Results

In Figure 8 and Figure 9, we show more qualitative comparisons with the latest approaches on test/validation datasets of COCO [38], 3DPW [57] and Human3.6M [20]. Besides CLIFF [33] and ReFit [59] that have been compared with in the main paper, we additionally bring HybrIK [32], NIKI [31], ProPose [13] and $\text{EFT}_{\text{CLIFF}}$ into the comparison. In the shown examples, HybrIK produces misaligned right arm in Figure 8 column 2. NIKI produces wrong head orientations (Figure 8 column 2 and column 4). For ProPose, ReFit and CLIFF, please pay attention to the feet in Figure 8 column 2. $\text{EFT}_{\text{CLIFF}}$ produces misaligned feet in Figure 8 column 4 and Figure 9 column 3. Besides, CLIFF produces wrong left leg for the example in Figure 8 column 1. For these examples, our method produces visually better results.

A.9 Failure Cases

One kind of failure cases of our method are shown in Figure 8 and Figure 9, where there is still slight misalignment between projected mesh and the target 2D evidence, though the misalignment is smaller than that of previous approaches. For example in column 1 of Figure 8, the feet of Our† and Our$*$ do not exactly match with the target 2D feet in the image. We observe that the ground-truth meshes in training datasets after projection also show such artifacts. To tackle this problem, one may need to provide more accurate human annotations.

Figure 10 shows another kind of failure cases. In the example of row 1, our reconstructed mesh and the target 2D person are well-aligned. However, from a novel perspective, there is a misalignment between our 3D mesh and the ground truth. This discrepancy arises from the inherent ill-posedness of inferring a 3D mesh from 2D information in a monocular image. In the second row of Figure 10, we showcase a person with partial occlusion. Please notice the left foot of the person, where the person exhibits self-occlusion. In such a scenario, the accuracy of the corresponding 2D joints is not high, posing a challenge to our method.

Figure 11 shows that the 3D mesh projected onto the 2D image performs poorly in the foot region, probably because the SMPL model itself is not flexible enough to capture the large distortion of the two legs.

Due to the severe occlusion or since the distance from the person to the camera is too far in Figure 12, the quality of the estimated human mesh is poor, not well fitted with the target person.

