

GAN-based Garment Generation Using Sewing Pattern Images

Yu Shen, Junbang Liang, and Ming C. Lin

University of Maryland, College Park

[https://gamma.umd.edu/researchdirections/virtualtryon/
garmentgeneration/](https://gamma.umd.edu/researchdirections/virtualtryon/garmentgeneration/)

Abstract. The generation of realistic apparel model has become increasingly popular as a result of the rapid pace of change in fashion trends and the growing need for garment models in various applications such as virtual try-on. For such application requirements, it is important to have a general cloth model that can represent a diverse set of garments. Previous studies often make certain assumptions about the garment, such as the topology or *suited* body shape. We propose a unified method using the generative network. Our model is applicable to different garment topologies with different sewing patterns and fabric materials. We also develop a novel image representation of garment models, and a reliable mapping algorithm between the general garment model and the image representation that can regularize the data representation of the cloth. Using this special intermediate image representation, the generated garment model can be easily retargeted to another body, enabling garment customization. In addition, a large garment appearance dataset is provided for use in garment reconstruction, garment capturing, and other applications. We demonstrate that our generative model has high reconstruction accuracy and can provide rich variations of virtual garments.

1 Introduction

The generation of realistic garment is one of the most important steps during the garment design and manufacturing process. Usually, a garment model needs to be manually designed by an experienced designer—this step can be time-consuming and labor-intensive. The efficiency can be dramatically improved if a garment model can be generated automatically. The generation of garment model can also benefit certain virtual-reality applications such as the virtual try-on system. As e-commerce becomes more prevalent in the apparel industry, a rich and realistic virtual try-on system can considerably improve the user experience during online shopping, where garment model generation plays a central role.

However, there are many challenges in automatically generating garment models. First, garments usually have different topologies, especially for fashion apparel, that make it difficult to design a universal generation pipeline. Moreover, it is often not straightforward for the general garments design to be retargeted onto another body shape, making it difficult for customization. Some previous work started to address this problem using either user-assisted input [16] or cloths with fixed topology such as a T-shirt or a skirt [33].

We propose a learning-based parametric generative model to overcome the above difficulties. Given garment sewing patterns and human body shapes as inputs, we compute the *displacement image* on the UV space of the human body as a unified representation of the garment mesh. Different sizes and topologies of garments are represented by different values and geometric adjacencies in the image. The 2D displacement image, as the representation of the 3D garment mesh data, is given as input into a conditional Generative Adversarial Network (GAN) for latent space learning. Using this 2D representation for the garment mesh, on one hand, we can transform the irregular 3D mesh data to regular image data where a traditional CNN can easily learn; on the other hand, we can extract the relative geometric information with respect to the human body, enabling straightforward garment retargeting onto a different human body.

Our network can generate a series of garment models that meet the constraints of inputs with various appearances. The generated garments can be easily retargeted to another body shape using our 2D representation, while other generative methods [16,33] need to rerun the generative network and cannot ensure the same appearance as the original.

To train such a generative model, a sufficient amount of garment data is needed. However, there is no publicly available garment dataset which provides the appearances of the garments undergoing different motions and on varying human body shapes. Therefore, we generate a large dataset with different garment geometries for this specific task. We employ physically-based simulation of different garment meshes and fabric materials. Together with different human body motions, we can obtain a large variety of garment appearances on the body.

Overall, our contributions include:

- The first image-based garment generative model (Sec. 5) which supports most garment topologies and patterns (Sec. 6.3), human body shapes and sizes (Sec. 6.5), and garment materials (Appendix 2).
- A novel image representation for garments (Sec. 4) that can transfer to/from general 3D garment models with little information loss (Sec. 6.2), enabling garment retargeting (Sec. 6.5).
- A large garment appearance dataset for training (Appendix 2).

2 Related work

In this section, we survey related works in garment modeling, garment retargeting, and generative networks.

2.1 Garment modeling

Garment model generation has attracted attention these days due to its importance in both real-world and virtual garment design application. Although professional tools, such as Marvelous Designer [2018], can help design high-quality garment models, it may take an excessive amount of time to use it. Several studies have addressed this issue by introducing an automatic generation pipeline to improve the efficiency. Assuming different priors, most previous studies lie in three categories: sketch-based, image-based, and depth-based.

