# HumanMeshNet: Polygonal Mesh Recovery of Humans

Abbhinav Venkat
abbhinav.venkat@research.iiit.ac.in

Chaitanya Patel
chaitanya.patel@students.iiit.ac.in

Yudhik Agrawal
yudhik.agrawal@research.iiit.ac.in

Avinash Sharma
asharma@iiit.ac.in

International Institute of Information Technology, Hyderabad

## Abstract

*3D Human Body Reconstruction from a monocular image is an important problem in computer vision with applications in virtual and augmented reality platforms, animation industry, en-commerce domain, etc. While several of the existing works formulate it as a volumetric or parametric learning with complex and indirect reliance on reprojections of the mesh, we would like to focus on implicitly learning the mesh representation. To that end, we propose a novel model, HumanMeshNet, that regresses a template mesh's vertices, as well as receives a regularization by the 3D skeletal locations in a multi-branch, multi-task setup. The image to mesh vertex regression is further regularized by the neighborhood constraint imposed by mesh topology ensuring smooth surface reconstruction. The proposed paradigm can theoretically learn local surface deformations induced by body shape variations and can therefore learn high-resolution meshes going ahead. We show comparable performance with SoA (in terms of surface and joint error) with far lesser computational complexity, modeling cost and therefore real-time reconstructions on three publicly available datasets. We also show the generalizability of the proposed paradigm for a similar task of predicting hand mesh models. Given these initial results, we would like to exploit the mesh topology in an explicit manner going ahead.*

## 1. Introduction

Recovering a 3D human body shape from a monocular image is an ill-posed problem in computer vision with great practical importance for many applications, including virtual and augmented reality platforms, animation industry, e-commerce domain, etc. Some of the recent deep learning methods employ volumetric regression to recover the voxel grid reconstruction of human body models from a monocular image [31, 29]. Although volumetric regression enables recovering a more accurate surface reconstruction, they do so without an animatable skeleton [31], which limits their applicability for some of the aforementioned applications. [29] attempted to overcome this limitation by fitting a parametric body model on the volumetric reconstruction using a sillhoute reprojection loss. Nevertheless, in general, such methods yield reconstructions of low resolution at higher computational cost (regression over the cubic voxel grid) and often suffer from broken or partial body parts.

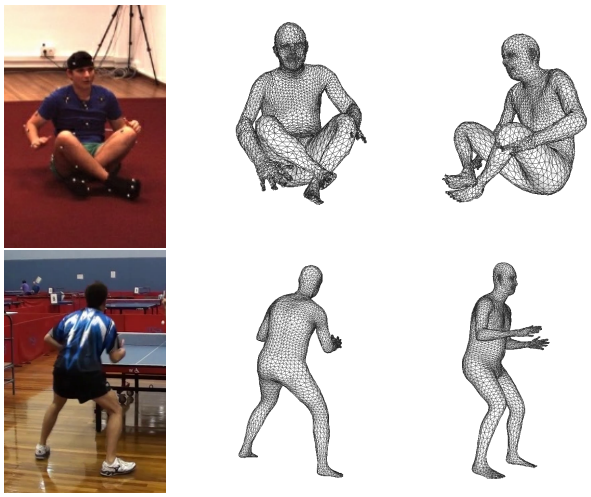


Figure 1: We present an early method to integrate Deep Learning with the sparse mesh representation, to successfully reconstruct the 3D mesh of a human from a monocular image

Alternatively, the parametric body model [2, 15, 23] based techniques address some of the above issues, how-

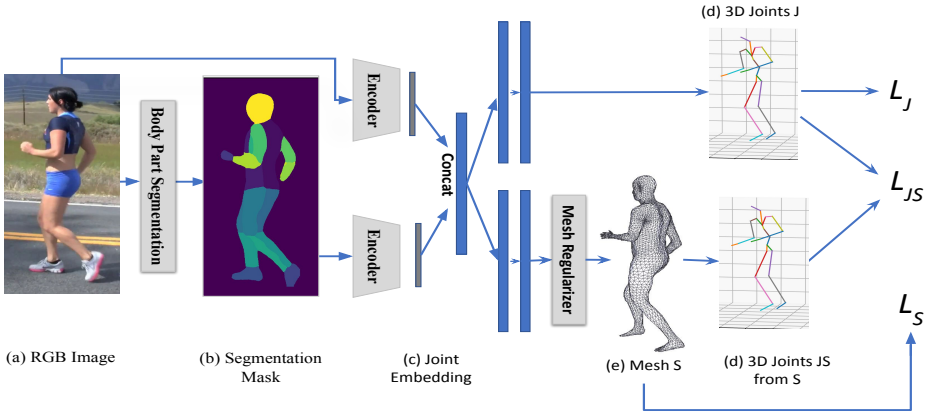



Figure 2: Overview of our Multi-Task 3D Human Mesh Reconstruction Pipeline. Given a monocular RGB image (a), we first extract a body part-wise segmentation mask using [1] (b). Then, using a joint embedding of both the RGB and segmentation mask (c), we predict the 3D joint locations (d) and the 3D mesh (e), in a multi-task setup. The 3D mesh is predicted by first applying a mesh regularizer on the predicted point cloud. Finally, the loss is minimized on both the branches (d) and (e).

