2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Keypoints-guided Lightweight Network for Single-view 3D Human Reconstruction

Yuhang Chen[1], Chenxing Wang[1*]

***Abstract*— Single-view 3D human reconstruction has been a hot topic due to the potential of wide applications. To achieve high accuracy, existing works usually take computationally intensive models as backbone for exhaustive underlying features and then directly estimate human mesh vertices. These factors lead to redundant parameters, large calculations and low efficiency, while lightweight solutions to address these challenges are relatively scarce. In this work, based on the problems studied above, we propose a keypoints-guided lightweight network with an encoding-decoding framework. As the input is an image, a lightweight backbone named multi-stage and global feature enhanced network is designed for 2D encoding, where some operations of multi-scale fusion and frequency domain filtering are performed to extract more informative but low-resolution features. As the output is mesh of human body, we construct a keypoints-based 3D human template, with which the 2D low-resolution features can be mapped to 3D space to guide the 3D decoding with high efficiency and high accuracy. Extensive experiments on popular benchmarks 3DPW and Human3.6M illustrate the favorable trade-off between the accuracy and complexity of our method. Our code is publicly available at https://github.com/ChrisChenYh/EfficientHuman.git.**

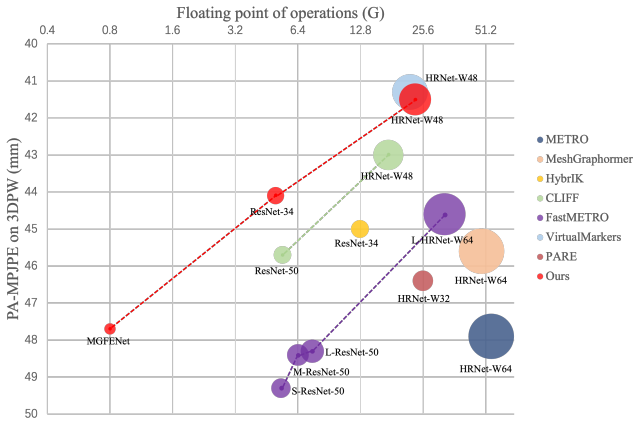

Fig. 1. Comparison of ours and SOTA methods on 3DPW [10] regarding accuracy, computational complexity and model size. The circles of different sizes for each method represent the use of different backbones. In ours, the three circles from small to large respectively represent using MGFENet, ResNet-34, HRNet-w48 as backbone. The circle size is related to model size.

I. Introduction

Single-view 3D human reconstruction models have been widely studied for many years due to their wide applications in AR/VR, behavior understanding and other fields. Although numerous methods [1], [2], [3], [4], [5] have achieved impressive progress on benchmarks, the high computational complexity and large model size make them difficult to be applied in practice.

A typical pipeline for single-view human reconstruction includes two parts, 2D encoding and 3D decoding. As the input is an image, the encoding module is conducted in 2D space, while most methods adopt computationally intensive networks as backbone to extract exhaustive 2D features and potential 3D features, resulting numerous parameters and high computational complexity. There have been some lightweight networks for 2D human pose estimation [6], [7], [8], but they cannot be simply used as the backbone of 3D reconstruction, due to the weak ability of extracting critical underlying features representing geometry structure. Therefore, our motivation is to design an efficient backbone to balance computational complexity, model size and accuracy.

For 3D decoding to output human mesh, existing methods can be divided into two types, model-based and model-free. Most model-based methods reconstruct the human mesh by estimating the parameters of the SMPL model [9], while model-free methods directly estimate the mesh vertices. Since mesh vertices contain complete human pose and shape information, the model-free methods that directly use vertex coordinates as supervision can easily achieve better accuracy. However, too much vertices participating the regression calculations must lead to high computational complexity and numerous parameters. In contrast, model-based methods are usually simple and lightweight, but it is difficult to directly estimate SMPL parameters from a monocular image without explicit supervision for human body. Therefore, our motivation is to combine the advantages of these two types of methods for a most optimized trade-off between accuracy and efficiency.

