XFormer: Fast and Accurate Monocular 3D Body Capture

Lihui Qian[1] , **Xintong Han**[1] , **Faqiang Wang**[1] , **Hongyu Liu**[2] , **Haoye Dong**[3] ,
Zhiwen Li[1] , **Huawei Wei**[4] , **Zhe Lin**[1] and **Cheng-Bin Jin**[1]
[1]Huya Inc
[2]Hong Kong University of Science and Technology
[3]Carnegie Mellon University
[4]Tencent
turtleduck1995@gmail.com, hanxintong@huya.com

Abstract

We present XFormer, a novel human mesh and motion capture method that achieves real-time performance on consumer CPUs given only monocular images as input. The proposed network architecture contains two branches: a keypoint branch that estimates 3D human mesh vertices given 2D keypoints, and an image branch that makes predictions directly from the RGB image features. At the core of our method is a cross-modal transformer block that allows information to flow across these two branches by modeling the attention between 2D keypoint coordinates and image spatial features. Our architecture is smartly designed, which enables us to train on various types of datasets including images with 2D/3D annotations, images with 3D pseudo labels, and motion capture datasets that do not have associated images. This effectively improves the accuracy and generalization ability of our system. Built on a lightweight backbone (MobileNetV3), our method runs blazing fast (over 30fps on a single CPU core) and still yields competitive accuracy. Furthermore, with an HRNet backbone, XFormer delivers state-of-the-art performance on Huamn3.6 and 3DPW datasets.

1 Introduction

3D body capture techniques play an essential role in a wide range of computer vision and graphics applications such as telepresence, VR chat, virtual YouTubers, and interactive gaming. However, accurate and temporally coherent body capture may require special (and usually expensive) devices, such as motion capture suits, multi-camera systems, and depth sensors, which highly hinder its large-scale applications. To resolve this issue, researchers have developed various methods that predict 3D body pose and mesh from monocular RGB images [Mehta *et al.*, 2017b; Kanazawa *et al.*, 2018; Kocabas *et al.*, 2020; Mehta *et al.*, 2020; Sun *et al.*, 2021; Lin *et al.*, 2021a; Kocabas *et al.*, 2021]. Despite remarkable progress has been made, these methods often fail to capture accurate body motion in challenging in-the-wild scenes, especially when real-time performance is desired.

One main challenge that remains in monocular 3D body capture is that acquiring training images with accurate 3D body annotations is hard. As a result, researchers attempt to use images with 2D annotations to facilitate the training. For example, most CNN-based approaches [Kanazawa *et al.*, 2018; Kolotouros *et al.*, 2019; Mehta *et al.*, 2020; Kanazawa *et al.*, 2019; Lin *et al.*, 2021a] leverage datasets annotated with 2D keypoints (*e.g.*, COCO [Lin *et al.*, 2014], LSP [Johnson and Everingham, 2010], MPII [Andriluka *et al.*, 2014]) and minimize the reprojection loss of keypoints in order to improve the accuracy on in-the-wild images. [Kocabas *et al.*, 2020; Kanazawa *et al.*, 2019] further extend the training modality to monocular videos with ground truth or pseudo 2D keypoint labels (PennAction [Zhang *et al.*, 2013], PoseTrack [Andriluka *et al.*, 2018], InstaVariety [Kanazawa *et al.*, 2019]) to exploit temporal information for boosting 3D motion estimation. Another line of research [Choi *et al.*, 2020; Martinez *et al.*, 2017; Zhao *et al.*, 2019] has shown that, without directly using image information, 2D keypoints alone provide essential (and sufficiently good) geometric information of 3D human pose/shape (*e.g.*, short 2D hip joint distance suggests skinny lower body). And directly regressing 3D joints/mesh vertices from 2D keypoints is more effective and easier than previously thought.

Intuitively, the semantic feature from images and 2D keypoints are complementary to each other, and it is interesting to integrate these two representations for a better pose and mesh reconstruction. This has been investigated by concatenating 2D keypoint heatmaps with intermediate image features [Mehta *et al.*, 2017b; Mehta *et al.*, 2020], adopting a trainable fusion scheme [Tekin *et al.*, 2017], and using a bilinear transformation for multi-modality fusion [Sun *et al.*, 2019]. However, these methods simply combine the multi-model features without explicitly exploiting their interactions. Meanwhile, these fusion strategies make the models unable to be trained on the Motion Capture data without paired images [Mahmood *et al.*, 2019], resulting in inadequate training.

To overcome the above limitations, we propose a real-time 3D body capture system, termed XFormer. The XFormer enforces knowledge transfer across image features and keypoints features with a proposed novel cross-modal attention module inspired by [Vaswani *et al.*, 2017; Lu *et al.*, 2019]. Specifically, as shown in Figure 1, XFormer consists of two branches, *i.e.*, the keypoint branch and the image branch. The

