Capturing Humans in Motion: Temporal-Attentive 3D Human Pose and Shape Estimation from Monocular Video

Wen-Li Wei*, Jen-Chun Lin*, Tyng-Luh Liu, and Hong-Yuan Mark Liao†
Institute of Information Science, Academia Sinica, Taiwan

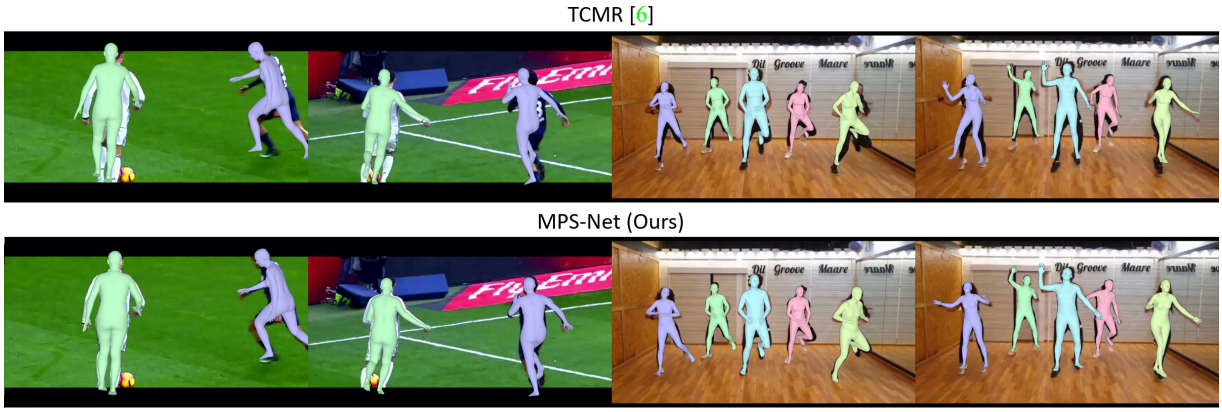

Figure 1. By coupling *motion continuity attention* with *hierarchical attentive feature integration*, the proposed MPS-Net can achieve more accurate pose and shape estimations (bottom row), when dealing with in-the-wild videos. For comparison, the results (top row) obtained by TCMR [6], the state-of-the-art video-based 3D human pose and shape estimation method, are included.

Abstract

Learning to capture human motion is essential to 3D human pose and shape estimation from monocular video. However, the existing methods mainly rely on recurrent or convolutional operation to model such temporal information, which limits the ability to capture non-local context relations of human motion. To address this problem, we propose a motion pose and shape network (MPS-Net) to effectively capture humans in motion to estimate accurate and temporally coherent 3D human pose and shape from a video. Specifically, we first propose a motion continuity attention (MoCA) module that leverages visual cues observed from human motion to adaptively recalibrate the range that needs attention in the sequence to better capture the motion continuity dependencies. Then, we develop a hierarchical attentive feature integration (HAFI) module to effectively combine adjacent past and future feature representations to strengthen temporal correlation and refine the feature representation of the current frame. By coupling the MoCA and HAFI modules, the proposed MPS-Net excels in estimating 3D human pose and shape in the video. Though conceptually simple, our MPS-Net not only outperforms the state-of-the-art methods on the 3DPW, MPI-INF-3DHP, and Human3.6M benchmark datasets, but also uses fewer network parameters. The video demos can be found at https://mps-net.github.io/MPS-Net/.

*Both authors contributed equally to this work
†Mark Liao is also a Chair Professor of Providence University

1. Introduction

Estimating 3D human pose and shape by taking a simple picture/video without relying on sophisticated 3D scanning devices or multi-view stereo algorithms, has important applications in computer graphics, AR/VR, physical therapy and beyond. Generally speaking, the task is to take a single image or video sequence as input and to estimate the parameters of a 3D human mesh model as output. Take, for example, the SMPL model [24]. For each image, it needs to estimate 85 (including pose, shape, and camera) parameters, which control the 6890 vertices that form the full 3D mesh of a human body [24]. Despite recent progress on 3D human pose and shape estimation, it is still a frontier challenge due to depth ambiguity, limited 3D annotations, and complex motion of non-rigid human body [6, 17, 20, 21].

Different from 3D human pose and shape estimation from a single image [11, 17, 21, 29, 31], estimating it from monocular video is a more complex task [6, 8, 18, 20, 25, 34]. It needs to not only estimate the pose, shape and camera parameters of each image, but also correlate the continuity of human motion in the sequence. Although existing single image-based methods can predict a reasonable output from a static image, it is difficult for them to estimate temporally coherent and smooth 3D human pose and shape in the video sequence due to the lack of modeling the continuity of human motion in consecutive frames. To solve this problem, several methods have recently been proposed to extend the single image-based methods to the video cases,

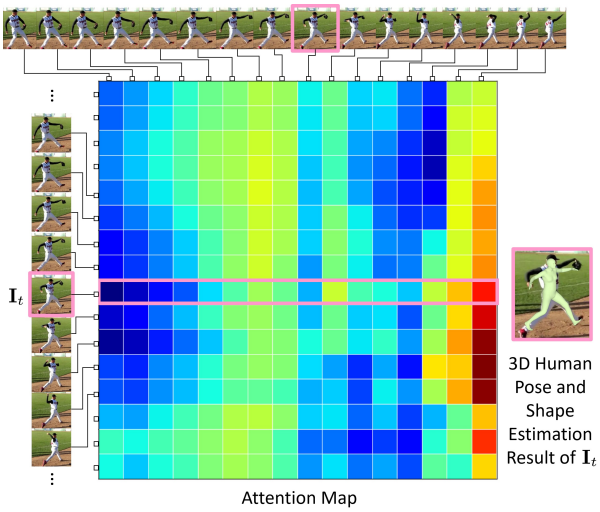

Figure 2. Visualization of the attention map generated by the self-attention module [38] in 3D human pose and shape estimation. The visualization shows that the attention map is easy to focus attention on less correlated temporal positions (*i.e.*, far apart frames with very different action poses) and lead to inaccurate 3D human pose and shape estimation (see frame $\mathbf{I}_t$). In the attention map, red indicates a higher attention value, and blue indicates a lower one.

which mainly rely on recurrent neural network (RNN) or convolutional neural network (CNN) to model temporal information (*i.e.*, continuity of human motion) for coherent predictions [6, 8, 18, 20, 25]. However, RNNs and CNNs are good at dealing with local neighborhoods [36, 38], and the models alone may not be effective for learning long-range dependencies (*i.e.*, non-local context relations) between feature representations to describe the relevance of human motion. As a result, there is still room for improvement for existing video-based methods to estimate accurate and smooth 3D human pose and shape (see Figure 1).

