STAF: 3D Human Mesh Recovery From Video With Spatio-Temporal Alignment Fusion

Wei Yao, Hongwen Zhang, Yunlian Sun, and Jinhui Tang, *Senior Member, IEEE*

***Abstract*— The recovery of 3D human mesh from monocular images has significantly been developed in recent years. However, existing models usually ignore spatial and temporal information, which might lead to mesh and image misalignment and temporal discontinuity. For this reason, we propose a novel Spatio-Temporal Alignment Fusion (STAF) model. As a video-based model, it leverages coherence clues from human motion by an attention-based Temporal Coherence Fusion Module (TCFM). As for spatial mesh-alignment evidence, we extract fine-grained local information through predicted mesh projection on the feature maps. Based on the spatial features, we further introduce a multi-stage adjacent Spatial Alignment Fusion Module (SAFM) to enhance the feature representation of the target frame. In addition to the above, we propose an Average Pooling Module (APM) to allow the model to focus on the entire input sequence rather than just the target frame. This method can remarkably improve the smoothness of recovery results from video. Extensive experiments on 3DPW, MPII3D, and H36M demonstrate the superiority of STAF. We achieve a state-of-the-art trade-off between precision and smoothness. Our code and more video results are on the project page https://yw0208.github.io/staf/.**

***Index Terms*— 3D human mesh recovery, temporal coherence, feature pyramid, attention model.**

I. Introduction

As a promising technology, video-based human mesh recovery can be used for many tasks such as motion monitoring, virtual try-on, VR, etc. It also contributes to traditional human-centered computer vision research, such as action recognition [2] and pose estimation [3], [4], [5]. Therefore, it has received wide attention from the research community and has been developed rapidly in recent years [6]. Especially after the emergence of parametric models that can describe the human body surface in detail (e.g., SMPL [7]), many excellent models have emerged and achieved good results with the development of deep learning.

Recovering the 3D human body from a video is a more complex problem than recovering it from a single image. Many video-based works tried to find effective methods to obtain temporal information. Currently, there are mainly convolutional neural network (CNN) and recurrent neural network (RNN) for learning temporal information [1], [8], [9], [10], [11]. It should be noted that both CNN and RNN are better at learning local information [12], [13] but have difficulty when handling long-range temporal dependencies. Therefore, finding a simple and efficient mechanism for acquiring temporal information is necessary. To leverage temporal cues, the mainstream methods simply fuse the global features extracted from ResNet [14] or HRNet [15] and then use this feature to get the final result. According to previous works [15], [16], [17], [18], [19], feature map tends to retain high-level information after reducing the spatial dimension while ignoring spatial information as well as local details. There are many studies attempting to solve this challenge using pixel-level information, such as body part segmentation [20], [21], [22], UV map [23], [24], [25], [26] and optical flow [27], [28], [29]. But these usually make the model too bloated and still challenging to learn the body structure prior and local details. Moreover, existing video-based and image-based models typically showed severe jitter when applied to video. And this jitter phenomenon cannot be effectively mitigated with the increase in recovery precision. Although there are previous works that attempted to solve this problem, they all sacrifice the recovery precision to some extent. So, achieving a better balance between precision and smoothness is still a difficult challenge.

To address these issues, we propose a spatio-temporal alignment fusion (STAF) model for recovering 3D human meshes from videos. In STAF, a feature pyramid is introduced into the video domain for 3D human reconstruction as the backbone to preserve the original information to the maximum extent. Based on this, we propose a temporal coherence fusion module (TCFM), a spatial alignment fusion module (SAFM) and an average pooling module (APM) for the three problems. In this way, STAF can fully utilize the spatio-temporal information of the input image sequence and achieve a breakthrough in both precision and smoothness with the support of APM. As shown in Fig. 1, STAF outperforms the previous SOTA method in both terms of precision and smoothness.

Specifically, TCFM no longer uses global features as input. Instead, we collect features as input by grid projection on high-dimensional spatial features. This method preserves the original spatial position information to a large extent. In the 3D human reconstruction task, the so-called temporal information refers more to the consistency of human shape and

Manuscript received 31 July 2023; revised 3 January 2024 and 27 May 2024; accepted 3 June 2024. Date of publication 6 June 2024; date of current version 27 November 2024. This work was supported by the National Natural Science Foundation of China under Grant 62332010 and Grant 62076131. This article was recommended by Associate Editor J. Jiao. *(Corresponding author: Yunlian Sun.)*

Wei Yao, Yunlian Sun, and Jinhui Tang are with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: wei.yao@njust.edu.cn; yunlian.sun@njust.edu.cn; jinhuitang@njust.edu.cn).

Hongwen Zhang is with the School of Artificial Intelligence, Beijing Normal University, Beijing 100875, China (e-mail: zhanghongwen@bnu.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2024.3410400.

Digital Object Identifier 10.1109/TCSVT.2024.3410400

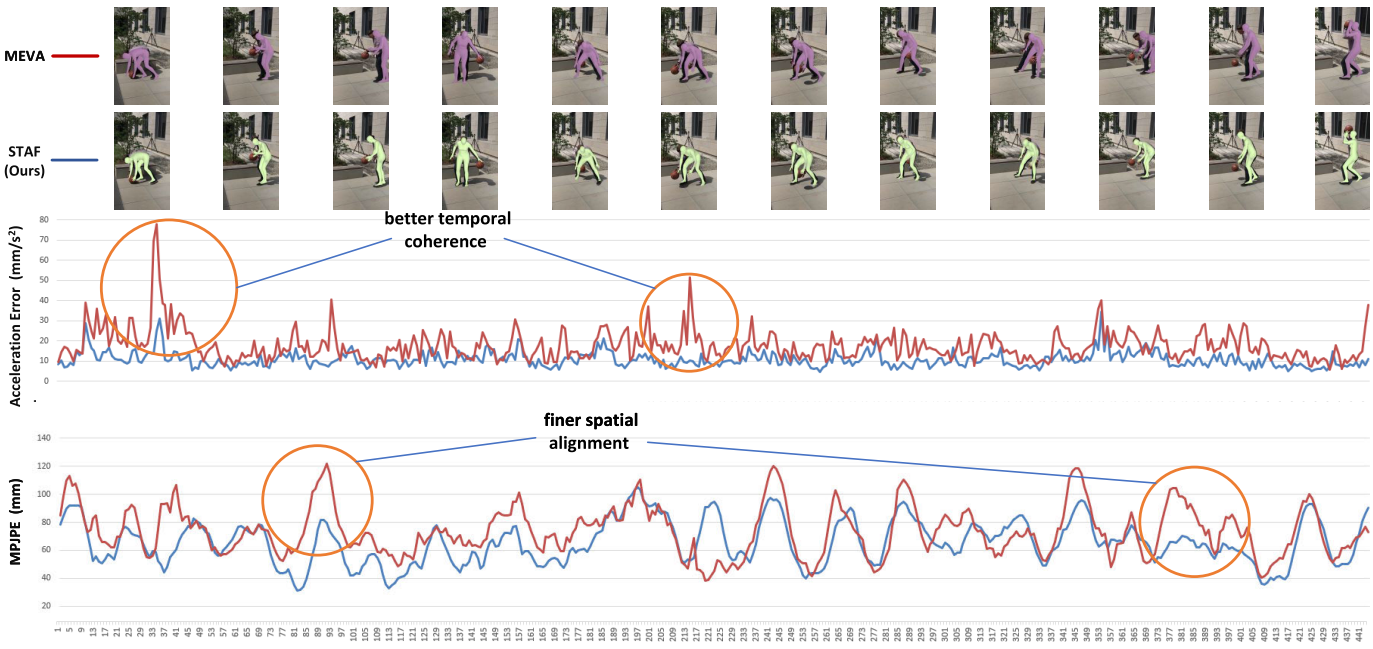

Fig. 1. Comparison with traditional video-based model MEVA [1]. We choose MPJPE and acceleration error to measure the model's performance in space and time. Thanks to our spatio-temporal fusion mechanism, our STAF surpasses MEVA in both metrics.

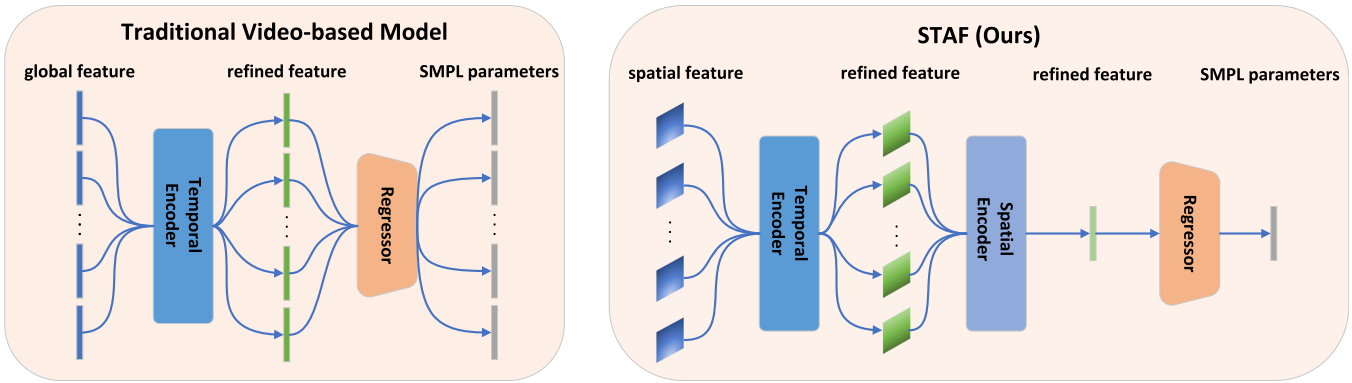

Fig. 2. The difference between traditional video-based models and our STAF. STAF has an additional spatial encoder compared to traditional video-based models. As a result, STAF can obtain more comprehensive refined features and achieve higher recovery precision.

the continuity of pose changes. Therefore, it is necessary to retain the original spatial position information for better learning of the temporal information. When choosing which network architecture to use for temporal encoding, we adopt a self-attention mechanism that is better at establishing long-range dependencies. However, the traditional self-attention module encodes the features before calculating the attention weights. As shown in M_{con} of Fig. 4, we find that this process could destroy the original feature space and instead make it difficult to establish the correct temporal dependencies. For this reason, we add the other self-similarity matrix M_{sim}, which can guide TCFM to encode the temporal information better and thus get more accurate initial human meshes. These initial human meshes enable SAFM to obtain better spatial information about the human body in the following step.