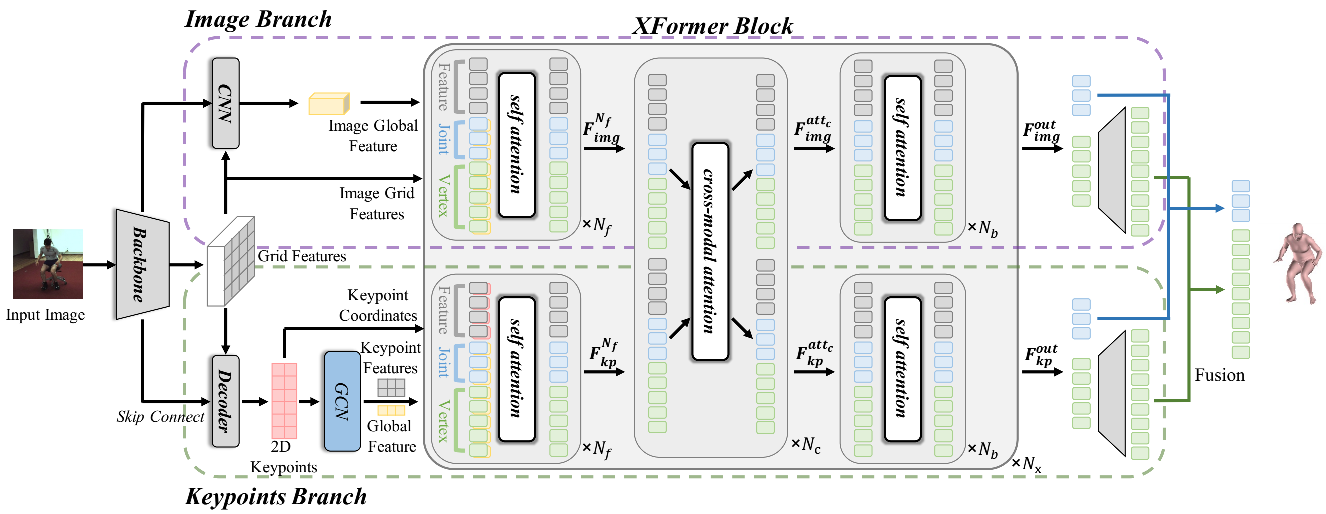

Figure 1: **XFormer system overview.** We predict 3D body joints and mesh with a network consisting of an image branch and a keypoint branch. These two branches interact with N_x XFormer blocks, with each block containing N_f front self-attention modules, N_c cross-modal attention modules, and N_b back self-attention modules. F_{img} and F_{kp} denote the image branch feature and the keypoint branch feature.

keypoint branch takes the 2D keypoints to regress 3D joints and mesh vertices of the SMPL model [Loper *et al.*, 2015], in which the 2D keypoints are predicted from the image by a keypoint detector or projected by the 3D keypoints from the MoCap data. The image branch directly predicts the same information from the input image feature. In order to effectively integrate the advantages of these two branches, we feed the keypoint representations together with the image feature map into the proposed cross-modal attention module. By exchanging key-value pairs in the multi-head attention, we obtain the cross attention between the 2D keypoints and the image features, which enforces information communication between these two modalities. Extensive experiments in Section 5.2 demonstrate that this framework significantly outperforms each individual branch that only captures a single modality.

XFormer also takes full advantage of the datasets with different supervision types. In the previous work like [Kocabas *et al.*, 2020; Kolotouros *et al.*, 2021; Rempe *et al.*, 2021], the MoCap data (*e.g.*, AMASS [Mahmood *et al.*, 2019]), as it does not have paired images, is only used in the discriminator or as human motion priors. Thus, the strong supervision of 2D-3D pose mapping is ignored. In contrast, our XFormer enjoys a "modality plug-in" characteristic, enabling the network to learn from MoCap or synthetic data without corresponding images. This is achieved by a modality switch, which uses an MLP to mimic the features learned by cross-modal attention. Therefore, 2D keypoint features can skip the cross-modal attention and directly forward to the following network during training when the image modality is unavailable. Benefiting from this design, we can train on massive MoCap sequences even if they do not consist of any images.

With the proposed cross-modal attention and leveraging MoCap datasets, Xformer significantly boosts the performance when the backbone is lightweight (MobileNetV3 [Howard *et al.*, 2019]) and achieves real-time performance on consumer CPUs. In contrast, the accuracy of previous methods [Kocabas *et al.*, 2020; Lin *et al.*, 2021b; Kocabas *et al.*, 2021] largely drops by replacing their original heavy backbones (*e.g.*, ResNet50 [He *et al.*, 2016], HRNet [Wang *et al.*, 2018]) with lighter ones.

In sum, the main contributions of this paper are as follows:

1) We introduce a novel network architecture that estimates body joints and mesh from 2D keypoints and RGB image features, whose interactions are captured with a proposed cross-modal attention module.

2) The proposed two-branch architecture and XFormer blocks are designed to leverage all types of training data (2D and 3D, with and without images) of different modalities, which further improves its accuracy.

3) Our system with a light backbone takes less than 7ms per frame for an input person on an Nvidia GTX 1660 GPU and 30ms with a single thread of Intel i7-8700 CPU, obtaining significant speedup while maintaining satisfactory accuracy.

4) Our proposed method achieves state-of-the-art quantitative performance with HRNet backbone on 3D pose estimation benchmarks and demonstrates significant qualitative results in challenging in-the-wild scenes.

2 Related Work

3D Human Pose Estimation. 3D human pose estimation can be categorized into model-free and model-based approaches. Model-free approaches predict 3D human pose by directly estimating 3D keypoint from images [Pavlakos *et al.*, 2018; Lin *et al.*, 2021a] or detected 2D human pose [Choi *et al.*, 2020]. Model-based methods estimate 3D pose by predicting body model (*e.g.*, SMPL) parameters [Kanazawa *et al.*, 2018; Kocabas *et al.*, 2020; Zanfir *et al.*, 2021] or meshes [Lin *et al.*, 2021a; Lin *et al.*, 2021b]. Our method directly predicts 3D mesh vertices of the SMPL model to leverage the strong non-local correlations among vertices. Few prior approaches capture 3D human pose in real-time (*e.g*., [Mehta *et al*., 2017b; Sun *et al*., 2021]), as most methods use computationally demanding deep neural networks or/and require time-consuming kinematic optimization for post-processing. Our method adopts a lightweight backbone with XFormer block heads, which runs much faster and occupies less memory, making it possible to run in real-time on even mobile devices while still performing on par with state-of-the-art methods.

