Encoder-decoder with Multi-level Attention for 3D Human Shape and Pose Estimation

Ziniu Wan[1*] Zhengjia Li[2*] Maoqing Tian[3] Jianbo Liu[4] Shuai Yi[3] Hongsheng Li[4]
[1] Carnegie Mellon University [2] Tongji University
[3] SenseTime Research [4] Chinese University of Hong Kong
ziniuwan@andrew.cmu.edu zjli1997@tongji.edu.cn tianmaoqing@sensetime.com
liujianbo@link.cuhk.edu.hk yishuai@sensetime.com hsli@ee.cuhk.edu.hk

Abstract

3D human shape and pose estimation is the essential task for human motion analysis, which is widely used in many 3D applications. However, existing methods cannot simultaneously capture the relations at multiple levels, including spatial-temporal level and human joint level. Therefore they fail to make accurate predictions in some hard scenarios when there is cluttered background, occlusion, or extreme pose. To this end, we propose Multi-level Attention Encoder-Decoder Network (MAED), including a Spatial-Temporal Encoder (STE) and a Kinematic Topology Decoder (KTD) to model multi-level attentions in a unified framework. STE consists of a series of cascaded blocks based on Multi-Head Self-Attention, and each block uses two parallel branches to learn spatial and temporal attention respectively. Meanwhile, KTD aims at modeling the joint level attention. It regards pose estimation as a top-down hierarchical process similar to SMPL kinematic tree. With the training set of 3DPW, MAED outperforms previous state-of-the-art methods by 6.2, 7.2, and 2.4 mm of PA-MPJPE on the three widely used benchmarks 3DPW, MPI-INF-3DHP, and Human3.6M respectively. Our code is available at https://github.com/ziniuwan/maed.

1. Introduction

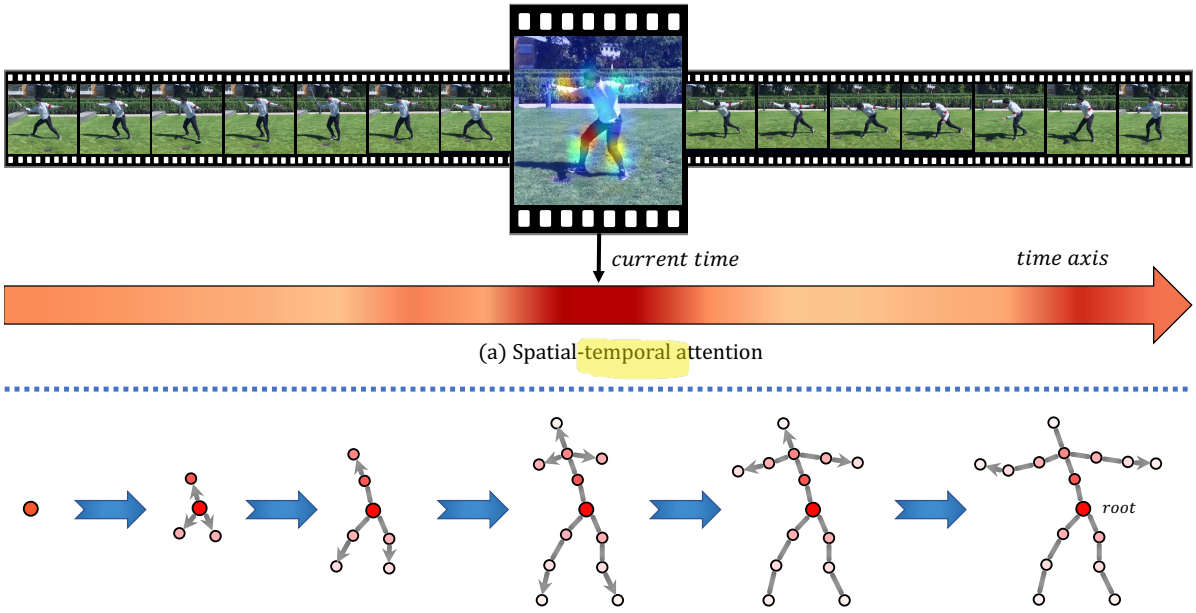
3D human shape and pose estimation from a single image or video is a fundamental topic in computer vision. It is difficult to directly estimate the 3D human shape and pose from monocular images without any 3D information. To tackle this problem, massive 3D labeled data and 3D parametric human body models [26, 30, 3] with prior knowledge are necessary. Tremendous works [16, 18, 20, 27, 21, 29] based on Deep Neural Network (DNN) have been made to increase the accuracy and robustness of this task.

However, existing DNN-based methods often fail in some challenging scenarios, including cluttered background, occlusion and extreme pose. To overcome these challenges, three intrinsic relations should be jointly modeled for the video-based 3D human shape and pose estimation: a). *Spatial relation*: For the pose estimation task, the human joints areas and the spatial correlations among body parts are directly related to the pose prediction. It is critical to carefully utilize the spatial relation, especially in the scene of cluttered background. b). *Temporal relation*: Everyone has particular temporal trajectory in a given video. In occlusion cases, this temporal relation should be exploited to infer the pose of current occluded frame from surrounding frames. c). *Human joint relation*: In the parametric 3D body model SMPL [26], human joints are organized as a kinematic tree. Once pose changes, the parent joint rotates first, and then rotates the children. When the pose amplitude is large, we argue that the prior of the dependence among joints is especially helpful for accurate pose estimation. However, none of the existing methods fully utilizes the above three relations in a unified framework.

Motivated by the above observations, we propose Multi-level Attention Encoder-Decoder Network (MAED) for video-based 3D human shape and pose estimation. MAED is the first work to explore the above three relations by exploiting corresponding multi-level attentions in a unified framework. It includes Spatial-Temporal Encoder (STE) for spatial-temporal attention and Kinematic Topology Decoder (KTD) for human joint attention.

