

Appearance Consensus Driven Self-Supervised Human Mesh Recovery

Jogendra Nath Kundu^{1*}, Mugalodi Rakesh^{1*}, Varun Jampani²,
Rahul Mysore Venkatesh¹, and R. Venkatesh Babu¹

¹Indian Institute of Science, Bangalore ²Google Research

Abstract. We present a self-supervised human mesh recovery framework to infer human pose and shape from monocular images in the absence of any paired supervision. Recent advances have shifted the interest towards directly regressing parameters of a parametric human model by supervising them on large-scale datasets with 2D landmark annotations. This limits the generalizability of such approaches to operate on images from unlabeled wild environments. Acknowledging this we propose a novel appearance consensus driven self-supervised objective. To effectively disentangle the foreground (FG) human we rely on image pairs depicting the same person (consistent FG) in varied pose and background (BG) which are obtained from unlabeled wild videos. The proposed FG appearance consistency objective makes use of a novel, differentiable *Color-recovery* module to obtain vertex colors without the need for any appearance network; via efficient realization of color-picking and reflectional symmetry. We achieve state-of-the-art results on the standard model-based 3D pose estimation benchmarks at comparable supervision levels. Furthermore, the resulting colored mesh prediction opens up the usage of our framework for a variety of appearance-related tasks beyond the pose and shape estimation, thus establishing our superior generalizability.

1 Introduction

Inferring highly deformable 3D human pose and shape from in-the-wild monocular images has been a longstanding goal in the vision community [12]. This is considered as a key step for a wide range of downstream applications such as robot interaction, rehabilitation guidance, animation industry, etc. Being one of the important subtasks, human pose estimation has gained considerable performance improvements in recent years [61, 45, 57], but in a fully-supervised setting. Such approaches heavily rely on large-scale 2D or 3D pose annotations. Following this, the parametric models of human body, such as SCAPE [3], SMPL [40], SMPL(-X) [49, 58] lead the way for a full 3D pose and shape estimation. Additionally, to suppress the inherent 2D-to-3D ambiguity, researchers have also utilized auxiliary cues of supervision such as temporal consistency [4, 62], multi-view image pairs [53, 20, 14], or even alternate sensor data from Kinect [67] or IMUs [44].

* Equal contribution. | Webpage: <https://sites.google.com/view/ss-human-mesh>

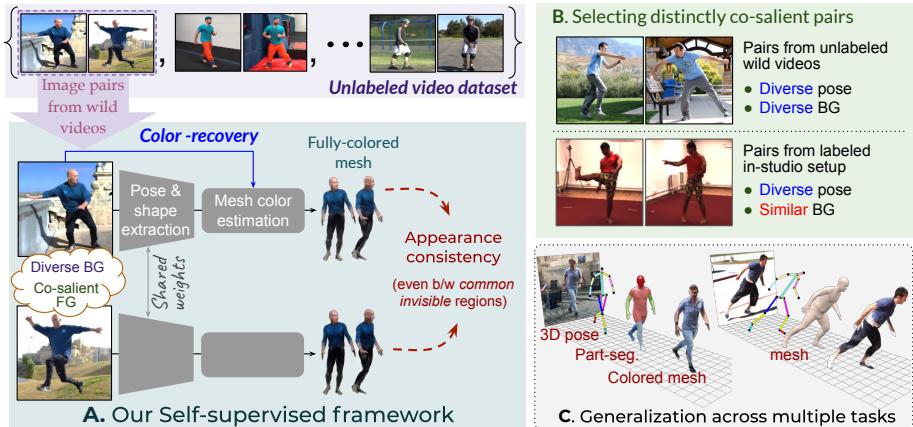


Fig. 1. Our framework disentangles the co-salient FG human from input image pairs. The resulting colored mesh prediction opens up its usage for a variety of tasks.

However, estimating 3D human pose and shape from a single RGB image without relying on any direct supervision remains a very challenging problem.

Early approaches [5,8,35] adopt iterative optimization techniques to fit a parametric human model (*e.g.* SMPL) to a given image observation. These works attempt to iteratively estimate the body pose and shape that best describe the available 2D observation, which is most often the 2D landmark annotations. Though these works usually get good body fits, such approaches are slow and heavily rely on the 2D landmark annotations [2,18,28] or predictions of an off-the-shelf, fully-supervised Image-to-2D pose networks. However, the recent advances in deep learning has shifted the interest towards data-driven regression based methods [21,64], where a deep network directly regresses parameters of the human model for a given input image [48,51,69] in a single-shot computation. This is a promising direction as the network can utilize the full image information instead of just the sparse landmarks to estimate human body shape and pose. In the absence of datasets having images with 3D pose and shape ground-truth (GT), several recent works leverage a variety of available paired 2D annotations [50,63] such as 2D landmarks or silhouettes [51]; alongside the unpaired 3D pose samples to instill the 3D pose priors [21] (*i.e.* to assure recovery of valid 3D poses). The strong reliance on paired 2D keypoint ground-truth limits the generalization of such approaches when applied to images from an unseen wild environment. Given the transient nature of human fashion, the visual appearance of human attire keeps evolving. This demands such approaches to periodically update their 2D pose dataset in order to retain their functionality.

In this work, the overarching objective is to move away from any kind of paired pose-related supervision for superior generalizability. Our aim is to explore a form of self-supervised objective which can learn both pose and shape from monocular images without accessing any paired GT annotations. We draw motivation from works [46,56,41,32] that aim to disentangle the fundamental factors of variations

from a given image. For human-centric images [33], these factors could be; a) pose, b) foreground (FG) appearance, and c) background (BG) appearance. Here, we leverage the full advantage of incorporating a parametric human model in our framework. Note that, this parametric model not only encapsulates the pose but also segregates the FG region from the BG, which is enabled by projecting the 3D mesh onto the image plane. Thus, the problem boils down to a faithful registration of the 3D mesh onto the image plane or in other words disentanglement of FG from BG. To achieve this disentanglement, we rely on image pairs depicting consistent FG appearance but varied 3D poses. Such image pairs can be obtained from videos depicting actions of a single person, which are abundantly available on the internet. Our idea stems from the concept of co-saliency detection [70,13] where the objective is to segment out the common, salient FG from a set of two or more images. Surprisingly, this idea works the best for image pairs sampled from wild videos as compared to videos captured in a constrained in-studio setup (static homogeneous background). This is because in wild scenarios, the commonness of FG is distinctly salient in relatively diverse BGs as a result of substantial camera movements (see Fig. 1B). Thus, in contrast to prior self-supervised approaches that either rely on videos with static BG [54] or operate under the assumption of BG commonness between temporally close frames [16]; our approach is more favorable to learn from wild videos hence better generalizable.

In the proposed framework, we first employ a CNN regressor to obtain the parameters (both pose and shape) of the SMPL model for a given input image. The human mesh model uses these parameters to output the mesh vertex locations. In contrast to the general trend [1,22], we propose a novel way of inferring mesh texture where the networks burden to regress vertex color or any sort of appearance representation (such as UV map) is entirely taken away. This is realized via a differentiable *Color-recovery* module which aims to assign color to the mesh vertices via spatial registration of the mesh over the image plane while effectively accounting for the challenges of mesh-vertex visibility like self and inter-part occlusions. To obtain a fully-colored mesh, we use a predefined, 4-way symmetry grouping knowledge (front-back and left-right) to propagate the color from camera visible vertices to the non-visible ones in a fully differentiable fashion.

For a given image pair, we pass them through two parallel pathways of our colored mesh prediction framework (see Fig. 1A). The commonness of FG appearance allows us to impose an appearance consistency loss between the predicted mesh representations. In the absence of any paired supervision, this appearance consistency not only helps us to segregate the common FG human from their respective wild BGs but also discovers the required pose deformation in a fully self-supervised manner. The proposed reflectional symmetry module

Table 1. Characteristic comparison against prior-arts.

Model-based methods	2D keypoint supervision	Temporal supervision	Colored mesh prediction
[21,26,27,51,48]	Yes	No	No
[62,4,23]	Yes	Yes	No
Ours(self-sup.)	No	No	Yes

brings in a substantial advantage in our self-supervised framework by allowing us to impose appearance consistency even between body parts which are “*commonly invisible*” in both the images. Recognizing the unreliability of consistent raw color intensities which can easily be violated as result of illumination changes, we propose a *part-prototype* consistency objective. This aims to match a higher level appearance representation beyond the raw color intensities which is enabled by operating the *Color-recovery* module on convolutional feature maps instead of the raw image. Additionally, to regularize the self-supervised framework, we also impose a shape consistency loss alongside the imposition of 3D pose prior learned from a set of unpaired MoCap samples. Note that at test time, we perform single image inference to estimate 3D human pose and shape.