To address the aforementioned issue, we propose a motion pose and shape network (MPS-Net) for 3D human pose and shape estimation from monocular video. Our key insights are two-fold. First, although a self-attention mechanism [36, 38] has recently been proposed to compensate (*i.e.*, better learn long-range dependencies) for the weaknesses of recurrent and convolutional operations, we empirically find that it is not always good at modeling human motion in the action sequence. Because the attention map computed by the self-attention module is often unstable, which is easy to focus attention on less correlated temporal positions (*i.e.*, far apart frames with very different action poses) and ignore the continuity of human motion in the action sequence (see Figure 2). To this end, we propose a motion continuity attention (MoCA) module to achieve the adaptability to diverse temporal content and relations in the action sequence. Specifically, the MoCA module contributes in two points. First, a normalized self-similarity matrix (NSSM) is developed to capture the structure of temporal similarities and dissimilarities of visual representations in the action sequence, thereby revealing the continuity of human motion. Second, NSSM is regarded as the a *priori* knowledge and applied to guide the learning of the self-attention module, which allows it to adaptively recalibrate the range that needs attention in the sequence to capture the motion continuity dependencies. In the second insight, motivated by the temporal feature integration scheme in 3D human mesh estimation [6], we develop a hierarchical attentive feature integration (HAFI) module that utilizes adjacent feature representations observed from past and future frames to strengthen temporal correlation and refine the feature representation of the current frame. By coupling the MoCA and HAFI modules, our MPS-Net can effectively capture humans in motion to estimate accurate and temporally coherent 3D human pose and shape from monocular video (see Figure 1). We characterize the main contributions of our MPS-Net as follows:

- We propose a MoCA module that leverages visual cues observed from human motion to adaptively recalibrate the range that needs attention in the sequence to better capture the motion continuity dependencies.
- We develop a HAFI module that effectively combines adjacent past and future feature representations in a hierarchical attentive integration manner to strengthen temporal correlation and refine the feature representation of the current frame.
- Extensive experiments on three standard benchmark datasets demonstrate that our MPS-Net achieves the state-of-the-art performance against existing methods and uses fewer network parameters.

2. Related work

3D human pose and shape estimation from a single image. The existing single image-based 3D human pose and shape estimation methods are mainly based on parametric 3D human mesh models, such as SMPL [24], *i.e.*, trains a deep-net model to estimate pose, shape, and camera parameters from the input image, and then decodes them into a 3D mesh of the human body through the SMPL model. For example, Kanazawa *et al.* [17] proposed an end-to-end human mesh recovery (HMR) framework to regress SMPL parameters from a single RGB image. They employ 3D to 2D keypoint reprojection loss and adversarial training to alleviate the limited 3D annotation problem and make the output 3D human mesh anatomically reasonable. Pavlakos *et al.* [31] used 2D joint heatmaps and silhouette as cues to improve the accuracy of SMPL parameter estimation. Similarly, Omran *et al.* [29] used a semantic segmentation scheme to extract body part information as a cue to estimate the SMPL parameters. Kolotouros *et al.* [21] proposed a self-improving framework that integrates the SMPL parameter regressor and iterative fitting scheme to better estimate 3D human pose and shape. Zhang *et al.* [41] designed a

pyramidal mesh alignment feedback (PyMAF) loop in the deep SMPL parameter regressor to exploit multi-scale contexts for better mesh-image alignment of the reconstruction.

Several non-parametric 3D human mesh reconstruction methods [22, 28, 35] have been proposed. For example, Kolotouros *et al*. [22] proposed a graph CNN, which takes the 3D human mesh template and image embedding (extracted from ResNet-50 [13]) as input to directly regress the vertex coordinates of the 3D mesh. Moon and Lee [28] proposed an I2L-MeshNet, which uses a lixel-based 1D heatmap to directly localize the vertex coordinates of the 3D mesh in a fully convolutional manner.

Despite the above methods are effective for static images, they are difficult to generate temporally coherent and smooth 3D human pose and shape in the video sequence, *i.e*., jittery, unstable 3D human motion may occur [6, 20].

3D human pose and shape estimation from monocular video. Similar to the single image-based methods, the existing video-based 3D human pose and shape estimation methods are mainly based on the SMPL model. For example, Kanazawa *et al*. [18] proposed a convolution-based temporal encoder to learn human motion kinematics by further estimating SMPL parameters in adjacent past and future frames. Doersch *et al*. [8] trained their model on a sequence of 2D keypoint heatmaps and optical flow by combining CNN and long short-term memory (LSTM) network to demonstrate that considering pre-processed motion information can improve SMPL parameter estimation. Sun *et al*. [34] proposed a skeleton-disentangling framework, which divides the task into multi-level spatial and temporal subproblems. They further proposed an unsupervised adversarial training strategy, namely temporal shuffles and order recovery, to encourage temporal feature learning. Kocabas *et al*. [20] proposed a temporal encoder composed of bidirectional gated recurrent units (GRU) to encode static features into a series of temporally correlated latent features, and feed them to the regressor to estimate SMPL parameters. They further integrated adversarial training strategy that leverages the AMASS dataset [26] to distinguish between real human motion and those estimated by its regressor to encourage the generation of reasonable 3D human motion. Luo *et al*. [25] proposed a two-stage model that first estimates the coarse 3D human motion through a variational motion estimator, and then uses a motion residual regressor to refine the motion estimates. Recently, Choi *et al*. [6] proposed a temporally consistent mesh recovery (TCMR) system that uses GRU-based temporal encoders with three different encoding strategies to encourage the network to better learn temporal features. In addition, they proposed a temporal feature integration scheme that combines the output of three temporal encoders to help the SMPL parameter regressor estimate accurate and smooth 3D human pose and shape.