In this work, we propose a keypoints-guided lightweight network for 3D human reconstruction. We firstly design a lightweight backbone based on MobileNetv3 [11], named Multi-stage and Global Features Enhanced Network (MGFENet), in order to extract low-resolution but informative features. These 2D low-resolution features are further mapped into 3D features to predict 3D keypoints presenting the human pose and shape. To guide the feature mapping, we design a keypoints-based 3D human template; further, to enhance the accuracy of 3D keypoints, we build a feature extractor using gated attention [12] to encode the local relationship of 3D features. The prediction of 3D keypoints are implemented with a coordinates classifier to conduct three classification tasks for estimating the vertical, horizontal, and

*Corresponding Author
[1]School of Automation, Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Southeast University, China, {chenyuhang, cxwang}@seu.edu.cn

depth coordinates independently, under the supervision of a residual log-likelihood estimation loss [13]. Finally, the 3D keypoints are used to guide the estimation of SMPL parameters for human mesh. We conduct a comprehensive comparison between the state-of-the-art (SOTA) methods [1], [2], [3], [4], [5], [14], [15], and our MGFENet, trained on the public 3DPW dataset [10], as illustrated in Fig. 1. Some methods are trained using different size of ResNet [16] or HRNet [17] as backbone, represented by varying circle sizes. The results indicate that our method achieves the best balance in terms of accuracy, complexity, and model size. Furthermore, it can maintain the same level of accuracy as the best method when both employ a large backbone.

The contribution of our method can be summarized as:

- We propose a lightweight model for 3D human reconstruction that can achieve competitive accuracy while saving 92.6% FLOPs and 51.8% parameters compared to the SOTA method FastMETRO [2] with backbone ResNet-50, as illustrated in Fig. 1.
- We design a lightweight backbone that can achieve better performance than ResNet-18 with only 33.6% FLOPs and 41.7% parameters, as listed in Table IV.
- We propose a lightweight 3D keypoints estimation network for accurate mesh reconstruction, where the novel keypoints-based 3D human template is also proposed first.

II. Related Work

Human mesh estimation. Methods for estimating 3D human mesh models from monocular RGB image can be divided into two types: model-based and model-free.

For model-based methods [14], [15], [18], the typical SMPL model [9] is used as a prior template, and some parameters are predicted from an input image and then used to drive the SMPL model to reconstruct human mesh. This parametric model can be lightweight but lose accuracy sometimes due to the neglect of spatial correlation. To make up for this shortcoming, several methods [5], [19] introduce human body structure in estimating SMPL parameters. Li et al. [5] take 3D skeleton as the structure priors to estimate the parameters of joints rotation more accurately. However, shape priors are still lacking with only the skeleton.

For model-free methods [2], [3], [4], [20] mesh vertices are estimated directly by the network trained with data of pseudo ground-truth human mesh generated using SMPL. Obviously, directly estimating a bulk of vertices is uncontrollable and can result in unconvincing meshes, with extremely high computational complexity. Ma et al. [1] propose a model-free method based on virtual markers, which reduces the calculation to a certain extent but is still far from the requirement of applications.

In this paper, to combine the advantages of both model-based and model-free methods, we propose an efficient approach for 3D human mesh estimation, with the design of a keypoints-based 3D human template.

Human keypoints estimation. Previous studies for estimating 3D keypoints from a monocular image usually employ single-stage or two-stage flowchart.

Single-stage methods can be divided into heatmap-based methods or regression directly. Heatmap-based methods render 3D joints as high-resolution maps with Gaussian probability distribution for supervised learning. This type of methods can achieve satisfactory accuracy, but they require a large number of calculations and large storage memory. Li et al. [21] decouple 2D heatmaps into 1D representations, reducing computation and memory while maintaining effects. Expanding this for 3D application is worth considering. Regression-based methods directly supervise the model to learn a small number of coordinates. Although they are lightweight, the performance of them is not good. Li et al. [13] propose an adaptive loss to learn the potential distribution of the output, which improves the performance comparable to the heatmap-based methods and so worth referring to.

Two-stage methods first estimate 2D joints and then lift them to 3D space. Recently, many lifting-based methods [22], [23], [24] use transformer or self-attention to effectively learn the structural features of human joints, which increases the accuracy further.

Combining the advantages of above methods, we propose a lightweight and accurate 3D keypoints estimation network.

Lightweight networks. Making a network lightweight has been a critical topic in many fields. In terms of convolution-based methods, a depth separable convolution is widely used for lightweight CNN vision tasks, such as MobileNets [11], [25], ShuffleNets [26], [27], etc. A disadvantage of these methods is that they only focus on local features. Then the transformer module [28] is introduced and several lightweight operators appear in order to enhance the attention to global features, such as the gated attention unit [12] and the global filter [29]. These operators are valuable to enhance the performance for a lightweight model and so considered in this paper.

However, the lightweight network for 3D human reconstruction is still rare. In this paper, we integrate the popular efficient ideas to design a lightweight backbone as well as some efficient modules to balance the accuracy and the calculations.

III. Method

Fig. 2 illustrates the overall architecture of our model. With a RGB image as input, we aim to infer SMPL parameters and generate 3D human mesh effectively through three parts. For image encoding, we leverage an efficient backbone pretrained on MSCOCO [30] to extract features from monocular image. For 3D keypoints estimation, we map image features to keypoints features learned with gated attention, and predict the 3D coordinates of keypoints by a designed coordinates classifier. For 3D mesh estimation, we apply multi-layer perceptron (MLP) and inverse kinematics solution (IK) to estimate the parameters that drive SMPL to generate human body mesh.