Specifically, the STE consists of several cascaded blocks, and each block uses two parallel branches to learn spatial and temporal attention respectively. We call the two branches Multi-Head Self-Attention Spatial (MSA-S) and Multi-Head Self-Attention Temporal (MSA-T), which are inspired by Multi-Head Self-Attention (MSA) mechanism

*Equal Contribution.



(a) Spatial-temporal attention

(b) Kinematic tree-based hierarchical regression

Figure 1: (a) Spatial-temporal attention: In current frame, the color of each pixel represents the spatial attention score, visualizing the importance of the spatial position. The color on the time axis represents the temporal attention score, visualizing the similarity between the corresponding frame and current one. Warmer color indicates higher attention score. (b) Kinematic tree-based hierarchical regression: Our model pays more attention to joints denoted by dots with warmer color.

in Transformer related works [38, 10, 11, 7, 6]. Derived from MSA, MSA-S and MSA-T have Transformer-like structures, but are different in the order of input features dimensions. As illustrated in Figure 1(a), MSA-S focuses on the critical spatial positions in image, highlighting significant features for pose estimation. Meantime, MSA-T concentrates on improving the prediction of current frame by exploiting frames that are informative to current one according to the calculated temporal attention scores.

On the other hand, existing methods usually use an iterative feedback regressor [16, 18] to regress the SMPL [26] parameters, in which the pose parameters of all joints are generated simultaneously. However, they ignore the human joint relation. To exploit the dependence among joints, we further propose KTD to simulate the SMPL kinematic tree for joint level attention modeling. In KTD, each joint is assigned a unique linear regressor to regress its pose parameters. As shown in Figure 1(b), these parameters are generated through a top-down hierarchical regression process. To estimate a joint, besides image feature, we also take the predicted pose parameters of its ancestors as the input of linear regressor. In this manner, the bias of the parent joint's estimation incurs substantial negative impact on the estimation of all its children, which forces the KTD to predict more accurate results for ancestor joints. In other words, although KTD does not explicitly allocate an attention score to each joint, the top-down regression process implicitly encourages the model to pay more attention to the parent joints with more children. As a result, the proposed KTD captures the inherent relation of joints and effectively reduce the prediction error.

We summarize the contributions of our method below:

- We propose Multi-level Attention Encoder-Decoder Network (MAED) for video-based 3D human shape and pose estimation. Our proposed MAED contains Spatial-Temporal Encoder (STE) and Kinematic Topology Decoder (KTD). It learns different attentions at spatial level, temporal level and human joint level in a unified framework.
- Our proposed STE leverages the MSA to construct MSA-S and MSA-T to encode the spatial and temporal attention respectively in the given video.
- Our proposed KTD considers hierarchical dependence among human joints and implicitly captures human joint level attention.

2. Related Works

2.1. 3D Human Shape and Pose Estimation

Recent works have made significant advances in 3D human pose and shape estimation due to the parametric 3D

CIS) ; VIBE

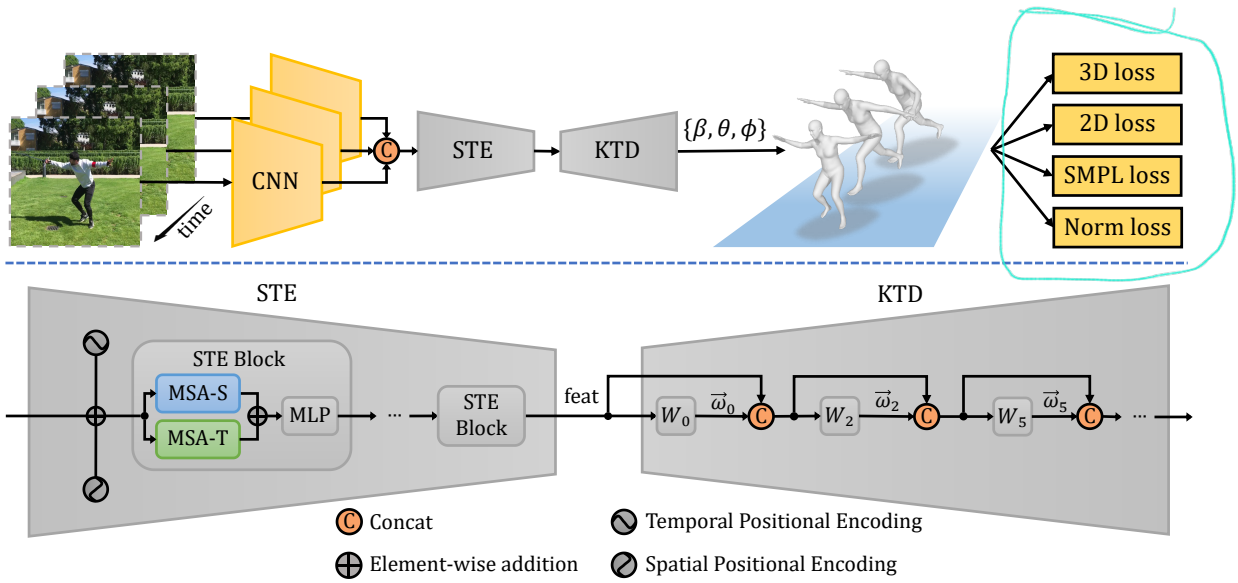


Figure 2: Overview of the proposed method MAED. The upper part shows the pipeline of the model and the lower part presents the structures of our proposed Spatial-Temporal Encoder and Kinematic Topology Decoder.

human body models, such as SMPL [26], SMPL-X [30] and SCAPE [3], which utilize the statistics of human body and provide 3D mesh based on few hyper-parameters. Later, various studies focus on estimating the hyper-parameters of 3D human model directly from image or video input.

