# Disentangling 3D Attributes from a Single 2D Image: Human Pose, Shape and Garment

Xue Hu[2†]
xue.hu17@imperial.ac.uk
Xinghui Li[3†]
xinghui@robots.ox.ac.uk
Benjamin Busam[4]
b.busam@tum.de
Yiren Zhou[1]
zhouyiren@huawei.com
Ales Leonardis[1]
ales.leonardis@huawei.com
Shanxin Yuan[1‡]
shanxinyuan@gmail.com

[1] Huawei Noah's Ark Lab
[2] Imperial College London
[3] University of Oxford
[4] Technical University of Munich

**Abstract**

For visual manipulation tasks, we aim to represent image content with semantically meaningful features. However, learning implicit representations from images often lacks interpretability, especially when attributes are intertwined. We focus on the challenging task of extracting disentangled 3D attributes only from 2D image data. Specifically, we focus on human appearance and learn implicit pose, shape and garment representations of dressed humans from RGB images. Our method learns an embedding with disentangled latent representations of these three image properties and enables meaningful re-assembling of features and property control through a 2D-to-3D encoder-decoder structure. The 3D model is inferred solely from the feature map in the learned embedding space. To the best of our knowledge, our method is the first to achieve cross-domain disentanglement for this highly under-constrained problem. We qualitatively and quantitatively demonstrate our framework's ability to transfer pose, shape, and garments in 3D reconstruction on virtual data and show how an implicit shape loss can benefit the model's ability to recover fine-grained reconstruction details.

## 1 Introduction

If you reconstruct a 3D model from a single image, you also want the power to control its content in a meaningful way. This wish has created a wide range of applications that leverage learning of implicit representations by disentanglement, such as face identity swapping [10, 46], hairstyle transfer [16, 38] and pose and shape transfer [17, 28, 47]. The main objective


† Equal contribution
‡ Corresponding author

of implicit representation learning is to disentangle and encode information regarding each characteristic of input signals such that a new sample can be generated by manipulating learned representations. For example, if we separate pose and shape information from the 3D model of a person, we can achieve pose transfer by simply replacing the pose information while keeping the original shape information.

However, the current literature only discusses cases where input and output are from the same domain. This means that if the input is an image or a 3D mesh, the output would also take the same form as the input and learned representations can only control the output in the same domain. Due to this restriction, current works are difficult to be directly applied to 2D-to-3D tasks such as Augmented Reality (AR), where 3D information is often inferred from easily acquired 2D images [8, 9, 20, 26, 41].

In this paper, we propose a solution to this highly under-constrained problem. We focus on learning pose, shape and garment representations of dressed human bodies from 2D RGB images and use these representations to manipulate corresponding 3D models. Our inspiration is drawn from the fact that the 3D mesh of a dressed human can be solely estimated from the feature map of its images [35, 36] by training a shape prior. A key deduction from this observation is that the feature map of an object's 2D signal contains sufficient information to construct its 3D model if a shape prior is provided. Therefore, if we disentangle and reconstruct feature maps rather than input signals themselves, we can subsequently control the final 3D models which are inferred from the feature maps using the shape prior.

Our method consists of three parts: a feature extractor, a multi-head encoder-decoder and an MLP, as illustrated in Fig. 1. A feature extractor firstly extracts a feature map from the input image. Further, a consecutive feature extractor learns an embedding that encodes disentangled representations for pose, shape and garment into three respective latent codes. The feature map is then recovered from the latent codes, and finally, an MLP is used as a shape prior to construct the 3D model based on the pixel-aligned feature interpolated from the feature map. To change the pose, shape or garment of the 3D model individually while keeping the other properties, we can change the corresponding latent code, which consequently changes the generated 3D mesh.

To the best of our knowledge, our model is the first method to use representations learned from 2D input to control 3D output. Although we exemplify the power of the method with dressed human bodies, such a principle can be generalised to any class of object if the shape prior can be properly trained. In contrast to standard human modelling methods [2, 19, 23, 34], we are able to control the model without using a template. To summarize, our contributions are twofold:

- We propose a method to disentangle 3D shape attributes from 2D image data. We exemplify the principle by learning pose, shape and garment representations of dressed humans from 2D images and allow expressive control of reconstructed 3D model from disentangled feature sub-manifolds. Our method is the first to achieve 2D-to-3D representation learning and output manipulation.

- We analyse design choices and provide experimental evidence of controlled feature manipulation for pose, shape and garment representations from 2D input on a publicly available dataset [32].

## 2 Related Work

**Disentanglement** The objective of disentanglement is to find underlying latent representations that control the variation of the data. The pioneering work in this area is InfoGAN [7] which is based on Generative Adversarial Networks (GANs) [11] and aims to maximize the mutual information between the latent codes and generator distribution. $\beta$-VAE [16] and its variant [6] use a Variational Autoencoder (VAE) [21] rather than GAN model and penalize a KL-Divergence term to enhance the independence within latent space.