As shown in Fig. 2, compared to traditional models, STAF goes beyond the fusion of temporal information, and further incorporates spatial information. There are two crucial points about SAFM: the extraction of human spatial features, and the other is how to enhance the feature representation of the target frame. We use the projection of the initial human meshes on the feature maps to obtain human spatial features. This has two advantages. First, the mesh alignment cues can be used to correct the result parameters effectively. More importantly, since the features are extracted only in the human body region of the feature map, the model can obtain richer semantic information and focus more on informative human areas by reducing interference from the background. After getting the human spatial features, we need to use them to enhance the feature representation of the target frame. Considering that adjacent images' human shape and pose are more similar, we adopt a multi-stage attention-based adjacent feature fusion mechanism, as shown in Fig. 5. The human spatial information enables STAF to obtain a more precise recovery mesh of the target frame.

But as mentioned earlier, like other traditional models, even though STAF utilizes spatio-temporal information to improve the accuracy further, the smoothness is still not sufficiently

improved. The reason for this is that the model cannot take into account the whole input sequence but focuses only on improving the recovery precision of the target frame. This leads to a lack of transition from frame to frame, which eventually causes frequent and noticeable jitter in the recovered human body. For this reason, we propose the APM that allows the model to focus on the entire input sequence, using each frame's information to generate results that match the human motion in the sequence. This module can significantly improve the smoothness without affecting accuracy. It is worth mentioning that it is also applicable to many existing models. At last, we summarize our contributions as follows:

- For the first time, multi-scale spatial features are introduced into the 3D human mesh recovery task in the video domain. We propose a novel spatio-temporal alignment fusion model to exploit both spatial and temporal information. We propose an effective spatio-temporal feature interaction and integration mechanism that enables the model to take full advantage of motion continuity cues and human spatial information to recover more precise 3D human mesh.
- We find an effective method to significantly improve the smoothness of the estimated mesh sequences from the video. We find that the main reason for the discontinuity of recovered human motion is that traditional models usually focus only on the target frame but not the overall sequence. With our proposed APM, we achieve a remarkable reduction in acceleration error and demonstrate experimentally that the method is somewhat generalizable.
- Extensive experiments on three standard benchmark datasets show that STAF achieves state-of-the-art performance with a better trade-off between precision and smoothness.

II. Related Work

A. Image-Based 3D Human Mesh Recovery

Research on 3D human reconstruction started early and saw explosive growth after the emergence of human parametric models [7], [30], [31] and human datasets with 3D labels [32], [33], [34]. The first works in this field were based on optimization. These optimization-based methods let the parametric model constantly fit the obtained 2D labels (including silhouettes, 2D joint points, part segmentation, etc.) together with the human pre-existing prior [35], [36], [37]. In 2018, Kanazawa et al. proposed the HMR model [38], which was the first end-to-end regression-based model with a single monocular image as input. Using ResNet50 [14] to extract features, HMR used an Iterative Error Feedback (IEF) loop regressor to get the final result and further adopted an action discriminator to ensure the reasonableness of the output 3D reconstruction. Since regression-based models have an absolute speed advantage as well as broader applicability than optimization-based models, a large number of excellent regression-based models [23], [39], [40], [41], [42], [43], [44], [45], [46] have emerged since then. However, image-based methods have their inherent limitations. Even compared to the latest PQ-GCN [46], which was carefully designed, our video-based STAF not only exceeds it in terms of accuracy but also offers much better smoothness.

While regression-based models have proliferated, optimization-based models have not fallen out of favor. Instead, they are combined with regression-based models to obtain models that can generate more accurate human body mesh [47], [48]. Such models generally used regression-based models to generate better initial results and then used the optimization process to obtain more accurate results. Researchers usually use such models to add 3D pseudo-labels to 2D training datasets, which can significantly facilitate the training of their models.

B. Video-Based 3D Human Mesh Recovery

In terms of practical applications, the application of 3D human reconstruction will be more based on videos, and the continuity of human motion contains rich temporal information that can be used. As a result, several video-based models have emerged in recent years. Currently, there are two main categories: sequence-to-sequence [1], [9], [11], [28], [49], where multiple images are input andes all the corresponding human meshes are output, and sequence-to-single-frame [8], [10], [50], [51], where multiple images are input but only the result of the target frame is output. Earlier, there was Arnab et al. [28], which used the entire video as input, generated initial results using an off-the-shelf 2D joint keypoint detector [52] and a 3D human reconstruction model [38], and then continuously optimized the results using temporal coherence. There are also methods extracting features that allow models to learn temporal information adaptively. For example, HMMR [10] used full convolutional networks to encode temporal information, and MEVA [1] adopted recurrent neural networks to learn. Among them, a classic work is VIBE [11], which added a GRU-based module to encode temporal information based on HMR [38] and further designed a temporal version of motion discriminator to ensure the rationality of the output human mesh. With the rise of Transformer [12], the model MAED [49], which used an attention mechanism to learn the continuity of each joint movement, achieved excellent performance.

C. Temporal Continuity of 3D Human Mesh Recovery

When 3D human reconstruction is transferred from a single image to a video, it is not enough to emphasize only the accuracy of the reconstructed mesh. In fact, the visual discomfort caused by the incoherence of human motion is even more pronounced than the inaccuracy. Since the acceleration error proposed by HMMR [10] to measure the smoothness of the recovery results, there have been many works [1], [8], [11], [49], [51] adopted this measure. Theoretically, the more accurate the human reconstruction results are, the lower the acceleration error and the smoother the estimated human motion. However, in practice, with the current accuracy, it is not yet possible to significantly reduce the acceleration error by increasing accuracy. Let us see two structurally similar models, MEVA [1] and VIBE [11]. MEVA sacrificed

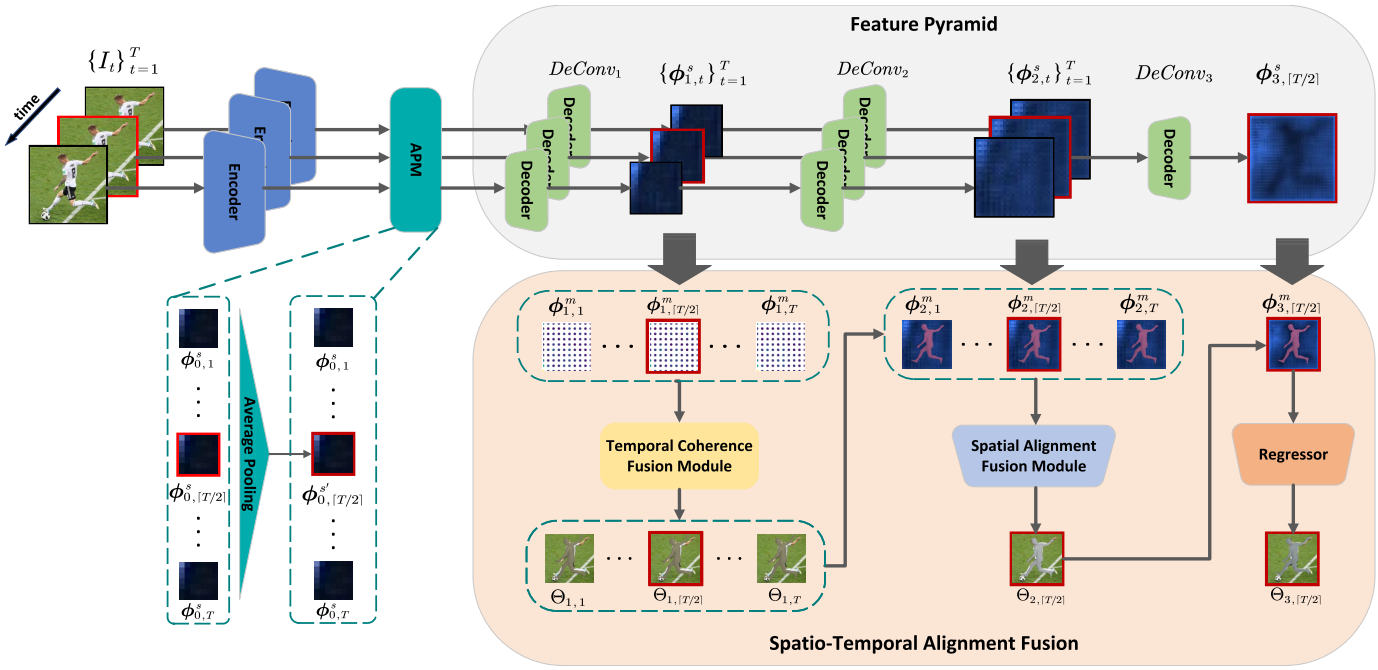

Fig. 3. The overall framework of STAF. We input T images and output the reconstruction result of the target frame $I_{\lceil T/2 \rceil}$ with a red border. We employ a feature pyramid to retain multi-scale spatial information and use projection down-sampling to obtain fine-grained local information. Also, to make full use of the spatio-temporal information, we add an average pooling module, a temporal coherence fusion module and a spatial alignment fusion module. The temporal coherence fusion module is described in Sec. III-C.1, and the spatial alignment fusion module is in Sec. III-C.2. Please refer to Sec. III-D for the entire process of our method.

reconstruction accuracy to improve smoothness, and VIBE improved accuracy but dramatically increased acceleration error. As far as the latest work is concerned, MAED [49] improved the recovery accuracy to a very high level, but the fluency was much poorer than TCMR [8] and MPS-Net [51]. These two works improved the smoothness to an unprecedented level without reducing accuracy. On the one hand, TCMR provided a method to remove the residual connections of features and reduce the feature dependence on the current frame. On the other hand, MPS-Net experimentally demonstrated that its feature integration module named HAFI could significantly reduce acceleration error. Inspired by the above two works, we go a step further and propose a more straightforward method to reduce the dependence on the target frame and significantly improve the smoothness without compromising accuracy.