ever, at the cost of accurate surface information [24, 10, 3, 14]. Recently, several end-to-end deep learning solutions for estimating the 3D parametric body model from a monocular image have been proposed [12, 27, 28, 20, 19, 33]. They all attempt to estimate the pose (relative axis-angles) and shape parameters of the SMPL [15] body model, which is a complex non-linear mapping. To get around this complex mapping, several methods transform them to rotation matrices [19, 20] or learn from the 2D/3D keypoint and silhouettes projections (a function of the parameters) [19, 12, 20]. Additionally, [12] proposes an alternate method for training (Iterative Error Feedback) as well as a body joint specific adversarial losses, which takes upto 5 days to train. In other words, learning the parametric body model hasn't been straightforward and has resulted in complex and indirect solutions that actually rely on different projections of the underlying mesh.

Directly regressing to point cloud or mesh data from image(s) is a severely ill-posed problem and there are very few attempts in deep learning literature in this direction [32, 18]. With regard to point cloud regression, most of the attempts have focused on rigid objects, where learning is done in a class specific manner. Apart from a very recent work [13], learning a mesh hasn't been explored much for reconstruction, primarily because of lack of deep learning constructs to do so.

In this paper, we attempt to work in between a generic point cloud and a mesh - i.e., we learn an "implicitly structured" point cloud. We hypothesize that in order to perform parametric body model based reconstruction, instead of learning the highly non-linear SMPL parameters, learning its corresponding point cloud (although high dimensional) and enforcing the same parametric template topology on it is an easier task. This is because, in SMPL like body models, each of the surface vertices is a sparse linear combination of the transformations induced by the underlying joints i.e., implicitly learning the skinning function by which parametric models are constructed is easier than learning the non-linear axis-angle representation itself (parameters). Further, such models lack high-resolution local surface details as well. Therefore, there are far fewer "representative" points that we have to learn. In comparison with generic point cloud regression as well, this is an easier task because of this implicit structure that exists between these points.

Going ahead, attempting to produce high resolution meshes are a natural extension that is easier in 3D space than in the parametric one. Therefore, we believe that this is a direction worth exploring and we present an initial solution in that direction - HumanMeshNet that simultaneously performs shape estimation by regressing to template mesh vertices (by minimizing surface loss) as well receives a body pose regularisation from a parallel branch in multi-task setup. The image to mesh vertex regression is further explicitly conditioned on the neighborhood constraint imposed by the mesh topology, thus ensuring a smooth surface reconstruction. Figure 2 outlines the architecture of HumanMeshNet.

Ours is a relatively simpler model as compared to the majority of the existing methods for volumetric and parametric model prediction (e.g., [29]). This makes it efficient in terms of network size as well as feed forward time yielding significantly high frame-rate reconstructions. At the same time, our simpler network achieves comparable

accuracy in terms of surface and joint error w.r.t. majority of state-of-the-art techniques on three publicly available datasets. The proposed paradigm can theoretically learn local surface deformations induced by body shape variations which the PCA space of parametric body models can't capture. In addition to predicting the body model, we also show the generalizability of our proposed idea for solving a similar task with different structure - non-rigid hand mesh reconstructions from a monocular image.
To summarize, the key contributions of this work are:

- We propose a simple end-to-end multi-branch, multi-task deep network that exploits a "structured point cloud" to recover a smooth and fixed topology mesh model from a monocular image.
- The proposed paradigm can theoretically learn local surface deformations induced by body shape variations which the PCA space of parametric body models can't capture.
- The simplicity of the model makes it efficient in terms of network size as well as feed forward time yielding significantly high frame-rate reconstructions, while simultaneously achieving comparable accuracy in terms of surface and joint error, as shown on three publicly available datasets.
- We also show the generalizability of our proposed paradigm for a similar task of reconstructing the hand mesh models from a monocular image.

## 2. Related Work

**Estimating 3D Body Models:** The traditional approach for parametric body model fitting entails iteratively optimizing an objective function with 2D supervision in the form of silhouettes, 2D key points etc [24, 10, 3, 14]. However, they often involve manual intervention and are time-consuming to solve as well as susceptible to converge at local optima.