In summary, we make the following main contributions:

- We propose a self-supervised learning technique to perform simultaneous pose and shape estimation which uses image pairs sampled from in-the-wild videos in the absence of any paired supervision.
- The proposed *Color-recovery* module completely eliminates the networks burden to regress any appearance-related representation via efficient realization of color-picking and reflectional symmetry. This best suits our self-supervised framework which relies on FG appearance consistency.
- We demonstrate generalizability of our framework to operate on *unseen* wild datasets. We achieve *state-of-the-art* results against the prior model-based pose estimation approaches when tested at comparable supervision levels.

2 Related Work

Vertex-color reconstruction. In literature, we find different ways to infer textured 3D mesh from a monocular RGB image. Certain approaches [34, 60] train a deep network to directly regress 3D features (RGB colors) for individual vertices. In the second kind, a fully convolutional deep network is trained to map the location of each pixel to the corresponding continuous UV-map coordinate parameterization [1]. In the third kind, the deep model is trained to directly regress the UV-image [22]. Note that, the spatial structure of the UV image is much different from that of the input image which prevents employing a fully-convolutional network for the same. Recently proposed, Soft-Rasterizer [38] uses a color-selection and color-sampling network whose outputs are processed to obtain the final vertex colors. All the above approaches adopt a learnable way to obtain the mesh color (*i.e.* obtained as neural output). In such cases, the deep network requires substantial training iterations to instill the knowledge of pre-defined UV mapping conventions. We believe this is an additional burden for the network specifically in absence of any auxiliary paired supervisions.

Model-based human mesh estimation. Recently, parametric human models [3, 40] have been used as the output target for the simultaneous pose and shape estimation task. Such a well-defined mesh model with ordered vertices provides a direct mapping to the corresponding 3D pose and part segments. Both optimization [5, 35, 68] and regression [21, 48, 51, 69] based approaches estimate the

body pose and shape that best describes the available 2D observations such as 2D keypoints [21], silhouettes [51], body/part segmentation [48] etc. Due to the lack of datasets having wild images with 3D pose and shape GT, most of the above approaches fully rely on the availability of 2D keypoint annotations [2,37] followed by different variants of a 2D reprojection loss [63,64] (see Table 1).

Use of auxiliary supervision. In the absence of any shape supervision, certain prior works also leverage full mesh supervision available from synthetically rendered human images [66] or images with fairly successful body fits [35]. Furthermore, multi-view image pairs have also been used for 3D pose [54] and shape estimation [11,36] via enforcing consistency of canonical 3D pose across multiple views. Liang *et al.* [36] use a multi-stage regressor for multi-view images to further reduce the projection ambiguity in order to obtain a better performance for 3D human body under clothing. To inculcate strong 3D pose prior, Zhou *et al.* [71] makes use of left-right symmetric bone-length constraint for the skeleton based 3D pose estimation task. Further, to assure recovery of valid 3D poses for the model-based pose estimation task, Kanazawa *et al.* [21] enforce learning based human pose and shape prior via adversarial networks using unpaired sample of plausible 3D pose and shape. With the advent of differentiable renderers [10,24] certain methods supervise 3D shape and pose estimation through a textured mesh prediction network to encourage matching of the rendered texture image with the image FG [22], alongside the 2D keypoint supervision [50].

3 Approach

We aim to discover the 3D human pose and shape from unlabeled image pairs of consistent FG appearance. During training, we assume access to a parametric human mesh model to aid our self-supervised paradigm. The mesh model provides a low dimensional parametric representation of variations in human shape and pose deformations. However, by design, this model is unaware of the plausibility restrictions of human pose and shape. Thus, it is prone to implausible poses and self-penetrations specifically in the absence of paired 3D supervision [21]. Therefore, to constrain the pose predictions, we assume access to a pool of human 3D pose samples to learn a 3D pose prior.

Fig. 2 shows an overview of our training approach. For a given image pair, two parallel pathways of shared CNN regressors predict the human shape and pose parameters alongside the required camera settings to segregate the co-salient FG human. Moreover, to realize a colored mesh representation, we develop a differentiable *Color-recovery* module which infers mesh vertex colors directly from the given image without employing any explicit appearance extraction network.

3.1 Representation and notations

Human mesh model. We employ the widely used SMPL body model [40] which parameterizes a triangulated human mesh of $K = 6890$ vertices. This model factorizes the mesh deformations into shape $\beta \in \mathbb{R}^{10}$ and pose $\theta \in \mathbb{R}^{3J}$

with $J = 23$ skeleton joints [21]. We use the first 10 PCA coefficients of the shape space as a compact shape representation inline with [21]. And, the pose is parameterized as parent-relative rotations in the axis-angle representation. This differentiable SMPL function outputs mesh vertex locations in a canonical 3D space which is represented as $V \in \mathbb{R}^{K \times 3} = \mathcal{M}(\theta, \beta)$. Here, the corresponding 3D pose (*i.e.* 3D location of J joints) is obtained using a pre-trained linear regressor, *i.e.* $Y \in \mathbb{R}^{J \times 3} = W_p V$ parameterized by $W_p \in \mathbb{R}^{J \times K}$. RGB color corresponding to the mesh vertices, V is denoted as $C \in \mathbb{R}^{3 \times K} = CRM(V, I)$, where *CRM* is the *Color-recovery* module. For each vertex id k , $C^{(k)}$ stores the corresponding RGB color intensities. As shown in Fig. 2, we use subscripts a and b to associate the terms with the respective input images, I_a and I_b .

Camera model. We define a weak perspective camera model using a global orientation $R \in \mathbb{R}^{3 \times 3}$ in axis-angle representation (3 angle parameters), a translation $t \in \mathbb{R}^2$ and a scale $s \in \mathbb{R}$. Given these parameters, the 2D camera space coordinates of the 3D mesh vertices with vertex index k is obtained as $v^{(k)} = \pi(V^{(k)}) = s\pi(RV^{(k)}) + t; v^{(k)} \in \mathcal{U}$, where π denotes orthographic projection and $\mathcal{U} \subset \mathbb{R}^2$ denotes the space of image coordinates. Similarly, the camera projected 2D joint locations (2D pose) is expressed as $y \in \mathbb{R}^{J \times 2} = \pi(Y)$.

3.2 Mesh estimation architecture

For a given monocular image, I as input, we first employ a CNN regressor to predict the SMPL parameters (*i.e.* θ and β) alongside the camera parameters, (R, s, t) . This is followed by the *Color-recovery* module. The prime functionality of this module is to assign color to the 3D mesh vertices, $C^{(k)}; k = 1, 2, \dots, K$ based on the corresponding image space coordinates obtained via camera projection. However, a reliable color assignment requires us to segregate the vertices based on the following two important criteria.

a) **Non-camera-facing vertices:** First, the camera-facing vertices are separated from the non-camera-facing ones using the mesh vertex normals. Here, the vertex normal is computed as the normalized average of the surface normals of the faces connected to a given vertex. We first transform these normals from the default canonical system to the camera coordinate system. Following this, Z-component of the *camera-space-normals*, $N^{(k)} \in \mathbb{R}$ are used to segregate the non-camera-facing vertices via a *sigmoid* operation, as shown in Fig. 2.

b) **Camera-facing, self-occluded vertices:** Note that, $N^{(k)}$ can not be used to select all the camera-visible vertices in presence of inter-part occlusions (see Fig. 2). As, in such scenario, there exist mesh vertices which face the camera but are obscured by other camera-facing vertices which are closer to the camera in 3D. This calls for modeling the relative depth of mesh-vertices as the second criteria to reliably select the vertices which are closer to the camera among all the camera-facing vertices projected to a certain spatial region. To realize this, we utilize *camera-space-depths*, $Z^{(k)} \in \mathbb{R}$ which stores the Z-component (or depth) of the vertex location in the camera transformed space.