III. Method

The whole framework of STAF is shown in Fig. 3. With features extracted from input images, we first go through APM to weaken the influence of the target frame but strengthen the model's dependence on the whole sequence. After that, TCFM is designed to learn the temporal information to get initial human meshes. With these initial body meshes, we can obtain finer spatial alignment cues. Next, we propose SAFM to fully integrate these cues to strengthen the target frame's body spatial representation and further correct its recovery result. Finally, the fine-grained local information is extracted by projection sampling and fed into the regressor to obtain the final result. In this section, we present the details of STAF. We first introduce some basic knowledge, including the SMPL model and feature sampling. Then we show two crucial submodules, TCFM and SAFM, and summarize the whole framework at last.

A. 3D Human Representation

In this work, a parametric model called SMPL [7] is used to encode the 3D surface of the human body, which is one of the most widely used 3D human models. In total, the SMPL model parameters Θ consist of three parts: shape $\boldsymbol{\beta}$, pose $\boldsymbol{\theta}$, and camera $\boldsymbol{\pi}$. The shape parameters $\boldsymbol{\beta} \in \mathbb{R}^{10}$ consist of the first 10 coefficients of the PCA shape space, including the body weight, height, and the proportion of each limb. The pose parameters $\boldsymbol{\theta} \in \mathbb{R}^{3J}$ use the 3D rotation of each joint point relative to its parent joint to describe the pose of the human body, where $J = 23$. After obtaining $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$, we can input them to a pre-trained function to obtain $M(\boldsymbol{\theta}, \boldsymbol{\beta}) \in \mathbb{R}^{3\times N}$, which represents the 3D coordinates of the N vertices of the body surface, where $N = 6890$. From this, we get a precise description of the human surface. Also, a global rotation $R \in \mathbb{R}^{3\times 3}$, scale $s \in \mathbb{R}^1$, and translation $t \in \mathbb{R}^2$ can be obtained using camera parameters $\boldsymbol{\pi}$ based on weak perspective camera model. These three parameters are mainly used to project the 3D object onto the 2D image. The 3D object can be the human mesh vertices or 3D joints. Its specific usage will be described in detail in later sections.

B. Feature Down-Sampling

To facilitate understanding, the feature down-sampling of our work is first introduced with a single frame input as an example.

As shown in Fig. 3, we first input the image I into the feature extractor without the last average pooling to get the feature $\boldsymbol{\phi}_0^s \in \mathbb{R}^{C_0 \times W_0 \times H_0}$. After that, the spatial features $\boldsymbol{\phi}_0^s$ are fed into a set of deconvolutional networks $\{DeConv_k\}_{k=1}^3$ to obtain $\left\{\boldsymbol{\phi}_k^s \in \mathbb{R}^{C_k \times W_k \times H_k}\right\}_{k=1}^3$, i.e.,

$$\boldsymbol{\phi}_k^s = DeConv_k\left(\boldsymbol{\phi}_{k-1}^s\right), \text{ for } k > 0. \tag{1}$$

Then we use the 2D projection X_k of the obtained 3D human mesh vertices $M\left(\boldsymbol{\theta}_k, \boldsymbol{\beta}_k\right)$ onto the feature map $\boldsymbol{\phi}_k^s$ to obtain point-wise features $\boldsymbol{\phi}_k^p \in \mathbb{R}^{C_k}$, i.e.,

$$\boldsymbol{\phi}_k^m = \oplus\left\{f\left(\boldsymbol{\phi}_k^p\left(x_{k-1}\right)\right), \text{ for } x_{k-1} \text{ in } X_{k-1}\right\}, \ k > 1 \tag{2}$$

where $\oplus$ represents concatenation, $\boldsymbol{\phi}_k^p\left(x_{k-1}\right)$ denotes acquiring $\boldsymbol{\phi}_k^p$ according to x_{k-1} using bilinear sampling, and $f(\cdot)$ is the MLP that reduces the channel dimension from C_k to C_m. Then we get the feature $\boldsymbol{\phi}_k^m \in \mathbb{R}^{C_m * \tilde{N}}$, where $\tilde{N}$ is the number of mesh vertices.

When $k = 0$, it is worth noting that the information density of $\boldsymbol{\phi}_0^s$ is very high. As illustrated in Fig. 7, the 2D projection of the initial human mesh Θ_0 obviously does not match the actual human body area. Performing projection down-sampling on $\boldsymbol{\phi}_1^s$ thus cannot help the model to focus more on the human body area. In addition, the global information of the image is crucial to estimate the camera parameters. So we choose the grid sampling method to extract global features, when $k = 1$. Grid sampling is that we define a 21×21 grid to acquire point-wise features $\boldsymbol{\phi}_1^p$. The other steps are the same as projection down-sampling.

As for how X_k is obtained, you can refer to this formula

$$X_k = \Pi\left(\mathcal{D}\left(M\left(\boldsymbol{\theta}_k, \boldsymbol{\beta}_k\right)\right)\right), \text{ for } k > 1, \tag{3}$$

where Π is an orthographic projection function based on camera parameters $\boldsymbol{\pi}_k$, and $\mathcal{D}(\cdot)$ represents down-sampling $\tilde{N}$ vertices from N human mesh vertices.

C. Spatio-Temporal Alignment Fusion

1) Temporal Coherence Fusion Module: For video-based models, an important design is how to implement feature interaction to capture temporal coherence effectively. Inspired by the non-local module in [13] and [51], we introduce a lightweight temporal coherence fusion module, as illustrated in Fig. 4. TCFM is a further improvement on the commonly used transformer structure. The main difference is that we add an extra correlation matrix M_{sim}. The traditional transformer usually encodes the features before computing the correlation matrix, as we get M_{con} in Fig. 4. However, from the visualization of M_{con}, the network does not correctly establish the temporal coherence. The traditional transformer does not work as expected but only focuses on some frames with more information. Therefore, we additionally add M_{sim} for steering the model so that each frame is more dependent on frames closer to itself and less on frames further away. As shown by M_{sim} and M_g in Fig. 4, the diagonal region is brighter, meaning TCFM learns the temporal coherence between frames more efficiently than traditional transformer.

The interaction objects of our temporal coherence fusion module are $\left\{\boldsymbol{\phi}_{1,t}^m\right\}_{t=1}^T$, where T is the number of input frames.

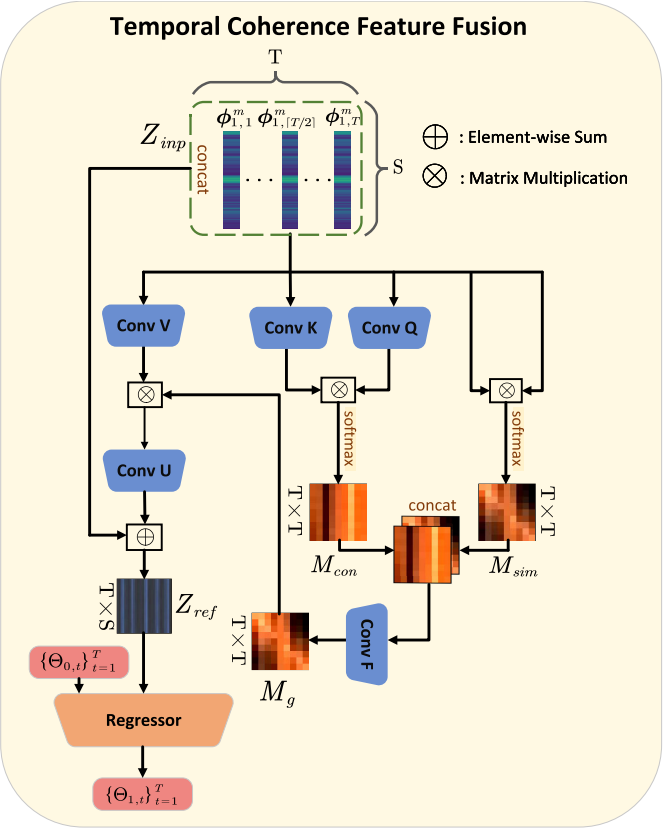

Fig. 4. The structure of the temporal coherence fusion module. With T features as input, the module outputs T temporal refined features. We use TCFM to get initial human meshes. Note $\{\Theta_{0,t}\}_{t=1}^T$ is set as the mean $\overline{\Theta}$ following [38]. As for the correlation matrix, it calculates the coherence between the frames by multiplying two feature matrices. The correlation matrix is a $T \times T$ matrix. The element of the i-th row and j-th column represent the coherence between the i-th frame and the j-th frame. Larger values indicate stronger coherence. The brighter color indicates a larger value.

As shown in the Fig. 4, the feature matrix $Z_{inp} \in \mathbb{R}^{T \times S}$ composed of $\left\{\boldsymbol{\phi}_{1,t}^m\right\}_{t=1}^T$ is input to the module, where S is the feature length of $\boldsymbol{\phi}_{1,t}^m$. We first input Z_{inp} to three convolutional networks $\mathcal{Q}$, $\mathcal{K}$, $\mathcal{V}$ to obtain refined feature matrices $\{Z_q, Z_k, Z_v\} \in \mathbb{R}^{T \times \frac{S}{m}}$. After that, we obtain two correlation matrices

$$\begin{cases} M_{con} = softmax\left(Z_q Z_k^T\right) \\ M_{sim} = softmax\left(Z_{inp} Z_{inp}^T\right) \end{cases} \in \mathbb{R}^{T \times T}. \tag{4}$$

From the visualization of the correlation matrix M_{con} in Fig. 4, we can see that each frame's features are almost equally similar to the other frames', which is clearly not intuitive. Theoretically, every single frame should be more similar to the frames closer to itself. Therefore, for better feature refinement, higher weights should be given to more similar frames in the correlation matrix. In our work, in addition to M_{con}, we further use M_{sim} to guide temporal coherence learning, i.e.,

$$M_g = softmax\left(\mathcal{F}\left(concat\left(M_{con}, M_{sim}\right)\right)\right), \tag{5}$$

where $\mathcal{F}(\cdot)$ is a CNN that make $concat\left(M_{con}, M_{sim}\right) \in \mathbb{R}^{2 \times T \times T}$ downscale to $\mathbb{R}^{T \times T}$. This module effectively enhances the ability of the model to learn long-range temporal

features. Finally, we get the refined features Z_{ref} by the following formula

$$Z_{ref} = Z_{inp} + \mathcal{U}\left(M_g Z_v\right), \tag{6}$$

where $\mathcal{U}(\cdot)$ is a convolutional layer, which let $M_g Z_v \in \mathbb{R}^{T \times \frac{S}{m}}$ upscale to $\mathbb{R}^{T \times S}$. With that, we can use the residual connection as in previous works. After that, we divide Z_{ref} into T features and feed them into the regressor separately to obtain a set of initial body meshes. These initial body meshes will be used to obtain spatial alignment clues and human spatial information for the next module SAFM.