Sketch-based methods. Generating garment models with sketches is one of the most popular ways. Turquin *et al.* [30] and Decaudin *et al.* [10] developed some of the early work in this area. They used grid and geometric methods to generate garment models with sketches. Later, Robson *et al.* [28] proposed a context-aware method to make the generated garment model more realistic based on a set of observations on key factors which could affect the shapes of garments. Jung *et al.* [18] proposed a method to model 3D developable surfaces with a multi-view sketch input. Bartle *et al.* [3] proposed a physics-driven pattern adjustment method for direct 3D garment editing. FoldSketch [20] supports simple and intuitive fold and pleat design. Recently, Huang *et al.* [16] proposed a realistic 3D garment generation algorithm based on front and back image sketches. Wang *et al.* [33] proposed a method that can achieve retargeting easily.

In addition, a common limitation of these methods is the domain knowledge requirement on garment sketching, while our method does not require any domain knowledge.

Image-based or depth-based methods. Other information such as images can also be used to generate a garment model. Bradley *et al.* [6] and Zhou *et al.* [36] researched early on garment modeling using multi-view images and single-view image, respectively. Jeong *et al.* [17] created the garment model with a single photograph by detecting the landmark points of the garment. Yang [34] made a full use of garment and human body databases to generate the garment models from image. Daněček *et al.*'s [9] method can estimate the 3D garment shape from a single image using deep neural networks. Recently, Tex2Shape [1], PIFu [29], DeepHuman [35], Gabeur *et al.* [12] proposed models for detailed clothed full-body geometry reconstruction. MGN [5] predicts body shape and clothing, layered on top of the SMPL [23] model from a few (typically 1 - 8) frames of a video. Depth information can also be useful. Chen *et al.* [8] proposed a method to generate garment models given an RGBD sequence of a worn garment.

However, these methods require photos or depth images from a real garment, which means they cannot generate a garment model from size parameters only. In contrast, our model is able to generate 3D garment meshes directly from sewing patterns and sizing parameters by using the generative network.

2.2 Garment Retargeting

Retargeting the garment model from one body to another is often needed due to different body shapes. Retargeting can save computational costs if it can be done efficiently. Brouet *et al.* [7] introduced a fully automatic method for design-preserving transfer of garments among characters with different body shapes. In contrast, Guan *et al.* [14] used a learning-based architecture to replace the expensive simulation process and present retargeting examples. GarNet [15] presented a two-stream architecture to fit a 3D garment template to a 3D body. TailorNet [26] predicts clothing deformation given the pose and shape of human and garment model.

In our method, by making use of the image representation of the garment, we can easily retarget one generated garment model from one body shape to another, without additional computations.

2.3 Generative Network

Generative networks have been becoming more popular due to their impressive performance. There are several well-known generative networks, such as Generative Adver-

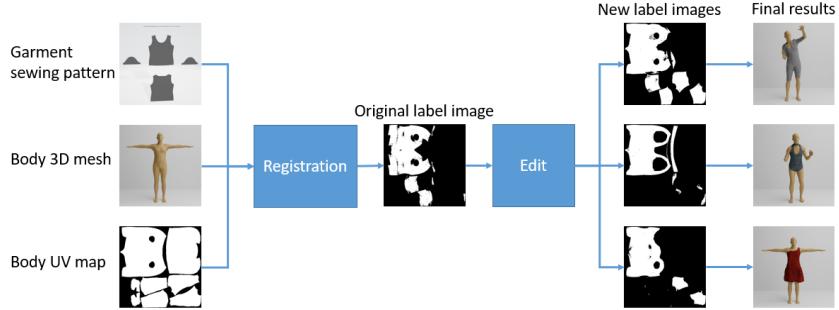


Fig. 1. Label image generation process. We first generate the label image with the pattern configuration registered on the body mesh and mapped to the body UV map. Then we can edit the original label image to new, different label images, which will lead to different garment topologies in the final results.

sarial Network (GAN) [13] and Variational Auto-Encoder (VAE) [11]. With the development of the neural network research, new variants of generative networks have been proposed, such as Pix2PixHD [31] based on GAN or VQ-VAE [27] based on VAE. In our algorithm, we design the network architecture based on the Pix2PixHD network architecture due to its high accuracy and efficiency.

3 Method Overview

Our objective for this work is to develop a GAN-based generator that creates different types of garment meshes, given the garment design (or sewing) patterns. The overall pipeline is shown in Fig. 1.