Human Pose Datasets. 3D pose datasets are often created in controlled environments [Ionescu *et al*., 2013] or in a relatively small scale [Mehta *et al*., 2017a; von Marcard *et al*., 2018]. MoCap datasets [Mahmood *et al*., 2019; Adobe, 2020], on the other hand, provide massive 3D human motion sequences, but no corresponding images are available. Additionally, there are several datasets with pseudo 3D shape and pose labels. Intuitively, using all available datasets of different modalities (images/videos with annotated 2D/3D keypoints, fitted 3D models on unlabeled images, MoCap data without images, *etc*.) could improve performance. To allow the model to train on images with only 2D annotations, previous approaches commonly optimize the reprojection loss of keypoints [Kolotouros *et al*., 2019; Mehta *et al*., 2020]. VIBE further takes advantage of 2D video datasets for training a temporal model and uses MoCap data in the discriminator to force the output pose more temporally coherent. MoCap data is also used to build human motion priors for producing plausible and accurate motions [Rempe *et al*., 2021; Kolotouros *et al*., 2021]. Similar to these methods, our method also aims to adopt data from different modalities. Nevertheless, we directly involve the MoCap data instead of just in a discriminator or prior, therefore fully utilizing the 3D information incorporated in this data.

Transformers. Our cross-modal attention is built on transformers [Vaswani *et al*., 2017]. In the context of 3D pose modeling, transformer-based models are used to lift 2D keypoints to 3D [Li *et al*., 2021; Zhao *et al*., 2022; Li *et al*., 2022; Shan *et al*., 2022], jointly model vertex-vertex and vertex-joint interactions [Lin *et al*., 2021a; Lin *et al*., 2021b], and focus on image regions that are relevant to the pose estimation [Zanfir *et al*., 2021]. These methods usually adopt heavy backbones, and we empirically find that the performance significantly drops with a lightweight backbone or fewer transformer layers. In contrast, we use transformers from a different perspective in that information is transferred across keypoints and image features with cross-modal attention. This enables our model to maintain good performance with a lightweight backbone and a single-layer transformer encoder.

3 Method

Figure 1 summarizes our system. A lightweight feature extraction backbone (Section 3.1) takes the image as input, followed by an image branch (Section 3.1) and a keypoint branch (Section 3.2), both of which predict 3D joints and mesh vertices. The two branches interact with the XFormer blocks structure (Section 3.3, Section 3.4) to exchange information between the keypoint modality and the image modality. Finally, the outputs of two branches can be fused to bring about more precise and stable results (Section 3.5).

3.1 Feature Extraction and Image Branch

As shown in Figure 1, we first feed the person image $I \in \mathbb{R}^{H \times W \times 3}$ into a CNN to get the grid features and a pooled image global feature. Following [Lin *et al*., 2021a; Lin *et al*., 2021b], we tokenize these features together with the 3D coordinates of each mesh vertex and body joint of a coarse template mesh for positional encoding to obtain the image feature F_{img}. F_{img} is then input to the image branch to recover 3D body mesh.

3.2 Keypoint Branch

The grid features extracted from the backbone are shared by the keypoint branch. We then adopt a heatmap-based method to estimate the 2D human pose. Following the common practice [Papandreou *et al*., 2017; Bazarevsky *et al*., 2020], a keypoint decoder inputs the grid features and low-level features from the backbone to predict keypoint heatmaps and offset maps of all 2D body keypoints, where K is the number of body joints. Each heatmap $\mathcal{H}_k$ represents the confidence map of the corresponding keypoint, and the offset map $\mathcal{O}_k$ represents the offset ranging in $[-2, 2]$ to compensate for the lost accuracy since the heatmap width and height are a quarter of the resolution of the input image. The final predicted keypoint coordinates $C \in \mathbb{R}^{K \times 2}$ are calculated by summing the coordinates of the maximum response value in the heatmap and the corresponding offset map value. We then regress the 2D pose to obtain keypoint features using GCNs [Zhao *et al*., 2019]. The final output of the GCN contains K keypoint features, which are then concatenated with their corresponding 2D coordinates. We further use a mean-pooling layer to get the keypoint global feature and concatenate it with vertices and joints from the template mesh as in the image branch. Combining the aforementioned features, we obtain the keypoint branch feature F_{kp} before inputting to the XFormer block.

3.3 XFormer Block

We propose a model based on transformers [Vaswani *et al*., 2017] to encourage the information exchange between keypoints and image branches. Unlike previous transformer-based models [Lin *et al*., 2021b; Lin *et al*., 2021a; Li *et al*., 2021; Li *et al*., 2022], we explicitly capture the attention between each 2D keypoint and the feature of each image location. Moreover, these two modalities are extracted from a shared backbone. Therefore, XFormer can be regarded as a cross-attention mechanism (for multiple modalities) and a self-attention mechanism (for the input image).

An XFormer block contains two types of attention modules, *self-attention* modules, and *cross-modal attention* modules. The self-attention module is a vanilla transformer encoder with multi-head attention that extracts the self-attended features of each branch. The structure of our cross-modal attention module is illustrated in Figure 2. Specifically, the image branch feature F_{img} that represents the visual features of the corresponding image spatial locations, is taken as one input to the cross-modal attention module. The other input is