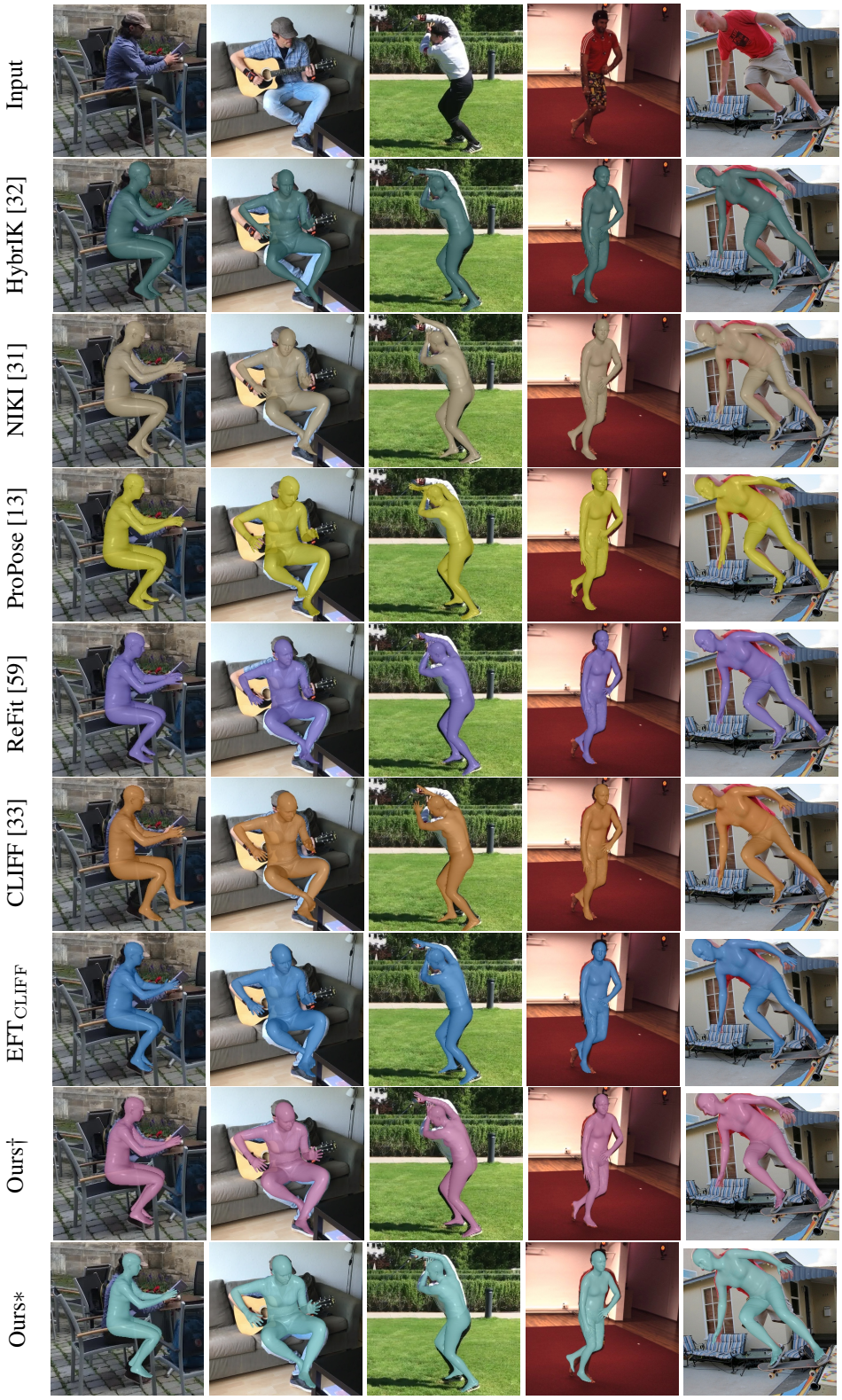


Figure 8: **More qualitative comparisons with SOTA methods.** We show results produced by HybrIK [32], NIKI [31], ProPose [13], ReFit [59], CLIFF [33], EFT$_{\text{CLIFF}}$, and our method (†: OpenPose, ∗: RSN).