Despite the success of RNNs and CNNs, both recurrent and convolutional operations can only deal with local neighborhoods [36, 38], which makes it difficult for them to learn long-range dependencies (*i.e*., non-local context relations) between feature representations in the action sequence. Therefore, existing methods are still struggling to estimate accurate and smooth 3D human pose and shape.

Attention mechanism. The attention mechanism has enjoyed widespread adoption as a computational module for natural language processing [2, 7, 32, 36, 40] and vision-related tasks [5, 9, 14, 15, 33, 38, 39] because of its ability to capture long-range dependencies and selectively concentrate on the relevant subset of the input. There are various ways to implement the attention mechanism. Here we focus on self-attention [36, 38]. For example, Vaswani *et al*. [36] proposed a self-attention-based architecture called *Transformer*, in which the self-attention module is designed to update each sentence's element through the entire sentence's aggregated information to draw global dependencies between input and output. The *Transformer* entirely replaces the recurrent operation with the self-attention module, and greatly improves the performance of machine translation. Later, Wang *et al*. [38] showed that self-attention is an instantiation of non-local mean [3], and proposed a non-local block for the CNN to capture long-range dependencies. Like the self-attention module proposed in *Transformer*, the non-local operation computes the correlation between each position in the input feature representation to generate an attention map, and then performs the attention-guided dense context information aggregation to draw long-range dependencies.

Despite the self-attention mechanism performs well, we empirically find that the attention map computed by the self-attention module (*e.g*., non-local block) is often unstable, which means that it is easy to focus attention on less correlated temporal positions (*i.e*., far apart frames with very different action poses) and ignore the continuity of human motion in the action sequence (see Figure 2). In this work, we propose the MoCA module, which extends the learning of the self-attention module by introducing the a *priori* knowledge of NSSM to adaptively recalibrate the range that needs attention in the sequence, so as to capture motion continuity dependencies. The HAFI module is further proposed to strengthen the temporal correlation and refine the feature representation of each frame through its neighbors.

3. Method

Figure 3 shows the overall pipeline of our MPS-Net. We elaborate each module in MPS-Net as follows.

3.1. Temporal encoder

Given an input video sequence $\mathbf{V} = \{\mathbf{I}_t\}_{t=1}^{T}$ with T frames. We first use ResNet-50 [13] pre-trained by Kolotouros *et al*. [21] to extract the static feature of each frame to

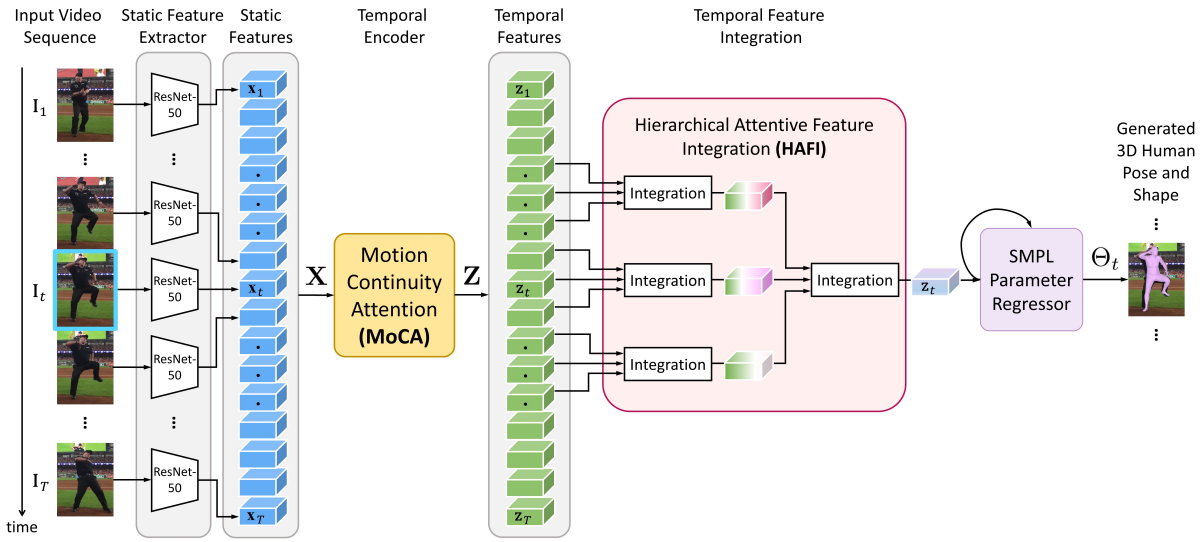

Figure 3. Overview of our motion pose and shape network (MPS-Net). MPS-Net estimates pose, shape, and camera parameters Θ in the video sequence based on the static feature extractor, temporal encoder, temporal feature integration, and SMPL parameter regressor to generate 3D human pose and shape.

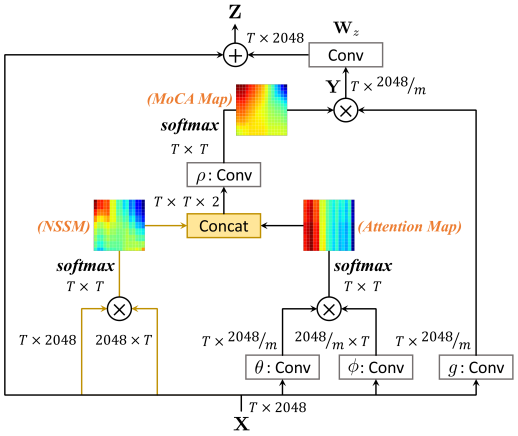

Figure 4. A MoCA module. $\mathbf{X}$ is shown as the shape of $T \times 2048$ for 2048 channels. g, ϕ, θ, and ρ denote convolutional operations, $\otimes$ denotes matrix multiplication, and $\oplus$ denotes element-wise sum. The computation of softmax is performed on each row.

form a static feature representation sequence $\mathbf{X} = \{\mathbf{x}_t\}_{t=1}^{T}$, where $\mathbf{x}_t \in \mathbb{R}^{2048}$. Then, the extracted $\mathbf{X}$ is sent to the proposed MoCA module to calculate the temporal feature representation sequence $\mathbf{Z} = \{\mathbf{z}_t\}_{t=1}^{T}$, where $\mathbf{z}_t \in \mathbb{R}^{2048}$.