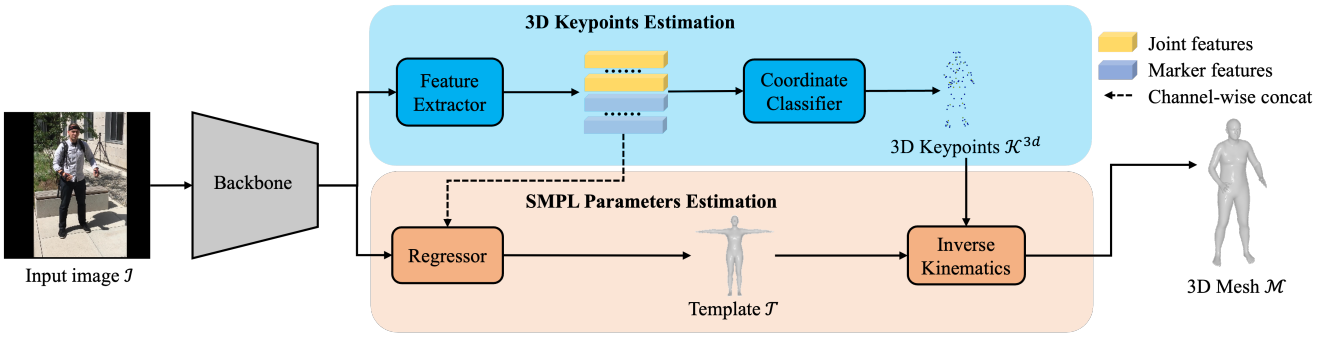

Fig. 2. Overview of our model architecture. Given an input image, it first estimates the 3D coordinates of the keypoints. Then, guided by the keypoints features, it estimates the SMPL template with corresponding shape. Finally, taking the template and 3D keypoints as input, it recovers the complete human mesh using inverse kinematics solutions.

A. *Lightweight backbone design and pretrain*

Through preliminary study, we find that quite a quantity of model parameters come from the backbone, which motivates us to design a lightweight backbone firstly.

Design motivation and choices. A common backbone network adopts ordinary convolutions (Conv) to ensure the extraction of generalized and diverse features, but this needs intensive calculations. Some lightweight backbone networks are proposed, with MobileNetv2 [25] the most classical and commonly used one. It combines the point-wise convolution (PW-Conv) and depth-wise separable convolution (DW-Conv) jointly to replace the ordinary convolution. This operation reduces computation and parameters, which also reduces performance. Therefore, based on MobileNetv3 [11] and inspired by DenseNet [31], we design a multi-stage feature concatenation structure. Features from different scales of multiple stages are concatenated successively, where the combination of PW-Conv and DW-Conv is still applied for feature extraction. To enhance the global feature extraction, we also add a global filter [29] in each stage. All these operations ensure the extracted features can maintain low-resolution and diverse information. The specific comparison of prevalent operators mentioned above is listed in Table I, illustrating the calculation superiority of the DW-Conv and PW-Conv as well as the global filter, where h, w are the size of input image features and c is the number of feature channels.

TABLE I
COMPARISONS OF PREVALENT OPERATIONS IN DEEP MODELS.

Method	FLOPs	#Param
Ordinary convolution	$\mathcal{O}(k^2hwc^2)$	k^2c^2
Depth-wise convolution	$\mathcal{O}(k^2hwc)$	k^2c
Point-wise convolution	$\mathcal{O}(hwc^2)$	c^2
Self-attention	$\mathcal{O}(hwc^2 + h^2w^2c)$	$4c^2$
Global filter	$\mathcal{O}(hwc[\log_2(hw)] + hwc)$	hwc

Backbone structure. Fig. 3 displays the framework of our MGFENet, which contains four stages. Features from every stage will be cascaded for feature extraction of next stage. There are several MGFE blocks designed in each stage, where the number is set the same to MobileNetv3.

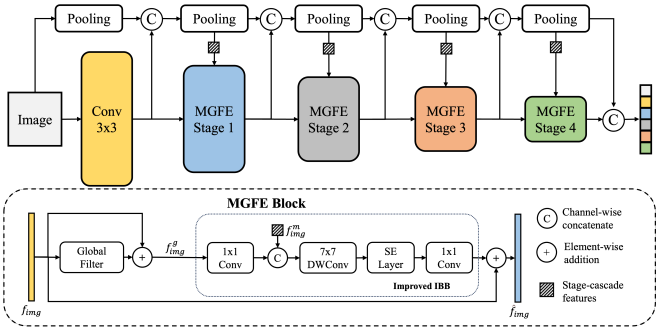

Fig. 3. MGFENet framework.