https://doi.org/10.52202/079017-0371

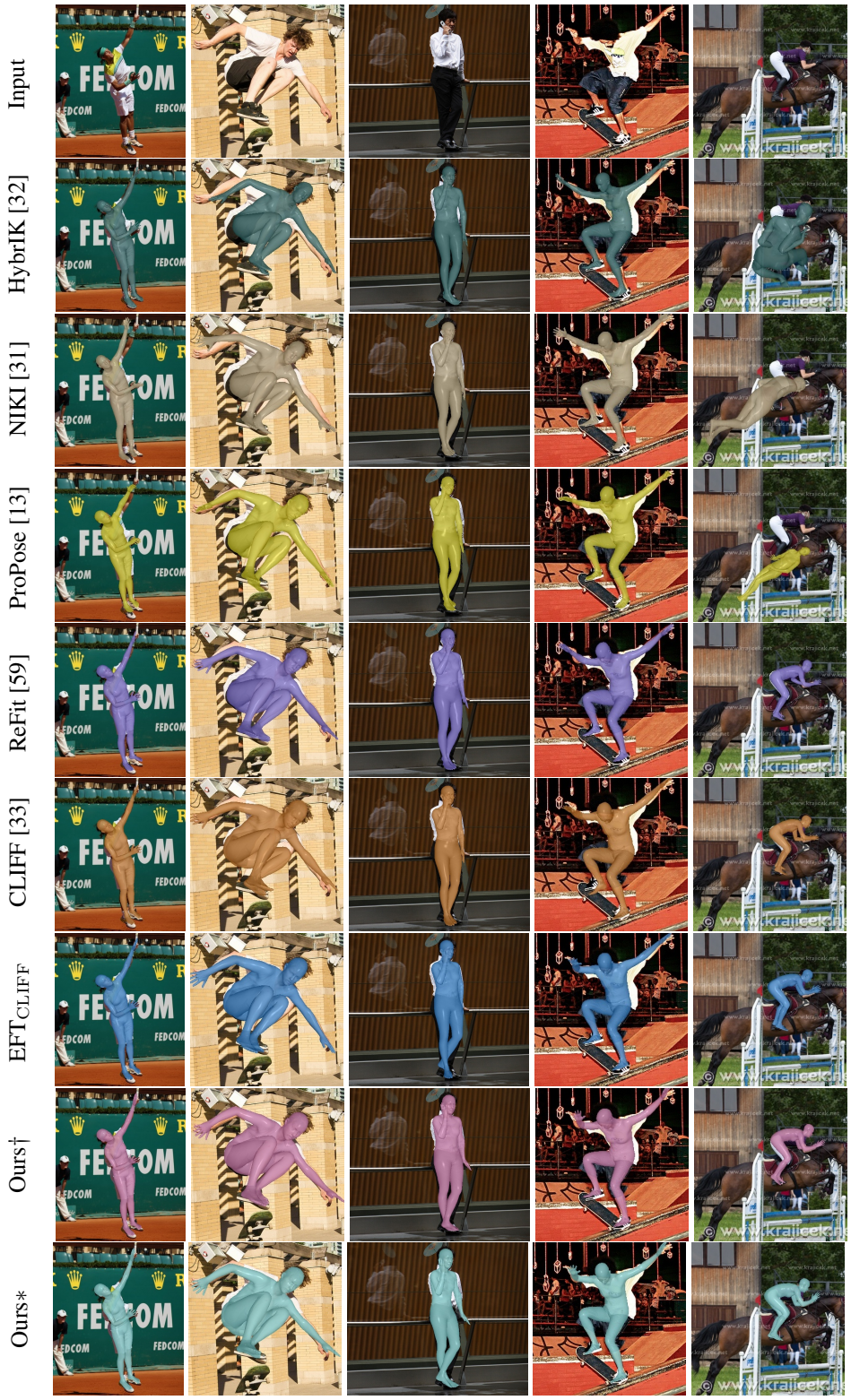


Figure 9: **More qualitative comparisons with SOTA methods.** We show results produced by HybrIK [32], NIKI [31], ProPose [13], ReFit [59], CLIFF [33], EFT$_{\text{CLIFF}}$, and our method (†: OpenPose, ∗: RSN).

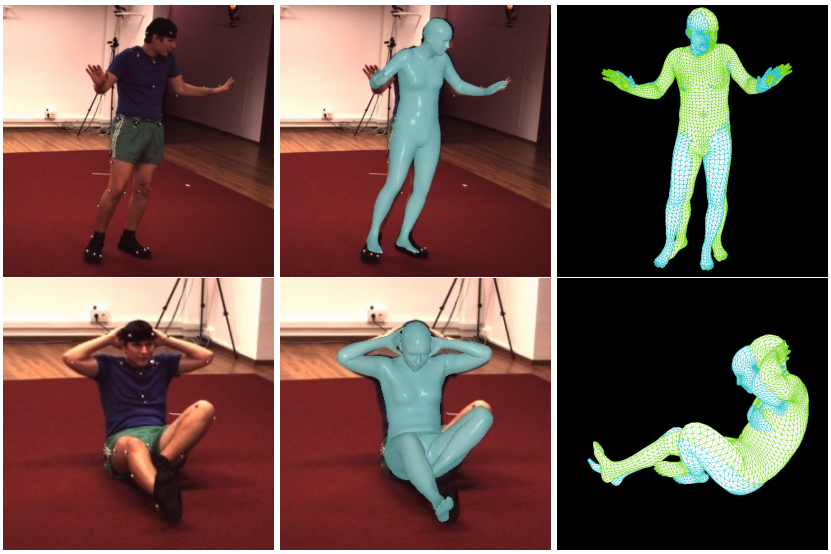


Figure 10: **Failure cases.** The projected mesh matches with the target 2D human body exactly, but there is misalignment in the 3D space. Green meshes represent ground-truth.