MoCA Module. We propose a MoCA operation to extend the non-local operation [38] in two ways. First, we introduce an NSSM to capture the structure of temporal similarities and dissimilarities of visual representations in the action sequence to reveal the continuity of human motion. Second, we regard NSSM as the a *priori* knowledge and combine it with the attention map generated by the non-local operation to adaptively recalibrate the range that needs attention in the action sequence.

We formulate the proposed MoCA module as follows (see Figure 4). Given the static feature representation sequence $\mathbf{X} \in \mathbb{R}^{T\times 2048}$, the goal of the MoCA operation is to obtain a non-local context response $\mathbf{Y} \in \mathbb{R}^{T\times \frac{2048}{m}}$, which aims to capture the *motion continuity dependencies* across the whole representation sequence by weighted sum of the static features at all temporal positions,

$$\mathbf{Y} = \rho([f(\mathbf{X}, \mathbf{X}), f(\theta(\mathbf{X}), \phi(\mathbf{X}))])g(\mathbf{X}), \tag{1}$$

where m is a reduction ratio used to reduce computational complexity [38], and it is set to 2 in our experiments. $g(\cdot)$, $\phi(\cdot)$, and $\theta(\cdot)$ are learnable transformations, which are implemented by using the convolutional operation [38]. Thus, the transformations can be written as

$$g(\mathbf{X}) = \mathbf{X}\mathbf{W}_g \in \mathbb{R}^{T\times \frac{2048}{m}}, \tag{2}$$

$$\phi(\mathbf{X}) = \mathbf{X}\mathbf{W}_\phi \in \mathbb{R}^{T\times \frac{2048}{m}}, \tag{3}$$

and

$$\theta(\mathbf{X}) = \mathbf{X}\mathbf{W}_\theta \in \mathbb{R}^{T\times \frac{2048}{m}}, \tag{4}$$

parameterized by the weight matrices $\mathbf{W}_g$, $\mathbf{W}_\phi$, and $\mathbf{W}_\theta \in \mathbb{R}^{2048\times \frac{2048}{m}}$, respectively. $f(\cdot,\cdot)$ represents a pairwise function, which computes the affinity between all positions. We use dot product [38] as the operation for f, *i.e.*,

$$f(\theta(\mathbf{X}), \phi(\mathbf{X})) = \theta(\mathbf{X})\phi(\mathbf{X})^{\mathsf{T}}, \tag{5}$$

where the size of the resulting pairwise function $f(\theta(\mathbf{X}), \phi(\mathbf{X}))$ is denoted as $\mathbb{R}^{T\times \frac{2048}{m}} \times \mathbb{R}^{\frac{2048}{m}\times T} \rightarrow \mathbb{R}^{T\times T}$, which encodes the mutual similarity between temporal positions under the transformed static feature representation sequence. Then, the softmax operation is used to normalize it into an attention map (see Figure 4).

We empirically find that although calculating the similarity in the transformed feature space provides an opportunity for insight into implicit long-range dependencies, it may sometimes be unstable and lead to attention on less

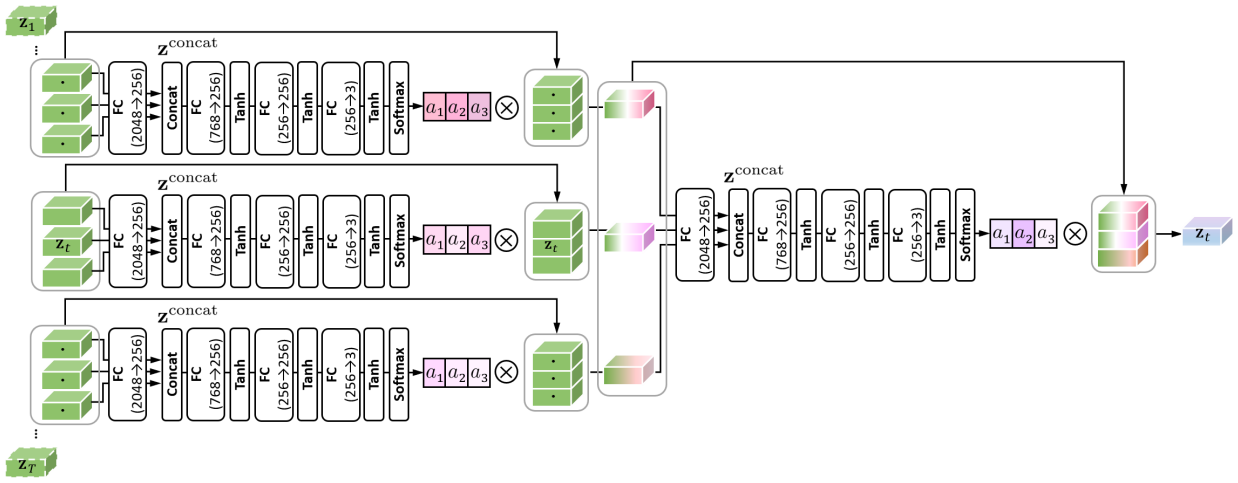

Figure 5. A HAFI module. It utilizes the temporal features observed from the past and future frames to refine the temporal feature of the current frame $\mathbf{z}_t$ in a hierarchical attentive integration manner. Where $\otimes$ denotes matrix multiplication.

correlated temporal positions (see Figure 2). To this end, we introduce NSSM into the MoCA operation to enable the MoCA module to learn to focus attention on a more appropriate range of action sequence.

Regarding NSSM construction, unlike the non-local operation [38], we directly use the static feature representation sequence $\mathbf{X}$ extracted from the input video to reveal the explicit dependencies between the frames through the self-similarity matrix [10] construction $f(\mathbf{X}, \mathbf{X}) = \mathbf{X}\mathbf{X}^{\mathsf{T}} \in \mathbb{R}^{T\times T}$. In this way, the continuity of human motion in the input video can be more straightforwardly revealed. Similarly, we normalize the resultant self-similarity matrix through the softmax operation to form an NSSM (see Figure 4) to facilitate subsequent combination with the attention map.