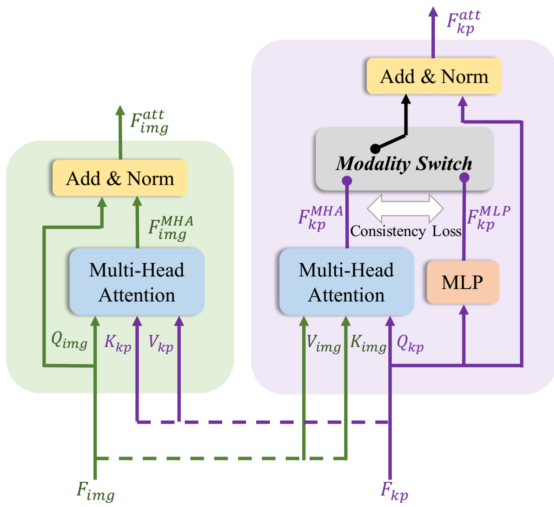

Figure 2: **Architecture of our proposed cross-modal attention module.** We exchange the key-value pairs in the multi-head attention block of two modalities to model the interactions between image spatial features and 2D keypoints. Our cross-modal attention has a "modality plug-in" characteristic such that we can still train our network when the image modality is missing. When the image data is unavailable, we can switch the cross-modal feature F_{kp}^{MHA} to the simulated feature F_{kp}^{MLP} (see the "modality switch" block) without breaking the training data flow.

the keypoint branch feature F_{kp} of 2D keypoints. We first obtain the query, key, and value matrices for each modality (*i.e.*, Q_{img}, K_{img}, V_{img} of the image modality and Q_{kp}, K_{kp}, V_{kp} of the keypoint modality). Then, we exchange key-value pairs in the multi-head attention block of two modalities to get the feature F_{img}^{MHA} and F_{kp}^{MHA}:

$$F_{img}^{MHA} = \text{softmax}\left(Q_{img}K_{kp}^{T}/\sqrt{C_t}\right)V_{kp}, \tag{1}$$

$$F_{kp}^{MHA} = \text{softmax}\left(Q_{kp}K_{img}^{T}/\sqrt{C_t}\right)V_{img}, \tag{2}$$

where C_t is a scaling factor [Vaswani *et al.*, 2017]. With the proposed XFormer block, the cross-modal attention matrix in the image branch provides rich spatial information to guide the network to focus on relevant regions given the keypoint coordinates. Meanwhile, the cross-modal attention matrix in the keypoint branch provides depth and semantic cues embedded in the image that helps regress better human body mesh.

3.4 Modality Switch

To enable training with MoCap data that does not have the image modality, we design a novel modality switch mechanism shown in Figure 2. More specifically, we first input the feature F_{kp} to an MLP ($F_{kp}^{MLP} = \text{MLP}(F_{kp})$). When image data is available, we apply a consistency loss (Eqn. (9)) between F_{kp}^{MLP} and F_{kp}^{MHA} to supervise the MLP simulating the cross-modal attention. When training with MoCap data without images in the keypoint branch, we switch off the cross-modal attention and only train the MLP layer. Thus, the final attended features F_{img}^{att} and F_{kp}^{att} can be written as:

$$F_{img}^{att} = \text{LN}\left(F_{img}^{MHA} + F_{img}\right), \tag{3}$$

$$F_{kp}^{att} = \begin{cases} \text{LN}\left(F_{kp}^{MHA} + F_{kp}\right), & F_{img} \text{ is available,} \\ \text{LN}\left(F_{kp}^{MLP} + F_{kp}\right), & \text{otherwise,} \end{cases} \tag{4}$$

where LN denotes the layer normalization used in the transformer. As a result, the Xformer block does not rely on the presence of both modalities for training. We can drop the image branch and use projected 2D keypoints as input to the keypoint branch to train on MoCap data without paired images, while training on such data is impracticable for existing multi-modal 3D pose estimation approaches like [Mehta *et al.*, 2017b; Mehta *et al.*, 2020; Tekin *et al.*, 2017; Sun *et al.*, 2019].

3.5 Final Ensemble Result

Finally, F_{kp}^{att} and F_{img}^{att} are further passed through self-attention modules to predict human mesh vertices and joints, as shown in Figure 1. The two branches both predict 3D joint locations, a coarse 3D body mesh with 431 vertices, and weak-perspective camera parameters. Similar to [Lin *et al.*, 2021a; Lin *et al.*, 2021b], we upsample the coarse mesh with MLPs to obtain the full SMPL mesh with 6,890 vertices. The estimations of two branches are fused to produce the final joints J^{3D} and vertices V^{3D}: $J^{3D} = \lambda J_{kp}^{3D} + (1-\lambda)J_{img}^{3D}$, $V^{3D} = \lambda V_{kp}^{3D} + (1-\lambda)V_{img}^{3D}$, where λ is simply set to 0.5.

4 Training

4.1 Loss Functions

Our network is end-to-end trained by minimizing a total loss L_{total} consisting of 2D keypoint detector loss L_{map}, keypoint branch loss L_{kp}, image branch loss L_{img}, and the consistency loss L_{cons}.

2D Keypoint Detector Loss. We use a heatmap-based method to predict 2D keypoints in our keypoint branch. For the k-th keypoint, we create its ground-truth heatmap $\bar{\mathcal{H}}_k$ by a Gaussian distribution with mean as the keypoint coordinate and standard variation $\sigma = 2$. Each element of the ground-truth offset map $\bar{\mathcal{O}}_k$ is set to be the offset value w.r.t. the corresponding keypoint location when their distance is less than 2, otherwise, it is set to zero. We minimize the L1 distance of the ground-truth and prediction: $L_{map} = \frac{1}{K}\sum_{i=1}^{K} w_k \left|\left|\bar{\mathcal{H}}_k - \mathcal{H}_k\right|\right|_1 + \frac{1}{K}\sum_{i=1}^{K} w_k \left|\left|\bar{\mathcal{O}}_k - \mathcal{O}_k\right|\right|_1$, where w_k indicates the weight of each keypoint, and it is set to zero for the invisible keypoint.