On the deep learning front, [12] proposes an iterative regression with 3D and 2D joint loss as a feedback and an adversarial supervision for each joint. However, this architecture has a large number of networks and takes 5 days to train. [19] predicts a colour-coded body segmentation that is used as a prior to predict the parameters. Similarly, in [20], 2D heatmaps and silhouettes are predicted first, which are then used to predict the pose and shape parameters. All of the above methodologies calculate the loss on 2D keypoints or silhouette projections of the rendered mesh, which significantly slows down training time (due to model complexity), in addition to requiring additional supervision. [29] proposes a complex multi-task network with a total of six networks (having respective losses computed on 2D and 3D joint locations, 2D segmentation mask, volumetric grid and silhouette reprojection of volumetric and SMPL model). This makes it a significantly heavy network with a longer feed forward time. The focus of reconstruction is to retrieve the boundary of the subject in 3D space. However, in a volumetric representation, predicting the volume within the surface is counterproductive. On the other hand, we focus on direct image to mesh vertex regression for recovering the surface points. The most recent state-of-the-art work proposed in [13] also recovers sparse surface points using Graph Neural Network(GCN). However GCNs experience troubles learning the global structure because of its neighbourhood aggregation scheme [34].

**Estimating Hand Models:** While most of the hand recovery methods typically estimate the 3D pose from one or multiple RGB/Depth images, hand shape estimation hasn't been extensively explored. For a detailed survey of the field, we refer to [26, 35]. Recent effort in [17] was the first attempt to predict both the pose and the vertex based full 3D mesh representation (surface shape) from a single depth image. The recently proposed MANO [23] model is an SMPL like model that describes both the shape and pose, and is learned from thousands of high resolution scans. [5] predicted the MANO parameters from a monocular RGB image, but, they don't show much shape variations. [8] use a graph CNN to recover the hand surface from monocular RGB image of the hand.

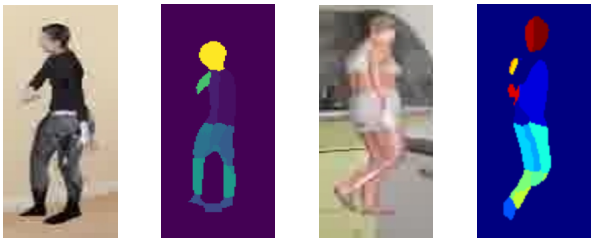
## 3. Proposed Method: HumanMeshNet

In order to learn this structured point cloud, we use an encoder- and multi-decoder model, which we describe in this section. Figure 2 gives an overview of our end-to-end pipeline. Our model consists of three primary phases:

**Phase 1 - RGB to Partwise Segmentation:** Given an input RGB image of size 224x224, we first predict a discrete body part label for each pixel in the input image (for a total of 24 body parts) using just the body part labeling network from [1]. A part-wise segmentation enables a tracking of the human body in the image, making it easier for shape estimation.

**Phase 2 - Image Encoders and Joint Embedding:** Both the RGB image and segmentation mask are passed through separate encoders, each a Resnet-18, and their respective CNN feature vectors, each of dimension 1000 are concatenated together to obtain a joint embedding.

Such fusion of RGB and segmentation mask was employed to combine complementary information from each modality. This is important as a segmentation mask predictions can be very noisy in many scenarios (see Figure 3), e.g., low lighting, distance of the person from the camera, sensing noise, etc., leading to failures like interchanged limbs or missing limbs.



(a) (b) (c) (d)

Figure 3: Noisy Segmentation Masks predicted from images (a) and (c) in Phase 1. The figure shows (b) missing body part masks (d) confusing between leg limbs.

**Phase 3 - Multi-branch Predictions:** From our concatenated feature embedding, we branch out into two complementary tasks via Fully Connected layers (FCs). Each branch consists of two FCs, each of dimension 1000 followed by the respective output dimensions for the 3D joints, and 3D surface respectively. It is to be noted that our predictions are in the camera frame.

**Loss Function.** We use a multi-branch loss functions to train our network i.e, $L_S$, $L_J$ and $L_{JS}$. We regularized the loss functions such that they contribute equally to the overall loss. This translates to Equation 1.

$$L = L_S + (\lambda_1 * L_J) + (\lambda_2 * L_{JS}) \tag{1}$$

The surface loss $L_S$ in Equation 2 gives the vertex-wise Euclidean distance between the predicted vertices $V_i$ and ground truth vertices $\hat{V}_i$ for the 3D mesh prediction branch in Figure 2 (e).

$$L_S = \sum_{\forall V_i} ||V_i - \hat{V}_i||_2 \tag{2}$$

However, this loss does not ensure prediction of smooth surfaces as each vertex is independently predicted.