For the combination of NSSM and attention map, we first regard NSSM as the a *priori* knowledge to concatenate the attention map through the operation $[\cdot, \cdot]$, and then use the learnable transformation $\rho(\cdot)$, *i.e.*, 1×1 convolution to recalibrate the attention map by referring to NSSM (see Figure 4 and Eq. (1)). The resultant $\rho(\cdot)$ is then normalized through the softmax operation, which is called the MoCA map. By jointly considering the characteristics of the NSSM and the attention map, the MoCA map can reveal the non-local context relations related to the human motion of the input video in a more appropriate range. To this end, the non-local context response $\mathbf{Y} \in \mathbb{R}^{T\times \frac{2048}{m}}$ can be calculated from the linear combination between the matrices resulted from $\rho(\cdot)$ and $g(\cdot)$.

Finally, as in the design of the non-local block [38], we use residual connection [13] to generate the output temporal feature representation sequence $\mathbf{Z} \in \mathbb{R}^{T\times 2048}$ (see Figure 4) in the MoCA module as follows:

$$\mathbf{Z} = \mathbf{Y}\mathbf{W}_z + \mathbf{X}, \tag{6}$$

where $\mathbf{W}_z$ is a learnable weight matrix implemented by using the convolutional operation [38], and the number of channels in $\mathbf{W}_z$ is scaled up to match the number of channels (*i.e.*, 2048) in $\mathbf{X}$. "$+\mathbf{X}$" denotes a residual connection. The residual connection allows us to insert the MoCA module into any pre-trained network, without breaking its initial behavior (*e.g.*, if $\mathbf{W}_z$ is initialized as zero). As a result, by further considering the non-local context response $\mathbf{Y}$, $\mathbf{Z}$ will contain rich temporal information, so $\mathbf{Z}$ can be regarded as enhanced $\mathbf{X}$.

3.2. Temporal feature integration

Given the temporal feature representation sequence $\mathbf{Z} \in \mathbb{R}^{T\times 2048}$, the goal of the HAFI module is to refine the temporal feature of the current frame $\mathbf{z}_t$ by integrating the adjacent temporal features observed from past and future frames to strengthen their temporal correlation and obtain better pose and shape estimation, as shown in Figure 3.

HAFI Module. Specifically, we use $T/2$ adjacent frames (*i.e.*, $\{\mathbf{z}_{t\pm\frac{T}{4}}\}$) to refine the temporal feature of the current frame $\mathbf{z}_t$ in a hierarchical attentive integration manner, as shown in Figure 5. For each branch in the HAFI module, we consider the temporal features of three adjacent frames as a group (adjacent frames between groups do not overlap), and resize them from 2048 dimensions to 256 dimensions respectively through a shared fully connected (FC) layer to reduce computational complexity. The resized temporal features are concatenated ($\mathbf{z}^{\text{concat}} \in \mathbb{R}^{768}$) and passed to three FC layers and a softmax activation to calculate the attention values $\mathbf{a} = \{a_k\}_{k=1}^{3}$ by exploring the dependencies among them. Then, the attention value is weighted back to each corresponding frame to amplify the contribution of important frames in the temporal feature integration to obtain the aggregated temporal feature (see Figure 5). The aggregated temporal features produced by the bottom branches will be passed to the upper layer and integrated in the same way to produce the final refined $\mathbf{z}_t$. By gradually integrating temporal features in adjacent frames to strengthen temporal

correlation, it will provide opportunities for the SMPL parameter regressor to learn to estimate accurate and temporally coherent 3D human pose and shape.

In this work, like Kocabas *et al*. [20], we use the SMPL parameter regressor proposed in [17, 21] as our regressor to estimate pose, shape, and camera parameters $\Theta_t \in \mathbb{R}^{85}$ according to each refined $\mathbf{z}_t$ (see Figure 3). In the training phase, we initialize the SMPL parameter regressor with pre-trained weights from HMR [17, 21].

3.3. Loss functions

In terms of MPS-Net training, for each estimated Θ_t, following the method proposed by Kocabas *et al*. [20], we impose $\mathcal{L}_2$ loss between the estimated and ground-truth SMPL parameters and 3D/2D joint coordinates to supervise MPS-Net to generate reasonable real-world poses. The 3D joint coordinates are obtained by forwarding the estimated SMPL parameters to the SMPL model [24], and the 2D joint coordinates are obtained through the 2D projection of the 3D joints using the predicted camera parameters [20]. In addition, like Kocabas *et al*. [20], we also apply adversarial loss $\mathcal{L}_{\text{adv}}$, *i.e*., using the AMASS [26] dataset to train a discriminator to distinguish between real human motion and those generated by MPS-Net's SMPL parameter regressor to encourage the generation of reasonable 3D human motion.

4. Implementation details

Following the previous works [6, 20], we set $T = 16$ as the sequence length. We use ResNet-50 [13] pre-trained by Kolotouros *et al*. [21] to serve as our static feature extractor. The static feature extractor is fixed and outputs a 2048-dimensional feature for each frame, *i.e*., $\mathbf{x}_t \in \mathbb{R}^{2048}$. The SMPL parameter regressor has two FC layers, each with 1024 neurons, and followed an output layer to output 85 pose, shape, and camera parameters Θ_t for each frame [17, 21]. The discriminator architecture we use is the same as [20]. The parameters of MPS-Net and discriminator are optimized by the Adam solver [19] at a learning rate of 5×10^{-5} and 1×10^{-4}, respectively. The mini-batch size is set to 32. During training, if the performance does not improve within 5 epochs, the learning rate of both the MPS-Net and the discriminator will be reduced by a factor of 10. We use an NVIDIA Titan RTX GPU to train the entire network for 30 epochs. PyTorch [30] is used for code implementation.

5. Experiments

We first illustrate the datasets used for training and evaluation and the evaluation metrics. Then, we compare our MPS-Net against other state-of-the-art video-based methods and single image-based methods to demonstrate its advantages in addressing 3D human pose and shape estimation. We also provide an ablation study to confirm the effectiveness of each module in MPS-Net. Finally, we visualize some examples to show the qualitative evaluation results.