Disentanglement can be applied to both 2D image inputs and 3D inputs, to achieve either attribute transfer like face identity swapping [10, 46] and hairstyle transfer [16, 38] on 2D images, or pose and shape transfer [17, 28, 47] on 3D inputs. Our method is more related to [47] where pose and shape representations are learned in an unsupervised manner. The most significant differences between our method and most existing pipelines are that our input and output are from different domains (2D input and 3D output), and we disentangle the image's feature map which is an abstract representation rather than the raw data. These differences make our task much more challenging.

**Parametric Human Modelling** Parametric models targeting humans initially focus on the parameterization of the naked human body. Various body templates, and skeleton hands [44, 45], have been proposed in the past decade [2, 14, 23, 30]. Such a parameterization has recently been generalized to the modelling of the garment [4, 13, 22, 24, 25, 32, 39]. Compared to the naked human body, the parameterization of the garment is much more difficult due to the complicated local details such as wrinkles and foldings. The garment models are usually associated with the naked body model such as SMPL. They are either represented as separated SMPL-like templates [32, 39, 48] or displacement fields to the body model [1, 24]. Some works [3, 32] additionally model the intra-class garment variation using statistical tools such as principle component analysis (PCA).

**Implicit Neural Representation** Implicit neural representation aims to represent a 3D object using an implicit function which typically takes the form of an MLP. The pioneering works are OccNet [27] and DeepSDF [31] which encode the objects as an occupancy field and a signed distance field, respectively. Compared with traditional 3D representation such as voxel or mesh, neural representation allows a continuous surface representation which circumvents the loss of accuracy due to discretization. Following these two works, many variations of implicit functions [5, 18, 33, 40] and training losses [12] have been proposed. The aforementioned methods mainly focus on 3D input such as point clouds, but PiFU [35] generalizes the implicit function to 2D images by proposing a pixel-aligned implicit function. Instead of a 3D coordinate, the function takes in the corresponding 2D image feature of the 3D location and the depth value, and it can infer a wide variety of human poses and shapes. DISN [42] also has a similar structure to construct static objects. Our method is closely related to PiFU as we disentangle pose and shape information from the image's feature map.

## 3 Method

Our method intends to disentangle pose, shape and garment representations of a dressed human from a single RGB image and simultaneously infer the 3D mesh of it. Existing works [35, 36] have already demonstrated that 3D mesh can be inferred solely from a single image

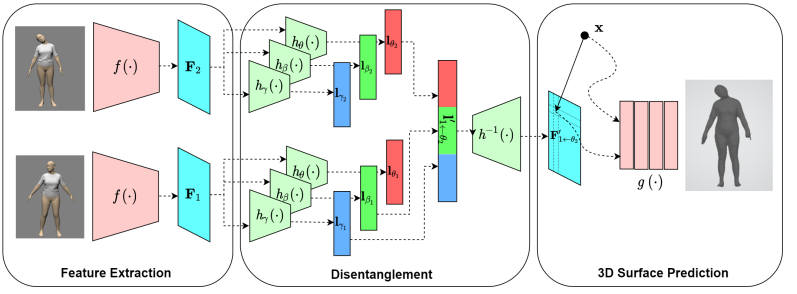


Figure 1: This figure illustrates the architecture of our method. The feature map is $\mathbf{F}$ is extracted from the image by the feature extractor $f(\cdot)$. The feature map is then disentangled into three latent codes $\mathbf{l}_\theta$, $\mathbf{l}_\beta$ and $\mathbf{l}_\gamma$ which correspond to pose, shape and garment representations, respectively. By swapping the latent code, a feature map $\mathbf{F}'$ with swapped information is recovered by $h^{-1}(\cdot)$, from which the 3D mesh with swapped property is inferred as the implicit expression by $g(\cdot)$, where $g(\cdot)$ takes both a 3D coordinate $\mathbf{x}$ and the aligned feature of the projection of $\mathbf{x}$ on the image.

by pixel-aligned implicit function based on the feature map of the image. This indicates that such a feature map contains all information regarding the target mesh; hence, pose, shape and garment representation can be disentangled. Therefore, our method consists of three parts: a feature extractor that extracts the feature map from the image, a multi-head encoder-decoder that disentangles and reconstructs the feature map, and an MLP which infers the 3D mesh from the feature map. We illustrate this pipeline in Fig. 1.

## 3.1 Feature Disentanglement

After feature extraction, the feature disentanglement is achieved by a multi-head autoencoder module. Given an image $\mathbf{I}_1$, its feature map $\mathbf{F}_1 \in \mathbb{R}^{H\times W\times C}$ can be extracted using the feature extractor $f(\cdot)$. Such a feature map encodes all the necessary information to construct a 3D model, as the missing depth information would be compensated by the implicit function, which acts like a 3D human shape prior [35]. Hence, we only need to disentangle the feature map so that a new feature map and thus the 3D human body can be reconstructed after latent editing using the encoded information from an image of the desired human model.