Nevertheless, each mesh vertex has a neighborhood structure that can be used to further refine the estimate of individual vertex. Here we make use of smoothing regularisation [25] (as shown in Equation 3), where the position of each vertex, $V_i$, is replaced by the average position, of its neighbours $N(V_j)$.

$$V_i = \frac{1}{|N(V_i)|} \sum_{V_j \in N(V_i)} V_j \qquad \forall V_i \tag{3}$$

This is achieved by first applying the smoothness mesh regularization given by Equation 3 and then calculating $L_S$. This helps in limiting the number of surface jitters or irregularities.

In order to enforce 3D joints consistency, we minimize joint loss $L_J$ defined in Equation 4, which gives the euclidean distance between the predicted joints $J_i$ and ground truth joints $\hat{J}_i$ in the 3D joint prediction branch as shown in Figure 2(d).

$$L_J = \sum_{\forall J_i} ||J_i - \hat{J}_i||_2 \tag{4}$$

The 3D joints $JS_i$ under the surface are recovered using the SMPL joint regressor [15]. We also minimize the loss $L_{JS}$ defined in Equation 5 which gives the euclidean distance between the joints $J_i$ predicted from the joints branch and the joints $JS_i$ from the surface branch. It helps both the branches to learn consistently with each other.

$$L_{JS} = \sum_{\forall J_i} ||J_i - JS_i||_2 \tag{5}$$

**Network Variants:** We define two different variants of HumanMeshNet in order to perform an extensive analysis:

(a) HumanMeshNet (HMNet) - The base version which uses an "off-the-shelf" body part segmentation network ( [1]).

(b) HumanMeshNetOracle (HMNetOracle) - A refined version using a more accurate body part segmentation given by the dataset. However, in some datasets (e.g., UP-3D, [14]), these segmentation masks can be noisy due to manual annotations.

## 4. Experiments & Results

In this section, we show a comprehensive evaluation of the proposed model and benchmark against the state-of-the-art optimization and deep learning based Parametric (P), Volumetric (V) and Surface based (S) reconstruction algorithms. It is to be noted that we train on each dataset separately and report on its given test sets. All of the trained models and code shall be made publicly available, along with a working demo. Please view our supplementary video for more results.

### 4.1. Datasets

**SURREAL [30]**: This dataset provides synthetic image examples with 3D shape ground truth. The dataset draws poses from MoCap [11] and body shapes from body scans [22] to generate valid SMPL instances for each image. Although this dataset is synthetically generated, it emulates complex real poses and shapes, coupled with challenging input images that contain background clutter and are reflective with low resolution. It has a total of 1.6 million training and 15,000 test samples.

**UP-3D [14]**: It is a recent dataset that collects color images from 2D human pose benchmarks and uses an extended version of SMPLify [3] to provide 3D human shape candidates. The candidates were evaluated by human annotators to select only the images with good 3D shape

(a) (b) (c) (a) (b) (c)

Figure 4: This figure depicts the quality of ground truth fits provided on UP-3D. (a) The input RGB image is fit using SMPLify [3] to give (b) the ground truth. Our fit (c) makes use of more accurate markers or keypoints in a multi-branch setup, to account for noisy ground truth mesh data.

fits. It comprises of 8,515 images, where 7,126 are used for training and 1,389 for testing. However, the ground truth meshes are sometimes inaccurately generated as shown in Figure 4. We separately train the network and report results on full test set of UP-3D.

**Human3.6M [11]**: It is a large-scale pose dataset that contains multiple subjects performing typical actions like "eating" and "walking" in a lab environment. It consists of a downsampled version of the original data with 300,000 image-3D joint pairs for training and 100,000 such for testing. Since ground truth 3D meshes for any of the commonly reported protocols [4] for evaluation aren't available anymore, we finetune SURREAL-pretrained network using joint loss only. We report the joint reconstruction error (trained as per Protocol 2 of [4]) and therefore compare with those methods that don't use mesh supervision for this dataset in Table 3.

### 4.2. Implementation Details

**Data Pre-processing** We use the ground truth bounding boxes from each of the datasets to obtain a square crop of the human. This is a standard step performed by most comparative 3D human reconstruction models.

**Network Training** We use Nvidia's GTX 1080Ti, with 11GB of VRAM to train our models. A batch size of 64 is used for SURREAL and Human 3.6M datasets and a batch size of 16 for UP3D dataset. We use the ADAM optimizer having an initial learning rate of $10^{-4}$, to get optimal performance. Attaining convergence on the SURREAL and Human3.6M takes 18 hours each, while on UP-3D takes 6 hours. We use the standard splits given by the datasets, for benchmarking, as indicated in Section 4.1.

**Procrustes Analysis (PA)** In order to evaluate the quality of the reconstructed mesh, we also report results after solving the Orthogonal Procrustes problem [9], in which we scale the output to the size of the ground truth and solve for rotation. Additionally, we also quantitatively evaluate without this alignment.