Previous parametric 3D human body model based methods are split into two categories: optimization-based methods and regression-based methods. The optimization-based methods fit the parametric 3D human body models to pseudo labels, like 2D keypoints, silhouettes and semantic mask. SMPLify [26], one of the first end-to-end optimization-based methods, uses strong statistics priors to guide the optimization supervised by 2D keypoints. The work [23] utilizes silhouettes along with 2D keypoints to supervise the optimization. On the other hand, regression-based methods train deep neural network to regress the hyper-parameters directly. HMR [16] is trained with the supervision of re-projection keypoints loss along with adversarial learning of human shape and pose. SPIN [20] exploits SMPLify [26] in the training loop to provide more supervision. VIBE [18] is a video-based method that employs adversarial learning of the motions.

2.2. Transformer in Computer Vision

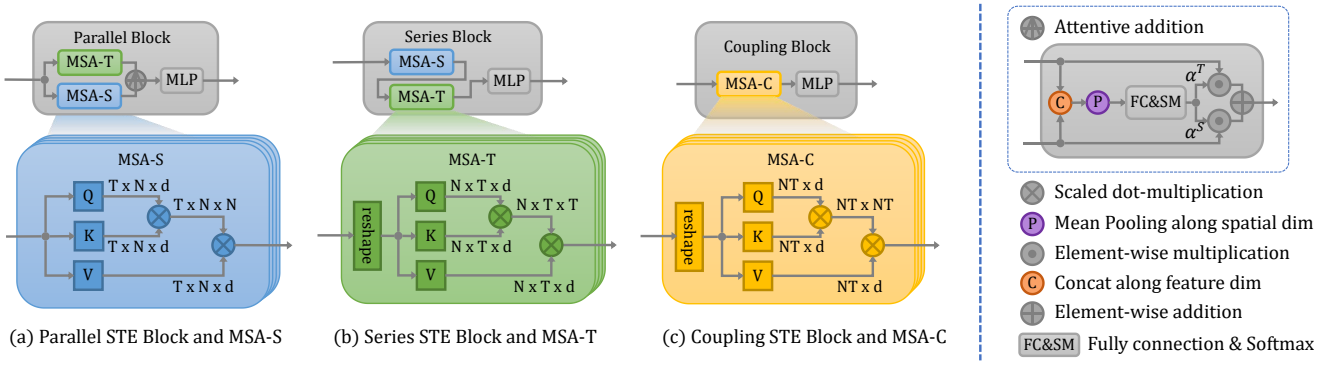
Transformer [38] is first proposed in NLP field. It is an encoder-decoder model, completely replacing commonly used recurrent neural networks with Multi-Head Self-Attention mechanism, and later achieves great success in various NLP tasks [10, 31, 32, 33, 22, 24]. Motivated by the achievements of Transformer in NLP, various works start to apply Transformer to computer vision tasks. Vision Transformer (ViT) [11] views an image as a 16x16 patch sequence, and trains a Transformer for image classification. The work [37] explores distillation to use smaller datasets to get more efficient ViT. Some works [41, 35] study various Transformer structures which are more suitable for visual classification tasks. In addition, Transformer has also achieved impressive results in many downstream computer vision tasks, including denoising [7], object detection [6, 46], video action recognition [12], 3D mesh reconstruction [43], panoptic segmentation [40], *etc*. In this paper, we focus on using Transformer to fully exploit the spatial-temporal level attention from video for better human pose and shape estimation.

3. Methods

In this section, we first revisit the parametric 3D human body model (SMPL [26]). Secondly, we give an overview of our proposed framework. Finally, we describe the proposed STE and KTD in detail.

3.1. SMPL

SMPL [26] is a classical parametric human body model with $N = 6890$ vertices and $K = 23$ joints. It provides a function $\mathcal{M}(\vec{\beta}, \vec{\theta})$ that takes as input the shape parameters $\vec{\beta} \in \mathbb{R}^{10}$ and the pose parameters $\vec{\theta} \in \mathbb{R}^{72}$, and returns the body mesh $M \in \mathbb{R}^{N\times 3}$. $\vec{\beta}$ are the first 10 coefficients of a PCA shape space, controlling the shape of the body (*e.g.*, height, weight, etc). $\vec{\theta} = \left[\vec{\omega}_0^T, \ldots, \vec{\omega}_K^T\right]^T$ controls the pose of the body, where $\vec{\omega}_k \in \mathbb{R}^3$ denotes the axis-angle representation of the relative rotation of joint k with respect to its parent in the kinematic tree. $\vec{\theta}$ is defined by $|\vec{\theta}| =$



(a) Parallel STE Block and MSA-S (b) Series STE Block and MSA-T (c) Coupling STE Block and MSA-C

Figure 3: STE block variants and MSA variants.

$3 \times 23 + 3 = 72$ parameters, *i.e.*, 3 for each joint plus 3 for the root orientation. These joints can be calculated by a linear regressor J_{reg}, *i.e.*, $J_{3d} \in \mathbb{R}^{K\times 3} = J_{reg}M$.

3.2. Framework Overview

Figure 2 shows the architecture of our proposed network. It takes a video clip of length T as input, and adopts a CNN backbone to extract the basic feature for each frame. The global pooling layer at the end of the CNN is omitted, resulting in T feature maps of size $(h \times w \times d)$, where h/w/d denotes the height/width/channel size of feature map. We reshape each feature map into 1D sequence of size $(hw \times d)$, and prepend a trainable embedding to each sequence (Following [11], we denote a token in the sequence as a patch). Thus, the CNN outputs a matrix of size $(T \times N \times d)$, where $N = hw + 1$. Then our proposed Spatial-Temporal Encoder (STE) is used to perform spatial-temporal modeling on these basic features. The encoded vector corresponding to the prepended embedding serves as the output of STE. Finally, our proposed Kinematic Topology Decoder (KTD) is employed to estimate shape $\vec{\beta}$, pose $\vec{\theta}$ and camera $\vec{\phi}$ parameters from the output of STE. These predicted parameters allow us to utilize SMPL to calculate 3D joints and their 2D projection, $J_{2d} = \Pi_{\vec{\phi}}(J_{3d})$, where $\Pi_{\vec{\phi}}(.)$ is the projection function.