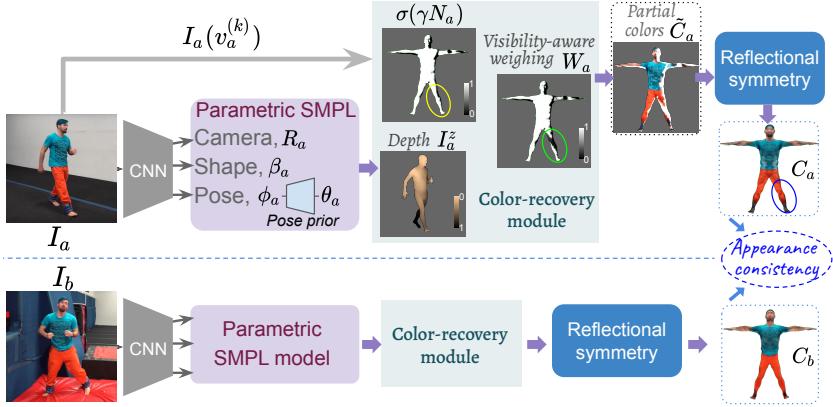


Fig. 2. The proposed self-supervised framework makes use of a differentiable *Color-recovery* module to recover the fully colored mesh vertices. *Yellow-circle*: camera-facing vertices does not account for inter-part occlusion. *Green-circle*: W_a accounts for the inter-part occlusion. *Blue-circle*: Fully colored mesh vertices via reflectional symmetry.

3.2.1 Color-recovery module. In absence of any appearance related features, we plan to realize a spatial depth map using a fast differentiable renderer [10] where the camera-space-depth of the mesh vertices, Z is treated as the color intensities for the rendering pipeline. The resultant depth-map is represented as $I^z(u)$, where u spans the space of spatial indices. The general idea is to use this depth-map as a margin. More concretely, for effective color assignment, one must select the spatially modulated mesh vertices which have the least absolute depth difference with respect to the above defined depth margin. To realize this, we compute a depth difference $D^{(k)}$ as $|I^z(v^{(k)}) - Z^{(k)}|$, where $I^z(v^{(k)})$ is computed by performing bilinear sampling on $I^z(u)$. In accordance with the above discussion, we formulate a *visibility-aware-weighing* which takes into account both the above mentioned criteria required for an effective mesh vertex selection.

$$W^{(k)} \in [0, 1] = \exp(-\alpha D^{(k)}) \sigma(\gamma N^{(k)}), \text{ where } D^{(k)} = |I^z(v^{(k)}) - Z^{(k)}|$$

Here, $\exp(-\alpha D^{(k)})$ performs a soft selection by assigning a higher weight value (close to 1) for mesh vertices, k whose *camera-space-depth* $Z^{(k)}$ is in agreement with $I^z(v^{(k)})$ and vice-versa. In the second term, σ denotes a sigmoid function with a higher steepness γ to reject the non-camera-facing mesh vertices by attributing a low (close to 0) weighing value. Refer Fig. 2 for visual illustration.

Intermediate vertex color assignment. The above defined *visibility-aware-weighing* is employed to realize a primary vertex color assignment. We denote $\tilde{C} \in \mathbb{R}^{3 \times K}$ as the intermediate vertex color, where $\tilde{C}^{(k)}$ stores the corresponding RGB color intensities acquired from the given input image I . Thus, the primary vertex colors are obtained as, $\tilde{C}^{(k)} = I(v^{(k)}) (2W^{(k)} - 1)$, where $I(v^{(k)})$ stores the RGB color intensities at the spatial coordinates $v^{(k)}$ realized via performing

bilinear sampling on the input RGB image I . The scaled weighing function $(2W^{(k)} - 1)$ assigns negative weight to the vertices having low visibility. This assigns a negative color intensity for the corresponding vertices thereby allowing a distinction between the *less-bright* (near-black) colors versus *unassigned* vertices.

3.2.2 Vertex color assignment via reflectional symmetry. Here, the prime objective is to propagate the reliable color intensities from the assigned vertices to the unreliable/unassigned ones. The idea is to use reflectional symmetry as a prior knowledge by accessing a predefined set of reflectional groups. For each group-id $g = 1, 2, \dots, G$, a set of 4 vertices are identified according to left-right and front-back symmetry which would have the same color property (except the vertices belonging to the head where only left-right symmetry is used). This symmetry knowledge is stored as a multi-hot encoding denoted as $S^{(g)} \in \{0, 1\}^K$ which constitutes of four ones indicating vertex members in the symmetry group g . All the symmetry groups are combined in a symmetry-encoding matrix represented as $S \in \{0, 1\}^{G \times K}$. This multi-hot symmetry group representation helps us to perform a fully-differentiable vertex color assignment for all the vertices including the occluded and non-camera facing ones.

To realize the final vertex colors C , we first estimate a group-color for each group g which is denoted by $\mathcal{C}^{(g)} \in \mathbb{R}^3 = (S^{(g)} \circ \text{ReLU}(\tilde{C})) / (S^{(g)} \circ \text{ReLU}(2W - 1))$. Here, \circ denotes dot product between the K -dimensional vectors. The group color can be interpreted as a combination of the intermediate vertex colors weighted by their visibility weighing W . This effectively handles the cases when only one or more of the vertices in a group are initially colored (visible). That is, when visibility is active only for a single vertex among the four vertices in a symmetry set; and when visibility is active for all the 4 vertices in a symmetry set; and also the intermediate cases. Finally, the group color is directly propagated to all the mesh vertices using the following matrix multiplication operation, *i.e.* $C = S^T * \mathcal{C}$, where $\mathcal{C} \in \mathbb{R}^{G \times 3} = [\mathcal{C}^{(1)}, \mathcal{C}^{(2)}, \dots, \mathcal{C}^{(G)}]$ (see Suppl for more details).

3.3 Self-supervised learning objectives

For a given image pair, denoted as I_a and I_b (depicting the same person in diverse pose and BGs), we forward them through two parallel pathways of our colored mesh estimation architecture (see Fig. 2). The commonness of FG appearance allows us to impose an appearance consistency loss between the predicted fully colored mesh representations.

a) **Color consistency.** First, we impose the following consistency loss,

$$\mathcal{L}_{CC} = \mathcal{L}_C + \lambda \mathcal{L}_{\tilde{C}}, \text{ where } \mathcal{L}_C = \|C_a - C_b\| \text{ and } \mathcal{L}_{\tilde{C}} = \|W_a \odot W_b \odot (\tilde{C}_a - \tilde{C}_b)\|$$

Here, \odot denotes element-wise multiplication. Note that, $\mathcal{L}_{\tilde{C}}$ enforces a vertex-color consistency on the co-visible mesh vertices (computed as $(W_a \odot W_b)$), *i.e.* the vertices which are visible in both the mesh representations obtained from the image pair, (I_a, I_b) . However, \mathcal{L}_C enforces full vertex color consistency. Here, \mathcal{L}_{CC} combines both of the losses thereby providing a higher weightage to the

co-visible vertex colors as compared to the approximate full color representation, considering the approximate nature of the symmetry assumption.

b) Part-prototype consistency. The proposed *Color-recovery* module can also be applied on the convolutional feature maps. For a given vertex k and a convolutional feature map $H \in \mathbb{R}^{\tilde{w} \times \tilde{h} \times \tilde{d}}$, we sample $\mathcal{H}^{(k)} \in \mathbb{R}^{\tilde{d}} = H(v^{(k)})$. Note that, we define a fixed vertex to part-segmentation mapping represented as $Q^{(l)}$, which stores a set of vertex indices for each part $l = 1, 2, \dots, L$. Now, one can use the vertex visibility weighing $W^{(k)}$ to obtain a prototype appearance feature for each body-part l , which is computed as; $\mathcal{F}^{(l)} = (\sum_{k \in Q^{(l)}} W^{(k)} \mathcal{H}^{(k)}) / (\sum_{k \in Q^{(l)}} W^{(k)})$. Following this, we enforce a prototype consistency loss between the image pairs as $\mathcal{L}_P = \sum_l \|\mathcal{F}_a^{(l)} - \mathcal{F}_b^{(l)}\| / L$. Note that, the prototype feature computation is inherently aware of the inter-part occlusions as a result of incorporating the visibility weighing $W^{(k)}$. As compared to enforcing vertex-color consistency, \mathcal{L}_{CC} (*i.e.* the raw color intensities), the part-prototype consistency aims to match a higher-level semantic abstraction (*e.g.* checkered regular patterns versus just plain individual colors) of the part appearances extracted from the image pairs. This also helps us to overcome the unreliability of raw vertex colors which could arise due to illumination differences. Motivated by the perceptual loss idea [17], we obtain H_a and H_b as the *Conv2-1* features corresponding to I_a and I_b from an ImageNet trained (frozen) VGG-16 network [59].

c) Shape-consistency. We also enforce a shape consistency loss between the shape parameters obtained from the image pair, *i.e.* $\mathcal{L}_\beta = |\beta_a - \beta_b|$. Almost all the prior works [21, 51, 50] utilize an *unpaired* human shape dataset to enforce plausibility of the shape predictions via adversarial prior. However, in the proposed self-supervised framework we do not access any human shape dataset. To regularize the shape parameters during the initial training iterations we enforce a loss on shape predictions with respect to a fixed mean shape as a regularization. However, after gaining a decent mesh estimation performance we gradually reduce weightage of this loss by allowing shape variations beyond the mean shape driven by the proposed appearance and shape consistency objectives.

d) Enforcing validity of pose predictions. Additionally, to assure validity of the predicted pose parameters we train an adversarial auto-encoder [42] to realize a continuous human pose manifold [29, 30] mapped from a latent pose representation, $\phi \in [-1, 1]^{32}$. This is trained using an unpaired 3D human pose dataset. The frozen pose decoder obtained from this generative framework is directly employed as a module, with instilled human 3D pose prior. More concretely, a *tanh* non-linearity on the pose-prediction head of the CNN regressor (inline with the latent pose ϕ) followed by the frozen pose decoder prevents implausible pose predictions during our self-supervised training. In contrast to enforcing an adversarial pose prior objective [21, 50], the proposed setup greatly simplifies our training procedure (devoid of discriminator training).