In each MGFE block, a global filter is added at first to process the input feature $f_{\text{img}} \in \mathbb{R}^{h\times w\times c_1}$ as:

$$f^g_{\text{img}} = \mathcal{F}^{-1}\left[\boldsymbol{K} \odot \boldsymbol{F}_{\text{img}}\right] + f_{\text{img}}, \tag{1}$$

where $\boldsymbol{K} \in \mathbb{C}^{h\times w\times c_1}$ is a learnable spectral filter, $\boldsymbol{F}_{\text{img}}$ is the spectra of f_{img} by Fourier transform, $\odot$ denotes element-wise multiplication, $\mathcal{F}^{-1}(\cdot)$ is the inverse Fourier transform. The output of the global filter is added to f_{img} to emphasize the global features with few parameters. Then, an improved inverted bottleneck block (IBB) is used to extract diverse local features, which improves the ordinary IBB [25] by concatenating each stage-cascade feature $f^m_{\text{img}} \in \mathbb{R}^{h\times w\times c_2}$ into IBB and using 7×7 larger kernels to expand the receptive field for more informative local features. The output features from the improved IBB are added to f_{img} to form the final output $\hat{f}_{\text{img}} \in \mathbb{R}^{h\times w\times c_2}$ for a MGFE block.

For each stage, the output feature $\tilde{f}_{\text{img}}$ is also aggregated into f^m_{img} to obtain an updated stage-cascade feature $f^{m+1}_{\text{img}} = \left[f^m_{\text{img}}, \tilde{f}_{\text{img}}\right] \in \mathbb{R}^{h\times w\times(c_1+c_2)}$.

Pretrain. Most 3D human pose estimation models are pretrained on ImageNet [32] that is for classification tasks and involves a wide variety of objects, making it challenging for the regression of 3D human keypoints. In this paper, we first try to pretrain our backbone on the MSCOCO 2D human pose dataset [30]. An encoder-decoder network is employed for pretraining, where our backbone serves as the encoder and some deconvolutions as the decoder. The task of our pretrain is to progressively predict the coordinates of 2D keypoints, while only the extracted features from our

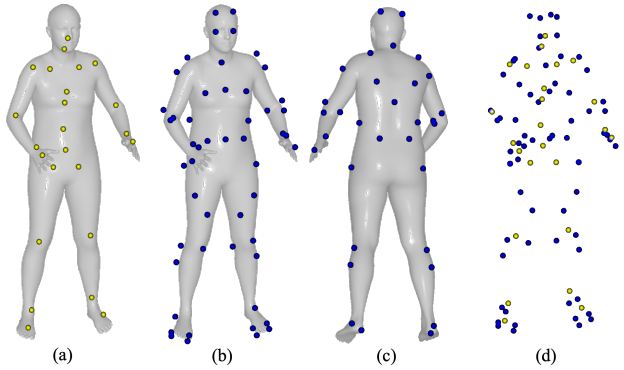

Fig. 4. The 3D human keypoints representations. (a) Front view of human mesh and joints (yellow). (b) Front view of Human mesh and marker points (blue). (c) Back view of human mesh and marker points (blue). (d) Front view of 3D joints (yellow) and marker points (blue).

backbone are taken into use in the following human mesh estimation.

B. Efficient human mesh estimation network

With the features from backbone, we propose to obtain the 3D keypoints firstly to guide estimating the SMPL parameters, then we generate human mesh with these parameters.

Keypoints-based 3D human template. To guide the prediction of 3D keypoints, we first propose to design a keypoints-based 3D human template. Before, the skeleton of 3D human joints has been taken as a template, which only contains 24 joints [5], as shown in Fig. 4(a). This template primarily describes the human pose information and lacks of shape information. The SMPL human mesh is another type of template containing 6890 mesh vertices, which can describe the human body shape better [3]. However, 6890 points for model inference may bring large calculations. To pick out the representative points, Loper et al. [33] evaluate 67 marker points that are very useful for estimating human mesh, as displayed in Fig. 4(b) and (c), where 47 points are determined when using a motion capture system and another 20 points are selected after using the greedy method. In this paper, we propose a 91 keypoints based 3D human template combing the 24 joints and 67 mesh vertices, as shown in Fig. 4(d). These points are combined to form the set of 3D keypoints to guide the down-stream tasks.

Efficient 3D keypoints estimation module. As mentioned above, the features from our backbone need to be mapped to predict the features of 91 3D keypoints firstly. Inspired by the 2D keypoints work SimCC [21], we regarding this task as the classification task of 3D coordinates. The proposed efficient 3D keypoints estimation module (3DKEM) is shown in Fig. 5, which consists of three stages: feature mapping, feature extraction, and coordinate classification.