The disentanglement network follows a similar design to the work of Zhou *et al.* [47]. To disentangle the pose, shape and garment properties, the feature map $\mathbf{F}_1$ is past through three separated encoder modules $h_\theta(\cdot)$, $h_\beta(\cdot)$ and $h_\gamma(\cdot)$, which downsample, flatten and project the feature map into three latent codes: $\mathbf{l}_{\theta_1}$, $\mathbf{l}_{\beta_1}$ and $\mathbf{l}_{\gamma_1}$, which represent pose, shape and garment information of the 3D model, respectively. With three latent codes, the feature map can be reconstructed by concatenating the latent codes together into a longer latent vector $\mathbf{l}'_1$, where $\mathbf{l}'_1 = \texttt{cat}[\mathbf{l}_{\theta_1}, \mathbf{l}_{\beta_1}, \mathbf{l}_{\gamma_1}]$. Such a vector is then passed through the decoder module $h^{-1}(\cdot)$ of the encoder-decoder, which upsamples the vector and then reconstructs the feature map $\mathbf{F}'_1$. The pose, shape or garment transfer can be achieved by simply replacing the corresponding latent code. For example, if we want to transfer the pose of the person in $\mathbf{I}_1$ to another pose of the person in $\mathbf{I}_2$, we just need to extract the latent codes $\mathbf{l}_{\theta_2}$, $\mathbf{l}_{\beta_2}$ and $\mathbf{l}_{\gamma_2}$ from $\mathbf{I}_2$ and replace the latent code $\mathbf{l}_{\theta_1}$ with $\mathbf{l}_{\theta_2}$. As a result, the feature map $\mathbf{F}'_{1\leftarrow\theta_2}$ reconstructed from concatenated latent code $\mathbf{l}'_{1\leftarrow\theta_2} = \texttt{cat}[\mathbf{l}_{\theta_2}, \mathbf{l}_{\beta_1}, \mathbf{l}_{\gamma_1}]$ would contain the shape and garment information of $\mathbf{I}_1$

but the pose information of $\mathbf{I}_2$. According to $\mathbf{F}'_{1\leftarrow\theta_2}$, the implicit surface expression can be predicted by the surface prediction MLP.

## 3.2 3D Reconstruction

In recent years, the implicit neural representation has exhibited the exceptional capability in 3D rendering [31, 35, 37, 40]. The object or scene is often encoded into an MLP as an occupancy field (OCC) or a signed distance field (SDF). Mathematically, it is defined as:

$$s_{\mathbf{x}} = g(\mathbf{x}) \tag{1}$$

where $g(\cdot)$ is the MLP and $s_{\mathbf{x}}$ is the learned implicit surface expression at the sampling position $\mathbf{x} \in \mathbb{R}^3$. The object's surface is represented by the level set of the implicit function $g(\cdot)$, from which the 3D geometry can be easily constructed by Marching Cube. Normally, a MLP could only store one scene or object [31, 37, 40]. However, a single MLP $g(\cdot)$ can be generalized to multiple objects by taking feature map of the image as an input [35, 36]:

$$s_{\mathbf{x}} = g(\mathbf{F}_x, \mathbf{x}_z) \tag{2}$$

where $x$ is the projection of $\mathbf{x}$ on to the image plane, $\mathbf{F}_x \in \mathbb{R}^C$ is the value of encoded feature map $\mathbf{F}$ at a sampling point $x$, and $\mathbf{x}_z$ is the depth value of $\mathbf{x}$. By taking the pixel-aligned feature $\mathbf{F}_x$, multiple objects can be inferred from a single MLP and objects can be controlled by editing the feature map.

For 2D-3D human reconstruction task, the formulation in Eq. 2 is commonly supervised using the ground truth occupancy values [35]. However, compared with occupancy, we consider the SDF as a better supervision signal during training because using SDF we can not only supervise the distance value but also the surface orientation. Unfortunately, the formulation in Eq. 2 is not suitable for SDF supervision as the output is not differentiable to the $\mathbf{x}$ hence the surface norm is unable to be computed. In order to supervise the surface norm, and being inspired by [43], we change the formulation in Eq. 2 into:

$$s_{\mathbf{x}} = g(\mathbf{F}_x, \sigma(\mathbf{x})) \mathrel{\hat{=}} g(\mathbf{F}, \mathbf{x}) \tag{3}$$

where $\sigma(\cdot)$ is the positional encoding function defined in [43]. For the sake of simplification, we denote the MLP expression $g(\mathbf{F}_x, \sigma(\mathbf{x}))$ as $g(\mathbf{F}, \mathbf{x})$. At here, we explicitly take the 3D location $\mathbf{x}$ as an input and hence the surface norm can be computed and supervised.

## 3.3 Loss

Our loss includes two parts: reconstruction loss and disentanglement loss.