After getting $\{\vec{\beta}, \vec{\theta}, J_{3d}, J_{2d}\}$, the model is supervised by the following 4 losses:

$$L = L_{3D} + L_{2D} + L_{SMPL} + L_{NORM} \tag{1}$$

where L_{2D}/L_{3D} denotes the 2D/3D keypoint loss, L_{SMPL} denotes the SMPL parameters loss, and L_{NORM} denotes the L_2-Normalization loss. $J_{3dgt}, J_{2dgt}, \vec{\theta}_{gt}, \vec{\beta}_{gt}$ represent the ground truth of $J_{3d}, J_{2d}, \vec{\theta}, \vec{\beta}$ respectively.

$$\begin{aligned} L_{3D} &= \sum_{k=1}^{K} \left\| J_{3d}^k - J_{3dgt}^k \right\|_2, \\ L_{2D} &= \sum_{k=1}^{K} \left\| J_{2d}^k - J_{2dgt}^k \right\|_2 \\ L_{SMPL} &= \|\vec{\theta} - \vec{\theta}_{gt}\|_2 + \|\vec{\beta} - \vec{\beta}_{gt}\|_2 \\ L_{NORM} &= \|\vec{\beta}\|_2 + \|\vec{\theta}\|_2 \end{aligned} \tag{2}$$

3.3. Spatial-Temporal Encoder

Transformer [38] is able to effectively model the interaction of tokens in a sequence. Recently, applying Transformer to model the temporal attention on global pooling feature of each frame is widely used in many video-based computer vision tasks. However, the global pooling operation will inevitably lose the spatial information in a frame, which makes it difficult to estimate detailed human pose. In our method, to perform spatial and temporal modeling simultaneously, we serialize the input video clip in multiple ways, and design three variants based on Multi-Head Self-Attention (MSA) [38]: Multi-Head Spatial Self-Attention (MSA-S), Multi-Head Temporal Self-Attention (MSA-T) and Multi-Head Self-Attention Coupling (MSA-C). Then we further design three forms of Spatial-Temporal Encoder (STE) Block as shown in Figure 3, which endows the encoder with both global spatial perception and temporal reasoning capability. Finally, we stack multiple STE Blocks to construct the STE.

MSA Variants. The standard MSA can only learn attention of one dimension, so the different order of input dimensions affects the meaning of learned attention. Our proposed three variants have similar model structure, but are different on the order of the input dimensions.

MSA-S aims at finding the key spatial information in a frame, such as joints and limbs of human body. It is shown

in the blue box in Figure 3(a), where each self-attention head outputs a heatmap of size $(T \times N \times N)$ computed by scaled dot-multiplication. However, in this setting, temporal relations among frames are not captured, as a patch in one frame does not interact with any patch in other frames.

MSA-T is pretty similar to MSA-S, except that it first reshapes the input matrix from size $(T \times N \times d)$ to $(N \times T \times d)$ as shown in the green box in Figure 3(b). Each head of MSA-T outputs the heatmap of size $(N \times T \times T)$, where each score reflects the attention of a patch to the patch in the same position in other frames. Although temporal semantics is modeled explicitly, MSA-T ignores spatial relation of patches in the same frame.

MSA-C flattens patch sequence and frame sequence together, *i.e.*, reshape the input matrix from size $(T \times N \times d)$ to $(TN \times d)$, as shown in the yellow box in Figure 3(c). In this way, the heatmap of size $(TN \times TN)$ enables each patch interacts with any other patches in the video clip.

STE Blocks. As depicted in Figure 3, we design three kinds of STE Blocks based on these MSA variants. Coupling Block consists of a MSA-C followed by a Multi-Layer Perception (MLP) layer, modeling spatial-temporal information in a coupling fashion. However, it greatly increases the complexity since the complexity of dot-multiplication is quadratic to sequence length.

Parallel Block and Series Block connect MSA-S and MSA-T in parallel and in series respectively. For Parallel Block, a naive way of integrating two branches is to simply compute the element-wise mean of the outputs of MSA-S and MSA-T. In order to dynamically balance the temporal and spatial information, we compute attentive weights $\alpha^S, \alpha^T \in \mathbb{R}^{T \times 1 \times d}$ for the two branches. They represent attention scores for the temporal and spatial component along the feature channels of each frame respectively.

Connection of MSA-T and MSA-S makes it possible to combine image and video datasets to train more robust models. When it comes to image input, we simply bypass or disconnect the MSA-T in the blocks to ignore the non-existent temporal information.

Considering the trade-off between accuracy and speed, we empirically choose Parallel Block in our STE, as the Parallel Block is able to dynamically adjust the attentive weights between spatial and temporal attention and yields the best results compared with other variants. The quantitative comparison is discussed in Section 4.4.2 in detail.

Spatial-Temporal Positional Encoding. In order to locate the spatial and temporal position of a patch, we add two separated positional encodings to inject sequence information into the input, namely spatial positional encoding $\mathbf{E}^S_{pos} \in \mathbb{R}^{1 \times N \times d}$ and temporal positional encoding $\mathbf{E}^T_{pos} \in \mathbb{R}^{T \times 1 \times d}$. They are both trainable and added to the input sequence matrix.