**Evaluation Metric**

(a) Surface Error (mm): Gives the mean-per-vertex error between the ground truth and predicted mesh.

(b) Joint Error (mm): Gives the mean-per joint error between the ground truth and predicted joints. All reported results are obtained from the underlying joints of the mesh, rather than the alternate branch, unless otherwise mentioned.

(c) PA. Surface/Joint Error (mm): It is the surface/joint error after Procrustes Analysis (PA).

### 4.3. Comparison with State-of-the-art

**Baseline** We define our baseline as the direct prediction of a point cloud from an RGB image, using a Resnet-50. This enables us to show the novelty introduced by our pipeline and the usefulness of learning in this output space.

**Results & Discussion** For qualitative results on all of the three datasets refer to Figures 5, 6. A large amount of training data is required to learn a vast range of poses and shapes. However, [30, 20] show a good domain transfer to real data by training on the synthetic SURREAL dataset. Since our supervision is dominated by surface meshes, SURREAL plays an important role in benchmarking our method. We show comparable performance on it, as indicated by Table 2. In Table 2, we also show our results with a subsampled mesh (subsampled as per [13]) from 6890 to 1723 vertices with almost no change in reconstruction error. This is a good proof of our hypothesis that there are far fewer representative points to learn in this structured point cloud.

UP-3D is an "in the wild" dataset, however, has inaccurate ground truth mesh annotations, as shown in Figure 4. Most circumvent this issue, by avoiding 3D supervision altogether and projecting back to a silhouette or keypoints [12, 20]. Further, training on such a small dataset doesn't provide a good generalisation. Therefore, we observe a higher error in HMNet. However, HMNetOracle produces a significant increase in accuracy with the increase in quality of the input image and segmentation mask (Table 4). Similar to state-of-the-art methods [29, 31, 13], we rely on 3D body supervision and providing more supervision like silhouette and 2D keypoint loss like [29, 12] can improve the performance further. For Human3.6m, we compare with those that don't use mesh supervision (since this data is currently unavailable) and achieve comparable performance.


Figure 5: Qualitative Results on SURREAL [30] (first six columns) and UP-3D [14] (next six columns) where (a) represents the input view, (b) our mesh reconstruction aligned to the input view, and (c) aligned to another arbitrary view.

| Output | Method | Surface Error | Joints Error | PA. Surface Error | PA. Joint Error |
|---|---|---|---|---|---|
| P | Pavlakos *et al.* [20] | 117.7 | - | - | - |
| P | Lasner *et al.* [14] | 169.8 | - | - | - |
| P | NBF [19] | - | - | - | 82.3 |
| V | BodyNet [29] | 80.1 | - | - | - |
| S | Baseline | 151.4 | 130.8 | 93.8 | 83.7 |
| S | HMNet | 130.4 | 112.5 | 77.6 | 69.6 |
| S | HMNetOracle | **60.3** | **51.5** | **42.9** | **37.9** |

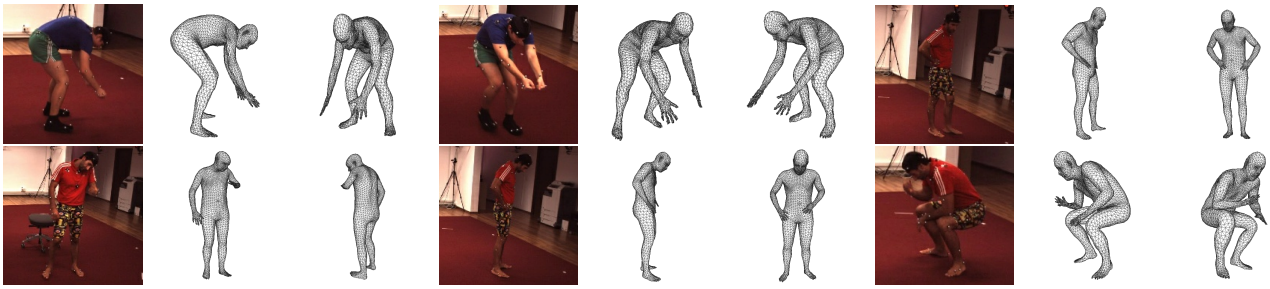
Table 1: Comparison with other methods on UP3D's full test set [14].

| Output | Method | Surface Error | Joint Error |
|---|---|---|---|
| P | Tung *et al.* [28] | 74.5 | 64.4 |
| | Pavlakos *et al.* [20] | 151.5 | - |
| | SMPLR [16] | 75.4 | 55.8 |
| V | BodyNet [29] | 65.8 | - |
| S | Baseline | 101 | 85.7 |
| | HMNet[subsampled] | 86.9 | 72.4 |
| | HMNet | 86.6 | 71.9 |
| | HMNetOracle | **63.5** | **49.1** |