**3D Reconstruction Loss** In order to learn the feature disentanglement from 2D images, we need to first initialize the feature extractor $f(\cdot)$ and the MLP modules $g(\cdot)$ for a 3D shape prior. $f(\cdot)$ followed by $g(\cdot)$ makes up the direct reconstruction pipeline similar to [35], which can be supervised by the 3D reconstruction loss according to ground-truth 3D meshes $\mathbf{M}$. We define the reconstruction loss as $\mathcal{L}_{recon}(\mathbf{F}, \mathbf{X}, \mathbf{M})$, where $\mathbf{X}$ is a set of 3D points that are sampled around the surface of $\mathbf{M}$. We explore two implicit representations: occupancy

and signed distance function (SDF). Specifically, following [35], the occupancy-based 3D reconstruction loss is defined as:

$$\mathcal{L}_{recon}(\mathbf{F},\mathbf{X},\mathbf{M}) = \sum_{\mathbf{x}\in\mathbf{X}} \|g(\mathbf{F},\mathbf{x}) - \mathbf{M}_{OCC}(\mathbf{x})\|_2^2 \tag{4}$$

where $\mathbf{X}$ is a set of off-surface points that are sampled tightly around the surface in a way such that the half of the samples are outside the surface and the other half are inside the surface. $\mathbf{M}_{OCC}(\mathbf{x})$ is the ground truth occupancy value of $\mathbf{M}$ at 3D position $\mathbf{x}$ and $\mathbf{x}\in\mathbf{X}$.

For SDF-based reconstruction, we adopt the implicit geometric regularization [12], which can learn the SDF surface from point clouds without ground truth SDF supervision, as the 3D reconstruction loss. We combine the sampling strategy in [40] and [35]: a set of points $\mathbf{X} = \mathbf{X}_{on} \cup \mathbf{X}_{off}$ are randomly sampled such that half number of points are on the surface of the mesh, denoted as $\mathbf{X}_{on}$ and the remaining points are off the surface, denoted as $\mathbf{X}_{off}$. In particular, $\mathbf{X}_{off}$ are randomly sampled around the ground-truth mesh and within the cubic space with a ratio of 3:1. For each image, the reconstruction loss $\mathcal{L}_{recon}(\mathbf{F},\mathbf{X},\mathbf{M})$ consists of three parts: the level set loss $\mathcal{L}_{ls}$, the Eikonal regularization loss $\mathcal{L}_{igr}$ and off surface loss $\mathcal{L}_o$:

$$\mathcal{L}_{recon}(\mathbf{F},\mathbf{X},\mathbf{M}) = \lambda_{ls}\mathcal{L}_{ls} + \lambda_{igr}\mathcal{L}_{igr} + \lambda_o\mathcal{L}_o \tag{5}$$

$$\text{with} \quad \mathcal{L}_{ls} = \sum_{\mathbf{x}\in\mathbf{X}_{on}} \Big(\|g(\mathbf{F},\mathbf{x})\|_1 + 1 - \big\langle \nabla_{\mathbf{x}} g(\mathbf{F},\mathbf{x}), \nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x})\big\rangle\Big), \tag{6}$$

$$\mathcal{L}_{igr} = \sum_{\mathbf{x}} \|\nabla_{\mathbf{x}} g(\mathbf{F},\mathbf{x}) - 1\|_1, \tag{7}$$

$$\mathcal{L}_o = \sum_{\mathbf{x}\in\mathbf{X}_{off}} \exp\big(-\alpha\cdot\|g(\mathbf{F},\mathbf{x})\|_1\big), \quad \alpha \gg 0. \tag{8}$$

The level set loss $\mathcal{L}_{ls}$ forces the gradient of the MLP (*i.e.*, the predicted surface normal) $\nabla_{\mathbf{x}} g(\mathbf{x})$ to align with the ground-truth normal of the surface $\nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x})$. The Eikonal regularization loss $\mathcal{L}_{igr}$ regularize the MLP to satisfy Eikonal equation $\|\nabla_{\mathbf{x}} g(\mathbf{x})\| = 1$. The off surface loss $\mathcal{L}_o$ push the points off the surface away from the level set surface. The overall reconstruction loss $\mathcal{L}_{recon}$ is the sum of these three components, weighted by coefficients $\lambda_{ls}$, $\lambda_{igr}$ and $\lambda_o$ respectively. In the experiments, the coefficients are set to $\lambda_{ls} = \lambda_{igr} = 1$ and $\lambda_o = 0.1$.

**Disentanglement Loss** With the properly pre-trained feature extractor and MLP surface predictor, we then jointly train the encoder-decoder for feature disentanglement and MLP for surface refinement. In order to properly learn the disentanglement of pose, shape and garment, we adopt the control variate strategy. We construct a dataset which consists of pairs of images and their corresponding ground-truth 3D meshes. For each pair, the difference between the images is either in pose, shape or garment, while the other two are kept same. This control strategy allows the encoder-decoder to focus on one property at each time and hence interpret the difference in latent space. We provides more details about our dataset pairing in section 4.1.