In the stage of feature mapping, the features from backbone, denoted as $\bar{f}_{img} \in \mathbb{R}^{\left(h' \times w'\right) \times c}$, are mapped to form the features of 91 keypoints as:

$$f_{kp} = A\bar{f}_{\text{img}}, \tag{2}$$

where $A \in \mathbb{R}^{(K+N) \times \left(h' \times w'\right)}$ is a learnable parameter matrix.

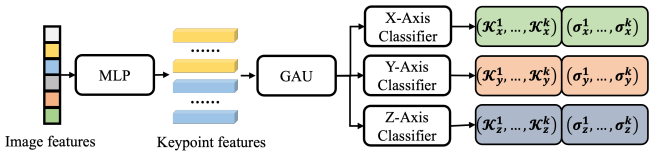

Fig. 5. 3D keypoints estimation module.

In the stage of feature extraction, the features of 91 keypoints are processed further by the gated attention unit (GAU) [29] to extract the global features. In contrast to conventional self-attention mechanism, the GAU employs gated units to refine attention weights, which can effectively reduce the impact of irrelevant information. Taking f_{kp} as input, the process of GAU is similar to the ordinary attention computation as follows:

$$U = \phi_u\left(f_{kp}W_u\right), V = \phi_v\left(f_{kp}W_v\right), Z = \phi_z\left(f_{kp}W_z\right), \tag{3}$$

where, $W_u, W_v, W_z \in \mathbb{R}^{c \times s}$ are all learnable matrices, ϕ is activation function where $SiLU$ is selected. The weakened attention mechanism formulas are:

$$\begin{aligned} B &= \mathrm{ReLU}^2\left(Q(Z)K(Z)^T + b\right), \\ \hat{V} &= BV, f'_{kp} = (U \odot \hat{V})W_o, \end{aligned} \tag{4}$$

where B is the attention weight matrix, $Q(\cdot)$ and $K(\cdot)$ are both linear transformations, b is a learnable bias, $W_o \in \mathbb{R}^{s \times c}$ is also a learnable matrix.

Finally, in the stage of coordinate classification, three groups of classifiers are used to estimate the coordinates as well as the degree of deviation between the output and the statistical distribution.

SMPL parameters estimation module. As well known, the SMPL model is effective to generate human mesh if the accuracy of the required parameters is ensured, where the parameters refer to the joints rotation angle $\theta \in \mathbb{R}^{24 \times 3}$ and shape $\beta \in \mathbb{R}^{10}$ [9]. The features from our 3DKEM are precisely used to guide the accurate estimation of these parameters. Referring to HybrIK [5], we calculate θ as:

$$\theta = IK(\mathcal{J}, T, \varphi), \tag{5}$$

where $\mathcal{J}$ denotes the 3D joints estimated by our 3DKEM, T is the initial template of 3D joints, φ is the joint twist angle predicted directly from image features, and $IK(\cdot)$ is the inverse kinematics equation. To estimate β, we employ a multi-layer perceptron (MLP):

$$\beta = MLP(f_{shape}), \tag{6}$$

where $f_{\text{shape}} = \left[f'_{kp}, \bar{f}_{\text{img}}\right] \in \mathbb{R}^{2c}$, which means a combination of f'_{kp} and $\bar{f}_{\text{img}}$, and f'_{kp} is first processed by a global average pooling to extract global keypoints features. Finally, the obtained θ and β are used to drive the SMPL model to generate human mesh with usual operation.

C. Loss design

For supervised training, the loss function is designed for estimating the 3D keypoints and SMPL parameters separately.

3D keypoints loss. Existing regression-based methods usually use L1 loss or L2 loss for supervision, which are only suitable for the assumption that the keypoint errors obey Laplace distribution or Gaussian distribution. However, the real human body keypoints error distribution does not strictly conform to these distributions. Therefore, we adopt residual log-likelihood estimation loss (RLE Loss) [13] to supervise the 3D keypoints estimation, which is described as:

$$\mathcal{L}_{3D} = \sum_{i=1}^{K+N} \left[-\log Q\left(\mu_i\right) - \log G_\phi\left(\mu_i\right) + \log \sigma_i \right], \quad (7)$$

where G_ϕ indicates the true error probability distribution predicted by a flow model [34], Q indicates the probability distribution, θ_i indicates the degree of deviation between the i^{th} keypoint and the statistical distribution, μ_i indicates the deviation between the prediction $x_i \in \mathbb{R}^3$ and ground-truth $\hat{x}_i \in \mathbb{R}^3$ of i^{th} keypoint $\mu_i = (x_i - \hat{x}_i)/\sigma_i$. In Eq. (7), we take the errors of all the 91 keypoints in the training stage as samples and learn the potential error probability distribution on the basis of the prior distribution using the flow model. Due to the hybrid training with 2D and 3D dataset, we set up two flow models to estimate the probability distribution of 2D data and 3D data respectively.