2) Spatial Alignment Fusion Module: Traditional video-based models often stop at exploiting temporal information. To overcome this issue, we propose SAFM to utilize the spatial information of each frame. "Spatial" is reflected in the fact that we do not directly use the full image information as input but further filter the spatial pixel alignment information for fusion. "Alignment" has two meanings. On the one hand, it means extracting features by aligning mesh vertices to feature maps, which include mesh-image alignment information. On the other hand, it implies the ability of SAFM to enhance the feature representation of target frames by aligning spatial information. The most significant difference between spatial information and the previous temporal information is also reflected here. When fusing the temporal information in the first stage, the focus we consider is the temporal coherence of the input frames. So, we take the full image information as input. However, in the second stage, we need more fine-grained information, i.e., spatial features of the human body. The pose of the human body tends to be different from frame to frame, but the shape is kept consistent. Meanwhile, the human body poses in neighboring frames tend to have some correlation. Based on the above discussion, we design a unique spatial feature fusion approach. SAFM can thus enhance the spatial feature representation of the target frame with supplement of the whole sequence, and obtain more accurate recovery results. The structure of SAFM is shown in Fig. 5.

We illustrate how this module works with an input sequence of 9 features $\left\{\boldsymbol{\phi}_{2,t}^m\right\}_{t=1}^{9}$, as we do in the final version of STAF. Following [51], we set each group to contain three frames, which has been shown to be the most efficient. As shown in Fig. 5, the feature sequences $\left\{\boldsymbol{\phi}_{2,1}^m, \boldsymbol{\phi}_{2,2}^m, \cdots, \boldsymbol{\phi}_{2,9}^m\right\}$ are first divided into three groups $\left\{\boldsymbol{\phi}_{2,t-1}^m, \boldsymbol{\phi}_{2,t}^m, \boldsymbol{\phi}_{2,t+1}^m\right\}$, where $t = 2,\ 5,\ 8$. We then input each group into an attention module to obtain the integrated features $\left\{\boldsymbol{\phi}_{2,p}^m, \boldsymbol{\phi}_{2,c}^m, \boldsymbol{\phi}_{2,f}^m\right\}$. $\left\{\boldsymbol{\phi}_{2,p}^m, \boldsymbol{\phi}_{2,c}^m, \boldsymbol{\phi}_{2,f}^m\right\}$ represent features of the **p**ast, **c**urrent and **f**uture frames, respectively. After that, we feed $\left\{\boldsymbol{\phi}_{2,p}^m, \boldsymbol{\phi}_{2,c}^m, \boldsymbol{\phi}_{2,f}^m\right\}$ into the attention module again to get the final refined feature $\boldsymbol{\phi}_{2,ref}^m$ for the target frame. Note that all the attention modules mentioned above share the same network architecture and weights when deployed in practice.

Next, we describe how the attention module works. With $\left\{\boldsymbol{\phi}_{2,t-1}^m, \boldsymbol{\phi}_{2,t}^m, \boldsymbol{\phi}_{2,t+1}^m\right\}$ as input, the attention module first reduces the dimension of each feature through a fully connected (FC) layer $fc(\cdot)$, i.e.,

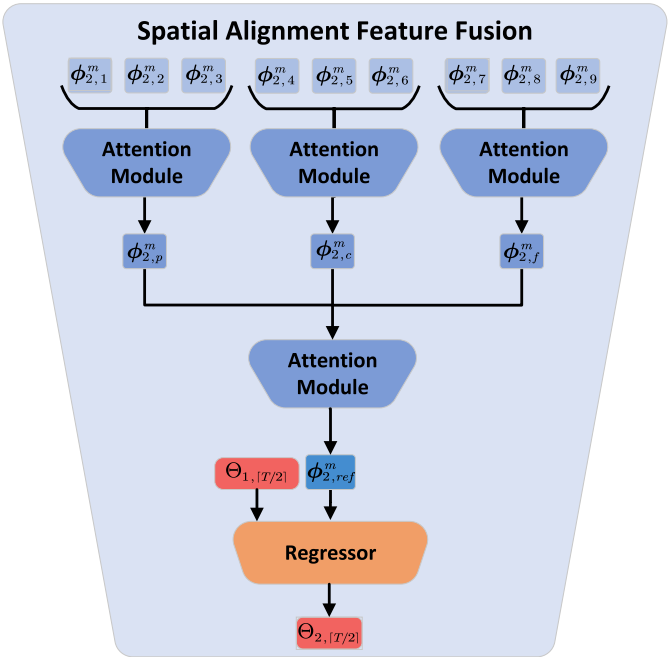

Fig. 5. The structure of spatial alignment feature fusion module. Take the example of entering nine features $\left\{\boldsymbol{\phi}_{2,1}^m, \boldsymbol{\phi}_{2,2}^m, \cdots, \boldsymbol{\phi}_{2,9}^m\right\}$. Start with a group of three features and integrate them into one feature through the attention module. Then the three integrated features $\left\{\boldsymbol{\phi}_{2,p}^m, \boldsymbol{\phi}_{2,c}^m, \boldsymbol{\phi}_{2,f}^m\right\}$ are integrated again into one feature $\boldsymbol{\phi}_{2,ref}^m$. We use $\boldsymbol{\phi}_{2,ref}^m$ to recover the 3D human mesh of the target frame.

$$\boldsymbol{\phi}_{2,concat}^m = \oplus\left\{fc\left(\boldsymbol{\phi}_{2,t-1}^m\right),\ fc\left(\boldsymbol{\phi}_{2,t}^m\right),\ fc\left(\boldsymbol{\phi}_{2,t+1}^m\right)\right\}, \tag{7}$$

where $\oplus$ represents concatenation. The resized feature $\boldsymbol{\phi}_{2,concat}^m$ is then passed through another three FC layers with tanh activation to reduce the channel size to 3. We add a softmax activation in the end to calculate attention weights $\{\alpha_1, \alpha_2, \alpha_3\}$. Finally, we get the integrated feature

$$\boldsymbol{\phi}_{2,integ}^m = \alpha_1 \boldsymbol{\phi}_{2,t-1}^m + \alpha_2 \boldsymbol{\phi}_{2,t}^m + \alpha_3 \boldsymbol{\phi}_{2,t+1}^m,\ t = 2,\ 5,\ 8 \tag{8}$$

where $\boldsymbol{\phi}_{2,integ}^m$ are the features $\left\{\boldsymbol{\phi}_{2,p}^m, \boldsymbol{\phi}_{2,c}^m, \boldsymbol{\phi}_{2,f}^m\right\}$ mentioned above. For example, $\boldsymbol{\phi}_{2,integ}^m$ is $\boldsymbol{\phi}_{2,p}^m$ when $t = 2$.

We use the features themselves to obtain attention weights, and then apply the attention weights to compute a weighted sum of the original features. This attention module fully preserves the spatial information of the original features, allowing this design to be embedded into other models without destroying the feature space. And this module can effectively tell the model which frame should be biased to integrate features better. It is worth mentioning that this multi-level integration approach considers only adjacent frames for each integration. Without such a multi-level design, it would get difficult to establish long-range spatial dependency. More importantly, model sizes would expand dramatically when the input sequence length gets too long. By adopting such a multi-level integration mechanism, SAFM can accommodate various input lengths.

D. The Overall Model

At the end of this section, we present the overall structure of STAF, as shown in Fig. 3. Given a sequence of images $\{I_t\}_{t=1}^{T}$, a set of spatial features $\left\{\boldsymbol{\phi}_{0,t}^{s} \in \mathbb{R}^{C_0 \times W_0 \times H_0}\right\}_{t=1}^{T}$ is obtained after a CNN-based encoder. We mark the target frame with red borders in Fig. 3. Then comes an essential operation, i.e.,

$$\boldsymbol{\phi}_{0,\lfloor T/2 \rfloor}^{s\prime} = Avg\left(\left\{\boldsymbol{\phi}_{0,t}^{s}\right\}_{t=1}^{T}\right), \tag{9}$$

where Avg means average pooling. This module APM enables the model to rely less on the feature of the target frame $I_{\lceil T/2 \rceil}$ but take full advantage of the information of each frame. The feature obtained after average pooling is used to replace the original feature of the target frame. For convenience, $\boldsymbol{\phi}_{0,\lfloor T/2 \rfloor}^{s\prime}$ continues to be named $\boldsymbol{\phi}_{0,\lfloor T/2 \rfloor}^{s}$.

As described in Fig. 3, the features $\left\{\boldsymbol{\phi}_{0,t}^{s}\right\}_{t=1}^{T}$ are fed into the deconvolution network to get features $\left\{\boldsymbol{\phi}_{1,t}^{s}\right\}_{t=1}^{T}$. And then $\left\{\boldsymbol{\phi}_{1,t}^{s}\right\}_{t=1}^{T}$ are sampled by the grid to obtain the features $\left\{\boldsymbol{\phi}_{1,t}^{m}\right\}_{t=1}^{T}$. Before sending the features into the regressor, we first feed them into the temporal coherence fusion module to fully learn the motion continuity dependencies. For more details, please refer to Sec. III-C.1. This allows STAF to achieve not only better initial mesh recovery $\left\{\Theta_{1,t}\right\}_{t=1}^{T}$, but also more accurate projection sampling used in the next step.