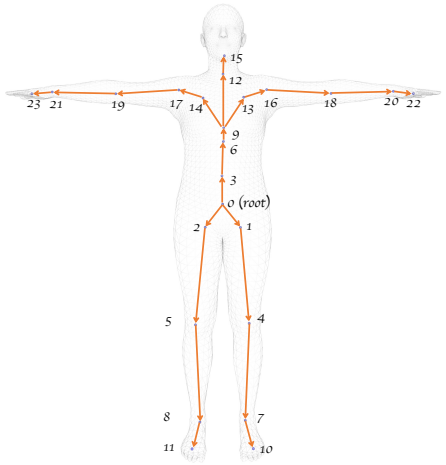


Figure 4: A demonstration of kinematic tree with 23 joints and a root. The arrow points from the parent to its child.

3.4. Kinematic Topology Decoder

As aforementioned, previous works ignore the inherent dependence among joints and regard them as equally important. Therefore, we design Kinematic Topology Decoder (KTD) to implicitly model the attention at the joint level.

As demonstrated in Figure 4, the pose of human body is controlled by 23 joints which are organized as a kinematic tree. We first revisit how the pose parameters rotate the joints in SMPL [26]. As shown in Eq (3), the world transformation of joint k denoted by $G_k(\mathcal{R}, \mathcal{T}) \in \mathbb{R}^{4 \times 4}$ equals the cumulative product of the transformation matrices of its ancestors in the kinematic tree.

$$G_k(\mathcal{R}, \mathcal{T}) = \prod_{i \in A(k)} \begin{pmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0} & \mathbf{1} \end{pmatrix} \tag{3}$$

where $\mathcal{R} = [\mathbf{R}_0, ..., \mathbf{R}_K], \mathcal{T} = [\mathbf{t}_0, ..., \mathbf{t}_K]$. Following SMPL, $\mathbf{R}_k \in \mathbb{R}^{3 \times 3}$ and $\mathbf{t}_k \in \mathbb{R}^{3 \times 1}$ denote the rotation matrix and translation vector of joint k respectively. $A(k)$ is the ordered set of joint k's ancestors, *e.g.*, $A(5) = \{0, 2\}$.

Therefore, the position of a joint is affected by its own and ancestral pose parameters. The more children a joint has, the greater its impact on the accuracy of the overall joint position estimation. Despite this, currently widely used iterative feedback regressor [16, 18] does not pay more attention to the parent joints, especially the root of kinematic tree. As a result, it can only get sub-optimal results. However, our proposed KTD can avoid the problem. In KTD, we first decode the shape/cam parameters with a matrix $\mathbf{W}_{\text{shape}}$/$\mathbf{W}_{\text{cam}}$ as shown in Eq (4).

$$\vec{\beta} = \mathbf{W}_{\text{shape}} \cdot \mathbf{x}, \quad \vec{\phi} = \mathbf{W}_{\text{cam}} \cdot \mathbf{x} \tag{4}$$

where $\mathbf{W}_{\text{shape}} \in \mathbb{R}^{10 \times d}$, $\mathbf{W}_{\text{cam}} \in \mathbb{R}^{3 \times d}$, and $\mathbf{x} \in \mathbb{R}^d$ is the image feature extracted by the STE.

Models	Input	3DPW				MPI-INF-3DHP		Human3.6M	
		PA-MPJPE	MPJPE	PVE	ACCEL	PA-MPJPE	MPJPE	PA-MPJPE	MPJPE
HMR[16] w/o 3DPW	image	81.3	130.0	-	37.4	89.8	124.2	56.8	88.0
GraphCMR[21] w/o 3DPW	image	70.2	-	-	-	-	-	50.1	-
STRAPS[34] w/ 3DPW	image	66.8	-	-	-	-	-	55.4	-
ExPose[9] w/o 3DPW	image	60.7	93.4	-	-	-	-	-	-
SPIN[20] w/o 3DPW	image	59.2	96.9	116.4	29.8	67.5	105.2	41.1	-
I2LMeshNet[29] w/o 3DPW	image	57.7	93.2	-	-	-	-	41.1	**55.7**
Pose2Mesh[8] w/o 3DPW	2D Pose	58.3	88.9	-	-	-	-	46.3	64.9
TemporalContext[4] w/o 3DPW	video	72.2	-	-	-	-	-	54.3	77.8
DSD-SATN[36] w/o 3DPW	video	69.5	-	-	-	-	-	42.4	59.1
MEVA[27] w/ 3DPW	video	54.7	86.9	-	**11.6**	65.4	96.4	53.2	76.0
VIBE[18] w/o 3DPW	video	56.5	93.5	113.4	27.1	63.4	97.7	41.5	65.9
VIBE[18] w/ 3DPW	video	51.9	82.9	99.1	23.4	64.6	96.6	41.4	65.6
Ours w/o 3DPW	video	**50.7**	**88.8**	**104.5**	18.0	**56.5**	**85.1**	**38.7**	56.3
Ours w/ 3DPW	video	**45.7**	**79.1**	**92.6**	17.6	**56.2**	**83.6**	**38.7**	56.4

Table 1: Performance comparison with the state-of-the-art methods on 3DPW, MPI-INF-3DHP and Human3.6M datasets. The bold font represents the best result.

Then we iteratively generate the pose parameter for each joint in hierarchical order according to the structure of kinematic tree. Take joints {0, 2, 5} in Figure 4 as an example. We first predict the pose parameters of root, namely the global body orientation, using the output feature of STE and a learnable linear regressor $\mathbf{W}_0 \in \mathbb{R}^{6\times d}$, *i.e.*, $\vec{\omega}_0 = \mathbf{W}_0 \cdot \mathbf{x}$. Here, following [20], we use the 6D rotation representation proposed in [45] for faster convergence. Then, for its child joint 2, we take the image feature $\mathbf{x}$ and $\vec{\omega}_0$ as the input of another linear regressor $\mathbf{W}_2 \in \mathbb{R}^{6\times(d+6)}$ which outputs the pose parameters $\vec{\omega}_2$, *i.e.*, $\vec{\omega}_2 = \mathbf{W}_2 \cdot Concat(\mathbf{x}, \vec{\omega}_0)$, where $Concat(\cdot)$ is the concatenate operation. Similarly for the grandson joint 5, $\vec{\omega}_5 = \mathbf{W}_5 \cdot Concat(\mathbf{x}, \vec{\omega}_0, \vec{\omega}_2)$, $\mathbf{W}_5 \in \mathbb{R}^{6\times(d+12)}$. This regression process is shown in Figure 2.