3D Reconstruction Loss. We optimize our framework by minimizing L_{kp} for the keypoint branch and L_{img} for the image branch. Specifically, we follow [Lin *et al.*, 2021a] and formulate L_{kp} as the sum of vertex loss L_{kp}^{V}, 3D joint loss L_{kp}^{J}, 3D joint regression loss $L_{kp}^{J_{reg}}$, and 2D re-projection $L_{kp}^{J_{proj}}$:

$$L_{kp}^{V} = \frac{1}{M}\sum_{i=1}^{M}\left|\left|V_{kp}^{3D} - \bar{V}^{3D}\right|\right|_1, \tag{5}$$

$$L_{kp}^{J} = \frac{1}{K}\sum_{i=1}^{K}\left|\left|J_{kp}^{3D} - \bar{J}^{3D}\right|\right|_1, \tag{6}$$

$$L_{kp}^{J_{reg}} = \frac{1}{K}\sum_{i=1}^{K}\left|\left|\mathcal{W}V_{kp}^{3D} - \bar{J}^{3D}\right|\right|_1, \tag{7}$$

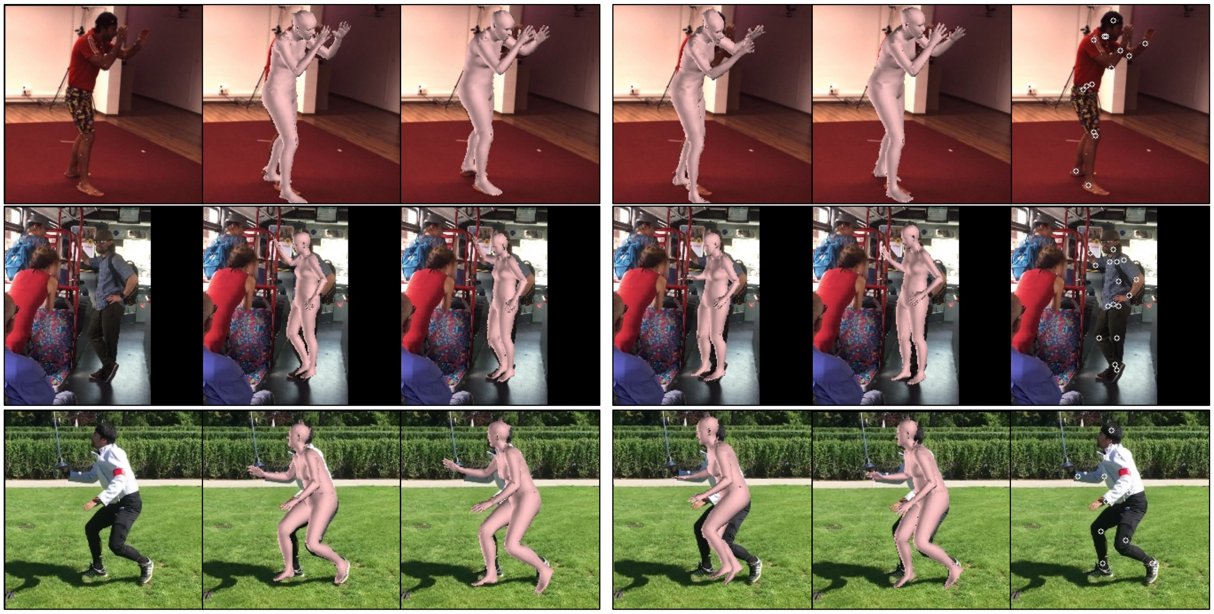

Figure 3: **Visual comparison** of our method against the previous state-of-the-art method Graphormer. XFormer performs slightly better than Graphormer with a large backbone. While with a small backbone, XFormer reconstructs human mesh more accurately than Graphormer. The 2D keypoints predicted by our keypoint decoder are also visualized.

$$L_{kp}^{J_{proj}} = \frac{1}{K}\sum_{i=1}^{K}\left|\left|\Pi_{kp}J_{kp}^{3D} - \bar{J}^{2D}\right|\right|_1, \tag{8}$$

where $\bar{V}^{3D}$, $\bar{J}^{3D}$, $\bar{J}^{2D}$ are the ground-truth 3D mesh vertex locations, the ground-truth 3D joint locations, and the ground-truth 2D keypoint coordinates. M is the number of the vertices, V_{kp}^{3D} denotes the output 3D vertex locations, and J_{kp}^{3D} is the output 3D joint locations. With a pre-trained linear regressor $\mathcal{W}$, the 3D locations of body joints can be inferred from the 3D vertices by $\mathcal{W}V_{kp}^{3D}$ [Loper *et al.*, 2015], and the 3D joint regression loss $L_{kp}^{J_{reg}}$ is their L1 distance to the ground-truth 3D locations. Π_{kp} is the weak-perspective camera parameters predicted by the keypoint branch, which is used to obtain 2D projections of the 3D joints. The image branch loss L_{img} is calculated in a similar fashion as L_{kp}.

Consistency Loss. As discussed in Section 3.3 and shown in Figure 2, we apply a consistency loss L_{cons} to make the MLP inside the cross-modal attention module simulate the cross-modal feature F_{kp}^{MHA}:

$$L_{cons} = \left|\left|F_{kp}^{MHA} - F_{kp}^{MLP}\right|\right|_2. \tag{9}$$

L_{cons} is only used in the keypoint branch when F_{kp}^{MHA} is available (*i.e.*, when training on the datasets with image data). This loss is conceptually similar to the idea of knowledge distillation [Hinton *et al.*, 2015]—we distill the knowledge in the cross-modal attention into an MLP layer to improve the model accuracy and robustness. But unlike knowledge distillation, we do not stop the gradient for F_{kp}^{MHA}, as the keypoint branch that learns from massive MoCap data and 3D labels provides valuable pose/shape priors, which enhances the performance of the image branch.