Datasets. Following the previous works [6, 20], we adopt batches of mixed 3D and 2D datasets for training. For 3D datasets, we use 3DPW [37], MPI-INF-3DHP [27], Human3.6M [16], and AMASS [26] for training, where 3DPW and AMASS provide SMPL parameter annotations, while MPI-INF-3DHP and Human3.6M include 3D joint annotations. For 2D datasets, we use PoseTrack [1] and InstaVariety [18] for training, where PoseTrack provides ground-truth 2D joints, while InstaVariety includes pseudo ground-truth 2D joints annotated using a 2D keypoint detector [4]. In terms of evaluation, the 3DPW, MPI-INF-3DHP, and Human3.6M datasets are used. Among them, Human3.6M is an indoor dataset, while 3DPW and MPI-INF-3DHP contain challenging outdoor videos. More detailed settings are in the supplementary material.

Evaluation metrics. For the evaluation, four standard metrics are used [6,20,25], including the mean per joint position error (MPJPE), the Procrustes-aligned mean per joint position error (PA-MPJPE), the mean per vertex position error (MPVPE), and the acceleration error (ACC-ERR). Among them, MPJPE, PA-MPJPE, and MPVPE are mainly used to express the accuracy of the estimated 3D human pose and shape (measured in millimeter (mm)), and ACC-ERR (mm/s^2) is used to express the smoothness and temporal coherence of 3D human motion. A detailed description of each metric is included in the supplementary material.

5.1. Comparison with state-of-the-art methods

Video-based methods. Table 1 shows the performance comparison between our MPS-Net and the state-of-the-art video-based methods on the 3DPW, MPI-INF-3DHP, and Human3.6M datasets. Following TCMR [6], all methods are trained on the training set including 3DPW, but do not use the Human3.6M SMPL parameters obtained from Mosh [23] for supervision. Because the SMPL parameters from Mosh have been removed from public access due to legal issues [25]. The values of the comparison method are from TCMR [6], but we validated them independently.

The results in Table 1 show that our MPS-Net outperforms the existing video-based methods in almost all metrics and datasets. This demonstrates that by capturing the motion continuity dependencies and integrating temporal features from adjacent past and future, performance can indeed be improved. Although TCMR [6] has also made great progress, it is limited by the ability of recurrent operation (*i.e*., GRU) to capture non-local context relations in the action sequence [36, 38], thereby reducing the accuracy of the estimated 3D human pose and shape (*i.e*., PA-MPJPE, MPJPE, and MPVPE are higher than MPS-Net). In addition, the number of network parameters and model size of TCMR are also about 3 times that of MPS-Net (see Table 2), which is relatively heavy. Regarding MEVA [25], as shown in Table 1, MEVA requires at least 90 input frames, which

Method	3DPW				MPI-INF-3DHP			Human3.6M			Number of
	PA-MPJPE ↓	MPJPE ↓	MPVPE ↓	ACC-ERR ↓	PA-MPJPE ↓	MPJPE ↓	ACC-ERR ↓	PA-MPJPE ↓	MPJPE ↓	ACC-ERR ↓	Input Frames
VIBE [20]	57.6	91.9	-	25.4	68.9	103.9	27.3	53.3	78.0	27.3	**16**
MEVA [25]	54.7	86.9	-	11.6	65.4	**96.4**	11.1	53.2	76.0	15.3	90
TCMR [6]	52.7	86.5	103.2	**6.8**	63.5	97.6	**8.5**	52.0	73.6	3.9	**16**
MPS-Net (Ours)	**52.1**	**84.3**	**99.7**	7.4	**62.8**	96.7	9.6	**47.4**	**69.4**	**3.6**	**16**

Table 1. Evaluation of state-of-the-art video-based methods on 3DPW [37], MPI-INF-3DHP [27], and Human3.6M [16] datasets. Following Choi *et al*. [6], all methods are trained on the training set including 3DPW, but do not use the Human3.6M SMPL parameters obtained from Mosh [23]. The number of input frames follows the original protocol of each method.

	#Parameters (M)	FLOPs (G)	Model Size (MB)
VIBE [20]	72.43	**4.17**	776
MEVA [25]	85.72	4.46	858.8
TCMR [6]	108.89	4.99	1073
MPS-Net (Ours)	**39.63**	4.45	**331**

Table 2. Comparison of the number of network parameters, FLOPs, and model size.

Method	3DPW			
	PA-MPJPE ↓	MPJPE ↓	MPVPE ↓	ACC-ERR ↓
MPS-Net - only Non-local [38]	54.1	87.6	103.1	24.1
MPS-Net - only MoCA	53.0	86.7	102.2	23.5
MPS-Net - MoCA + TF-intgr. [6]	52.4	86.0	101.5	10.5
MPS-Net (Ours) - MoCA + HAFI	**52.1**	**84.3**	**99.7**	**7.4**

Table 3. Ablation study for different modules of the MPS-Net on the 3DPW [37] dataset. The training and evaluation settings are the same as the experiments on the 3DPW dataset in Table 1.

	Method	3DPW			
		PA-MPJPE ↓	MPJPE ↓	MPVPE ↓	ACC-ERR ↓
single image -based	HMR [17]	76.7	130.0	-	37.4
	GraphCMR [22]	70.2	-	-	-
	SPIN [21]	59.2	96.9	116.4	29.8
	PyMAF [41]	58.9	92.8	110.1	-
	I2L-MeshNet [28]	57.7	93.2	110.1	30.9
video-based	HMMR [18]	72.6	116.5	139.3	15.2
	Doersch *et al*. [8]	74.7	-	-	-
	Sun *et al*. [34]	69.5	-	-	-
	VIBE [20]	56.5	93.5	113.4	27.1
	TCMR [6]	55.8	95.0	111.3	**6.7**
	MPS-Net (Ours)	**54.0**	**91.6**	**109.6**	7.5

Table 4. Evaluation of state-of-the-art single image-based and video-based methods on the 3DPW [37] dataset. All methods do not use 3DPW for training.

means it cannot be trained and tested on short videos. This greatly reduces the value in practical applications. Overall, our MPS-Net can effectively estimate accurate (lower PA-MPJPE, MPJPE, and MPVPE) and smooth (lower ACC-ERR) 3D human pose and shape from a video, and is relatively lightweight (fewer network parameters). The comparisons on the three datasets also show the strong generalization property of our MPS-Net.