By KTD, we establish the dependence between the parent joint and its children, which is consistent with kinematic tree structure. In traditional regressor, the error of the parent joint's pose estimation only affects itself. While in KTD, the error will be propagated to its children as well. This encourages the model to learn an attention at the joint level and pay more attention to parent joints, so as to achieve more accurate estimation results.

4. Experiments

4.1. Datasets

Training. Following previous works [16][20][18], we use mixed datasets for training, including 3D video datasets, 2D video datasets and 2D image datasets. For 3D video datasets, Human3.6M [14] and MPI-INF-3DHP [28] provide 3D keypoints and SMPL parameters in indoor scene. For 2D video datasets, PennAction [42] and PoseTrack [1] provide ground-truth 2D keypoints annotation, while InstaVariaty [17] provides pseudo 2D keypoints annotation using a keypoint detector [5, 19]. For image-based datasets, COCO [25], MPII [2] and LSP-Extended [15] are adopted, providing in-the-wild 2D keypoints annotation. Meanwhile, we conduct ablation study on the 3DPW [39] dataset.

Evaluation. We report the experiments results on Human3.6M [14], MPI-INF-3DHP [28] and 3DPW [39] evaluation set. We adopt the widely used evaluation metrics following previous works [16][20][18], including Procrustes-Aligned Mean Per Joint Position Error (PA-MPJPE), Mean Per Joint Position Error (MPJPE), Per Vertex Error (PVE) and ACCELeration error (ACCEL). We report the results with and without 3DPW [39] training set for fair comparison with previous methods.

4.2. Training Details

Data Augmentation. Horizontal flipping, random cropping, random erasing [44] and color jittering are employed to augment the training samples. Different frames of the same video input share consistent augmentation parameters.

Model Details. Following [11], we use a modified ResNet-50 [13] as the CNN backbone to extract the basic feature of an input image. For STE, 6 STE Parallel Blocks are stacked, and each block has 12 heads. We adopt the weights from [11] to initialize the ResNet-50 and STE.

The whole training process is divided into two stages. In the first stage, the model aims at accumulating sufficient spatial prior knowledge, and thus is trained with all image-based datasets and frames from Human3.6M and MPI-INF-3DHP. We fix the number of epochs as 100 and the mini-batch size as 512 for this stage. In the second stage, we use both video and image datasets for temporal modeling. For video datasets, we sample 16-frame clips at a interval of 8 as training instances. We train another 100 epochs with a mini-batch size of 32 for this stage. The model is optimized by

ACCEL would have been interesting here

Encoder	Decoder	3DPW	
		PA-MPJPE	MPJPE
CNN	Iterative	52.2	87.5
CNN	KTD	50.9	88.0
CNN+STE	Iterative	47.5	80.2
CNN+STE	KTD	**45.7**	**79.1**
CNN	Iterative	52.2	87.5
CNN+TE		51.1	84.5
CNN+SE		49.8	84.5
CNN+STE$_{\text{series}}$		48.5	83.6
CNN+STE$_{\text{parallel}}$v1		48.1	81.6
CNN+STE$_{\text{parallel}}$v2		**47.5**	**80.2**
CNN+STE$_{\text{coupling}}$		49.3	82.6
CNN+STE	Iterative	47.5	80.2
	Decoder$_{\text{vanilla}}$	47.7	80.7
	KTD	**45.7**	**79.1**
	KTD$_{\text{random}}$	47.7	82.5
	KTD$_{\text{reverse}}$	47.6	79.7

Table 2: Analytical experiment results with different encoders and decoders. CNN represents ResNet-50. "Iterative" represents the iterative feedback regressor.

Adam optimizer with an initial learning rate of 10^{-4} which is decreased by 10 at the 60-th and 90-th epochs. Finally, each term in the loss function has different weighting coefficients. Refer to Sup. Mat. for further details. All experiments are conducted on 16 Nvidia GTX1080ti GPUs.

4.3. Comparison to state-of-the-art results

In this section, we compare our method with the state-of-the-art models on 3DPW, MPI-INF-3DHP and Human3.6M, and the results are summarized in Table 1. On the 3DPW and MPI-INF-3DHP datasets, our method outperforms other competitors including image- and video-based methods by a large margin, whether or not using 3DPW training set. On Human3.6M, our method achieves results on-par with I2LMeshNet [29]. We also observe MEVA [27], an two-stage method that aims at producing both smooth and accurate results, ranks best in ACCEL metric on 3DPW. However, considering all indicators, our method achieves better performance overall.

These results validate our hypothesis that the exploitation of the attentions at spatial-temporal level and human joint level greatly helps to achieve more accurate estimation. The leading performance on these three datasets (especially the in-the-wild dataset 3DPW) demonstrates the robustness and the potential to real-world applications of our method.

4.4. Ablation Study

4.4.1 The effectiveness of STE and KTD

The upper part of Table 2 verifies the effectiveness of our proposed STE and KTD. Compared with CNN encoder+Iterative decoder, STE and KTD brings 4.7 and 1.3 mm improvement in PA-MPJPE metric respectively. Moreover, STE and KTD together further improves the performance by 6.5 mm. This proves the attention at different levels extracted by STE and KTD effectively complement rather than conflict each other.