4.2 Datasets

As discussed in Section 2 and summarized in Table 2, common datasets can be divided into the following categories: 1) Image datasets with 3D annotations, such as 3DPW, UP-3D [Lassner *et al.*, 2017], MuCo-3DHP [Mehta *et al.*, 2018]; 2) Image datasets with 2D keypoints annotations, such as COCO, MPII; 3) Image datasets with 2D keypoints annotations and pseudo 3D human labels, such as SPIN fits on COCO, Pose2Mesh fits on Human3.6M; 4) MoCap datasets without images, such as AMASS. Because of our system's "plug-in" characteristic, we can flexibly use all these kinds of datasets to train our network.

Each training sample is used to minimize corresponding losses according to dataset type. Specifically, we use the 2D keypoint datasets COCO and MPII to train the network by minimizing L_{map}, $L_{kp}^{J_{proj}}$, $L_{img}^{J_{proj}}$ and L_{cons}. As for the image datasets with 3D annotations, they are used to train our whole network by minimizing the total loss L_{total}. For datasets without image data, we first generate 3D joint locations $\bar{J}^{3D}$ and 3D vertices locations $\bar{V}^{3D}$ from the ground-truth SMPL parameters. Then, we obtain 2D keypoints $\bar{J}^{2D}$ with a random orthographic projection. This way, we generate paired 2D keypoints input and ground-truth 3D joints and vertices output. To improve the robustness of the keypoint branch, we employ the following data augmentations: 1) we apply random rotations ($[-30°, 30°]$, $[-30°, 30°]$, $[-60°, 60°]$ for row, pitch, and yaw, respectively) to the global rotation of the mesh to account for more projection variations; 2) we apply random global shifting ($[-20, 20]$ pixels) and scaling ($[0.9, 1.1]$) to the 2D keypoints. As we fully utilize abundant data of different modalities across different domains, our

Method	Backbone	Number of Blocks				Human3.6M		3DPW		
		N_f	N_c	N_b	N_x	MPJPE↓	PA-MPJPE↓	MPJPE↓	PA-MPJPE↓	PVE↓
HMR [Kanazawa *et al.*, 2018]	ResNet50					88.0	56.8	130.0	76.7	-
SPIN [Kolotouros *et al.*, 2019]	ResNet50					-	41.1	96.9	59.2	116.4
VIBE [Kocabas *et al.*, 2020]	ResNet50					65.6	41.4	82.9	51.9	99.1
XNect [Mehta *et al.*, 2020]	SelecSLS					63.6	-	134.2	80.3	-
ROMP [Sun *et al.*, 2021]	HRNet					-	-	76.7	47.3	93.4
Pose2Mesh [Choi *et al.*, 2020]	HRNet					64.9	47.0	89.2	58.9	-
PARE* [Kocabas *et al.*, 2021]	HRNet					54.4	38.2	75.8	48.3	91.2
METRO* [Lin *et al.*, 2021a]	HRNet	4	0	0	3	54.0	36.7	77.1	47.9	88.2
Graphormer* [Lin *et al.*, 2021b]	HRNet	4	0	0	3	53.3	36.1	76.5	46.9	87.9
Ours-Large w/o AMASS*	HRNet	1	1	2	3	53.0	35.7	75.8	46.0	87.5
Ours-Large	HRNet	1	1	2	3	**52.6**	**35.2**	**75.0**	**45.7**	**87.1**
PARE* [Kocabas *et al.*, 2021]	MobileNetV3					72.8	46.6	96.6	58.3	114.4
Graphormer* [Lin *et al.*, 2021b]	MobileNetV3	1	0	0	1	74.6	50.1	92.4	59.0	112.8
Ours-Small w/o AMASS*	MobileNetV3	0	1	0	1	71.1	45.8	87.5	56.8	105.5
Ours-Small	MobileNetV3	0	1	0	1	**68.2**	**44.2**	**84.7**	**55.1**	**102.6**

Table 1: Performance comparison with state-of-the-art methods."-" denotes the results that are not available. Methods with "*" are trained with the same datasets (*i.e.*, the first three types of datasets described in Section 4.2) for a fair comparison.

modal can predict more plausible and accurate body mesh and joint locations. This is different from existing approaches that only use MoCap data to train a discriminator [Kanazawa *et al.*, 2018; Kocabas *et al.*, 2020] or as shape and pose priors for regularization [Bogo *et al.*, 2016].

5 Experiments

5.1 Main Results

Evaluation Protocol. We evaluate on the Human3.6M and 3DPW datasets following the protocols in [Kanazawa *et al.*, 2018; Kolotouros *et al.*, 2019] and report Procrustes-aligned mean per joint position error (PA-MPJPE), mean per joint position error (MPJPE) and per-vertex error (PVE).

Quantitative Evaluation. Table 1 compares our method with the prior works on Human3.6M and 3DPW. We compare these methods with small and large backbones. The original PARE trains with more powerful EFT-fitted [Joo *et al.*, 2021] SMPL parameters on COCO, MPII, LSPET [Johnson and Everingham, 2011] as pseudo 3D labels. To get a fair comparison, we train PARE with the same image datasets as Ours. As for the comparison with Graphormer, we use 3 full XFormer blocks in Ours-Large, which has similar network parameters. The results of Graphormer are reproduced with the official code released by the authors. We report the best performance of Graphormer and Xformer by running the experiments three times, and we find the results of Graphormer are more stochastic (on Human3.6M, Graphormer has a standard deviation of PA-MPJPE of 0.16, while that of Ours-Large is only 0.05). We attribute this to the fact that XFormer benefits from the stable complementary information provided by the keypoint branches, but Graphormer only relies on single modality input. Note that we do not manage to reproduce the results of Graphormer reported in the paper (PA-MPJPE 34.5, MPJPE 51.2 on Human3.6M) in all three experiments. For the methods with a small backbone (*e.g.*, MobileNetV3), Ours-Small outperforms state-of-the-art methods Graphormer and PARE with the same backbone by a clear margin and running at a higher speed, as shown in Table 4.