First, we unify the common garment pattern configurations to a body mask that shows the region of garment coverage. To do this, we mark the sizes of each pattern pieces from the 2D sewing pattern and register each piece to its corresponding body part. We can then obtain the label map by coloring the covered body part according to the registration. As an auxiliary step, we may edit the label image to vary the sizes and the connectivity of different parts, leading to different garment styles and topologies in the final results.

We model the garment mesh using a 2D image representation in the UV space of the corresponding human body (Fig. 11), which shares the same space as the label map that we obtained from the pattern input. This step regularizes the input mesh onto a CNN-friendly format that is independent to the original mesh resolution. We compute the correspondence between 3D points of the mesh and the 2D pixels of the image using non-rigid ICP and a Voronoi diagram, as later discussed in Sec. 4.

We then train a deep GAN to learn the distribution of the representative images. We use a state-of-the-art conditional GAN to learn a mapping between a topology label mask and the final image representation, conditioned on the human pose, shape and a random noise, as shown in Fig. 2. We define a set of loss functions to provide smooth

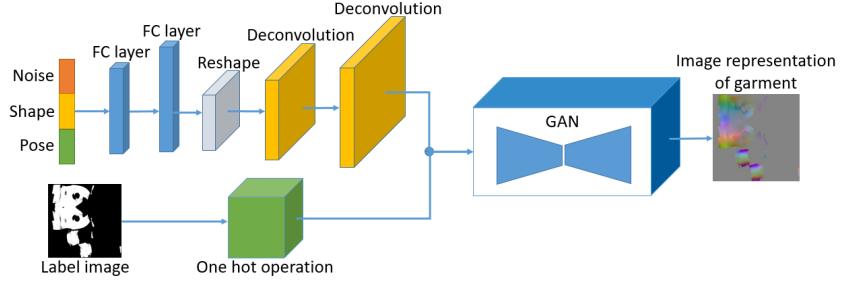


Fig. 2. Our network architecture. We first encode one dimensional input to match the sizes to the label image (upper branch). It is then concatenated with one hot labelled image (bottom branch) and fed into the GAN network. Finally, the network outputs the image representation of the garment (right).

results and avoid mode collapse (Sec. 5.1). To train the network model, we create a large dataset consisting of different garments, human body shapes and motions using cloth simulation. Our dataset not only covers most of the commonly seen garment shapes and geometries, but also assigns different fabric materials to the garments so that the simulated garment motions may vary noticeably even with the same clothing geometry (Sec. 5.2).

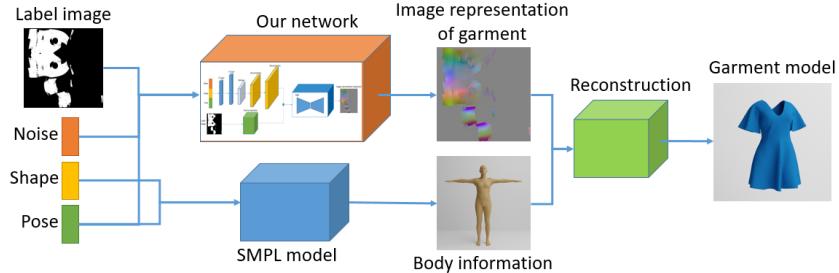


Fig. 3. Our inference pipeline. The upper branch generates the image representation of the garment, while the bottom branch generates the body mesh. Finally, we recover the garment mesh by decoding the image representation of the garment given the body mesh.

The inference pipeline of our method is shown in Fig. 3. We use the previously obtained label mask as input to constrain and control the topology of the output mesh. Given the label mask, we can generate a set of different image representations of the garment by varying the human pose and shape parameters, as well as the noise vector. As the last step, we recover the 3D garment mesh using its image representation and the corresponding human body. The final garment mesh can naturally fit onto the given

human body shape due to the nature of our representation model (See Sec. 4), and can provide realistic details depending on the body pose and shape.

4 Garment Representation in UV Space

As stated before, there are several challenges involved in modeling garments. First, garment meshes are graph data with nonuniform structures. Different meshes usually have different numbers of vertices and connections. It is difficult to set up a uniform and vectorized graph representation for all garments. Also, in contrast to other graph data, subdivision does not change the geometric information of the mesh. Graph representation cannot easily account for this ambiguity or redundancy of the mesh. Next, there are many kinds of garments that have different topologies. Shoulder style can provide a large variety of garment looks, not to mention the difference between skirts and pants. This makes high-level parameterization (*e.g.*, sleeve length) impossible without predefined classification.