In absence of paired supervision, parameters of the shared CNN regressor is trained by directly enforcing the above consistency losses, *i.e.* \mathcal{L}_{CC} , \mathcal{L}_P , and \mathcal{L}_β .

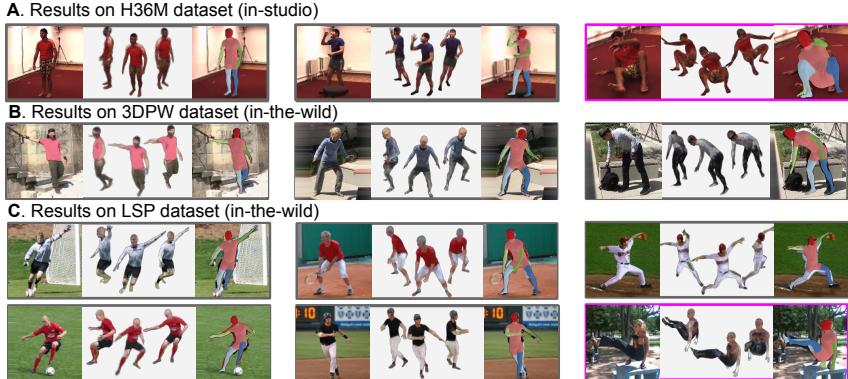


Fig. 3. Qualitative results. In each panel, 1st column depicts the input image, 2nd column depicts our colored mesh prediction, and 3rd column shows the model-based part segments. Our model fails (in magenta) in presence of complex inter-part occlusions.

4 Experiments

We perform thorough experimental analysis to demonstrate the generalizability of our framework across several datasets on a variety of tasks.

Implementation details. We use Resnet-50 [9] initialized from ImageNet as the base CNN network. The average pooled last layer features are forwarded through a series of fully-connected layers to regress the pose (latent pose encoding ϕ), shape and camera parameters. Note that, the series of differentiable operations post the CNN regressor do not include any trainable parameters even to estimate the vertex colors. During training, we optimize individual loss terms at alternate training iteration using Adam optimizer [25]. We enforce prediction of the mean shape for initial 100k training iterations. We also impose a silhouette loss on the predicted human mesh with respect to a pseudo silhouette ground-truth obtained either by using an unsupervised saliency detection method [72] or by using a background estimate as favourable for static camera scenarios [54].

Datasets. We sample image pairs with diverse BG (pairs with large L_2 distance) from the following standard datasets, *i.e.* Human3.6M [15], MPII [2], MPI-INF-3DHP [47] and an in-house collection of wild YouTube videos. In contrast to the in-studio datasets with hardly any camera movement implying static BG [15], the videos collected from YouTube have diverse camera movements (*e.g.* Parkour and Free-running videos). We prune the raw video samples using a person-detector [52] to obtain reliable human-centric crops as required for the mesh estimation pipeline (see Suppl). The unpaired 3D pose dataset required to train the 3D pose prior is obtained from CMU-MoCap (also used in MoSh [39]).

a) Human3.6M This is a widely used dataset consisting of paired image with 3D pose annotations of actors imitating various day-to-day tasks in a controlled in-studio environment. Adhering to well established standards [21] we consider subjects S1, S6, S7, S8 for training, S5 for validation and S9, S11 for evaluation, in both Protocol-1 [54,55] and Protocol-2 [21].

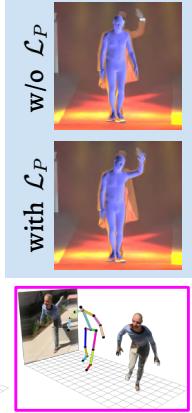
A. Results on YouTube, LSP and 3DPW dataset (in-the-wild)**B. Ablation results**

Fig. 4. A. Qualitative results on single image colored human mesh recovery. The model fails in presence of complex inter-limb occlusions (in magenta box). **B.** Qualitative analysis demonstrating importance of incorporating \mathcal{L}_P to extract relevant part-semantics.

b) LSP A standard 2D pose dataset consisting of wild athletic actions. We access the LSP test-set with silhouette and part segment annotations as given by Lassner *et al.* [35]. In absence of any standard shape evaluation dataset, segmentation results are considered as a proxy for the shape fitting performance [21,27].

c) 3DPW We also evaluate on the 3D Poses in the Wild dataset [43]. We do not train on 3DPW and use it only to evaluate our cross-dataset generalizability [31]. We compute the mean per joint position error (MPJPE) [15], both before and after rigid alignment. Rigid alignment is done via Procrustes Analysis [7]. MPJPE computed post Procrustes alignment is denoted by PA-MPJPE.

4.1 Ablative study

To analyze effectiveness of individual self-supervised consistency objectives, we perform ablations by removing certain losses as shown in Table 2. First, we train *Baseline-1* by enforcing \mathcal{L}_C and \mathcal{L}_β . Following this, in *Baseline-2* we enforce \mathcal{L}_{CC} by incorporating $\mathcal{L}_{\tilde{C}}$ which further penalizes color inconsistency between the vertices which are commonly visible in both the mesh representations. This results in marginal improvement of performance. Moving forward, we recognize a clear limitation in our assumption of FG color consistency (raw RGB intensities) which can easily be violated by illumination differences. Further, the assumption of left-right and front-back symmetry in apparel color can also be violated specifically for asymmetric upper body apparel. As a solution, the proposed part-prototype consistency objective, \mathcal{L}_P tries to match a higher level appearance representation beyond just raw color intensities (see Fig. 4B), thus resulting in a significant performance gain (*Ours(unsup)* in Table 2). Note that, \mathcal{L}_P is possible as a consequence of the proposed differentiable *Color-recovery* module.

Table 2. Ablative study (on Human3.6M) to analyze importance of self-supervised objectives (first 3 rows), and results at varied degree of paired supervision (last 3 rows). P1 and P2 denote MPJPE and PA-MPJPE in Protocol-1 and Protocol-2 respectively.

Methods	P1(\downarrow)	P2(\downarrow)
<i>Baseline-1</i> ; ($\mathcal{L}_C + \mathcal{L}_\beta$)	127.1	101.2
<i>Baseline-2</i> ; ($\mathcal{L}_{CC} + \mathcal{L}_\beta$)	119.6	97.4
<i>Ours(unsup.)</i> ; ($\mathcal{L}_{CC} + \mathcal{L}_\beta + \mathcal{L}_p$)	110.8	90.5
<i>Ours(multi-view-sup)</i>	102.1	74.1
<i>Ours(weakly-sup)</i>	86.4	58.2
<i>Ours(semi-sup)</i>	73.8	48.1

Table 3. Evaluation on wild 3DPW dataset in a *fully-unseen* setting. Note that, in contrast to Temporal-HMR [23] we do not use any temporal supervision. Methods in first 5 rows use equivalent 2D and 3D pose supervision, thus directly comparable.