SMPL parameters loss. We use L2 loss to supervise the SMPL parameters, i.e., θ and β, and joint twist angle φ_i:

$$\begin{gathered} \mathcal{L}_\theta = \|\theta - \hat{\theta}\|_2, \mathcal{L}_\beta = \|\beta - \hat{\beta}\|_2, \\ \mathcal{L}_\varphi = \frac{1}{K} \sum_{i=1}^{K} \left\| (\cos\varphi_i, \sin\varphi_i) - (\cos\hat{\varphi}_i, \sin\hat{\varphi}_i) \right\|_2, \end{gathered} \quad (8)$$

where $\hat{\theta}$, $\hat{\beta}$, $\hat{\varphi}_i$ are the ground-truths of θ, β and φ_i, respectively.

In summary, the overall loss of training the 3D human mesh estimation network is:

$$\mathcal{L} = \varepsilon_1 \mathcal{L}_{3D} + \varepsilon_2 \mathcal{L}_\theta + \varepsilon_3 \mathcal{L}_\beta + \varepsilon_4 \mathcal{L}_\varphi, \quad (9)$$

where ε_1, ε_2, ε_3, and ε_4 are hyperparameters.

IV. Experiments

A. Datasets and metrics

Human3.6M [35] is an indoor 3D human dataset with 3D joints annotations but without SMPL annotations. So, we use the SMPL pseudo annotations generated by SMPLify-X [36] for the 3D mesh supervision. We use objects S1, S5, S6, S7, and S8 for training, and objects S9, S11 for testing.

3DPW [10] is an outdoor 3D human dataset with 3D joints annotations and SMPL annotations, and so is used for training and testing.

MPI-INF-3DHP [37] contains indoor scenes and complex outdoor scenes, with 3D joints annotations but without SMPL parameter annotations, which is used for training only.

MSCOCO [30] is a 2D dataset of large-scale outdoor scenes, which is used for training.

The network performances are evaluated using PA-MPJPE / MPVPE as the errors of 3D joint / vertex position, and AP as the errors of 2D joint. Additionally, we measure the computational complexity with FLOPs(G) and parameter quantity with #Param(M).

B. Implementation details

We train the proposed network in two stages: 2D pretraining and 3D training. For 2D pretraining, we use the AdamW optimizer with batch size of 256. All backbones in our experiments are trained for 210 epochs. The initial learning rate is 4×10^{-3}, and from the 105^{th} epoch, we use cosine annealing algorithm to gradually reduce the learning rate to 2×10^{-4}. For 3D training, we use the Adam optimizer with batch size of 64. All networks in our experiments are trained for 30 epochs. The learning rate is 1×10^{-4}. The hyperparameters in Eq. (9) are: $\varepsilon_1 = 1$, $\varepsilon_2 = 0.01$, $\varepsilon_3 = 0.1$, and $\varepsilon_4 = 0.01$.

C. Comparison to state-of-the-art methods

As shown in Table II, we compare our method to the SOTA methods on popular benchmarks 3DPW and Human3.6M from the perspective of joints error (PA-MPJPE), computational complexity (FLOPs), parameters amount (#Param), respectively. Since previous works use ResNet or HRNet as backbone, we also train our approach on these two backbones, and compare them with the SOTA methods. Some qualitative results from wild images are shown in Fig. 6.

When we use the same backbone as the SOTA methods, our method can achieve comparable accuracy to them, but with lower computational complexity and smaller model size. For example, when we use ResNet-34 as the backbone, compared with HybrIK that also uses ResNet-34 as the backbone, our model achieves lower PA-MPJPE (-0.9mm) by only costing 39.4% FLOPs of HybrIK.

When employing our MGFENet as a lightweight backbone, our method demonstrates a notable reduction in both calculations and parameters, while maintaining competitive accuracy compared to SOTA methods. Compared with FastMETRO that also has light scale of backbone, our method achieves the similar accuracy but a 92.6% reduction in FLOPs and a 51.8% reduction in #Param.

Fig. 6. Qualitative results from wild images.

D. Ablation studies

Efficient backbone. We use MobileNetv3 as the baseline, which adopts the inverted residual module to encode image

TABLE II

COMPARISON TO THE STATE-OF-THE-ART METHODS ON 3DPW AND HUMAN3.6M.