Ablation analysis. To analyze the effectiveness of the MoCA and HAFI modules in MPS-Net, we conduct ablation studies on MPS-Net under the challenging in-the-wild 3DPW dataset. Specifically, we evaluate the impact on MPS-Net by replacing the MoCA module with the non-local block [38], considering only the MoCA module (without using HAFI), and replacing the HAFI module with the temporal feature integration scheme proposed by Choi *et al*. [6]. For performance comparison, it is obvious from Table 3 that the proposed MoCA module (*i.e*., MPS-Net-only MoCA) is superior to non-local block (*i.e*., MPS-Net-only Non-local) in all metrics. The results confirm that by further introducing the a *priori* knowledge of NSSM to guide self-attention learning, the MoCA module can indeed improve 3D human pose and shape estimation. On the other hand, the results also show that our HAFI module (*i.e*., MPS-Net-MoCA+HAFI) outperforms the temporal feature integration scheme (*i.e*., MPS-Net-MoCA+TF-intgr.), which demonstrates that the gradual integration of adjacent features through a hierarchical attentive integration manner can indeed strengthen temporal correlation and make the generated 3D human motion smoother (*i.e*., lower ACC-ERR). Overall, the ablation analysis confirmed the effectiveness of the proposed MoCA and HAFI modules.

Single image-based and video-based methods. We further compare our MPS-Net with the methods including single image-based methods on the challenging in-the-wild 3DPW dataset. Notice that a number of previous works [6,8,17,18,20–22,28,34,41] did not use the 3DPW training set to train their models, so in the comparison in Table 4, all methods are not trained on 3DPW.

Similar to the results in Table 1, the results in Table 4 demonstrate that our MPS-Net performs favorably against existing single image-based and video-based methods on the PA-MPJPE, MPJPE, and MPVPE evaluation metrics. Although TCMR achieves the lowest ACC-ERR, it tends to be overly smooth, thereby sacrificing the accuracy of pose and shape estimation. Specifically, when TCMR reduces ACC-ERR 0.8 mm/s^2 compared to MPS-Net, MPS-Net reduces PA-MPJPE, MPJPE, and MPVPE by 1.8 mm, 3.4 mm, and 1.7 mm, respectively. Table 4 further confirms the importance of considering temporal information in consec-

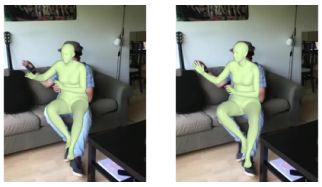

Figure 6. Qualitative comparison of TCMR [6] (left) and our MPS-Net (right) on the challenging in-the-wild 3DPW [37] dataset (the 1st and 2nd clips) and MPI-INF-3DHP [27] dataset (the 3rd clip). *This is an embedded video, please refer to our arxiv paper to view the video.*

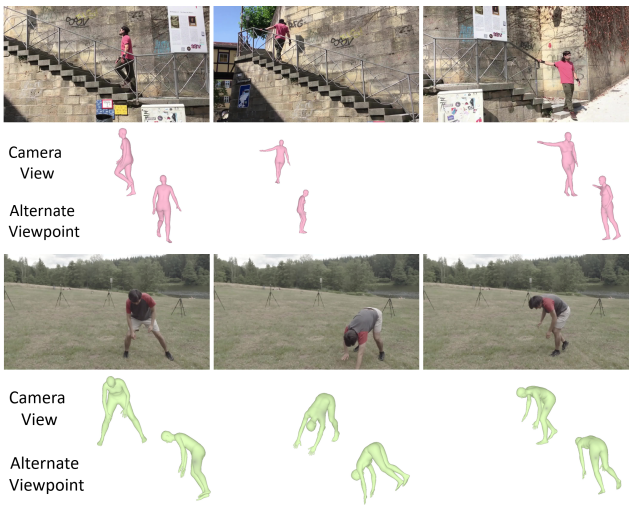

Figure 7. Qualitative results of MPS-Net on the challenging in-the-wild 3DPW [37] dataset and MPI-INF-3DHP [27] dataset. For each sequence, the top row shows input images, the middle row shows the estimated body mesh from the camera view, and the bottom row shows the estimated mesh from an alternate viewpoint.

utive frames, *i.e.*, compared with single-image-based methods, video-based methods have lower ACC-ERR. In summary, MPS-Net achieves a better balance in the accuracy and smoothness of 3D human pose and shape estimation.

5.2. Qualitative evaluation

We present 1) visual comparisons with the TCMR [6], 2) visual effects of MPS-Net in alternative viewpoints, and 3) visual results of the learned human motion continuity.

Visual comparisons with the TCMR. The qualitative comparison between TCMR and our MPS-Net on the 3DPW and MPI-INF-3DHP datasets is shown in Figure 6. From the results, we observe that the 3D human pose and shape estimated by MPS-Net can fit the input images well, especially on the limbs. TCMR seems to be too focused on generating smooth 3D human motion, so the estimated pose has relatively small changes from frame to frame, which limits its ability to fit the input images.

Visual effects of MPS-Net in alternative viewpoints. We visualize the 3D human body estimated by MPS-Net from different viewpoints in Figure 7. The results show that MPS-Net can estimate the correct global body rotation. This is quantitatively demonstrated by the improvements in the PA-MPJPE, MPJPE, and MPVPE (see Table 1).

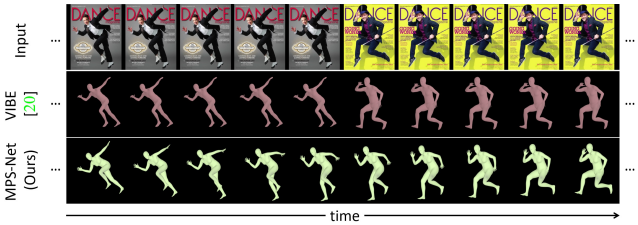

Figure 8. An example of visualization of the VIBE [20] and our MPS-Net on the continuity of human motion.