The features $\left\{\boldsymbol{\phi}_{1,t}^{s}\right\}_{t=1}^{T}$ continue to be fed into the decoder consisting of deconvolution to obtain the feature sequence $\left\{\boldsymbol{\phi}_{2,t}^{s}\right\}_{t=1}^{T}$. Unlike the former step, for the features $\left\{\boldsymbol{\phi}_{2,t}^{m}\right\}_{t=1}^{T}$ obtained by projection sampling, we input them into the spatial alignment fusion module to obtain the feature $\boldsymbol{\phi}_{2,ref}^{m}$ for the target frame. Owing to further deconvolution and projection sampling, the features $\left\{\boldsymbol{\phi}_{2,t}^{m}\right\}_{t=1}^{T}$ contain rich fine-grained local information. The operation of multi-level adjacency integration can effectively enhance the mesh-alignment cues and enrich the human body information of $\boldsymbol{\phi}_{2,\lceil T/2 \rceil}^{m}$. Finally, we feed $\boldsymbol{\phi}_{2,ref}^{m}$ into the regressor together with the SMPL parameters $\Theta_{1,\lceil T/2 \rceil}$ obtained in the previous step to get the recovery result $\Theta_{2,\lfloor T/2 \rfloor}$ of the target frame.

For the last update of the SMPL parameters, we first send the features $\boldsymbol{\phi}_{2,\lceil T/2 \rceil}^{s}$ into the decoder to get the features $\boldsymbol{\phi}_{3,\lceil T/2 \rceil}^{s}$. Then we apply projection down-sampling to it to get the features $\boldsymbol{\phi}_{3,\lceil T/2 \rceil}^{m}$. Finally, $\boldsymbol{\phi}_{3,\lceil T/2 \rceil}^{m}$ concatenated with SMPL parameters $\Theta_{2,\lceil T/2 \rceil}$ is passed through the regressor to get the final result $\Theta_{3,\lceil T/2 \rceil}$.

E. Loss Function

For model training, we use three basic loss functions within the 3D human mesh recovery domain. Following TCMR [8], the first is the loss function L_{smpl} of the SMPL parameters. It calculates the $L2$ loss between the predicted and ground-truth SMPL parameters. It should be noted that the datasets with the ground-truth SMPL parameters are very scarce. In order to take the vast datasets with ground-truth 2D and 3D joint coordinates into consideration, we introduce the other loss functions L_{2D} and L_{3D}. The 3D joint coordinates can be obtained directly from the SMPL parameters, i.e., $X(\boldsymbol{\theta}, \boldsymbol{\beta}) \in \mathbb{R}^{3\times P}$, where P is the number of joints. For the 2D joint coordinates x, we adopt the projection of 3D joints as follows:

$$x = s\Pi\left(RX\left(\boldsymbol{\theta}, \boldsymbol{\beta}\right)\right) + t \tag{10}$$

where Π is a projection function and R, s, t are obtained from camera parameters. In conclusion, our loss function can be summarized as

$$L = \lambda_{smpl}\left\|\Theta - \hat{\Theta}\right\|_2 + \lambda_{3D}\left\|X - \hat{X}\right\|_2 + \lambda_{2D}\left\|x - \hat{x}\right\|_2 \tag{11}$$

where λ_{smpl}, λ_{3D} and λ_{2D} are weights and would be 0 when relevant annotation is unavailable.

IV. Experiments

In this section, we describe the implementation details and experimental results. A series of experimental demonstrations and visualization results are also reported to prove the validity of the innovative points in our work.

A. Datasets

1) COCO: Common Objects in Context [53] is a large-scale image dataset widely used for various computer vision tasks such as object detection, image segmentation, and image captioning. It is provided by Microsoft and consists of over 330,000 images, each with detailed annotations. For our task, we primarily utilize the part of the COCO that focuses on human subjects. COCO provides 2D joint location labels. And based on this, we use EFT [48] to add pseudo labels, such as 3D joint positions and SMPL parameters, to the dataset. Since COCO is not a video dataset, we use it to train our base model only, allowing the base model to acquire the initial ability to extract features about the human body.

2) LSP & LSP-Extended: Leeds Sports Pose [54] is a classic benchmark dataset used for human pose estimation. It consists of training and test sets, each containing 1,000 images with 2D joint labels. Later, LSP-Extended [55] is introduced, which adds an additional 10,000 images for training. We only use the training set of LSP and LSP-Extended for training our base model. Additionally, we utilize pseudo-labels generated by EFT to enhance supervision.

3) Human 3.6M: As a widely used 3D human body dataset, Human 3.6M [32] has been used as a benchmark dataset by many works for its large data volume and rich 3D labels. It should be noted that this dataset is collected indoors. It is thus often used together with in-the-wild datasets. However, its abundant 3D labels, stable objects and scenes are all useful to get human prior. Following previous work, we use subsets S1, S5, S6, S7, S8 for training and S9, S11 for testing. Since the videos in Human 3.6M are at 50 fps, which causes data redundancy, we extract frames at a frame rate of 25 fps. Note that the SMPL parameters obtained from Mosh are no longer publicly available due to legal reasons. Therefore, we use the pseudo SMPL labels provided by NeuralAnnot [56] to supervise the training following [8] and [51].

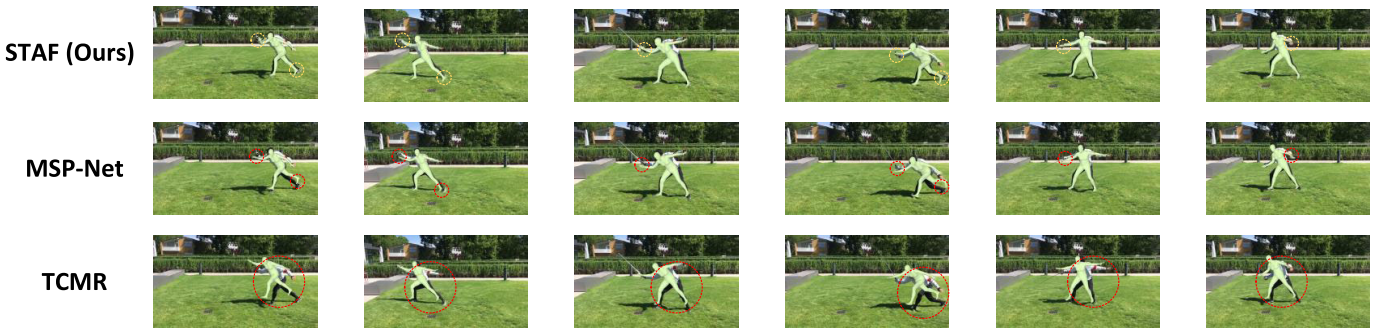

Fig. 6. Qualitative comparison between STAF and two latest works (MPS-Net [51] and TCMR [8]). Traditional video-based models usually pursue only temporal coherence but miss spatial information, which might result in misalignment between the recovered mesh and image. Our STAF instead can effectively solve this problem.

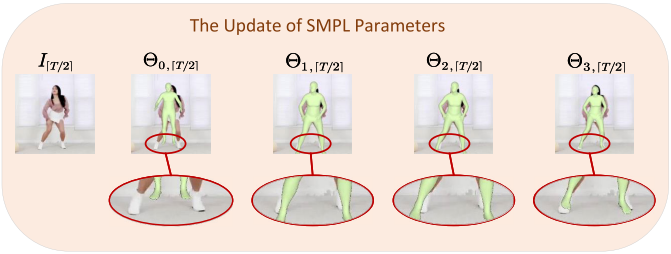

Fig. 7. Visualization of output results of STAF regressors for each stage. It shows how the final results are obtained from SMPL mean parameters after adjustment by the three regressors.

4) MPII: MPII (Max Planck Institute for Informatics) [57] is a large-scale image dataset used for human pose estimation. This dataset is provided by the Max Planck Institute for Informatics in Germany and contains approximately 25,000 images along with corresponding pose annotations. We only use the images with complete 2D joint labels for training.

5) 3DPW: 3D Human Pose in the Wild [33] is a challenging dataset, since its data is collected from both indoors and outdoors. This dataset provides 3D joint coordinates, so we use it to enhance the model's adaptability to complex situations. Also, because it is very challenging, it is the main dataset for our experimental evaluation. We test both models trained with and without 3DPW to demonstrate the generalization ability of STAF.

6) MPII3D: MPI-INF-3DHP [34] is also a dataset with 3D joints coordinates. It acquires ground-truth labels through a multi-camera marker-less motion capture system. It includes data obtained from indoors and outdoors, which is also a very tough dataset. And more and more works use it to perform experimental evaluation. In our experiment, we use MPII3D for both training and testing.

7) Insta: InstaVariety [10] is a very large dataset with 2D labels, although its 2D joint coordinates are pseudo-labels generated by OpenPose. Its videos are collected from Instagram, so it is very content-rich and can complement the shortage of other datasets. We use it to perform weakly supervised training and enhance the generalization ability of the model.

8) PoseTrack: PoseTrack [58] is a multi-person video-based dataset with 2D labels. Although it is intended to provide a benchmark for pose estimation and multi-person tracking, we use it for training to increase the amount of training data.

Due to the two-stage training process of our model, the datasets used in each stage are not entirely the same. In the first stage, we train the base model with single-frame inputs, allowing us to utilize some nonvideo datasets. Following [59], we use COCO [53], LSP [54], LSP-Extended [55], Human 3.6M [32], MPII [57], and MPII3D [34]. In the second training stage, we begin training the complete version of STAF, which requires video datasets. In addition to the previously mentioned Human 3.6M and MPII3D, we also incorporate 3DPW [33], Insta [10], and PoseTrack [58] for training, aiming to complement the limited training data. Overall, our training data volume remains consistent with previous works. Following previous works, we evaluate our approach in 3 classic benchmarks, i.e., 3DPW, MPII3D and Human 3.6M.

B. Implementation Details

We choose Resnet50 [14] without the last average pooling as the encoder, which takes 9 images as input. It is worth mentioning that, in order to recover human meshes for all frames of a video, we choose to use a repeated set of 9 images as input for the first 4 frames and the final 4 frames. Since the image size is 224×224, the size of initial spatial features $\left\{\boldsymbol{\phi}^s_{0,t}\right\}_{t=1}^{9}$ is $2048 \times 7 \times 7$. As for $\left\{\boldsymbol{\phi}^s_{1,t}, \boldsymbol{\phi}^s_{2,t}, \boldsymbol{\phi}^s_{3,t}\right\}_{t=1}^{9}$, we keep their channel length constant, but their width and height are $\{14 \times 14,\ 28 \times 28,\ 56 \times 56\}$.