Methods	MPJPE(\downarrow)	PA-MPJPE(\downarrow)
Martinez <i>et al.</i> [45]	-	157.0
SMPLify [5]	199.2	106.1
TP-Net [6]	163.7	92.3
Temporal-HMR [23]	127.1	80.1
<i>Ours(semi-sup)</i>	125.8	78.2
<i>Ours(weakly-sup)</i>	153.4	89.8
<i>Ours(unsup)</i>	187.1	102.7

Further, maintaining a fair comparison ground against the prior weakly supervised approaches, we train 3 variants of the proposed framework by utilizing increasing level of paired supervisions alongside our self-supervised objectives.

a) ***Ours(multi-view-sup)*** Under multi-view supervision, we impose additional consistency loss on the canonically aligned (view-invariant) 3D mesh vertices (*i.e.* $\|V_a - V_b\|$) and the 3D pose (*i.e.* $\|Y_a - Y_b\|$) for the time synchronized multi-view pairs, (I_a, I_b) . Inline with Rhodin *et al.* [54], we also use full 3D pose supervision only for S1 while evaluating on the standard Human3.6M dataset. We outperform Rhodin *et al.* [54] by a significant margin as reported in the Table 4. This is beyond the usual trend of weaker performance in non-parametric approaches against the model-based parametric ones. Thus, we attribute this performance gain to the proposed appearance consensus driven self-supervised objectives.

b) ***Ours(weakly-sup)*** In this setting, we access image datasets with paired 2D landmark annotations, inline with the supervision setting of prior model-based approaches [21]. Alongside the proposed self-supervised objectives, we impose a direct 2D landmark supervision loss (*i.e.* $\|y - y_{gt}\|$) with respect to the corresponding ground-truths but only on samples from specific datasets, such as LSP, LSP-extended [19] and MPII [2]. Certain prior arts, such as HMR [21], use even more images with paired 2D landmark annotations from COCO [37].

c) ***Ours(semi-sup)*** In this variant, we access paired 3D pose supervision on the widely used in-studio Human3.6M [15] dataset alongside the 2D landmark supervision as used in *Ours(weakly-sup)*. Note that, a better performance on Human3.6M (with limited BG and FG diversity as a result of the in-studio data collection setup) does not translate to the same on wild images as a result of the significant domain gap. As we impose the above supervisions alongside the proposed self-supervised objective on unlabeled wild images, such a training is expected to deliver improved performance by successfully overcoming the domain-shift issue. We evaluate this on the wild 3DPW dataset.

Table 4. Evaluation on Human3.6M (Protocol-2). Methods in first 9 rows use equivalent 2D and 3D pose supervision hence are directly comparable. Same analogy applies for the rows 10-11 and 12-13.

No.	Methods	PA-MPJPE(\downarrow)
1.	Lassner <i>et al.</i> [35]	93.9
2.	Pavlakos <i>et al.</i> [51]	75.9
3.	Omran <i>et al.</i> [48]	59.9
4.	HMR [21]	56.8
5.	Temporal HMR [23]	56.9
6.	Arnab <i>et al.</i> [4]	54.3
7.	Kolotouros <i>et al.</i> [27]	50.1
8.	TexturePose [50]	49.7
9.	<i>Ours(semi-sup)</i>	48.1
10.	HMR unpaired [21]	66.5
11.	<i>Ours(weakly-sup)</i>	58.2
12.	Rhodin <i>et al.</i> [54]	98.2
13.	<i>Ours(multi-view-sup)</i>	74.1

Table 5. Evaluation of FG-BG and 6-part segmentation on LSP test set. It reports accuracy (Acc.) and F1 score values of ours against the prior-arts. **First group:** Iterative, *optimization-based* approaches. **Last 3 groups:** *Regression-based* methods grouped based on comparable supervision levels.

Methods	FG-BG Seg.		Part Seg.	
	Acc.(\uparrow)	F1(\uparrow)	Acc.(\uparrow)	F1(\uparrow)
SMPLify <i>oracle</i> [5]	92.17	0.88	88.82	0.67
SMPLify [5]	91.89	0.88	87.71	0.64
SMPLify on [51]	92.17	0.88	88.24	0.64
Bodynet [65]	92.75	0.84	-	-
HMR [21]	91.67	0.87	87.12	0.60
Kolotouros <i>et al.</i> [27]	91.46	0.87	88.69	0.66
TexturePose [50]	91.82	0.87	89.00	0.67
<i>Ours(semi-sup)</i>	91.84	0.87	89.08	0.67
HMR unpaired [21]	91.30	0.86	87.00	0.59
<i>Ours(weakly-sup)</i>	91.70	0.87	87.12	0.60
<i>Ours(unsup)</i>	91.46	0.86	87.26	0.64

4.2 Comparison with the state-of-the-art

Evaluation on Human3.6M. Table 4 shows a comparison of different variants of the proposed framework against the prior-arts which are grouped based on the respective supervision levels. We clearly outperform in all the three groups *i.e.* while accessing comparable a) 3D pose supervision, b) 2D landmark supervision, and c) multi-view supervision. Except Rhodin *et al.* [54] all the prior works mentioned in Table 4 use parametric human model for the human mesh estimation task. Note the significant performance gain specifically in absence of any 3D pose supervision, *i.e.* for *Ours(weakly-sup)* and *Ours(multi-view-sup)* against the relevant counterparts as reported in the last 4 rows.

Evaluation on 3DPW. Table 3 reports a comparison of different variants of the proposed framework against the prior-arts which use comparable pose supervision as used in *Ours(semi-sup)* (except certain methods, such as HMR [21] which use even more supervision on 3D pose from the MPI-INF-3DHP [47] dataset). It is worth noting that none of our model variants is trained on the samples from 3DPW dataset (not even in self-supervised paradigm). A better performance in such *unseen* setting highlights our superior cross-dataset generalizability.

Evaluation of part-segmentation. We also evaluate our performance on FG-BG segmentation and body part-segmentation tasks which are considered as a proxy to quantify the shape fitting performance. In presence of 2D landmark annotation, iterative model fitting approaches have a clear advantage over the single-shot regressor based approaches as shown in Table 5. At comparable supervision, *Ours(semi-sup)* not only outperforms the relevant regression based prior arts but also performs competitive to the iterative model fitting based approaches with a significant advantage on inference time (1 min vs 0.04 sec.).

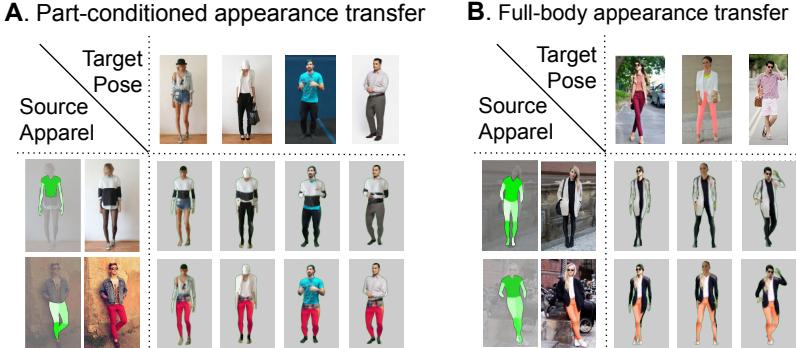


Fig. 5. Qualitative results on **A.** Part-conditioned, and **B.** Full-body appearance transfer. This is enabled as a result of our ability to infer the colored mesh representation.

Note that, *Ours(unsup)* performs competitive to the prior supervised regression-based approaches, thus establishing the importance of FG appearance consistency for accurate shape recovery.

4.3 Qualitative results

The proposed mesh recovery model not only infers pose and shape but also outputs a colored mesh representation as a result of the proposed *reflectional-symmetry* procedure. To evaluate effectiveness of the recovered part appearance we perform 2 different tasks a) part-conditioned appearance transfer, and b) full-body appearance transfer as shown in Fig. 5. On the top, we show the target images whose pose and shape (network predicted) is combined with part appearances recovered from the source image (only for the highlighted parts) shown on left, to realize a novel synthesized image. Note that, in case of *part-conditioned* appearance transfer, appearance of the non-highlighted parts are taken from the target image shown on the top. For instance, in the first row, the synthesized image depicts upper-body apparel of the person in the source image combined with the lower-body apparel from the target (and in the target image pose). Qualitative results of *Ours(semi-sup)* model on other primary tasks are shown in Fig. 3 and Fig. 4 with highlighted failure scenarios (see Suppl.).

5 Conclusion

We introduce a self-supervised framework for model-based human pose and shape recovery. The proposed appearance consistency not only helps us to segregate the common FG human from their respective wild BGs but also discovers the required pose deformation in a fully self-supervised manner. However, extending such a framework for human centric images with occlusion by external objects or truncated human visibility, remains to be explored in future.

Acknowledgements. We thank Qualcomm Innovation Fellowship India 2020.

Supplementary Material

Apearance Consensus Driven Self-Supervised Human Mesh Recovery

In this supplementary, we summarize the proposed differentiable colored-mesh recovery procedure followed by additional implementation details and qualitative results. Follow our project page¹ for more details.

The supplementary material is organized as follows:

- Section 1: Differentiable operations in the proposed framework
- Section 2: Sampling image pairs with diverse background
- Section 3: Reflectional symmetry groups
- Section 4: Qualitative evaluation

Table 1. A list of notations and their size as used in the main paper.