To overcome these difficulties, we employ displacement maps on human body UV space as a unified representation of the garments. The geometric information of the mesh can be preserved, as long as the map resolution is sufficient. The key idea is that the garment mesh, as a 2D manifold, can be non-rigidly deformed onto the human body surface, and the UV space of the human body surface preserves most of the adjacency and connectivity of the 3D space. Also, this representation is independent to the resolution of the original mesh. No matter how the mesh is subdivided, the underlying representation will remain the same.

The method of using displacements from the human body surface as a way to represent clothes has been adopted in previous works [19,5]. However, in their work, the clothes are fixed to a template mesh. The representations are thus forced to be separated into a set of different clothes, since they have different templates. In contrast, we do not rely on specific clothing templates. Our model not only unifies different cloth types, but also generates clothes with new topologies.

4.1 Encoding Process

To create a displacement map of a certain garment, we first use non-rigid ICP [2] to register the cloth surface to the body surface, which makes the cloth tight-fit to the body. We then subdivide the cloth surface according to the Voronoi regions of body vertices to assign garment surface to body vertices. Finally, for each point on the body UV map, we compute the corresponding 3D position on the body surface, match it to the point on the cloth with the interpolated normal vector (of the garment surface that is assigned to the region), and fill in pixel value of the map using the displacement.

Specifically, we first register the cloth surface $\mathcal{G} = (\mathcal{V}_G, \mathcal{E}_G)$ to the body surface $\mathcal{B} = (\mathcal{V}_B, \mathcal{E}_B)$ by optimization:

$$\mathbf{X} = \arg \min_{\mathbf{X}} E(\mathbf{X}) = \sum_{\mathbf{v}_i \in \mathcal{V}_G} d^2(\mathcal{B}, \mathbf{X}_i \mathbf{v}_i) + \alpha \sum_{(\mathbf{v}_i, \mathbf{v}_j) \in \mathcal{E}_G} \|(\mathbf{X}_i - \mathbf{X}_j) \mathbf{D}\|_F^2 \quad (1)$$

where \mathbf{X} is the set of affine matrices for all garment vertices, α and $\mathbf{D} = \text{diag}(1, 1, 1, \gamma)$ are importance weights, and $d()$ is the distance between a point to a mesh. We set α and γ to a small value (typically 0.1) to encourage non-rigidity so that the cloth is mapped onto the body surface without large global rigid transformation. Note that after the non-rigid ICP, there may still be some vertices that are far from the body surface because of the topology constraint (*e.g.* dresses). We then design an algorithm to create a correspondence mapping of the surfaces between the cloth and the body.

As a preliminary step, we create the correspondence between each face of the cloth mesh and the vertices of the body mesh according to the Euclidean distance. First, we subdivide the registered cloth mesh using the 3D Voronoi diagram of the body surface. Voronoi regions [22] of the body vertices cuts the garment surface into convex polygons, which can be easily triangulated. Given that computing the analytical intersection to Voronoi regions is challenging, we achieve the subdivision by repeatedly check if an edge of the cloth mesh belongs to multiple Voronoi regions:

$$\exists \mathbf{e} = (\mathbf{v}_0, \mathbf{v}_1) \in \mathcal{E}_G : V_{\min}(\mathcal{V}_B, \mathbf{v}_0) \cap V_{\min}(\mathcal{V}_B, \mathbf{v}_1) = \emptyset \quad (2)$$

where $V_{\min}(\mathbf{V}, \mathbf{u})$ computes the subset of \mathbf{V} that has the closest distance to \mathbf{u} :

$$V_{\min}(\mathbf{V}, \mathbf{u}) = \{\mathbf{v} \in \mathbf{V} : \forall \mathbf{v}' \in \mathbf{V}, d(\mathbf{v}', \mathbf{u}) \geq d(\mathbf{v}, \mathbf{u})\} \quad (3)$$

If so, we subdivide the edge using the perpendicular bisector plane of the two vertices selected from $V_{\min}(\mathcal{V}_B, \mathbf{v}_0)$ and $V_{\min}(\mathcal{V}_B, \mathbf{v}_1)$, and its adjacent faces accordingly. Finally, we ensure that each face of the cloth mesh belongs to only one Voronoi region.