For the first regressor, since we use 21×21 grid sampling and further reduce the channel length from $C_k = 2048$ to $C_m = 5$, the size of the input features $\left\{\boldsymbol{\phi}^m_{1,t}\right\}_{t=1}^{9}$ gets $21 \times 21 \times 5 = 2205$. For the other two regressors, we adopt projection down-sampling to calculate the features. Since the standard SMPL model generates too many vertices (6890), it is impracticable to use all of them to perform projection down-sampling. Following [43], we down-sample 6890 vertices to get a sparse human body mesh with only 431 vertices. The length of input features thus becomes $431 \times 5 = 2155$. To summarize, the features $\left\{\boldsymbol{\phi}^m_{1,t}, \boldsymbol{\phi}^m_{2,t}, \boldsymbol{\phi}^m_{3,t}\right\}_{t=1}^{9}$ have lengths of $\{2205, 2155, 2155\}$, resp.

In the classical HMR [38] regressor, the input features are 2048 in length and go through three loops. Our regressors are consistent with the classical one but change the input scale. Considering that the HMR regressor takes three loops, we adopt a total of three regressors, too. Finally, in TCFM,

TABLE I
COMPARISON WITH SOTA METHODS ON 3DPW AND MPII3D (* INDICATES TRAINING WITH 3DPW)

	Model	Backbone	3DPW				MPI-INF-3D		
			PA-MPJPE ↓	MPJPE ↓	PVE ↓	Accel ↓	PA-MPJPE ↓	MPJPE ↓	Accel ↓
image-based	HMR [38] 2018	ResNet-50	76.7	130.0	-	37.4	89.8	124.2	-
	GraphCMR [39] 2019	ResNet-50	70.2	-	-	-	-	-	-
	SPIN [47] 2019	ResNet-50	59.2	96.9	116.4	29.8	67.5	105.2	
	I2L-MeshNet [60] 2020	ResNet-50	57.7	93.2	110.1	-	-	-	-
	PyMAF [23] 2021	ResNet-50	58.9	92.8	110.1	-	-	-	-
	PARE [59] 2021	HRNet-W32	50.9	82.0	97.9	-	-	-	-
video-based	HMMR [10] 2019	ResNet-50	72.6	116.5	139.3	15.2	-	-	-
	Sim2Real [9] 2019	ResNet-50	74.7	-	-	-	-	-	-
	Temporal Context [28] 2019	ResNet-50 (from HMR)	72.2	-	-	-	-	-	-
	DSD-SATN [50] 2019	ResNet-50	69.5	-	-	-	-	-	-
	MEVA* [1] 2020	ResNet-50 (from SPIN)	54.7	86.9	-	11.6	65.4	96.4	11.1
	VIBE* [11] 2020	ResNet-50 (from SPIN)	51.9	82.9	99.1	23.4	64.6	96.6	-
	VIBE [11] 2020	ResNet-50 (from SPIN)	56.5	93.5	113.4	27.1	63.4	97.7	-
	TCMR* [8] 2021	ResNet-50 (from SPIN)	52.7	86.5	102.9	7.1	63.5	97.3	8.5
	TCMR [8] 2021	ResNet-50 (from SPIN)	55.8	95.0	111.3	**6.7**	62.8	97.4	**8.0**
	MPS-Net* [51] 2022	ResNet-50 (from SPIN)	52.1	84.3	99.7	7.4	62.8	96.7	9.6
	MPS-Net [51] 2022	ResNet-50 (from SPIN)	54.0	91.6	109.6	7.5	-	-	-
	Ours*	ResNet-50 (pre-trained)	**48.0**	**80.6**	**95.3**	8.2	59.6	93.7	10.0
	Ours	ResNet-50 (pre-trained)	48.7	81.2	96.0	8.2	**58.8**	**92.4**	10.1

the three convolutional networks $\mathcal{Q}$, $\mathcal{K}$, $\mathcal{V}$ reduce the input dimension 2205 by half to 1102.

Our base model consists of an encoder, three decoders, a down-sampling network and three regressors. It serves as our baseline for validating the effectiveness of the proposed modules. Additionally, a pre-trained base model is also utilized to provide a good initialization for STAF.

C. Training Details

1) Stage 1: The base model is first trained on COCO [53] for 175 epochs with a batchsize of 64. In the second stage, we train the base model on a mixed dataset for 60 epochs. Pseudo SMPL labels produced by EFT [48] are used for supervising. The mixed dataset consists of Human 3.6M(50%), and MPII3D(20%). And the remaining 30% of the mixed dataset is composed of COCO, LSP, LSP-Extended and MPII. The whole process takes about 4 days.

2) Stage 2: When training STAF in our work, we use the pre-trained base model to initialize the parameters, except for the two modules TCFM and SAFM. Next, following [1], [8], [11], and [51], we train the network on a mixed dataset consisting of Insta, PoseTrack, Human 3.6M, 3DPW and MPII3D for 45 epochs with a mini-batchsize of 32. There are only 60% of the training data with 2D labels. Note that Resnet50 is frozen during this stage of training. Image preprocessing including cropping method is referenced from VIBE [11] and MEVA [1]. The training and testing video frame rate is 25 to 30 frames per second. Note that no data augmentation is applied in our work. The model weights are updated by the Adam optimizer with an initial learning rate of 0.00005. And the learning rate is reduced by a factor of 10 when the best performance is not updated for every 5 epochs. We train the model until it converges. In practical training, it typically takes around 18 hours.

All training is performed on a single RTX 3090. The code implementation relies on Pytorch [61].

D. Comparison With the State-of-the-Art Methods

To demonstrate the superiority of STAF, we first show its evaluation results on 3DPW, MPII3D, and Human 3.6M. We compare our model with other previous excellent models. The results are shown in Table I and Table III. Following [1], [8], [11], [51], we use four standard evaluation metrics. The most comprehensive and representative metric is the mean per joint position error (MPJPE). Another important metric is PA-MPJPE, which expresses the Procrustes-aligned mean per joint position error. It removes the error introduced by the camera model by forcing the alignment. Note that PA-MPJPE evaluates only the accuracy of the recovered joints. The Per Vertex Position Error (PVE) calculates the error of the mesh vertices, but it is so redundant that it often does not match the actual qualitative result of the model. The units of the above metrics are all in mm. Another key metric is the acceleration error (Accel), which is calculated as the acceleration error of the joint points in mm/s^2. It can be used to evaluate the smoothness of the reconstructed meshes. Note that the above joints and vertices are all in three dimensions.

We begin in Table I by summarizing the performance of some outstanding works over the past three years on 3DPW and MPII3D. These two datasets are chosen because they contain challenging in-the-wild data. The performance on these two challenging datasets can better demonstrate the model's robustness. As seen from Table I, the all-around performance of STAF exceeds that of many previous SOTA models. STAF achieves optimal performance on three key metrics: PA-MPJPE, MPJPE, and PVE. Compared with the latest work MPS-Net, STAF reduces the MPJPE by 3.7 mm and 3.0 mm on 3DPW and MPII3D, respectively. As mentioned earlier, in the past, it often has to sacrifice smoothness for precision, as in VIBE [11], or conversely, sacrifice precision for smoothness, as in MEVA [1]. STAF instead achieves a better trade-off between precision and smoothness. Our acceleration error remains very low while we achieve high reconstruction precision. In terms of smoothness, STAF far

TABLE II
COMPARISON OF NETWORK PARAMETERS AND MODEL SIZE

Model	#Parameters (M)	Model Size (MB)
VIBE	72.43	776
MEVA	85.72	858.8
TCMR	108.89	1073
Ours	**51.12**	**359.8**

TABLE III
COMPARISON WITH SOTA METHODS ON H36M

Model	Human 3.6M		
	PA-MPJPE ↓	MPJPE ↓	Accel ↓
HMMR [10]	56.9	-	-
VIBE [11]	53.3	78.0	27.3
MEVA [1]	53.2	76.0	15.3
TCMR [8]	52.0	73.6	3.9
MPS-Net [51]	47.4	**69.4**	**3.6**
Ours	**44.5**	70.4	4.8

exceeds image-based models and is second only to TCMR and MPS-Net among video-based models. In Fig. 1, we randomly select a video to test and plot the acceleration error. As shown, our model avoids severe jitter suffered by traditional video-based models and reaches a new level of overall smoothness.

In addition to this, STAF shows surprising generalizability. The * in Table I indicates that the 3DPW training set is used for training, and the absence of * indicates that it is not used. From Table I, we can see that the PA-MPJPE and MPJPE of the previous models on 3DPW increase by 3.65-5.88% and 8.7-12.8%, respectively, when the models are not trained with the 3DPW training set. However, the PA-MPJPE and MPJPE of STAF increase only by 1.5% and 0.7%. On one hand, this can be explained by the small percentage of 3DPW in our training set, which accounts for only 0.5%. On the other hand, it also demonstrates the stronger generalization ability of STAF, which can still achieve good evaluation results even without in-domain training data.

In order to further demonstrate the complexity and efficiency of STAF, we report the number of parameters and the model size of STAF compared to some other models in Table II. However, due to our input consisting of only 9 frames, direct comparisons of FLOPs with models that utilize 16-frame inputs may not be entirely fair. Therefore, we disregard FLOPs in our comparison. From the perspective of parameters and model size, our model is significantly smaller than models that employ RNN or CNN to learn temporal information. Therefore, STAF exhibits higher model efficiency.

Since H36M [32] no longer publicly provides ground-truth SMPL parameters from Mosh, it is not fair to compare STAF directly with those models that use SMPL parameters from H36M for training. Contrary to the common perception, H36M is not an "easy" dataset, although its data are collected indoors. As mentioned in EFT [48], many models that perform exceptionally well on H36M but poorly on 3DPW are often overfitting on the H36M training set. Therefore, we follow TCMR [8] and MPS-Net [51] and reproduce some models without using the ground-truth SMPL parameters of H36M.