Visual results of the learned human motion continuity. We use a relatively extreme example to show the continuity of human motion learned by MPS-Net. In this example, we randomly downloaded two pictures with different poses from the Internet, and copied the pictures multiple times to form a sequence. Then, we send the sequence to VIBE [20] and MPS-Net for 3D human pose and shape estimation. As shown in Figure 8, compared with VIBE, it is obvious from the estimation results that our MPS-Net produces a transition effect between pose exchanges, and this transition conforms to the continuity of human kinematics. It demonstrates that MPS-Net has indeed learned the continuity of human motion, and explains why MPS-Net can achieve lower ACC-ERR in the benchmark (action) datasets (see Table 1). This result is also similar to using a 3D motion predictor to estimate reasonable human motion in-betweening of two key frames [12]. In contrast, VIBE relies too much on the features of the current frame, making it unable to truly learn the continuity of human motion. Thus, its ACC-ERR is still high (see Table 1).

For more results and video demos can be found at https://mps-net.github.io/MPS-Net/.

6. Conclusion

We propose the MPS-Net for estimating 3D human pose and shape from monocular video. The main contributions of this work lie in the design of the MoCA and HAFI modules. The former leverages visual cues observed from human motion to adaptively recalibrate the range that needs attention in the sequence to capture the motion continuity dependencies, and the later allows our model to strengthen temporal correlation and refine feature representation for producing temporally coherent estimates. Compared with existing methods, the integration of MoCA and HAFI modules demonstrates the advantages of our MPS-Net in achieving the state-of-the-art 3D human pose and shape estimation.

Acknowledgment: This work was supported in part by MOST under grants 110-2221-E-001-016-MY3, 110-2634-F-007-027 and 110-2634-F-002-050, and Academia Sinica under grant AS-TP-111-M02.

References

[1] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. PoseTrack: A benchmark for human pose estimation and tracking. In *CVPR*, 2018. 6

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015. 3

[3] Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. A non-local algorithm for image denoising. In *CVPR*, 2005. 3

[4] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. *CVPR*, 2017. 6

[5] Ding-Jie Chen, He-Yen Hsieh, and Tyng-Luh Liu. Adaptive image transformer for one-shot object detection. In *CVPR*, 2021. 3

[6] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3D human pose and shape from a video. In *CVPR*, 2021. 1, 2, 3, 6, 7, 8

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019. 3

[8] Carl Doersch and Andrew Zisserman. Sim2real transfer learning for 3D human pose estimation: Motion to the rescue. In *NeurIPS*, 2019. 1, 2, 3, 7

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

[10] Jonathan Foote. Visualizing music and audio using self-similarity. In *ACM Multimedia*, 1999. 5

[11] Georgios Georgakis, Ren Li, Srikrishna Karanam, Terrence Chen, Jana Košecká, and Ziyan Wu. Hierarchical kinematic human mesh recovery. In *ECCV*, 2020. 1

[12] Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM ToG*, 39(4):60:1–60:12, 2020. 8

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3, 5, 6

[14] Ting-I Hsieh, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. One-shot object detection with co-attention and co-excitation. In *NeurIPS*, 2019. 3

[15] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks. *IEEE TPAMI*, 42(8):2011–2023, 2020. 3

[16] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE TPAMI*, 36(7):1325–1339, 2014. 6, 7

[17] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 1, 2, 6, 7

[18] Angjoo Kanazawa, Jason Y. Zhang, Panna Felsen, and Jitendra Malik. Learning 3D human dynamics from video. In *CVPR*, 2019. 1, 2, 3, 6, 7

[19] Diederik P. Kingma and Jimmy Lei Ba. Adam: a method for stochastic optimization. In *ICLR*, 2015. 6

[20] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. VIBE: Video inference for human body pose and shape estimation. In *CVPR*, 2020. 1, 2, 3, 6, 7, 8

[21] Nikos Kolotouros, G. Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 1, 2, 3, 6, 7

[22] Nikos Kolotouros, G. Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019. 3, 7

[23] Matthew Loper, Naureen Mahmood, and Michael J. Black. MoSh: Motion and shape capture from sparse markers. *ACM ToG*, 33(6):220:1–220:13, 2014. 6, 7

[24] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: a skinned multi-person linear model. *ACM ToG*, 34(6):248:1–248:16, 2015. 1, 2, 6

[25] Zhengyi Luo, S. Alireza Golestaneh, and Kris M. Kitani. 3D human motion estimation via motion compression and refinement. In *ACCV*, 2020. 1, 2, 3, 6, 7

[26] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *ICCV*, 2019. 3, 6

[27] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3D human pose estimation in the wild using improved CNN supervision. In *3DV*, 2017. 6, 7, 8

[28] Gyeongsik Moon and Kyoung Mu Lee. I2L-MeshNet: Image-to-lixel prediction network for accurate 3D human pose and mesh estimation from a single RGB image. In *ECCV*, 2020. 3, 7

[29] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter V. Gehler, and Bernt Schiele. Neural Body Fitting: Unifying deep learning and model-based human pose and shape estimation. In *3DV*, 2018. 1, 2

[30] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NeurIPS Workshop on Autodiff*, 2017. 6

[31] G. Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3D human pose and shape from a single color image. In *CVPR*, 2018. 1, 2

[32] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. Technical report, OpenAI, 2018. 3

[33] Abhijit Guha Roy, Nassir Navab, and Christian Wachinger. Recalibrating fully convolutional networks with spatial and channel 'squeeze & excitation' blocks. *IEEE T-MI*, 38(2):540–549, 2019. 3

[34] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, Yili Fu, and Tao Mei. Human mesh recovery from monocular images via a skeleton-disentangled representation. In *ICCV*, 2019. 1, 3, 7

[35] Gül Varol, Duygu Ceylan, Bryan Russell, Jimei Yang, Ersin Yumer, Ivan Laptev, and Cordelia Schmid. BodyNet: Volumetric inference of 3D human body shapes. In *ECCV*, 2018. 3

[36] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2, 3, 6

[37] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3D human pose in the wild using IMUs and a moving camera. In *ECCV*, 2018. 6, 7, 8

[38] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 2, 3, 4, 5, 6, 7

[39] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional block attention module. In *ECCV*, 2018. 3

[40] Adams Wei Yu, David Dohan, Minh-Thang Luong, R. Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. QANet: Combining local convolution with global self-attention for reading comprehension. In *ICLR*, 2018. 3

[41] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. PyMAF: 3D human pose and shape regression with pyramidal mesh alignment feedback loop. In *ICCV*, 2021. 2, 7