Table 2: Comparison with state-of-the-art methods on SURREAL's test set [30].

| 3D mesh Supervision | Method | PA. Joint Error |
|---|---|---|
| No | Ramakrishnan *et al.* [21] | 157.3 |
| | Zhou *et al.* [36] | 106.7 |
| | SMPLify [4] | 82.3 |
| | SMPLify 91 kps [14] | 80.7 |
| | Pavlakos *et al.* [20] | 75.9 |
| | HMR [12] | **56.8** |
| | HMNet(Ours) | 60.9 |
| Yes | NBF [19] | 59.9 |
| | SMPLR [16] | 56.4 |
| | CMR [13] | **50.1** |

Table 3: Joint Reconstruction error as per Protocol 2 of Bogo *et al.* [4] on Human 3.6M [11]. Refer to Section 4.1 for details on 3D mesh supervision.



(a) (b) (c) (a) (b) (c) (a) (b) (c)

Figure 6: Qualitative Results on Human3.6M, where (a) represents the input view, (b) our mesh reconstruction aligned to the input view, and (c) aligned to another arbitrary view.

### 4.4. Discussion

**Ablation Study:** Directly regressing the mesh from RGB leads to sub-par performance. Limbs are typically the origin of maximum error in reconstruction, and the segmentation mask provides a better tracking in scenarios such as leg-swap shown in Figure 7. The first two rows of Table 4 quantitatively explain this behaviour. Further, by having a more accurate segmentation mask, HMNetOracle achieves a significant reduction in surface error ($\downarrow$ $34.7mm$). In scenarios with inaccurate ground truth 3D (Figure 4), the regularisation 3D joint loss in our multi-branch setup helps us in recovering better fits (row 4 for UP3D). In datasets such as Human3.6m where accurate MoCap markers are given, this multi-branch loss provides a good boost - with and without joint loss, the joint reconstruction error is 60.9mm v/s 67.3mm respectively.

(a) (b) (c) (d) (a) (b) (c) (d)

Figure 7: Given an input RGB image (a), this figure depicts a comparison of the baseline (b), against our output, HMNet (d). The predicted part-wise segmentation mask (c) assists HMNet to track the body parts and therefore solve the confusion between the legs as well as complex poses.

**Effect of Mesh Regularisation** Our mesh regularization module adds a smoothing effect while training, therefore ensuring that the entire local patch should move towards the ground truth for minimizing the error. Although intrinsic geometry based losses can also be used here, we hypothesize that they have a larger impact when more complex local surface deformations (e.g., facial expressions) are present. Figure 8 shows the impact of this regularization.

| Config. | Input | PA. Surface Error | PA. Joint Error |
|---|---|---|---|
| Baseline | RGB | 93.8 | 83.7 |
| Single Task | $SM_{DP}$ | 82.9 | 74.6 |
| Single Task | RGB+$SM_{DP}$ | 79.2 | 71.04 |
| HMNet | RGB+$SM_{DP}$ | 77.6 | 69.6 |
| HMNetOracle | RGB+$SM_{GT}$ | **42.9** | **37.9** |

Table 4: Effect of each network module on the reconstruction error on UP-3D dataset. $SM_{DP}$ and $SM_{GT}$ denotes segmentation obtained from Densepose and groundtruth respectively.

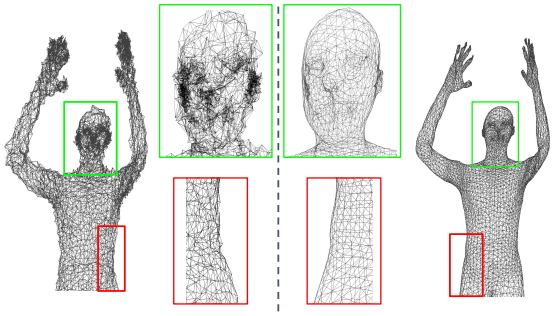


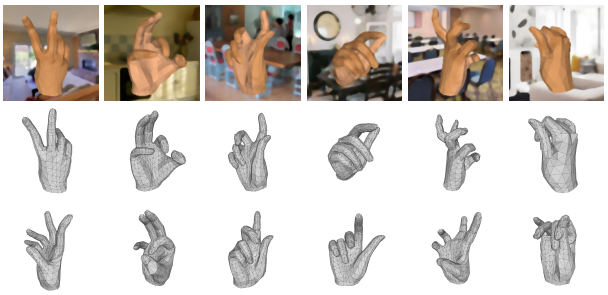
Figure 8: Results showing the effect of our mesh regularization module while learning. The figure on the left shows the irregularities in the mesh reconstructed, without our regularization, while the one on the right shows the smoothness induced by our regularizer.