Notations	Description	Size
I	Input image	$224 \times 224 \times 3$
θ	View invariant SMPL pose	$3J (J=23)$
β	SMPL Shape parameter	10
ϕ	Pose embedding	32
V	3D vertex locations	6890×3
C	Vertex colors (RGB)	6890×3
v	Image-projected vertex locations	6890×2
N	Z-component of Camera-space normals	6890×1
Z	Camera-space depth of mesh vertices	6890×1
$I^z(u)$	Rendered depth image	$224 \times 224 \times 3$
W	Visibility-aware-weighting	6890×1
\tilde{C}	Intermediate Vertex colors (RGB)	6890×3
S	Vertex to symmetry group mapping	$(1575+295) \times 6890$
C	Group color for symmetry groups	1870×3
Q	Vertex to part-segmentation mapping	-
H	$Conv2\text{-}1$ output of pre-trained VGG-16	$112 \times 112 \times 128$
\mathcal{H}^k	Sampled feature from H at $v^{(k)}$	$\tilde{d} = 128$
\mathcal{F}	Part-prototype appearance feature	128
k	Index over mesh vertices	$K=6890$
g	Index over symmetry groups	$G=1870$
l	Index over body parts	$L=14$
a, b	Indicating association with inputs I_a, I_b	-

1 Differentiable operations in the proposed framework

We propose three completely differentiable modules in order to realize our self-supervised approach namely the color-recovery module, part-prototype module

¹ Project-page: <https://sites.google.com/view/ss-human-mesh>

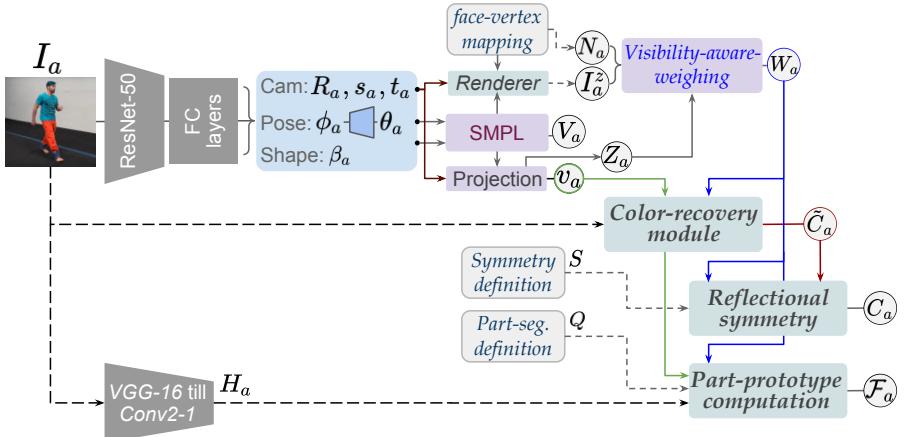


Fig. 1. The series of differentiable computations and their interdependence as employed in the proposed self-supervised mesh recovery framework (see Table 1 for the notations).

and the reflectional symmetry module. See Fig. 1 for an illustration of the differentiable computations and their interdependence.

a) Obtaining *visibility-aware-weighing*, W : All the modules use a differentiable *visibility-aware-weighing*, W to softly segregate the 3D vertices based on their visibility for a given (or predicted) camera view. The computation of W relies on the fact that visible vertices are influenced by two factors (i) camera and (ii) human skeleton self-occlusion. We identify camera facing vertices using the z-component of the Normal (N) while we handle human self-occlusion by soft selection based on a camera-centric depth image and a margin in z-buffering.

$$W^{(k)} \in [0, 1] = \exp(-\alpha D^{(k)}) \sigma(\gamma N^{(k)}), \text{ where } D^{(k)} = |I^z(v^{(k)}) - Z^{(k)}|$$

b) Recovering intermediate color, \tilde{C} : Next, we obtain the intermediate, visibility-aware colors, \tilde{C} by weighting the raw picked colors (done by bilinear sampling of image I , given the vertex 2D projection $v^{(k)}$) as shown below.

$$\tilde{C}^{(k)} = I(v^{(k)}) (2W^{(k)} - 1), \text{ where } I(v^{(k)}) \text{ denotes RGB color at the } v^{(k)}$$

c) Applying reflectional symmetry to obtain the full vertex color, C : Next, we focus on propagating the color intensities from the visible vertices (as stored in the intermediate \tilde{C}) to the invisible ones (*i.e.* the vertices having low $W^{(k)}$). To realize a fully-colored mesh C , we use a predefined, 4-way symmetry grouping knowledge (front-back and left-right) as stored in S . First the group colors $\mathcal{C}^{(g)}$ are computed as a normalized combination of the intermediate vertex colors weighted by their visibility weighing W . Then, the group colors are directly propagated to all the mesh vertices using S as shown in the following equation.

$$C = S^T * \mathcal{C}, \text{ where } \mathcal{C}^{(g)} = (S^{(g)} \circ \text{ReLU}(\tilde{C})) / (S^{(g)} \circ \text{ReLU}(2W - 1))$$

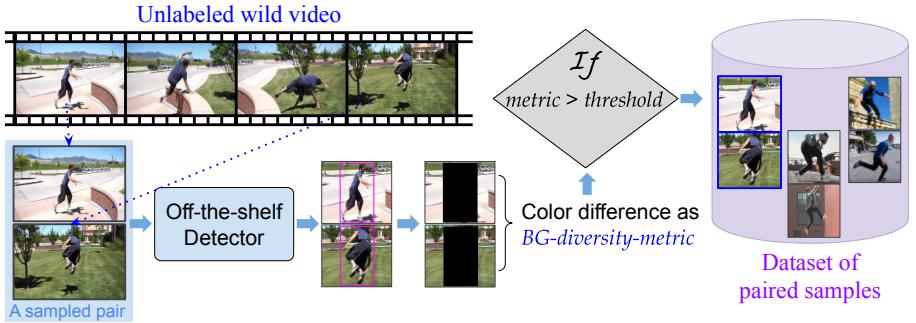


Fig. 2. An illustration of the adopted procedure to sample image pairs of diverse background. To built the dataset of image pairs as required by the proposed self-supervised framework, we chose image pairs depicting the same person in diverse pose (maintaining a considerable temporal gap) and background (via *BG-diversity-metric*).

d) Computation of part-prototype features, \mathcal{F} : Here, we reuse the color-recovery idea to realize part-prototype features. $\mathcal{F}^{(l)}$ is computed as the normalized weighted sum, of the recovered spatial features $\mathcal{H}^{(k)} = H(v^{(k)})$, over the vertices belonging to the part l (using vertex to part-segmentation mapping, Q).

$$\mathcal{F}^{(l)} = (\sum_{k \in Q^{(l)}} W^{(k)} \mathcal{H}^{(k)}) / (\sum_{k \in Q^{(l)}} W^{(k)}), \text{ where } \mathcal{H}^{(k)} = H(v^{(k)})$$

2 Sampling image pairs with diverse background

Given a video clip depicting actions of a single person, in consistent apparel, we aim to sample image pairs which would have diverge background (BG) appearance. To realize this, we first prune the video frames using an off-the-shelf person-detector [3] to obtain a reliable human-centric crop as required for the mesh estimation pipeline. Following this, we compute $L2$ distance (mean squared error) between image pairs, only for the regions outside the detector box, to obtain a *BG-diversity-metric* (see Fig. 2). Among all possible frame pairs (beyond 1 sec temporal gap), we choose the pairs having *BG-diversity-metric* greater than a certain threshold value. In contrast to the in-studio datasets with hardly any camera movement implying static BG [1], our in-house collection of YouTube videos have diverse camera movements (*e.g.* Parkour and Free-running videos). The wild camera movement inherently results in huge diversity in the sampled image pairs. Note that, in static camera scenarios BG diversity occurs when the person moves from one location to another. This is because, instead of taking the full video feed, we consider a square region around the detector output as the effective input to the CNN regressor.