Dataset Type	Datasets	PA-MPJPE			
Image datasets w/ 3D labels	Human3.6M UP-3D MuCo-3DHP 3DPW	✓	✓	✓	✓
2D keypoint datasets w/o pseudo 3D labels	COCO MPII	✓	✓	✓	✓
2D keypoint datasets w/ pseudo 3D label	SPIN fits Pose2Mesh fits			✓	✓
MoCap datasets w/o images	AMASS		✓		✓
Ours-Small		47.0	46.5	45.8	**44.2**
Ours-Large		36.9	36.4	35.7	**35.2**

Table 2: Ablation study on Human3.6M by varying the types of training datasets used in our method. With the proposed modality switch mechanism, we achieve performance improvement by taking advantage of different types of datasets.

For the methods with a large backbone (*e.g.*, ResNet50, SelecSLS, and HRNet), Ours-Large performs better than the other methods on pose and shape estimation. Note that our model still performs favorably against state-of-the-art methods even if we turn off the modality switch to train without the MoCap dataset (*i.e.*, Ours w/o AMASS), validating the effectiveness of our cross-modal attention. Powered with the ability of training with MoCap data, the performance of XFormer is further enhanced.

Qualitative Evaluation. We conduct qualitative comparison against previous methods, as shown in Figure 3. These visual comparisons verify that our method outperforms previous real-time methods in 3D human mesh recovery and gives comparable results to state-of-the-art offline methods.

5.2 Ablation Study

On Different Training Datasets. As the proposed XFormer can leverage datasets with different annotation types, we compare on different combinations of datasets in Table 2

Method	Human3.6M		3DPW	
	MPJPE	PA-MPJPE	MPJPE	PA-MPJPE
Image Branch Only	74.5	51.3	93.3	60.2
Keypoint Branch Only	72.8	50.5	94.5	61.6
w/o Consistency Loss	70.3	45.1	86.6	55.8
Single Branch with Both Tokens	71.4	49.2	91.2	58.5
Ours-Small (w/o AMASS)	71.1	45.8	87.5	56.8
Keypoint Output	68.8	45.0	85.9	56.3
Image Output	69.0	45.1	86.8	57.0
Ours-Small	**68.2**	**44.2**	**84.7**	**55.1**

Table 3: Ablation study on different settings of our small model. Without our two-branch structure, the models trained with a single branch drop significantly. Our cross-modal attention module effectively models the information exchange of two branches. A single-branch model with both tokens does perform better than individual branches but is still worse than a two-branch model with cross-modal attention. For a better understanding of the contribution of our cross-modal attention mechanism, we also put the results trained without the AMASS dataset. "w/o Consistency Loss" means we remove the consistency loss. Ours Keypoint Output and Image Output indicate that we do not use the ensemble strategy and inference with the full network (with cross-modal attention module) only with the output of the keypoint branch V_{kp}^{3D} or the image branch V_{img}^{3D}, respectively.

Method	GPU Speed (fps)	CPU Speed (fps)
VNect [Mehta *et al.*, 2017b]	30.3	3.1
SPIN [Kolotouros *et al.*, 2019]	126.6	7.9
VIBE [Kocabas *et al.*, 2020]	123.4	8.0
XNect [Mehta *et al.*, 2020]	29.8	2.9
ROMP [Sun *et al.*, 2021]	24.2	3.1
PARE [Kocabas *et al.*, 2021] (MobileNetV3)	103.1	27.6
Graphormer [Lin *et al.*, 2021b] (MobileNetV3)	142.1	32.4
Ours-Small (MobileNetV3)	**154.0**	**37.6**

Table 4: Inference time of state-of-the-art methods and XFormer. The CPU speeds of our methods are tested with a single thread, while VNect and XNect use multiple CPU cores.

to evaluate the effect of different types of datasets. As the first two types of datasets are commonly used in all related works, we always turn them on. It shows the MoCap datasets without images can bring about evident performance gain as the modality switch mechanism enables XFormer to train on such data. Note that state-of-the-art methods all leverage 3D pseudo-labeled datasets, and Table 2 also shows that such datasets benefit XFormer. Plus, dropping both the 3D pseudo labeled and MoCap data, we still outperform the best models trained without these types of data [Kocabas *et al.*, 2020; Mehta *et al.*, 2020].

On Two Branches and Cross-Modal Attention. To validate the effectiveness of our two-branch strategy and the cross-modal attention module, we test several single-branch models, including the model containing only an image branch (Image Branch Only), the model containing only a keypoint branch (Keypoint Branch Only), and the model which takes image and keypoint tokens sequentially as input (Single Branch with Both Tokens). The results are shown in Table 3. Furthermore, the output of each branch from the full two-branch network still works better than the corresponding single-branch network when inferencing. We attribute these observations to two reasons: 1) the cross-modal attention module effectively models the information interactions between the keypoint and image modalities; 2) the additional MoCap datasets improve the generalization of the keypoints branch, resulting in improved full network performance. The ensemble strategy further improves performance.

On Consistency Loss. As shown in Table 3, the performance drops if we remove the consistency loss and do not let the MLP mimic the cross-modal attention feature. The consistency loss can distill the knowledge from cross-modal attention into the MLP layer, which gains performance when training with both the image and MoCap data.

Discussions. We observe that the improvement of our XFormer is more notable for small backbones. We attribute this to that for heavy backbones, both branches have the network capacity of learning fairly good 3D body mesh, and adding cross-modal attention and making use of the MoCap dataset mildly improve the performance. As the model size decreases, small models have lower generalization ability and are more prone to appearance domain gap between limited controlled environment data [Mehta *et al.*, 2017a; von Marcard *et al.*, 2018] and large-scale in-the-wild images, making it hard to predict accurate 3D body shape directly from image features. 2D keypoints, which are easier to estimate thanks to well-established datasets and methods [Cao *et al.*, 2019; Bazarevsky *et al.*, 2020], provide complementary information to boost the small model's performance. This further validates that XFormer is suitable for light backbones in real-time scenarios while existing methods have severely degraded performance when the model capability decreases.