Method	Backbone	FLOPs(G)	#Params(M)	3DPW		Human3.6M	
				PA-MPJPE↓	MPVPE↓	PA-MPJPE↓	MPVPE↓
PARE [14]	HRNet-W32	25.5	36.0	46.4	88.6	-	-
METRO [3]	HRNet-W64	54.3	195.4	47.9	88.2	36.7	-
Mesh Graphormer [4]	HRNet-W64	48.8	180.6	45.6	87.7	34.5	-
HybrIK [5]	ResNet-34	12.7	27.6	45.0	86.5	34.5	65.7
CLIFF [15]	ResNet-50	5.4	27.0	45.7	85.3	35.1	-
	HRNet-W48	17.4	78.9	43.0	81.2	32.7	-
FastMETRO [2]	S-ResNet-50	5.3	32.7	49.3	91.9	39.4	-
	M-ResNet-50	6.4	40.6	48.4	91.2	38.6	-
	L-ResNet-50	7.5	48.4	48.3	90.6	37.3	-
	L-HRNet-W64	32.4	153.0	44.6	84.1	33.7	-
VirtualMarkers [1]	HRNet-W48	22.1	109.6	**41.3**	**77.9**	32.0	**58.0**
Ours	MGFENet	**0.8**	**10.9**	47.7	90.3	38.3	72.4
	ResNet-34	5.0	26.0	44.1	86.2	34.0	64.9
	HRNet-W48	23.4	89.0	41.5	80.3	**31.5**	59.1

features. Our goal is to enable the proposed backbone to achieve performance comparable to ResNet-18 with lower complexity and less parameters. With the same hyperparameters and training schedules, we improve the inverted residual module step by step: (1) replace the 3×3 convolution kernels and 5×5 kernels with larger 7×7 kernels; (2) add multi-stage feature concatenation (MFC); (3) add a global filter. We compare the performance of all models on the MSCOCO benchmark, using AP as the evaluation index, shown as Table III. Also, we count the computational complexity of each step to ensure that all improvements are efficient. As shown in Table IV, we finally achieve better performance (+1.1 AP) than ResNet-18 with only 33.6% FLOPs and 41.7% #Param.

TABLE III

ABLATION STUDIES OF MGFENET.

7×7Conv	MFC	Global Filter	AP↑	FLOPs(G)
−	−	−	60.3	**0.34**
+	−	−	61.1	0.41
+	+	−	62.6	0.45
+	+	+	**64.1**	0.46

TABLE IV

COMPARISON OF MGFENET WITH MOBILENETV3 AND RESNET-18.

Model	AP↑	FLOPs(G)	#Param(M)
MobileNetv3	60.3	**0.34**	**2.63**
ResNet-18	63.2	1.37	11.18
MGFENet	**64.1**	0.46	4.66

Efficient 3D keypoints estimation module. As the baseline, we directly regress the 3D joints from 2D image features and generate the human mesh under the guidance of the joints. With the same backbone (ResNet-34), hyperparameters and training schedules, we improve the baseline above by four steps: (1) replace L1 loss with RLE loss to supervise 3D keypoints; (2) replace the fully connect layer in the baseline with the proposed efficient 3DKEM; (3) pretrain the model on MSCOCO instead of ImageNet; (4) use the proposed keypoints-based 3D human template as the guidance of human mesh estimation instead of using 3D joints. We compare the performance of all models on the 3DPW benchmark, using PA-MPJPE as the evaluation index, shown as Table V. Also, we count the computational complexity of each model to ensure that all steps are efficient. As the result, PA-MPJPE of our method is lowered by 6.1mm at the expense of 0.2G FLOPs, which only accounts for 4.1% of the baseline.

TABLE V

ABLATION STUDIES OF 3D KEYPOINTS ESTIMATION MODULE.

RLE	3DKEM	Pretrain	Keypoints	PA-MPJPE↓	FLOPs(G)
−	−	−	−	50.2	**4.8**
+	−	−	−	48.1	4.8
+	+	−	−	46.7	4.9
+	+	+	−	44.6	4.9
+	+	+	+	**44.1**	5.0

V. CONCLUSION

In this work, we present a keypoints-guided lightweight network for 3D human reconstruction with satisfactory efficiency and accuracy. In the image encoding stage, we design a lightweight backbone that outperforms ResNet-18 with lower complexity and smaller model size. In the 3D mesh estimation stage, we design a simple yet effective keypoints-based 3D human template and propose an efficient 3D keypoints estimation network. The estimated keypoints can help the network to learn more structural features of the human body, thereby obtaining a more accurate mesh. Through extensive experiments on popular benchmarks 3DPW and Human3.6M, our proposed method exhibits favorable accuracy / complexity trade-offs. In the future work, we plan to extend this work to explore lightweight methods about human-object interactions, 3D virtual try-on, etc.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (61828501), and the science and technology project fundings of State Grid Jiangsu Electric Power Co., Ltd. (J2023031).