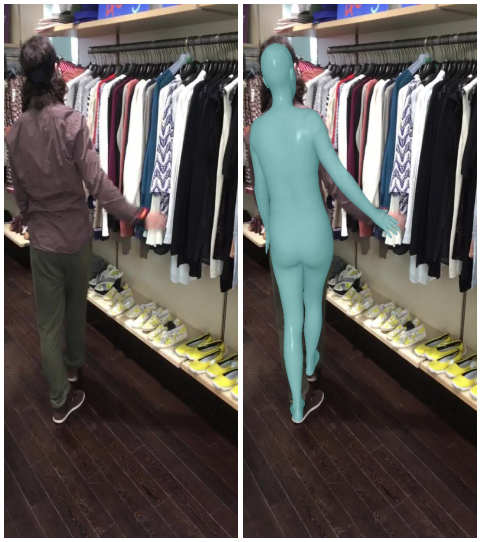


Figure 11: **Failure case**. The SMPL model's limited flexibility likely causes poor 3D mesh projection in the foot region.

Figure 12: **Failure case**. Severe occlusion or far camera distance results in a low-quality human mesh estimation.

https://doi.org/10.52202/079017-0371

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract presents the main idea, highlighting its innovative aspects that distinguish it from other works. The introduction clearly states the main contributions, which are supported by both theoretical analyses (Section 3) and experimental results (Section 4). The claims align with the results presented in the paper, and the scope is well-defined.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: While the main body of the paper does not include a dedicated section on limitations, we have discussed several failure cases(A.9) and potential limitations in the appendix. These cases highlight scenarios where there is still slight misalignment between projected mesh and the target 2D evidence. We chose to include these discussions in the appendix due to space constraints in the main text, prioritizing detailed descriptions of our methodology and results.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The theory of this paper, which includes test-time optimization, incorporation of testing-time into training, unifying training and test-time optimizations, and inference with dual networks, is discussed in Section (3). This section introduces the loss function, the architecture of the network, and the algorithm for updating gradients according to meta-learning.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides detailed descriptions of the experimental setup, including dataset descriptions that were used, hyperparameters, and evaluation metrics. All necessary steps to reproduce the experiments are included either in the main text or in the supplementary material. Additionally, it offers both qualitative and quantitative results for comparison.

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

https://doi.org/10.52202/079017-0371

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper has released the code and provided the download path for the data.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper relies on public datasets which have provided the information of the data splits along with the datasets. The paper provides detailed descriptions of the training and test experiments, including datasets, chosen hyperparameters, the architecture of network, and the criteria for their selection. These details are included in the section(3) and Experiments(4) in main text.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: The paper does not include error bars, confidence intervals, or statistical significance tests.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper specifies the types of compute resources used, including GPU models and memory specifications. This information is detailed in the section Experiments(4).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We ensured that all data and code used was obtained and processed ethically.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.

https://doi.org/10.52202/079017-0371

- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: Human pose estimation has several broader impacts. Positively, it enhances healthcare, optimizes athlete performance, improves human-computer interactions, and reduces workplace accidents. Negatively, it raises privacy concerns, risks misuse, and may lead to unfair outcomes if training data lacks diversity. Due to space limitations, this paper does not discuss these societal impacts in detail.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA] .

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All external assets used in the paper are properly credited. The licenses and terms of use for these assets are explicitly mentioned in the references section and supplementary materials. Human3.6M: CC BY-NC-ND 4.0; MPI-INF-3DHP: CC BY-NC 4.0; 3DPW: CC BY-NC-ND 4.0; COCO: CC BY 4.0; MPII: CC BY 4.0; LSP-Extended: CC BY-NC 4.0.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new assets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

https://doi.org/10.52202/079017-0371

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve research with human subjects requiring IRB Approvals.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.