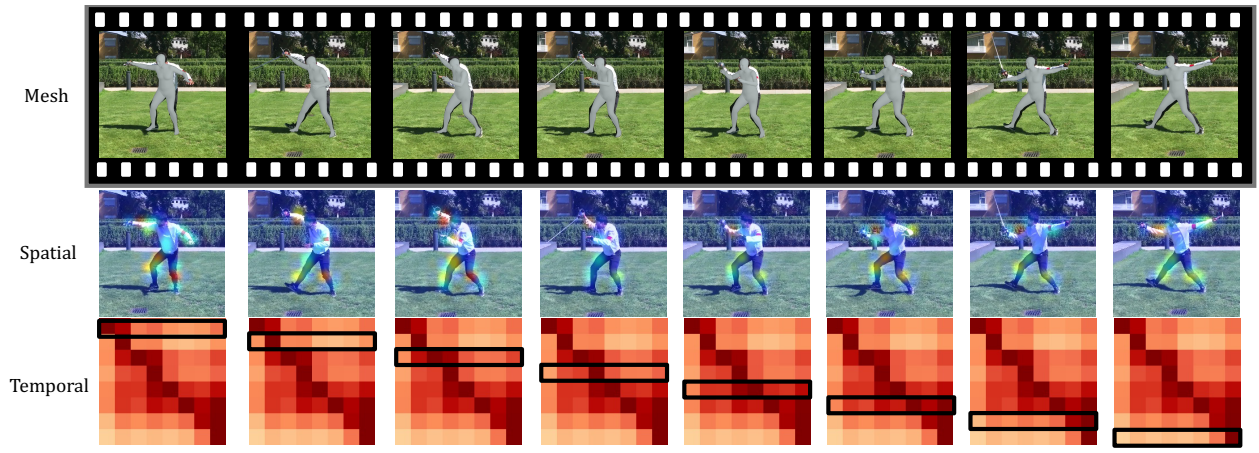
We can also observe that when using CNN encoder, the gain of KTD in PA-MPJPE metric is smaller than that when using CNN+STE encoder. Even there is a small decline in MPJPE metric. This is because the CNN loses too much spatial information due to the global pooling operation, and fails to provide detailed human body clue for KTD. However, with hard downsampling removed, STE not only preserves more spatial information, but also pay more attention to more informative locations, which makes KTD capture more precise attention between joints.

4.4.2 Influence of different encoders

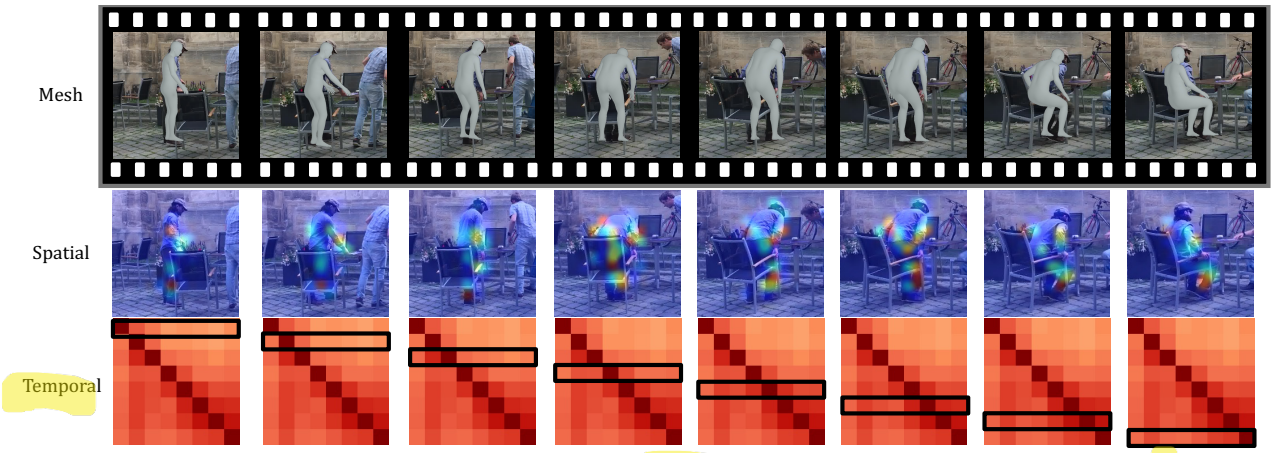
In the middle part of Table 2, we compare the performance of various forms of STE. SE denotes the encoder with only MSA-S. TE denotes the encoder with only MSA-T and CNN global pooling layer kept. STE$_{\text{parallel}}$v1 and STE$_{\text{parallel}}$v2 denote the Parallel Block w/o and w/ attentive addition respectively. We conclude that all the variants of STE benefit the model, while STE$_{\text{parallel}}$v2 yields the most significant gain. This is because the attentive weights dynamically computed in the Parallel Block effectively act as a valve which adjusts the proportion of temporal and spatial information passing through the network. When it comes to occlusion or ambiguity, the valve will allow more temporal information to pass through to complement the lack of information in current frame, and do otherwise when the current frame is clear. Surprisingly, STE$_{\text{coupling}}$ yields only modest improvement over encoder with only MSA-S (49.8→49.3), which has no temporal modeling capability. We also observe that STE$_{\text{coupling}}$ converges more slowly compared to other STE variants. We argue that flattening the spatial and temporal dimension together may harm human pose estimation mainly due to the extremely long sequence. Tremendous irrelevant patches (such as background and joints that are too far apart) overwhelm valid information, making it challenging for the current patch to allocate reasonable attention.