REFERENCES

[1] X. Ma, J. Su, C. Wang, W. Zhu, and Y. Wang, "3d human mesh estimation from virtual markers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 534–543, 2023.

[2] J. Cho, K. Youwang, and T.-H. Oh, "Cross-attention of disentangled modalities for 3d human mesh recovery with transformers," in *Proceedings of the European Conference on Computer Vision*, pp. 342–359, Springer, 2022.

[3] K. Lin, L. Wang, and Z. Liu, "End-to-end human pose and mesh reconstruction with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1954–1963, 2021.

[4] K. Lin, L. Wang, and Z. Liu, "Mesh graphormer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 12939–12948, 2021.

[5] J. Li, C. Xu, Z. Chen, S. Bian, L. Yang, and C. Lu, "Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3383–3393, 2021.

[6] C. Yu, B. Xiao, C. Gao, L. Yuan, L. Zhang, N. Sang, and J. Wang, "Lite-hrnet: A lightweight high-resolution network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10440–10450, 2021.

[7] Q. Li, Z. Zhang, F. Xiao, F. Zhang, and B. Bhanu, "Dite-hrnet: Dynamic lightweight high-resolution network for human pose estimation," pp. 1095–1101, 2022.

[8] Y. Wang, M. Li, H. Cai, W.-M. Chen, and S. Han, "Lite pose: Efficient architecture design for 2d human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13126–13136, 2022.

[9] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM Transactions on Graphics*, vol. 34, no. 6, 2015.

[10] T. Von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "Recovering accurate 3d human pose in the wild using imus and a moving camera," in *Proceedings of the European Conference on Computer Vision*, pp. 601–617, 2018.

[11] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1314–1324, 2019.

[12] W. Hua, Z. Dai, H. Liu, and Q. Le, "Transformer quality in linear time," in *International Conference on Machine Learning*, pp. 9099–9117, PMLR, 2022.

[13] J. Li, S. Bian, A. Zeng, C. Wang, B. Pang, W. Liu, and C. Lu, "Human pose regression with residual log-likelihood estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11025–11034, 2021.

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

[15] Z. Li, J. Liu, Z. Zhang, S. Xu, and Y. Yan, "Cliff: Carrying location information in full frames into human pose and shape estimation," in *Proceedings of the European Conference on Computer Vision*, pp. 590–606, Springer, 2022.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[17] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5693–5703, 2019.

[18] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7122–7131, 2018.

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

[20] N. Kolotouros, G. Pavlakos, and K. Daniilidis, "Convolutional mesh regression for single-image human shape reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4501–4510, 2019.

[21] Y. Li, S. Yang, P. Liu, S. Zhang, Y. Wang, Z. Wang, W. Yang, and S.-T. Xia, "Simcc: A simple coordinate classification perspective for human pose estimation," in *Proceedings of the European Conference on Computer Vision*, pp. 89–106, Springer, 2022.

[22] J. Zhang, Z. Tu, J. Yang, Y. Chen, and J. Yuan, "Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13232–13242, 2022.

[23] H. Ma, Z. Wang, Y. Chen, D. Kong, L. Chen, X. Liu, X. Yan, H. Tang, and X. Xie, "Ppt: token-pruned pose transformer for monocular and multi-view human pose estimation," in *Proceedings of the European Conference on Computer Vision*, pp. 424–442, Springer, 2022.

[24] W. Li, H. Liu, R. Ding, M. Liu, P. Wang, and W. Yang, "Exploiting temporal contexts with strided transformer for 3d human pose estimation," *IEEE Transactions on Multimedia*, vol. 25, pp. 1282–1293, 2022.

[25] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.

[26] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European Conference on Computer Vision*, pp. 116–131, 2018.

[27] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, 2018.

[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[29] Y. Rao, W. Zhao, Z. Zhu, J. Lu, and J. Zhou, "Global filter networks for image classification," *Advances in Neural Information Processing Systems*, vol. 34, pp. 980–993, 2021.

[30] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of the European Conference on Computer Vision*, pp. 740–755, Springer, 2014.

[31] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, 2017.

[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[33] M. Loper, N. Mahmood, and M. J. Black, "Mosh: motion and shape capture from sparse markers.," *ACM Transactions on Graphics*, vol. 33, no. 6, pp. 220–1, 2014.

[34] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," in *International Conference on Learning Representations*, 2016.

[35] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1325–1339, 2013.

[36] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. 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*, pp. 10975–10985, 2019.

[37] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, "Monocular 3d human pose estimation in the wild using improved cnn supervision," in *2017 International Conference on 3D Vision*, pp. 506–516, IEEE, 2017.