3 Reflectional symmetry groups

We define reflectional groups where each group constitutes a set of vertices which is assumed to have similar color property. Though this assumption does not hold

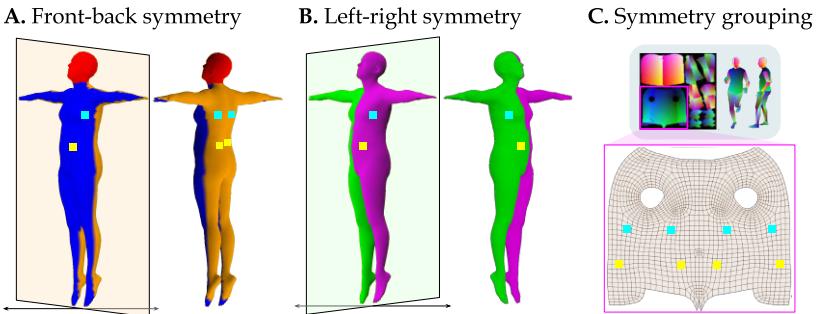


Fig. 3. An illustration of front-back and left-right symmetry. This is used to define the multi-hot encoding, $S^{(g)} \in \{0, 1\}^K$ which constitutes of four ones indicating vertex members in the symmetry group g . Here, "yellow" and "cyan" color patches show rough location of the vertices for two such symmetry groups. Note that, for the head region only left-right symmetry is used (red colored region in panel A).

true in presence of illumination difference and non-symmetric apparel design, we find this to be helpful in general because of the following reasons. Firstly, it is rare to encounter non-symmetric apparel with diverse color difference between the left-right or front-back. Secondly, although the luminosity property (*i.e.* intensity) is influenced in presence of illumination difference, the color property (*i.e.* hue) remains comparable. However, the consistency loss on the part-prototypes and also on the intermediate vertex color \tilde{C} helps us to effectively balance this shortcomings. Broadly, we define 2 types of symmetry groups; a) group sets of 2 members (vertex indices) only for the head region (295 groups), and b) group sets of 4 members (vertex indices) for rest of the body parts (1575 groups). See Fig. 3 for a rough illustration. 4-membered groups are obtained by applying both front-back and left-right symmetry. However, 2-membered groups represent only left-right symmetry. Note that all the group sets are mutually exclusive and exhaustive, *i.e.* $2*295 + 4*1575 = 6890$, where 6890 is the total number of vertices. This symmetry knowledge is stored as a multi-hot encoding denoted as $S^{(g)} \in \{0, 1\}^K$ which constitutes of four ones indicating vertex members in the symmetry group g . All the symmetry groups are combined in a symmetry-encoding matrix represented as $S \in \{0, 1\}^{G \times K}$. This multi-hot symmetry group representation helps us to perform a fully-differentiable vertex color assignment for all the vertices including the occluded and non-camera facing ones.

4 Qualitative evaluation

In order to evaluate the generalizability of our model, we visualize our model's 3D pose and shape performance on a variety of images sampled from different datasets. Fig 4 shows the predicted colored mesh and the corresponding 3D pose in aligned grid plots. Fig. 5 shows a qualitative analysis on the standard 6-part mesh overlay. Here, mesh overlay can be considered as a proxy to evaluate both shape and pose in a collective fashion.

A. Results on YouTube, LSP and 3DPW dataset (in-the-wild)

Fig. 4. Qualitative results on single image colored human mesh recovery.

Although, pose and shape are predicted correctly, background leaks could occur at misaligned locations that can be visualized using the coloured mesh reconstructions as shown in Fig 4. Also note that, SMPL [2] does not parameterize hand pose hence hands remain in a fixed mean pose (flat open hand). This tends to be a consistent location for background leakage. Background leakages are observed to generally occur at boundaries of hands and feet; *e.g.*, in row 2 last column of Fig 4 the green background leaks onto the appearance of the hand due to the limitation of the parametric human model in articulating the exact hand pose. Also, our model outputs sub-optimal results in cases with complex inter-limb occlusions as highlighted in magenta in Fig. 5.

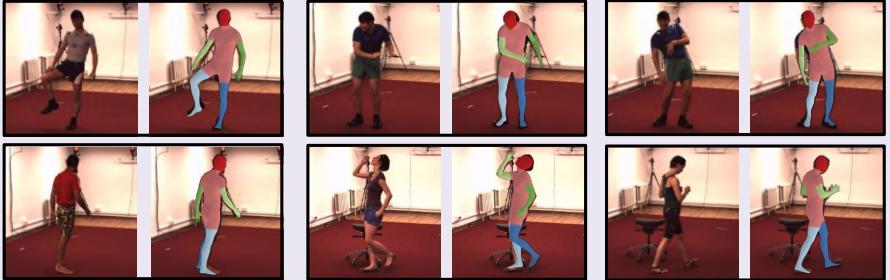
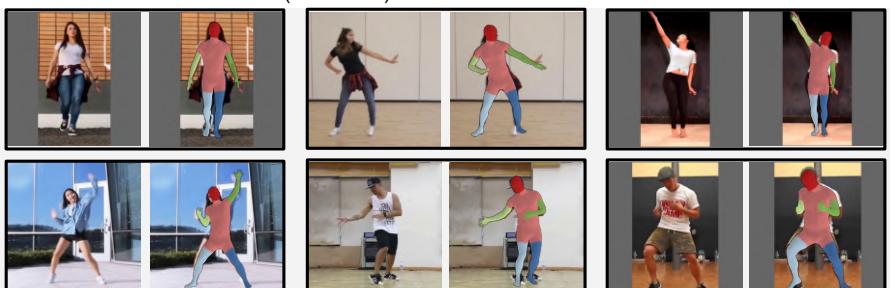
A. Results on H36M dataset (in-studio)**B. Results on 3DPW dataset (in-the-wild)****C. Results on LSP dataset (in-the-wild)****D. Results on YouTube dataset (in-the-wild)**

Fig. 5. Qualitative results. In each panel, 1st column depicts the input image, 2nd column shows the model-based part segments on **A.** Human3.6M (in-studio), **B.** 3DPW (in-the-wild) **C.** LSP (in-the-wild) **D.** YouTube (in-the-wild). The model fails in presence of complex inter-limb occlusions (in magenta box).

References

1. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence* (2013) [3](#)
2. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. *ACM transactions on graphics* (2015) [5](#)
3. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: *NeurIPS*. pp. 91–99 (2015) [3](#)

References

1. Alp Güler, R., Neverova, N., Kokkinos, I.: Densepose: Dense human pose estimation in the wild. In: CVPR (2018) [3](#), [4](#)
2. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: CVPR (2014) [2](#), [5](#), [10](#), [12](#)
3. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. In: ACM SIGGRAPH (2005) [1](#), [4](#)
4. Arnab, A., Doersch, C., Zisserman, A.: Exploiting temporal context for 3d human pose estimation in the wild. In: CVPR (2019) [1](#), [3](#), [13](#)
5. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In: ECCV (2016) [2](#), [4](#), [12](#), [13](#)
6. Dabral, R., Mundhada, A., Kusupati, U., Afaque, S., Sharma, A., Jain, A.: Learning 3d human pose from structure and motion. In: ECCV (September 2018) [12](#)
7. Gower, J.C.: Generalized procrustes analysis. Psychometrika **40**(1), 33–51 (1975) [11](#)
8. Guan, P., Weiss, A., Balan, A.O., Black, M.J.: Estimating human shape and pose from a single image. In: ICCV (2009) [2](#)
9. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: ECCV (2016) [10](#)
10. Henderson, P., Ferrari, V.: Learning single-image 3D reconstruction by generative modelling of shape, pose and shading. International Journal of Computer Vision (2019) [5](#), [7](#)
11. Hofmann, M., Gavrila, D.M.: Multi-view 3d human pose estimation combining single-frame recovery, temporal integration and model adaptation. In: CVPR (2009) [5](#)
12. Hogg, D.: Model-based vision: a program to see a walking person. Image and Vision computing **1**(1), 5–20 (1983) [1](#)
13. Hsu, K.J., Tsai, C.C., Lin, Y.Y., Qian, X., Chuang, Y.Y.: Unsupervised cnn-based co-saliency detection with graphical optimization. In: ECCV (2018) [3](#)
14. Huang, Y., Bogo, F., Lassner, C., Kanazawa, A., Gehler, P.V., Romero, J., Akhter, I., Black, M.J.: Towards accurate marker-less human shape and pose estimation over time. In: 3DV (2017) [1](#)
15. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE transactions on pattern analysis and machine intelligence (2013) [10](#), [11](#), [12](#)
16. Jakab, T., Gupta, A., Bilen, H., Vedaldi, A.: Unsupervised learning of object landmarks through conditional image generation. In: NeurIPS (2018) [3](#)
17. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016) [9](#)
18. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC (2010) [2](#)
19. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: BMVC (2010) [12](#)
20. Joo, H., Simon, T., Sheikh, Y.: Total capture: A 3d deformation model for tracking faces, hands, and bodies. In: CVPR (2018) [1](#)
21. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: CVPR (2018) [2](#), [3](#), [4](#), [5](#), [6](#), [9](#), [10](#), [11](#), [12](#), [13](#)