**Recovering Shape Variations** Most parametric models prediction work with a neutral template model [12], and would have to learn the gender from the image. In our method, a direct mesh regression can learn the local shape variations (as long as training data has such variations) which extend to inherently learning gender invariant meshes. Two such samples are showing in Figure 9.

(a) (b) (c) (a) (b) (c)

Figure 9: Sample Shape Variations recovered by our model given an input image (a), rendered from the recovered view (b) and another arbitrary view (c)

**Generalizability to Hand Mesh Models.** We show the generalizability of our model to a similar task with a different structure. First, we populated a SURREAL like synthetic hand dataset using the MANO hand model [23], similar to [8] with a total of 70,000 image-mesh pairs. We train our model on this dataset to predict hand surface and joints from an input RGB image using the same pipeline described in Figure 2. The training setting remains the same as earlier, and we obtain impressive qualitative results as shown in Figure 10. The average surface error across the test dataset is 1mm, which acts as a proof of concept that polygonal mesh reconstruction of non-rigid hands (although in a simplistic scenario), is feasible.



(a) (b) (c) (d) (e) (f)

Figure 10: Reconstruction Results on our Hand Mesh. Each column consists of the RGB image, its corresponding reconstruction from the same view, and from another arbitrary view.

**Network Runtime.** Table 5 list out run-time of various methods. Comparing this with HMNet with HMNetOracle, it is evident that a major part of HMNet's complexity arises from the multi-human pixel wise class prediction, which runs at around 30 FPS for an image of size 224x224. [6] is an accurate real time body part segmentation network which runs at 120 FPS, and can be incorporated into our system to produce accurate, real time reconstructions.

**Limitations and Future Work.** Since we do not enforce any volume consistency, skewing/thinning artifacts might be introduced in our meshes. We would like to account for these in a non-handcrafted anthropomorphically valid way by either learning the SMPL parameters on top of it using an MLP similar to [13] or by using a GAN to penalize fake/invalid human meshes. Further, we have made use of the mesh topology in two ways in this work - (a) implicitly, to make the learning easier and (b) for smoothing. Going ahead, we would like to make use of the mesh topology and geometry details is a more explicit manner, by using intrinsic mesh/surface properties. We believe that this is a largely unexplored space and applying such a regularization can result in better exploitation of surface geometry for reconstruction.

| Method | Output | FPS |
|---|---|---|
| SMPLify [3] | P | 0.01 |
| SMPLify, 91 kps [14] | | 0.008 |
| Decision Forests [14] | | 7.69 |
| HMR [12] | P | 25 |
| Pavlakos [20] | | 20 |
| Direct Prediction [14] | | 2.65 |
| Baseline | S | 175.4 |
| HMNet | | **28.01** |
| HMNetOracle | | **173.17** |
| Fusion4D [7] | S | 31 |

Table 5: Overview of the run time (in Frames Per Second, FPS) of various algorithms. Numbers have been picked up from the respective papers. All methods have used 1080Ti or equivalent GPU.

Figure 11: Failure Cases of Our Method.

## 5. Conclusion

We proposed a multi-branch multi-task HumanMeshNet network that simultaneously regresses to the template mesh vertices as well as body joint locations from a single monocular image. The proposed method achieves comparable performance with significantly lower modelling and computational complexity on three publicly available datasets. We also show the generalizability of the proposed architecture for a similar task of predicting the mesh of the hand. Looking forward, we would like to exploit intrinsic mesh properties to recover a more accurate surface reconstruction.

## References


[1] R. Alp Güler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7297–7306, 2018. 2, 3, 4

[2] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. *ACM Transaction on Graphics*, 24:408–416, 2005. 1

[3] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *European Conference on Computer Vision (ECCV)*, 2016. 2, 3, 4, 5, 8

[4] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *European Conference on Computer Vision*, pages 561–578. Springer, 2016. 5, 6

[5] A. Boukhayma, R. de Bem, and P. H. Torr. 3d hand shape and pose from images in the wild. *arXiv preprint arXiv:1902.03451*, 2019. 3

[6] J. J. Charles, I. Budvytis, and R. Cipolla. Real-time factored convnets: Extracting the x factor in human parsing. 2018. 8

[7] M. Dou, S. Khamis, Y. Degtyarev, P. Davidson, S. R. Fanello, A. Kowdle, S. O. Escolano, C. Rhemann, D. Kim, J. Taylor, P. Kohli, V. Tankovich, and S. Izadi. Fusion4d: Real-time performance capture of challenging scenes. *ACM Transaction on Graphics*, 35(4):114:1–114:13, July 2016. 8

[8] L. Ge, Z. Ren, Y. Li, Z. Xue, Y. Wang, J. Cai, and J. Yuan. 3d hand shape and pose estimation from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10833–10842, 2019. 3, 8