The next step is to match each subdivided face of the cloth mesh to the UV space of the corresponding Voronoi region. The intersection of the Voronoi region of a vertex and the body surface is bounded by the perpendicular bisector planes of each of its adjacent edges. We refer it as the ‘Voronoi surface’ of a vertex. Instead of further subdividing the cloth face into smaller faces and mapping them to different UV regions of the Voronoi surface, we iterate each pixel of the UV regions and shoot a ray out of the surface. To ensure an even sampling, the direction of each ray is computed by interpolating between the normal direction of face, edge and vertex (See description with figure in Appendix 5). An intersection of the ray and the cloth face creates a match between a pixel of the UV space and a point on the cloth surface. We enforce that the pixels on the edge of the pattern cuts are positioned on the body edge in 3D space. This ensures that their ray directions are the same, resulting in that the adjacent pairs of faces that are separated in the UV map have their common edge mapped onto the same garment edge in the 3D space, thus preserving connectivity. This property is used to reconstruct the 3D cloth mesh from the representation, as discussed in Sec. 4.2.

The quality of our mapping algorithm depends heavily on the load balance of the Voronoi regions. This is why we perform non-rigid ICP as pre-processing: it prevents loss of reconstruction details when the garment pieces are far from the body surface. Nonetheless, the non-rigid ICP may still not be able to handle extreme cases such as complex stacked garment layers. When multiple faces overlap on the same region, we choose the garment vertices that are the farthest from the body surface. This will result in smoother and simpler reconstructed garments in these challenging cases.

4.2 Decoding Process

Decoding the image representation back to the 3D cloth mesh is straightforward. Since adjacent pixels of the UV space correspond to adjacent points in the 3D space, we can simply connect adjacent pixels together to form the mesh. The only problem is that the connectivity will be lost when the cloth is cut into different UV regions. We solve this problem by ensuring that the two edges at different sides of the cut boundary are mapped to the same garment edge, as discussed in the encoding process. After fusing the duplicated 3D edge, the surface will be faithfully reconstructed.

5 Latent Space Learning

We apply a GAN-based model to learn the latent space of the representation image. Our network structure is shown in Fig. 2.

Since the pixel values in the representation image are related to the human body pose and shape, we add them as the conditional input in the network. Additionally, we provide a label map that indicates the overall topology of the garment to further constrain the generated image. The noise vector here serves mostly as the encoded detailed appearance, such as wrinkles and tightness of the cloth. We re-format the label image to one-hot version, and concatenate it with the encoded features of the other 1D input. Currently we only have binary information for the garment label map, but we can also support labels of different garment parts, as long as the corresponding data is provided. We use Pix2PixHD [32] as our backbone network, but other state-of-the-art methods can also work in practice.

5.1 Loss Functions and Training Process

Because we cannot simply enumerate every possible garment and simulate them on every possible human pose, the trained model can easily have mode collapse problems, which is not ideal. To deal with this problem, we use a two-phase learning process. First, we train the model with the usual GAN loss and the feature loss:

$$\mathcal{L} = \mathcal{L}_{GAN} + \lambda_0 \mathcal{L}_{feat} \quad (4)$$

$$\mathcal{L}_{GAN} = \|D(I_{real}) - 1\|_1 + \|D(I_{fake})\|_1 + \|D(G(I_{fake})) - 1\|_1 \quad (5)$$

$$\mathcal{L}_{feat} = \|D^*(I_{real}) - D^*(I_{fake})\|_1 + \|VGG^*(I_{real}) - VGG^*(I_{fake})\|_1 \quad (6)$$

In the above equations, \mathcal{L} is the total loss, \mathcal{L}_{GAN} is the GAN loss, and \mathcal{L}_{feat} is the feature loss. $D()$ is the discriminator, $G()$ is the generator, and $VGG()$ is the pretrained VGG network. I_{real} and I_{fake} are the real and the fake images. D^* and VGG^* means the concatenation of the activations in all layers. After the first phase, the network can learn a conditional mapping between the input label and the output image, but it lacks variation from the noise vector.

Next, we fine-tune the model using the GAN loss and the new smoothness loss only:

$$\mathcal{L} = \mathcal{L}_{GAN} + \lambda_1 \mathcal{L}_{smooth} \quad (7)$$

$$\mathcal{L}_{smooth} = \left\| \frac{\partial I_{fake}}{\partial x} \right\|_1 + \left\| \frac{\partial I_{fake}}{\partial y} \right\|_1 \quad (8)$$

where \mathcal{L}_{smooth} is introduced to the GAN model to enforce the smoothness of the representation image. Since the paired supervision from the feature loss is removed, the model will gradually become diverse to include more plausible but unseen results. We show later in our experiments that our learned model can generate clothing styles that are not from the training dataset.