For each image pair, denote the different property between them as $d$, where $d\in\{\theta,\beta,\gamma\}$ and two remaining properties as $s1$ and $s2$, where $s1,s2\in\{\theta,\beta,\gamma\}\setminus\{d\}$. Let $\mathbf{F}_1 = f(\mathbf{I}_1)$ and $\mathbf{F}_2 = f(\mathbf{I}_2)$, then $\{\mathbf{l}_{s1,1},\mathbf{l}_{s2,1},\mathbf{l}_{d1}\} = h(\mathbf{F}_1)$ and $\{\mathbf{l}_{s1,2},\mathbf{l}_{s2,2},\mathbf{l}_{d2}\} = h(\mathbf{F}_2)$. We define two procedures:

- *Disentangled self-reconstruction* If no latent code swapping is applied, we should be able to reconstruct the feature maps $\mathbf{F}_1$ and $\mathbf{F}_2$ by $\mathbf{F}'_1 = h^{-1}(\mathbf{l}'_1)$ and $\mathbf{F}'_2 = h^{-1}(\mathbf{l}'_2)$,

where $\mathbf{l}'_1 = \texttt{concat}[\mathbf{l}_{s1,1}, \mathbf{l}_{s2,1}, \mathbf{l}_{d1}]$ and $\mathbf{l}'_2 = \texttt{concat}[\mathbf{l}_{s1,2}, \mathbf{l}_{s2,2}, \mathbf{l}_{d2}]$. We should also be able to recover $\mathbf{M}_1$ and $\mathbf{M}_2$ from the reconstructed feature map $\mathbf{F}'_1$ and $\mathbf{F}'_2$.

- *Disentangled cross-reconstruction* In the case where the latent codes of property $d$ is swapped, the modified latent codes become: $\mathbf{l}'_{1\leftarrow d2} = \texttt{concat}[\mathbf{l}_{s1,1}, \mathbf{l}_{s2,1}, \mathbf{l}_{d2}]$, then two new feature maps can be reconstructed as $\mathbf{F}'_{1\leftarrow d2} = h^{-1}(\mathbf{l}'_{1\leftarrow d2})$ and $\mathbf{F}'_{2\leftarrow d1} = h^{-1}(\mathbf{l}'_{2\leftarrow d1})$. Since there is only one different property between two images, the reconstructed feature map of one image should be the same as the original feature maps of the other and so are their 3D meshes.

We define the feature map reconstruction loss before and after disentanglement during self- and cross-reconstruction:

$$\mathcal{L}_{feat} = \|\mathbf{F}_1 - \mathbf{F}'_1\|_2^2 + \|\mathbf{F}_2 - \mathbf{F}'_2\|_2^2 + \|\mathbf{F}_1 - \mathbf{F}'_{2\leftarrow d1}\|_2^2 + \|\mathbf{F}_2 - \mathbf{F}'_{1\leftarrow d2}\|_2^2 \quad (9)$$

We also add the latent identity loss $\mathbf{L}_{latent}$ which matches the compressed latent code of two invariant properties in each image pair:

$$\mathbf{L}_{latent} = \|\mathbf{l}_{s1,1} - \mathbf{l}_{s1,2}\|_2^2 + \|\mathbf{l}_{s2,1} - \mathbf{l}_{s2,2}\|_2^2 \quad (10)$$

We define a surface reconstruction loss (either with occupancy or SDF) between the ground truth mesh $\mathbf{M}$ and the predicted output $g(\cdot)$. This term direct supervises the quality of the 3D models generated from the reconstructed feature maps:

$$\begin{aligned}\mathcal{L}_{recon} = \mathcal{L}_{recon}(\mathbf{F}'_1, \mathbf{X}_1, \mathbf{M}_1) + \mathcal{L}_{recon}(\mathbf{F}'_2, \mathbf{X}_2, \mathbf{M}_2) \\ + \mathcal{L}_{recon}(\mathbf{F}'_{2\leftarrow d1}, \mathbf{X}_1, \mathbf{M}_1) + \mathcal{L}_{recon}(\mathbf{F}'_{1\leftarrow d2}, \mathbf{X}_2, \mathbf{M}_2) \quad (11)\end{aligned}$$

Therefore, for each image pair, the overall disentanglement loss $\mathcal{L}_{disent}$ is defined as the weighted combination of the above three terms:

$$\mathcal{L}_{disent} = \mathcal{L}_{feat} + \mathcal{L}_{latent} + \mathcal{L}_{recon} \quad (12)$$