5.3 Running Time Analysis

In Table 4, we profile the running time of our method on a desktop with an Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz and an Nvidia GeForce GTX 1660. All CPU models are accelerated with OpenVino. Our network achieves real-time performance (154.0fps on GPU and 37.6fps on CPU). We observe that our method achieves a good balance between effectiveness and efficiency. Our method gives a comparable reconstruction error and runs much faster compared with most approaches (*e.g.*, [Kolotouros *et al.*, 2019; Kocabas *et al.*, 2020]). We have a similar speed to Graphormer (MobileNetV3) while obtaining much more accurate estimations.

6 Conclusion and Limitations

In this paper, we have described a fast and accurate approach to capturing the 3D human body from monocular RGB images. We utilize all available datasets of different modalities by designing an effective two-branch network to predict 3D body joints and mesh jointly. The information incorporated in these two branches interacts through a novel cross-modal attention module. Experiments have demonstrated that our system runs at more than 30fps on consumer CPU cores while still achieving accurate motion capture performance.

Contribution Statement

X. Han and L. Qian contribute equally. X. Han is the corresponding author.

References

[Adobe, 2020] Adobe. Mixamo. https://www.mixamo.com, 2020. Accessed: 2021-3-8.

[Andriluka *et al.*, 2014] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In *CVPR*, June 2014.

[Andriluka *et al.*, 2018] 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.

[Bazarevsky *et al.*, 2020] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. Blazepose: On-device real-time body pose tracking. *arXiv preprint arXiv:2006.10204*, 2020.

[Bogo *et al.*, 2016] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*, 2016.

[Cao *et al.*, 2019] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *TPAMI*, 2019.

[Choi *et al.*, 2020] 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 *ECCV*, 2020.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[Hinton *et al.*, 2015] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[Howard *et al.*, 2019] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *ICCV*, 2019.

[Ionescu *et al.*, 2013] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *TPAMI*, 2013.

[Johnson and Everingham, 2010] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *BMVC*, 2010.

[Johnson and Everingham, 2011] Sam Johnson and Mark Everingham. Learning effective human pose estimation from inaccurate annotation. In *CVPR*, 2011.

[Joo *et al.*, 2021] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation. In *3DV*, 2021.

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

[Kanazawa *et al.*, 2019] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *CVPR*, 2019.

[Kocabas *et al.*, 2020] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, 2020.

[Kocabas *et al.*, 2021] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. Pare: Part attention regressor for 3d human body estimation. In *ICCV*, 2021.

[Kolotouros *et al.*, 2019] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019.

[Kolotouros *et al.*, 2021] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *ICCV*, 2021.

[Lassner *et al.*, 2017] 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 *CVPR*, 2017.

[Li *et al.*, 2021] Wenhao Li, Hong Liu, Runwei Ding, Mengyuan Liu, and Pichao Wang. Lifting transformer for 3d human pose estimation in video. *arXiv preprint arXiv:2103.14304*, 2021.

[Li *et al.*, 2022] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In *CVPR*, 2022.

[Lin *et al.*, 2014] 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 *ECCV*, 2014.

[Lin *et al.*, 2021a] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *CVPR*, 2021.

[Lin *et al.*, 2021b] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *ICCV*, 2021.

[Loper *et al.*, 2015] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *TOG*, 2015.

[Lu *et al.*, 2019] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019.

[Mahmood *et al.*, 2019] 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.

[Martinez *et al.*, 2017] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, 2017.

[Mehta *et al.*, 2017a] 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.

[Mehta *et al.*, 2017b] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *TOG*, 2017.

[Mehta *et al.*, 2018] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb. In *3DV*, 2018.

[Mehta *et al.*, 2020] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Mohamed Elgharib, Pascal Fua, Hans-Peter Seidel, Helge Rhodin, Gerard Pons-Moll, and Christian Theobalt. Xnect: Real-time multi-person 3d motion capture with a single rgb camera. *TOG*, 2020.

[Papandreou *et al.*, 2017] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, 2017.

[Pavlakos *et al.*, 2018] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3D human pose estimation. In *CVPR*, 2018.

[Rempe *et al.*, 2021] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J Guibas. Humor: 3d human motion model for robust pose estimation. In *ICCV*, 2021.

[Shan *et al.*, 2022] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. P-stmo: Pre-trained spatial temporal many-to-one model for 3d human pose estimation. In *ECCV*, 2022.

[Sun *et al.*, 2019] 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.

[Sun *et al.*, 2021] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *ICCV*, 2021.

[Tekin *et al.*, 2017] Bugra Tekin, Pablo Márquez-Neila, Mathieu Salzmann, and Pascal Fua. Learning to fuse 2d and 3d image cues for monocular body pose estimation. In *ICCV*, 2017.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. NeurIPS*, 2017.

[von Marcard *et al.*, 2018] 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.

[Wang *et al.*, 2018] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018.

[Zanfir *et al.*, 2021] Mihai Zanfir, Andrei Zanfir, Eduard Gabriel Bazavan, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Thundr: Transformer-based 3d human reconstruction with markers. *arXiv preprint arXiv:2106.09336*, 2021.

[Zhang *et al.*, 2013] Weiyu Zhang, Menglong Zhu, and Konstantinos G Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *ICCV*, 2013.

[Zhao *et al.*, 2019] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *CVPR*, 2019.

[Zhao *et al.*, 2022] Weixi Zhao, Yunjie Tian, Qixiang Ye, Jianbin Jiao, and Weiqiang Wang. Graformer: Graph convolution transformer for 3d pose estimation. In *CVPR*, 2022.