TABLE IV
ABLATION RESULTS ON 3DPW

model	3DPW			
	PA-MPJPE ↓	MPJPE ↓	PVE ↓	Accel ↓
base	49.3	83.5	99.5	27.5
base+APM	48.4	81.8	96.9	**8.1**
base+STAF	48.8	82.3	97.8	24.7
base+APM+STAF (Ours)	**48.0**	**80.6**	**95.3**	8.2

TABLE V
ABLATION RESULTS OF AVERAGE POOLING MODULE ON 3DPW

Model	3DPW			
	PA-MPJPE ↓	MPJPE ↓	PVE ↓	Accel ↓
HMR	54.3	91.4	107.2	29.1
HMR+HAFI	54.2	90.9	107.3	29.1
HMR+APM	53.1	87.9	103.3	**7.7**
MPS-Net w/o HAFI	53.0	**86.7**	**102.2**	23.5
MPS-Net w/ HAFI	53.2	87.7	102.9	8.0
MPS-Net w/ APM	**52.8**	87.5	102.7	7.8

The evaluation results are summarized in Table III, where some of the results are from [8] and [51]. It can be seen that STAF still achieves competitive results with equivalent training sets. STAF reduces PA-MPJPE by 2.9 mm compared to MPS-Net, which indicates STAF produces more precise human meshes.

E. Ablation Study

In this section, we demonstrate the contribution of our work. First, we validate the effectiveness of each module added to the base model. Then, we verify the applicability of APM to other models. Finally, we show how we determine the optimal way to combine TCFM and SAFM.

1) Ablation Experiments: We conduct a series of experiments on 3DPW to show the contribution of each module of STAF. The results are summarized in Tabel IV. In the table, APM denotes the average pooling module, STAF represents the combination of TCFM and SAFM, and base indicates the base model. For a fair comparison, the base model is also trained with the same second training stage as the subsequent ablation experiments. The evaluation results of the base model indicate that the acceleration error is still high, although the precision of human mesh recovery reaches a high level. This is a pain point that is difficult to be solved by many image-based models. Even many video-based models cannot improve the smoothness much. With the addition of the average pooling module, the acceleration error is easily reduced by 70.5%, but the precision is not affected too much and even increased. STAF also brings an all-round improvement, with PA-MPJPE, MPJPE, PVE, and Accel reduced by 1.3 mm, 1.2 mm, 4.2 mm and 2.8 mm/s^2, respectively. From the last row, we can see that the combination of APM and STAF achieves the best results. Although the acceleration error increases by 0.1 mm/s^2 compared to base+APM, the precision is improved. A good balance between precision and smoothness is achieved.

2) Effect of Average Pooling Module: Next, we demonstrate the effect of our average pooling module, and the related

results are in Table V. Our inspiration is drawn from the HAFI module of MPS-Net [51]. The evaluation results of MPS-Net w/o HAFI are from [51]. The evaluation results of MPS-Net w/ HAFI are reproduced by ourselves and are similar to the results of [51]. It can be found that MPS-Net achieves such a low acceleration error relying mainly on the HAFI module.

However, we do not achieve the same effect when adding HAFI to the classic model HMR [38]. Since the output of HAFI is a weighted sum of the features, we output the weights obtained from both the pre-trained MPS-Net and HMR+HAFI. Note that HMR+HAFI represents taking a sequence as input to HAFI and then sending the integrated features to HMR. As shown in Fig. 8, compared to HMR+HAFI, MPS-Net does not focus on the target frame effectively but on the whole input sequence.

Obviously, HMR+HAFI is more reasonable since our goal is to get the result for the target frame, which typically is the middle frame of the input sequence. So, the middle weight deserves to be the largest. However, our experiment results demonstrate that it is the over-reliance on the feature of the target frame that leads to high acceleration error. A similar point has been mentioned in TCMR [8]. The evaluation results of MPS-Net w/ APM and MPS-Net w/ HAFI in Table V also prove our point. Next, we replace the HAFI module of MPS-Net with our APM and find the acceleration error is still reduced. Therefore, we can conclude that HAFI's ability to minimize the acceleration error sharply is not attributed to its attention module design but benefits from the equal treatment of each frame, i.e., attaching similar weights to features of each frame. Another weak point of HAFI is its poor generalization ability. In most cases, its attention module is still automatically biased toward the target frame during training. Our APM, instead, is easier to generalize because it forces the model to handle each frame equally. We also test it on the classical model HMR [38] to verify its generalizability. The effect is noticeable, with a 73.5% drop in acceleration error. Hence, we believe APM is a simple and effective way to improve smoothness and can be easily embedded in both image-based and seq2frame video-based models.

3) Ablation Study of TCFM and SAFM: The final ablation experiment is to find the best combination of TCFM and SAFM. First, we introduce the meaning of the first column in Table VI. TCFM refers to **T**emporal **C**oherence **F**usion **M**odule and SAFM refers to **S**patial **A**lignment **F**usion **M**odule. And the n in TCFMn and SAFMn indicates that the input features of this module are $\{\boldsymbol{\phi}_{n,t}^{m}\}_{t=1}^{9}$. Note that the output feature sequence of TCFMn is the input of SAFMm, when $n = m$. Because SAFM must come after TCFM, and to avoid bloat, we do not consider module reuse. So there are $4+3+2+3=12$ combinations in total. The average pooling module is also applied during this experiment.

The evaluation results of all combinations are presented in Table VI. The best combination is finally found, i.e., TCFM1+SAFM2. As for why this is the case, it is explainable. First, in an iterative error feedback loop, the latter regressor outputs a smaller $\triangle\Theta$, i.e., the lower-level features have less impact on the final result. As shown in Fig. 7, the output

Fig. 8. The attention weights generated by the attention module of each model. MPS-Net is [51]. HMR+HAFI refers to a seq2frame video-based model composed of a classical single-frame model [38] plus HAFI. As for TCFM1+SAFM1 and TCFM1+SAFM2, please refer to Sec. IV-E.3. As we can see, neither MPS-Net nor TCFM1+SAFM1 can focus on the target frame correctly. HMR+HAFI instead focuses too much on the target frame and cannot take into account the temporal coherence. Our STAF, however, can focus on the whole input sequence with a slight bias towards the target frame so as to obtain a better balance between precision and smoothness.

TABLE VI
ABLATION RESULTS OF SPATIO-TEMPORAL FUSION MODULE ON 3DPW

Model	3DPW			
	PA-MPJPE ↓	MPJPE ↓	PVE ↓	Accel ↓
TCFM1	48.5	81.8	97.1	8.1
TCFM1+SAFM1	49.4	83.2	98.4	8.5
TCFM1+SAFM2	**48.0**	**80.6**	**95.3**	8.2
TCFM1+SAFM3	48.8	81.6	96.6	8.3
TCFM2	48.9	82.7	98.5	8.2
TCFM2+SAFM2	48.8	81.8	97.4	8.3
TCFM2+SAFM3	48.9	82.2	97.6	8.4
TCFM3	49.2	83.2	99.1	**8.0**
TCFM3+SAFM3	49.0	82.0	97.3	8.1
SAFM1	48.4	81.3	96.2	8.2
SAFM2	48.5	80.7	95.7	8.2
SAFM3	48.8	82.1	97.0	**8.0**

of the first regressor is very close to the final result, and the last two regressors just need to do a little fine-tuning on the details. Therefore, the benefit of refining higher-level features is supposed to be greater. The evaluation results also prove this point. The evaluation metrics of TCFM1 to TCFM3 and SAFM1 to SAFM3 in Table VI both show an increasing trend. To answer why TCFM1+SAFM2 is better than TCFM1+SAFM1, we output the weights generated by SAFM in TCFM1+SAFM2 and TCFM1+SAFM1. As shown in Fig. 8, the attention weights generated by TCFM1+SAFM2 are in line with our expectation that the attention model is only slightly biased towards the middle frame. However, the attention weights generated by TCFM1+SAFM1 are obviously unreasonable.

We believe that this is because SAFM1 adopts the refined feature of TCFM1 as input. But TCFM1 destroys the spatial structure of the original features, making SAFM1 difficult to learn them correctly. More importantly, if TCFM1+SAFM1

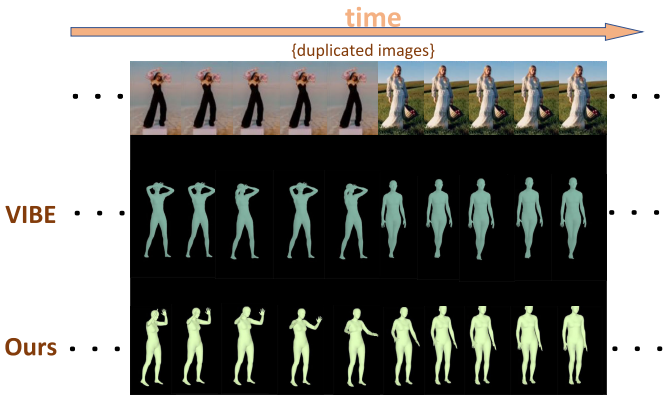

Fig. 9. Visualization of an extreme example, where the human pose in the video suddenly changes dramatically. Compared to VIBE [11], STAF can estimate a smoother human motion process.

is used, SAFM cannot well use the human spatial information as well as mesh-alignment cues of each frame to enhance the feature representation of the target frame. TCFM1+SAFM1 is similar to traditional video-based models because they all just apply a temporal encoding of the features. Although TCFM1+SAFM2 is not the lowest in acceleration error, it is optimal in all other evaluation metrics. So, it is finally chosen under comprehensive consideration.

V. Discussion

In this section, we would like to discuss the issue of over-smooth in STAF. As can be seen in Fig. 9, we design an extreme example in which we forcefully merge two individuals with different poses into a single video as input. While models like VIBE [11], which prioritize accuracy, generate poses without any transition, STAF generates smooth transitions from one pose to another. On one hand, this demonstrates that our model does indeed exhibit over-smooth in such extreme cases. On the other hand, this example also showcases the capability of STAF to estimate smooth results.

To address this issue, we adopt a shorter input sequence in our model. This is done to prevent an excessive sequence length, which could impact the precision of recovery from the target frame. It is also essential to be aware that if sequences are too brief, it may be challenging to acquire enough temporal information.

Taking these factors holistically into account, we choose to use a 9-frame input sequence, striking a balance between smoothness and precision. For more qualitative results, please refer to our project page. We also encourage you to run our program to generate video demos.

VI. Conclusion

In this paper, we presented a novel seq2frame video-based model for 3D human mesh recovery. We proposed spatio-temporal alignment fusion to preserve spatial information and further exploit both temporal and spatial information. We introduced the temporal coherence fusion module that takes full advantage of the motion coherence without destroying the original feature space. In addition to the temporal encoder, we proposed the spatial alignment fusion module. We cleverly used spatial information and alignment cues to further correct the recovery result of the target frame. Except for the above, we revealed the cause of the temporal discontinuity that previous works suffer from, i.e., over-reliance on the target frame. We thus proposed the averaging pooling module, which reduces the model's reliance on the target frame and enhances the overall attention of the input sequence. It improved the smoothness substantially without affecting the recovery precision and can be easily embedded in other image-based and seq2frame video-based models. Compared with the previous 3D human mesh recovery models, STAF achieved a better trade-off between precision and smoothness.