# 4 Experiment

## 4.1 Dataset

It is extremely challenging to obtain a real-world dataset that contains two humans with different shapes and garments but in the same pose. Therefore, we modified a public physically simulated dataset, TailorNet dataset [32] to evaluate our method. The original dataset consists of physically simulated 3D sequences of motions of dressed humans, which are modelled by the SMPL model with an additional garment layer on top of it. The dataset includes humans with a range of body shapes and several types of garments, where each type of garment also has intra-class variation. We make the following modifications to the dataset: (i) We construct the image pairing by comparing the SMPL parameters of the human body and garment type. In the end, we obtain 1369 pose-vary pairs, 51 shape-vary pairs and 761 garment-varying pairs. (ii) Since the combined ground-truth meshes of dressed humans have multiple layers, the normals are in the opposite direction at overlapping regions. This may confuse the network during training with SDF-based supervision loss. Therefore, we convert them into single-layer water-tight meshes using the Manifold package. (iii) For each mesh, we render the front view using Blender as the input to our method. The final output is the three subsets that contain pairwise data variation. The three subsets are padded to the same size of 1369 and split into the training, validation and test dataset with a ratio of 6:2:2.

| Experiment | Implicit Func | Chamfer (mm) | P2S (mm) | normal (mm) |
|---|---|---|---|---|
| self | SDF/OCC | **1.43**/3.65 | **11.21**/22.24 | 11.16/**8.93** |
| cross - pose | SDF/OCC | **1.26**/4.73 | **12.12**/21.63 | 10.63/**8.72** |
| cross - shape | SDF/OCC | **2.46**/3.09 | **12.25**/19.88 | 11.46/**9.28** |
| cross - garment | SDF/OCC | **1.26**/3.36 | **9.35**/22.61 | 11.53/**8.86** |

Table 1: Reconstruction error (median) after swapping the latent code of body shape, pose and garment style, using SDF and occupancy.

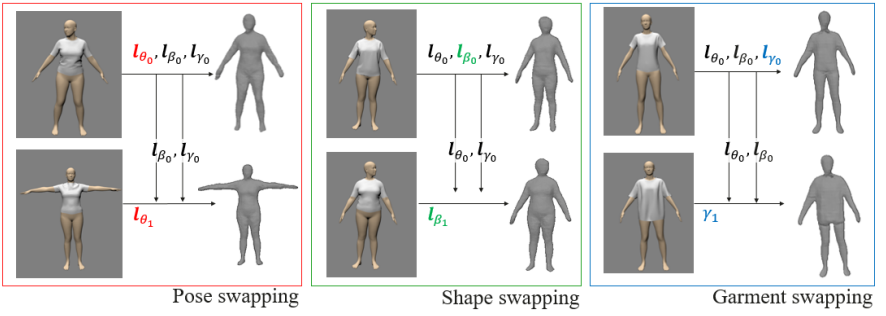


Figure 2: Example results of surface reconstruction after pairwise swapping the encoded latent components. Note that the pose and shape swapping are achieved on SDF-based pipeline, while the garment swapping is achieved on occupancy-based pipeline.

## 4.2 Implementation Details

Our method is implemented using PyTorch. For the feature extractor $f(\cdot)$, we follow the choice in [45] and use the Hourglass Network [29]. For the encoder $h(\cdot)$ and decoder modules $h^{-1}(\cdot)$, we use a ResNet18 [15] implemented in Bolts library [8]. We change the input dimension of the first convolutional layer to match the dimension of the feature map. The latent size for $\mathbf{l}_\theta$, $\mathbf{l}_\beta$ and $\mathbf{l}_\gamma$ are are all set to be 128 according to the ablation study (see details in supplementary material). For the surface predictor, we follow the design in [43] which is a fully connected MLP with residual link between every two linear layers.

We first pre-train the feature extractor and surface predictor on the direct reconstruction pipeline. For both occupancy and SDF supervision, we use the RMSprop optimizer with a learning rate of $1 \times 10^{-4}$, the number of sampling is 1000 and batch size is 8. After epoch 150, the learning rate is reduced to $1 \times 10^{-5}$ to ensure smooth convergence. During the training for feature disentanglement, the feature extractor is frozen while the encoder-decoder and surface predictor are tuned together. The same optimizer is used except that the learning rate is kept as $1 \times 10^{-4}$ until the convergence reaches around 360 epochs.

## 4.3 Results

### 4.3.1 Reconstruction with latent representation

Our proposed encoder-decoder structure allows us to disentangle and modify the latent code $\mathbf{l}$ by replacing its components $\mathbf{l}_\theta$, $\mathbf{l}_\beta$ and $\mathbf{l}_\gamma$. We test the quality of self-reconstruction and pairwise cross-reconstruction defined in Section 3.3. The reconstruction quality is measured using three metrics: the Chamfer distance between reconstructed and ground-truth surfaces,