22. Kanazawa, A., Tulsiani, S., Efros, A.A., Malik, J.: Learning category-specific mesh reconstruction from image collections. In: ECCV (2018) [3](#), [4](#), [5](#)
23. Kanazawa, A., Zhang, J.Y., Felsen, P., Malik, J.: Learning 3d human dynamics from video. In: CVPR (2019) [3](#), [12](#), [13](#)
24. Kato, H., Ushiku, Y., Harada, T.: Neural 3d mesh renderer. In: CVPR (2018) [5](#)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) [10](#)
26. Kolotouros, N., Pavlakos, G., Black, M.J., Daniilidis, K.: Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In: ICCV (2019) [3](#)
27. Kolotouros, N., Pavlakos, G., Daniilidis, K.: Convolutional mesh regression for single-image human shape reconstruction. In: CVPR (2019) [3](#), [11](#), [13](#)
28. Kundu, J.N., Ganeshan, A., MV, R., Prakash, A., Babu, R.V.: iSPA-Net: Iterative semantic pose alignment network. In: ACM Multimedia (2018) [2](#)
29. Kundu, J.N., Gor, M., Babu, R.V.: BiHMP-GAN: Bidirectional 3d human motion prediction gan. In: AAAI (2019) [9](#)
30. Kundu, J.N., Gor, M., Uppala, P.K., Babu, R.V.: Unsupervised feature learning of human actions as trajectories in pose embedding manifold. In: WACV (2019) [9](#)
31. Kundu, J.N., Patravali, J., Babu, R.V.: Unsupervised cross-dataset adaptation via probabilistic amodal 3d human pose completion. In: WACV (2020) [11](#)
32. Kundu, J.N., Seth, S., Jampani, V., Rakesh, M., Babu, R.V., Chakraborty, A.: Self-supervised 3d human pose estimation via part guided novel image synthesis. In: CVPR (2020) [2](#)
33. Kundu, J.N., Seth, S., Rahul, M., Rakesh, M., Babu, R.V., Chakraborty, A.: Kinematic-structure-preserved representation for unsupervised 3d human pose estimation. In: AAAI (2020) [3](#)
34. L Navaneet, K., Mandikal, P., Jampani, V., Babu, V.: Differ: Moving beyond 3d reconstruction with differentiable feature rendering. In: CVPR Workshops (2019) [4](#)
35. Lassner, C., Romero, J., Kiefel, M., Bogo, F., Black, M.J., Gehler, P.V.: Unite the people: Closing the loop between 3d and 2d human representations. In: CVPR (2017) [2](#), [4](#), [5](#), [11](#), [13](#)
36. Liang, J., Lin, M.C.: Shape-aware human pose and shape reconstruction using multi-view images. In: ICCV (2019) [5](#)
37. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV (2014) [5](#), [12](#)
38. Liu, S., Li, T., Chen, W., Li, H.: Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In: ICCV (2019) [4](#)
39. Loper, M., Mahmood, N., Black, M.J.: Mosh: Motion and shape capture from sparse markers. ACM Transactions on Graphics (TOG) **33**(6), 220 (2014) [10](#)
40. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. ACM transactions on graphics (2015) [1](#), [4](#), [5](#)
41. Ma, L., Sun, Q., Georgoulis, S., Van Gool, L., Schiele, B., Fritz, M.: Disentangled person image generation. In: CVPR (2018) [2](#)
42. Makhzani, A., Shlens, J., Jaitly, N., Goodfellow, I., Frey, B.: Adversarial autoencoders. arXiv preprint arXiv:1511.05644 (2015) [9](#)
43. von Marcard, T., Henschel, R., Black, M.J., Rosenhahn, B., Pons-Moll, G.: Recovering accurate 3d human pose in the wild using imus and a moving camera. In: ECCV (2018) [11](#)
44. von Marcard, T., Rosenhahn, B., Black, M.J., Pons-Moll, G.: Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. In: Computer Graphics Forum. vol. 36, pp. 349–360. Wiley Online Library (2017) [1](#)

45. Martinez, J., Hossain, R., Romero, J., Little, J.J.: A simple yet effective baseline for 3d human pose estimation. In: ICCV (2017) **1, 12**
46. Mathieu, M.F., Zhao, J.J., Zhao, J., Ramesh, A., Sprechmann, P., LeCun, Y.: Disentangling factors of variation in deep representation using adversarial training. In: NeurIPS. pp. 5040–5048 (2016) **2**
47. Mehta, D., Rhodin, H., Casas, D., Fua, P., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3d human pose estimation in the wild using improved cnn supervision. In: 3DV (2017) **10, 13**
48. Omran, M., Lassner, C., Pons-Moll, G., Gehler, P., Schiele, B.: Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In: 3DV (2018) **2, 3, 4, 5, 13**
49. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: CVPR (2019) **1**
50. Pavlakos, G., Kolotouros, N., Daniilidis, K.: Texturepose: Supervising human mesh estimation with texture consistency. In: ICCV (2019) **2, 5, 9, 13**
51. Pavlakos, G., Zhu, L., Zhou, X., Daniilidis, K.: Learning to estimate 3d human pose and shape from a single color image. In: CVPR (2018) **2, 3, 4, 5, 9, 13**
52. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: NeurIPS. pp. 91–99 (2015) **10**
53. Rhodin, H., Robertini, N., Casas, D., Richardt, C., Seidel, H.P., Theobalt, C.: General automatic human shape and motion capture using volumetric contour cues. In: ECCV (2016) **1**
54. Rhodin, H., Salzmann, M., Fua, P.: Unsupervised geometry-aware representation for 3d human pose estimation. In: ECCV (2018) **3, 5, 10, 12, 13**
55. Rhodin, H., Spörri, J., Katircioglu, I., Constantin, V., Meyer, F., Müller, E., Salzmann, M., Fua, P.: Learning monocular 3d human pose estimation from multi-view images. In: CVPR (2018) **10**
56. Rifai, S., Bengio, Y., Courville, A., Vincent, P., Mirza, M.: Disentangling factors of variation for facial expression recognition. In: ECCV (2012) **2**
57. Rogez, G., Weinzaepfel, P., Schmid, C.: Lcr-net: Localization-classification-regression for human pose. In: CVPR (2017) **1**
58. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM Transactions on Graphics (ToG) **36**(6), 245 (2017) **1**
59. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014) **9**
60. Song, S., Yu, F., Zeng, A., Chang, A.X., Savva, M., Funkhouser, T.: Semantic scene completion from a single depth image. In: CVPR (2017) **4**
61. Sun, X., Xiao, B., Wei, F., Liang, S., Wei, Y.: Integral human pose regression. In: ECCV (2018) **1**
62. Sun, Y., Ye, Y., Liu, W., Gao, W., Fu, Y., Mei, T.: Human mesh recovery from monocular images via a skeleton-disentangled representation. In: ICCV (2019) **1, 3**
63. Tan, V., Budvytis, I., Cipolla, R.: Indirect deep structured learning for 3d human body shape and pose prediction. In: BMVC (2017) **2, 5**
64. Tung, H.Y., Tung, H.W., Yumer, E., Fragkiadaki, K.: Self-supervised learning of motion capture. In: NIPS (2017) **2, 5**
65. Varol, G., Ceylan, D., Russell, B., Yang, J., Yumer, E., Laptev, I., Schmid, C.: Bodynet: Volumetric inference of 3d human body shapes. In: ECCV (2018) **13**
66. Varol, G., Romero, J., Martin, X., Mahmood, N., Black, M.J., Laptev, I., Schmid, C.: Learning from synthetic humans. In: CVPR (2017) **5**

67. Weiss, A., Hirshberg, D., Black, M.J.: Home 3d body scans from noisy image and range data. In: ICCV (2011) [1](#)
68. Zanfir, A., Marinoiu, E., Sminchisescu, C.: Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints. In: CVPR (2018) [4](#)
69. Zanfir, A., Marinoiu, E., Zanfir, M., Popa, A.I., Sminchisescu, C.: Deep network for the integrated 3d sensing of multiple people in natural images. In: NIPS (2018) [2](#), [4](#)
70. Zhang, D., Meng, D., Han, J.: Co-saliency detection via a self-paced multiple-instance learning framework. IEEE transactions on pattern analysis and machine intelligence **39**(5), 865–878 (2016) [3](#)
71. Zhou, X., Huang, Q., Sun, X., Xue, X., Wei, Y.: Towards 3d human pose estimation in the wild: a weakly-supervised approach. In: CVPR (2017) [5](#)
72. Zhu, W., Liang, S., Wei, Y., Sun, J.: Saliency optimization from robust background detection. In: CVPR (2014) [10](#)