[9] J. C. Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, Mar 1975. 5

[10] P. Guan, A. Weiss, A. O. Balan, and M. J. Black. Estimating human shape and pose from a single image. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1381–1388. IEEE, 2009. 2, 3

[11] 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*, 36(7):1325–1339, 2014. 4, 5, 6

[12] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. 2, 3, 5, 6, 7, 8

[13] N. Kolotouros, G. Pavlakos, and K. Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *CVPR*, 2019. 2, 3, 5, 6, 8

[14] C. Lassner, J. Romero, M. Kiefel, F. Bogo, M. J. Black, and P. V. Gehler. Unite the people: Closing the loop between 3d and 2d human representations. 2, 3, 4, 6, 8

[15] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 1, 2, 4

[16] M. Madadi, H. Bertiche, and S. Escalera. Smplr: Deep smpl reverse for 3d human pose and shape recovery. *arXiv preprint arXiv:1812.10766*, 2018. 6

[17] J. Malik, A. Elhayek, F. Nunnari, K. Varanasi, K. Tamaddon, A. Heloir, and D. Stricker. Deephps: End-to-end estimation of 3d hand pose and shape by learning from synthetic depth. *arXiv preprint arXiv:1808.09208*, 2018. 3

[18] P. Mandikal, N. K. L., M. Agarwal, and V. B. Radhakrishnan. 3d-lmnet: Latent embedding matching for accurate and diverse 3d point cloud reconstruction from a single image. In *British Machine Vision Conference*, page 55, 2018. 2

[19] M. Omran, C. Lassner, G. Pons-Moll, P. V. Gehler, and B. Schiele. Neural body fitting: Unifying deep learning and model-based human pose and shape estimation. *arXiv preprint arXiv:1808.05942*, 2018. 2, 3, 6

[20] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis. Learning to estimate 3d human pose and shape from a single color image. *arXiv preprint arXiv:1805.04092*, 2018. 2, 3, 5, 6, 8

[21] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3d human pose from 2d image landmarks. In *European Conference on Computer Vision*, pages 573–586. Springer, 2012. 6

[22] K. M. Robinette, S. Blackwell, H. Daanen, M. Boehmer, and S. Fleming. Civilian american and european surface anthropometry resource (caesar), final report. volume 1. summary. Technical report, SYTRONICS INC DAYTON OH, 2002. 4

[23] J. Romero, D. Tzionas, and M. J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics (TOG)*, 36(6):245, 2017. 1, 3, 8

[24] L. Sigal, A. Balan, and M. J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Advances in neural information processing systems*, pages 1337–1344, 2008. 2, 3

[25] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, and H.-P. Seidel. Laplacian surface editing. In *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 175–184. ACM, 2004. 4

[26] J. S. Supancic, G. Rogez, Y. Yang, J. Shotton, and D. Ramanan. Depth-based hand pose estimation: data, methods, and challenges. In *Proceedings of the IEEE international conference on computer vision*, pages 1868–1876, 2015. 3

[27] V. Tan, I. Budvytis, and R. Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. 2

[28] H.-Y. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki. Self-supervised learning of motion capture. In *Advances in Neural Information Processing Systems*, pages 5236–5246, 2017. 2, 6

[29] G. Varol, D. Ceylan, B. Russell, J. Yang, E. Yumer, I. Laptev, and C. Schmid. Bodynet: Volumetric inference of 3d human body shapes. *arXiv preprint arXiv:1804.04875*, 2018. 1, 2, 3, 5, 6

[30] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid. Learning from synthetic humans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 109–117, 2017. 4, 5, 6

[31] A. Venkat, S. S. Jinka, and A. Sharma. Deep textured 3d reconstruction of human bodies. *arXiv preprint arXiv:1809.06547*, 2018. 1, 5

[32] Y. Xia, Y. Zhang, D. Zhou, X. Huang, C. Wang, and R. Yang. Realpoint3d: Point cloud generation from a single image with complex background. *CoRR*, abs/1809.02743, 2018. 2

[33] D. Xiang, H. Joo, and Y. Sheikh. Monocular total capture: Posing face, body, and hands in the wild. *arXiv preprint arXiv:1812.01598*, 2018. 2

[34] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. 3

[35] S. Yuan, G. Garcia-Hernando, B. Stenger, G. Moon, J. Yong Chang, K. Mu Lee, P. Molchanov, J. Kautz, S. Honari, L. Ge, et al. Depth-based 3d hand pose estimation: From current achievements to future goals. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3

[36] X. Zhou, S. Leonardos, X. Hu, and K. Daniilidis. 3d shape estimation from 2d landmarks: A convex relaxation approach. In *proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4447–4455, 2015. 6