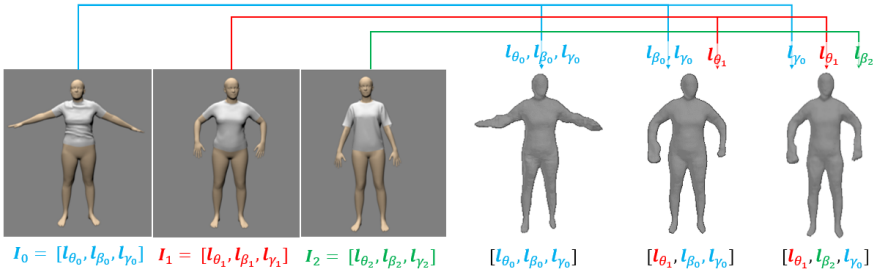


Figure 3: Example results of surface reconstruction after uncontrolled swapping of both body pose and shape. The synthetic mesh (rightmost) incorporates the tight garment style of $\mathbf{I}_0$, body pose of $\mathbf{I}_1$ and slimmer body shape of $\mathbf{I}_2$.

the Euclidean distance from predicted vertices to their closest point on the ground-truth surface (P2S), and the RMS difference between the predicted and ground-truth surface normal vectors. We summarise the results in Table 1, where "self " means self-recon and "cross" means cross-recon. The SDF-based pipeline achieves a higher accuracy than the occupancy-based pipeline. However, from visual inspections, we find the SDF-based method can better distinguish between different body poses and shapes than garment styles. By contrast, occupancy-based disentangled reconstruction capture all three variations, though the reconstruction quality is lower. The difference may be because the SDF approach tends to oversmoothen the predicted surface and the garment variation in the training dataset is too subtle. Fig. 2 shows the visual demonstration for how the mesh reconstruction is controlled through pairwise latent swapping.

Pairwise swapping can only change one property each time and requires the other two to be identical, which is hardly possible in reality. Therefore, given a specific garment, we test the uncontrolled swapping where the body pose and shape latent code are replaced by new values at the same time. As shown in Fig. 3, the body pose $\mathbf{l}_\theta$, shape $\mathbf{l}_\beta$ and garment information $\mathbf{l}_\gamma$ are respectively encoded from $\mathbf{I}_1$, $\mathbf{I}_2$ and $\mathbf{I}_0$, concatenated, and decoded into a new human body. We have included more examples in the supplementary material.

### 4.3.2 Interpolation test

Given a starting and ending frame with different features, we are able to interpolate the latent codes to achieve a smooth transition of body pose, shape or garment style in the reconstructed mesh, as shown in Fig. 4. While the pose interpolation can be used to achieve an animation effect, the body shape and garment interpolation is helpful in generating new virtual assets in an effortless way. More examples are provided in the supplementary material.

## 5 Discussion and Conclusion

In this paper, we propose the first and novel method for the task of extracting disentangled 3D attributes only from 2D image data. To illustrate the feasibility, we apply the principle to learn pose, shape, and garment representations of a human body directly from images. Our method disentangles and encodes the representations of three properties into latent codes using an encoder-decoder from the feature map of the input image and reconstructs the human

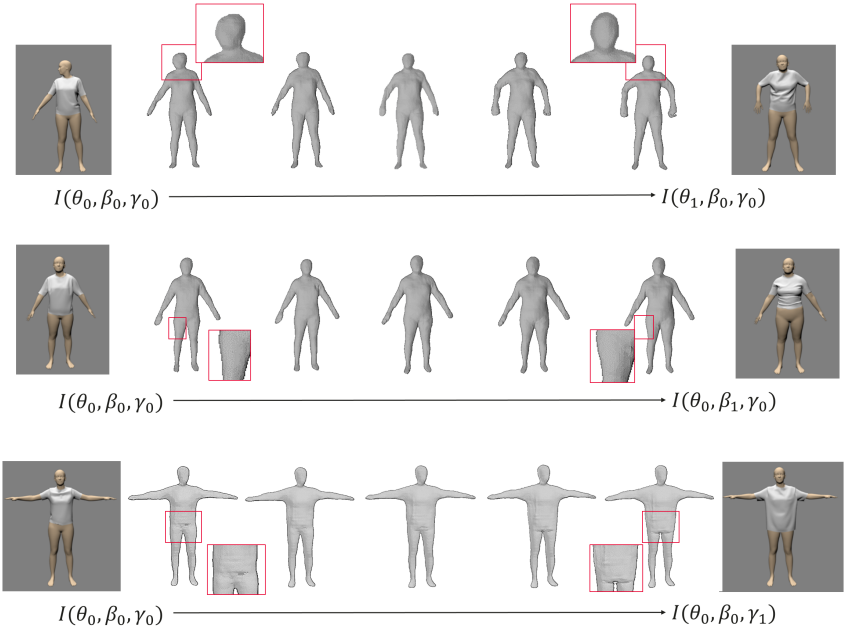


Figure 4: Example results of interpolation test: the latent code of source and target image are interpolated to reconstruct a dressed human body with new features.

body based on learned representations using the implicit function. We demonstrate the control to the reconstructed mesh by manipulating the code in the latent space. We believe that our novel approach towards 2D-to-3D disentanglement can pave the way to interpretable manipulations of 3D content from 2D images alone and therefore opens the path to more seamless and controllable interaction with 3D models without the need for explicit supervision. In our future work we plan to use TKFAC on MindSpore [1], which is a new deep learning computing framework.