References

[1] Z. Luo, S. A. Golestaneh, and K. M. Kitani, "3D human motion estimation via motion compression and refinement," in *Proc. Asian Conf. Comput. Vis.*, 2020, pp. 324–340.

[2] G. T. Papadopoulos and P. Daras, "Human action recognition using 3D reconstruction data," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1807–1823, Aug. 2018.

[3] G. Wei, C. Lan, W. Zeng, and Z. Chen, "View invariant 3D human pose estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 12, pp. 4601–4610, Dec. 2020.

[4] R. Gu, G. Wang, Z. Jiang, and J. Hwang, "Multi-person hierarchical 3D pose estimation in natural videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 11, pp. 4245–4257, Nov. 2020.

[5] T. Chen, C. Fang, X. Shen, Y. Zhu, Z. Chen, and J. Luo, "Anatomy-aware 3D human pose estimation with bone-based pose decomposition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 1, pp. 198–209, Jan. 2022.

[6] Y. Tian, H. Zhang, Y. Liu, and L. Wang, "Recovering 3D human mesh from monocular images: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 12, pp. 15406–15425, Dec. 2023.

[7] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–16, Oct. 2015.

[8] H. Choi, G. Moon, J. Y. Chang, and K. M. Lee, "Beyond static features for temporally consistent 3D human pose and shape from a video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 1964–1973.

[9] C. Doersch and A. Zisserman, "Sim2Real transfer learning for 3D human pose estimation: Motion to the rescue," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 12929–12941.

[10] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik, "Learning 3D human dynamics from video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5614–5623.

[11] M. Kocabas, N. Athanasiou, and M. J. Black, "VIBE: Video inference for human body pose and shape estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5253–5263.

[12] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.

[13] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[15] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 5693–5703.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, vol. 9351. Cham, Switzerland: Springer, 2015, pp. 234–241.

[17] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, 2016, pp. 483–499.

[18] T. Y. Lin, P. Dollàr, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.

[19] J. Zhang, D. Yu, J. H. Liew, X. Nie, and J. Feng, "Body meshes as points," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 546–556.

[20] M. Omran, C. Lassner, G. Pons-Moll, P. Gehler, and B. Schiele, "Neural body fitting: Unifying deep learning and model based human pose and shape estimation," in *Proc. Int. Conf. 3D Vis. (3DV)*, Sep. 2018, pp. 484–494.

[21] A. Zanfir, E. G. Bazavan, M. Zanfir, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "Neural descent for visual 3D human pose and shape," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14479–14488.

[22] M. Zanfir, A. Zanfir, E. G. Bazavan, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "THUNDR: Transformer-based 3D human reconstruction with markers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12951–12960.

[23] H. Zhang et al., "PyMAF: 3D human pose and shape regression with pyramidal mesh alignment feedback loop," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11446–11456.

[24] Y. Xu, S.-C. Zhu, and T. Tung, "DenseRaC: Joint 3D pose and shape estimation by dense render-and-compare," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 7759–7769.

[25] H. Zhang, J. Cao, G. Lu, W. Ouyang, and Z. Sun, "Learning 3D human shape and pose from dense body parts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2610–2627, May 2022.

[26] B. Huang, T. Zhang, and Y. Wang, "Pose2UV: Single-shot multiperson mesh recovery with deep UV prior," *IEEE Trans. Image Process.*, vol. 31, pp. 4679–4692, 2022.

[27] T. Alldieck, M. Kassubeck, B. Wandt, B. Rosenhahn, and M. Magnor, "Optical flow-based 3D human motion estimation from monocular video," in *Proc. German Conf. Pattern Recognit.* Basel, Switzerland: Springer, 2017, pp. 347–360.

[28] A. Arnab, C. Doersch, and A. Zisserman, "Exploiting temporal context for 3D human pose estimation in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3390–3399.

[29] Z. Li, B. Xu, H. Huang, C. Lu, and Y. Guo, "Deep two-stream video inference for human body pose and shape estimation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 637–646.

[30] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "SCAPE: Shape completion and animation of people," in *Proc. ACM SIGGRAPH Papers*, Jul. 2005, pp. 408–416.

[31] H. Xu, E. G. Bazavan, A. Zanfir, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "GHUM & GHUML: Generative 3D human shape and articulated pose models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6183–6192.

[32] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3D human sensing in natural environments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1325–1339, Dec. 2013.

[33] T. V. Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3D human pose in the wild using IMUs and a moving camera," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 601–617.

[34] D. Mehta et al., "Monocular 3D human pose estimation in the wild using improved CNN supervision," in *Proc. Int. Conf. 3D Vis. (3DV)*, Oct. 2017, pp. 506–516.

[35] L. Sigal, A. Balan, and M. Black, "Combined discriminative and generative articulated pose and non-rigid shape estimation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 20, 2007, pp. 1337–1344.

[36] P. Guan, A. Weiss, A. O. Bălan, and M. J. Black, "Estimating human shape and pose from a single image," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Oct. 2009, pp. 1381–1388.

[37] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 561–578.

[38] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7122–7131.

[39] N. Kolotouros, G. Pavlakos, and K. Daniilidis, "Convolutional mesh regression for single-image human shape reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4501–4510.

[40] P. Yao, Z. Fang, F. Wu, Y. Feng, and J. Li, "DenseBody: Directly regressing dense 3D human pose and shape from a single color image," 2019, *arXiv:1903.10153*.

[41] H. Zhang et al., "PyMAF-X: Towards well-aligned full-body model regression from monocular images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 10, pp. 12287–12303, Oct. 2023.

[42] Z. Li, J. Liu, Z. Zhang, S. Xu, and Y. Yan, "CLIFF: Carrying location information in full frames into human pose and shape estimation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 590–606.

[43] K. Lin, L. Wang, and Z. Liu, "Mesh graphormer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 12919–12928.

[44] N. Kolotouros, G. Pavlakos, D. Jayaraman, and K. Daniilidis, "Probabilistic modeling for human mesh recovery," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11585–11594.

[45] Y. Sun, Q. Bao, W. Liu, Y. Fu, M. J. Black, and T. Mei, "Monocular, one-stage, regression of multiple 3D people," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11159–11168.

[46] L. Wang, X. Liu, X. Ma, J. Wu, J. Cheng, and M. Zhou, "A progressive quadric graph convolutional network for 3D human mesh recovery," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 1, pp. 104–117, Jan. 2023.

[47] N. Kolotouros, G. Pavlakos, M. J. Black, and K. Daniilidis, "Learning to reconstruct 3D human pose and shape via model-fitting in the loop," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2252–2261.

[48] H. Joo, N. Neverova, and A. Vedaldi, "Exemplar fine-tuning for 3D human model fitting towards in-the-wild 3D human pose estimation," in *Proc. Int. Conf. 3D Vis. (3DV)*, Dec. 2021, pp. 42–52.

[49] Z. Wan, Z. Li, M. Tian, J. Liu, S. Yi, and H. Li, "Encoder–decoder with multi-level attention for 3D human shape and pose estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 13033–13042.

[50] Y. Sun, Y. Ye, W. Liu, W. Gao, Y. Fu, and T. Mei, "Human mesh recovery from monocular images via a skeleton-disentangled representation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5349–5358.

[51] W.-L. Wei, J.-C. Lin, T.-L. Liu, and H.-Y.-M. Liao, "Capturing humans in motion: Temporal-attentive 3D human pose and shape estimation from monocular video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 13211–13220.

[52] G. Papandreou et al., "Towards accurate multi-person pose estimation in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 4903–4911.

[53] T. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 740–755.

[54] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation," in *Proc. Brit. Mach. Vis. Conf.*, 2010, p. 5.

[55] S. Johnson and M. Everingham, "Learning effective human pose estimation from inaccurate annotation," in *Proc. CVPR*, Jun. 2011, pp. 1465–1472.

[56] G. Moon, H. Choi, and K. M. Lee, "NeuralAnnot: Neural annotator for 3D human mesh training sets," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2022, pp. 2298–2306.

[57] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 3686–3693.

[58] M. Andriluka et al., "PoseTrack: A benchmark for human pose estimation and tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5167–5176.

[59] M. Kocabas, C.-H. P. Huang, O. Hilliges, and M. J. Black, "PARE: Part attention regressor for 3D human body estimation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2021, pp. 11127–11137.

[60] G. Moon and K. M. Lee, "I2L-MeshNet: Image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 752–768.

[61] A. Paszke et al., "Automatic differentiation in PyTorch," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

Wei Yao received the B.E. degree from the University of South China, Hengyang, China, in 2021. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His research interests include computer vision, motion capture, and embodied intelligence.

Hongwen Zhang received the B.E. degree from the South China University of Technology, Guangzhou, China, in 2015, and the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2021. He has been working as a Post-Doctoral Researcher with Tsinghua University. He is currently an Associate Professor with the School of Artificial Intelligence, Beijing Normal University. His research interests include computer vision, computer graphics, and their applications in 3D human modeling.

Yunlian Sun received the M.E. degree in computer science and technology from Harbin Institute of Technology, China, in 2010, and the Ph.D. degree in ingegneria elettronica, informatica e delle telecomunicazioni from the University of Bologna, Italy, in 2014. After the Ph.D. degree, she worked as a Post-Doctoral Researcher with the National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences. She is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology, China. Her research interests include biometrics, pattern recognition, and computer vision.

Jinhui Tang (Senior Member, IEEE) received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2003 and 2008, respectively. He is currently a Professor with Nanjing University of Science and Technology, Nanjing, China. He has authored more than 200 papers in top-tier journals and conferences. His research interests include multimedia analysis and computer vision. He is a fellow of IAPR. He was a recipient of the Best Paper Awards in ACM MM 2007 and ACM MM Asia 2020 and the Best Paper Runner-Up in ACM MM 2015. He has served as an Associate Editor for IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.