4.4.3 Influence of different decoders

We choose CNN+STE as the encoder and report the results with different decoders in the lower part of Table 2. KTD$_{\text{random}}$ denotes the KTD on a randomly generated kinematic tree. KTD$_{\text{reverse}}$ denotes the KTD on the reverse kinematic tree, namely, exchange the relationship between parent joint and its children. Decoder$_{\text{vanilla}}$ denotes the standard decoder in [38] with 6 layers. It takes as input the



(a) Extreme Pose. Spatial attention is mainly concentrated on human joints, while temporal attention is mainly concentrated near the anchor frame.



(b) Back view under cluttered background and occlusion. The heatmap of temporal attention shows relatively higher variance, indicating the current frame needs to refer to more temporal information to make up for the infomation lack incurred by occlusion.

Figure 5: Qualitative visualization of MAED. More visualization results will be shown in Sup.Mat.

zero sequence of length 37 (24 for pose, 10 for shape and 3 for camera) and outputs SMPL parameters. We observe that KTD outperforms Iterative by a large margin. While $\text{KTD}_{\text{random}}$ and $\text{KTD}_{\text{reverse}}$ have no obvious improvement, even are slightly worse, proving unreasonable kinematic tree is useless prior knowledge, which brings difficulties to the optimization of the network. We also observe that $\text{Decoder}_{\text{vanilla}}$ brings no improvement. Although it can capture the relation between different joints with the self-attention mechanism, the predictions of all joints are generated simultaneously, not in the sequential way as KTD. As a result, it can not pay more attention to the parent joints.

4.5. Visualization Analysis

Figure 5 includes qualitative results of MAED from two representative scenarios. For these challenging cases including extreme pose in Figure 5(a) and cluttered background and occlusion in Figure 5(b), our model predicts reasonable spatial and temporal attention maps and further produce proper estimations.

5. Conclusion

This paper describes MAED, an approach that utilizes multi-level attentions at spatial-temporal level and human joint level for 3D human shape and pose estimation. We design multiple variants of MSA and STE Block to construct STE to learn spatial-temporal attention from the output feature of CNN backbone. In addition, we propose KTD, which simulates the process of joint rotation based on SMPL kinematic tree to decode human pose. MAED makes significant accuracy improvement on multiple datasets but also brings non-negligible computation overhead, which we explore further in the Sup. Mat. Thus, future work could consider reducing computation overhead or extending this method to capture the relation between multiple people.

References

[1] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5167–5176, 2018. 6

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pages 3686–3693, 2014. 6

[3] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *ACM SIGGRAPH 2005 Papers*, pages 408–416. 2005. 1, 3

[4] A. Arnab, C. Doersch, and A. Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 6

[5] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 6

[6] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 2, 3

[7] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer. *arXiv preprint arXiv:2012.00364*, 2020. 2, 3

[8] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *European Conference on Computer Vision*, pages 769–787. Springer, 2020. 6

[9] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J Black. Monocular expressive body regression through body-driven attention. In *European Conference on Computer Vision*, pages 20–40. Springer, 2020. 6

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 3

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2, 3, 4, 6

[12] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019. 3

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

[14] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 6

[15] Sam Johnson and Mark Everingham. Learning effective human pose estimation from inaccurate annotation. In *CVPR 2011*, pages 1465–1472. IEEE, 2011. 6

[16] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018. 1, 2, 3, 5, 6

[17] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5614–5623, 2019. 6

[18] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5253–5263, 2020. 1, 2, 3, 5, 6

[19] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 417–433, 2018. 6

[20] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2252–2261, 2019. 1, 3, 6

[21] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4501–4510, 2019. 1, 6

[22] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*, 2019. 3

[23] Christoph Lassner, Javier Romero, Martin Kiefel, Federica Bogo, Michael J Black, and Peter V Gehler. Unite the people: Closing the loop between 3d and 2d human representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6050–6059, 2017. 3

[24] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019. 3

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 6

[26] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 1, 2, 3, 5

[27] Zhengyi Luo, S Alireza Golestaneh, and Kris M Kitani. 3d human motion estimation via motion compression and refinement. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1, 6, 7

[28] 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 *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 6

[29] 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. *arXiv preprint arXiv:2008.03713*, 2020. 1, 6, 7

[30] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10975–10985, 2019. 1, 3

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

[32] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 3

[33] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 3

[34] Akash Sengupta, Ignas Budvytis, and Roberto Cipolla. Synthetic training for accurate 3d human pose and shape estimation in the wild. *arXiv preprint arXiv:2009.10013*, 2020. 6

[35] Aravind Srinivas, Tsung-Yi Lin, Niki Parmar, Jonathon Shlens, Pieter Abbeel, and Ashish Vaswani. Bottleneck transformers for visual recognition. *arXiv preprint arXiv:2101.11605*, 2021. 3

[36] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, Yili Fu, and Tao Mei. Human mesh recovery from monocular images via a skeleton-disentangled representation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5349–5358, 2019. 6

[37] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. *arXiv preprint arXiv:2012.12877*, 2020. 3

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 2, 3, 4, 7

[39] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 601–617, 2018. 6

[40] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European Conference on Computer Vision*, pages 108–126. Springer, 2020. 3

[41] Li Yuan, Yunpeng Chen, Tao Wang, Weihao Yu, Yujun Shi, Francis EH Tay, Jiashi Feng, and Shuicheng Yan. Tokens-to-token vit: Training vision transformers from scratch on imagenet. *arXiv preprint arXiv:2101.11986*, 2021. 3

[42] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2248–2255, 2013. 6

[43] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. Point transformer. *arXiv preprint arXiv:2012.09164*, 2020. 3

[44] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13001–13008, 2020. 6

[45] Xiaowei Zhou, Menglong Zhu, Spyridon Leonardos, Konstantinos G Derpanis, and Kostas Daniilidis. Sparseness meets deepness: 3d human pose estimation from monocular video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4966–4975, 2016. 6

[46] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 3