[1] https://www.mindspore.cn/

# References


[1] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single rgb camera. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[2] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *Proceedings of ACM Special Interest Group on GRAPHICS (SIGGRAPH)*, 2005.

[3] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.

[4] Benjamin Busam, Hyun Jun Jung, and Nassir Navab. I like to move it: 6d pose estimation as an action decision process. *arXiv preprint arXiv:2009.12678*, 2020.

[5] Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[6] Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2018.

[7] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2016.

[8] William Falcon and Kyunghyun Cho. A framework for contrastive self-supervised learning and designing a new approach. *arXiv preprint arXiv:2009.00104*, 2020.

[9] Daoyi Gao, Yitong Li, Patrick Ruhkamp, Iuliia Skobleva, Magdalena Wysocki, Hyun-Jun Jung, Pengyuan Wang, Arturo Guridi, and Benjamin Busam. Polarimetric pose prediction. In *European Conference on Computer Vision (ECCV)*, October 2022.

[10] Gege Gao, Huaibo Huang, Chaoyou Fu, Zhaoyang Li, and Ran He. Information bottleneck disentanglement for identity swapping. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2014.

[12] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of Intl. Conf. on Machine Learning (ICML)*, 2020.

[13] Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. Garnet: A two-stream network for fast and accurate 3d cloth draping. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[14] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and H-P Seidel. A statistical model of human pose and body shape. In *Computer graphics forum*, 2009.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[16] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *Proceedings of Intl. Conf. on Machine Learning (ICML)*, 2016.

[17] Boyi Jiang, Juyong Zhang, Jianfei Cai, and Jianmin Zheng. Disentangled human body embedding based on deep hierarchical neural network. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 2020.

[18] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3d scenes. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[19] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[20] HyunJun Jung, Patrick Ruhkamp, Guangyao Zhai, Nikolas Brasch, Yitong Li, Yannick Verdie, Jifei Song, Yiren Zhou, Anil Armagan, Slobodan Ilic, et al. Is my depth ground-truth good enough? hammer–highly accurate multi-modal dataset for dense 3d scene regression. *arXiv preprint arXiv:2205.04565*, 2022.

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[22] Zorah Lahner, Daniel Cremers, and Tony Tung. Deepwrinkles: Accurate and realistic clothing modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[23] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Trans. on Graphics (ToG)*, 2015.

[24] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to dress 3d people in generative clothing. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[25] Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. SCALE: Modeling clothed humans with a surface codec of articulated local elements. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[26] Fabian Manhardt, Gu Wang, Benjamin Busam, Manuel Nickel, Sven Meier, Luca Minciullo, Xiangyang Ji, and Nassir Navab. Cps++: Improving class-level 6d pose and shape estimation from monocular images with self-supervised learning. *arXiv preprint arXiv:2003.05848*, 2020.

[27] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[28] Jiteng Mu, Weichao Qiu, Adam Kortylewski, Alan Yuille, Nuno Vasconcelos, and Xiaolong Wang. A-sdf: Learning disentangled signed distance functions for articulated shape representation. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[29] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

[30] Ahmed A A Osman, Timo Bolkart, and Michael J. Black. STAR: A sparse trained articulated human body regressor. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[31] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[32] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[33] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[34] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael J Black. Dyna: A model of dynamic human shape in motion. *ACM Trans. on Graphics (ToG)*, 2015.

[35] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.

[36] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[37] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. SCANimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[38] Eduardo Hugo Sanchez, Mathieu Serrurier, and Mathias Ortner. Learning disentangled representations via mutual information estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[39] Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. Self-supervised collision handling via generative 3d garment models for virtual try-on. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[40] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.

[41] Pengyuan Wang, HyunJun Jung, Yitong Li, Siyuan Shen, Rahul Parthasarathy Srikanth, Lorenzo Garattoni, Sven Meier, Nassir Navab, and Benjamin Busam. Phocal: A multimodal dataset for category-level object pose estimation with photometrically challenging objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21222–21231, 2022.

[42] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In *Advances in Neural Information Processing Systems*, 2019.

[43] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[44] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[45] Shanxin Yuan, Guillermo Garcia-Hernando, Björn Stenger, Gyeongsik Moon, Ju Yong Chang, Kyoung Mu Lee, Pavlo Molchanov, Jan Kautz, Sina Honari, Liuhao Ge, et al. Depth-based 3d hand pose estimation: From current achievements to future goals. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[46] Jianfu Zhang, Yuanyuan Huang, Yaoyi Li, Weijie Zhao, and Liqing Zhang. Multi-attribute transfer via disentangled representation. In *Proceedings of AAAI Conference on Artificial Intelligience(AAAI)*, 2019.

[47] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[48] Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.