5.2 Data Preparation

To learn the network model with high accuracy and variety, a large dataset depicting the joint distribution between the garment geometry and the human body is required. Previous datasets such as Bhatnagar *et al.* [5] or Liang *et al.* [21] have limited garment styles and body motions and are thus not suitable for our needs. Therefore, we propose a physics-based simulated dataset to represent most common garment types, human motions, and cloth materials. We sample different human motions and body shapes using the Moshed CMU MoCap dataset [24]. Our garments are obtained from various online sources, which we will make public with the dataset. We initialize the human to a T-pose and dress the body with each of the garments. Then we use the cloth simulator [25] to generate the cloth motion along with the body motion. We notice that the cloth material of the garment can significantly alter the appearance, so we also vary the material parameters during data generation. For quantitative details, please refer to Sec. 6.3. We show examples of different garment data in Appendix 2.

6 Experimental Analysis

In this section, we will first introduce the implementation details of our method. Next, we show the effectiveness and performance of the key parts of our method by various experiments, including garment reconstruction, clothing style generation, and garment retargeting.

6.1 Implementation details

We collected 104 types of garment models, each with 10 materials, and chose one random body motion sequence out of the 10 most commonly seen sequences. Then we dressed the garment on the body and simulated it using a cloth simulator [25] to generate a series of garment meshes with different poses, thereby generating $104 * 10 * 250 = 260,000$ garment (split into 80%/20% for training/test). After that, we applied the representation transfer process on those garment instances and generated the image representation as well as the label mask. Next, we fed the images together with body shapes, poses, and the label images to the network for training. In practice, we randomly

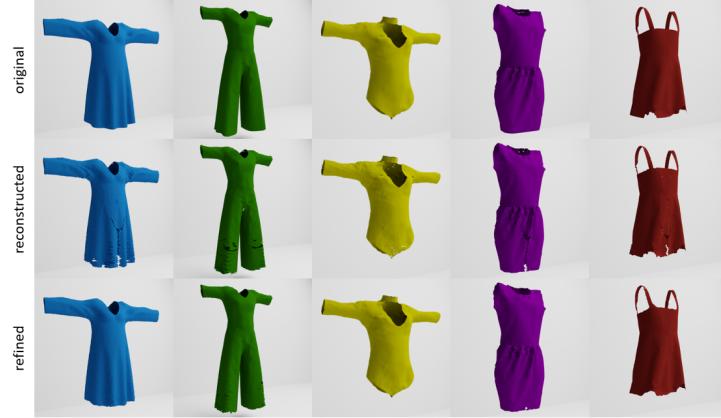


Fig. 4. Comparison between original mesh (first row), reconstructed mesh (second row), and refined mesh (third row). Our method is able to retain most of the original information, independent of the topology or the geometry of the garment mesh. The refined meshes indicate that the post-process is able to fix the small holes and gaps on the reconstructed meshes.

chose 2 materials in each epoch, to reduce the training time while making full use of the whole dataset.

We set λ_1 to 500, and the learning rate to 0.0002. We trained the model on an Nvidia GTX 1080 GPU. It took around 4 hours to train for each epoch, and we trained our model for 20 epochs in total.

6.2 Garment Reconstruction

Image representation of garments is one of the key contributions for the entire pipeline. We show the accuracy of the representation transfer process on our training data both qualitatively and quantitatively.

By transferring the 3D mesh of the garment to its 2D image representation and transferring back to a 3D mesh, we were able to recover the original 3D garment mesh. We randomly chose 5 different types of garments from the entire training dataset, chose 1 instance in each type, and generated the 3D mesh pair. The first row of Fig. 4 shows the original garments, while the second row shows the results of the recovered garments. As shown in the figure, our method is able to retain most of the original information when transferring between the 3D mesh and 2D image representation, under different types and topologies of garments. There might be small gaps or holes on the reconstructed meshes because of the resolution differences between two representations. We performed post-processing on the reconstructed meshes to resolve these small gaps/holes, as shown in the third row of Fig. 4. The post-processing method that we used is Ball Pivoting [4] on incomplete regions.

Since the regenerated vertices and edges of garments are aligned with those of the body mesh, it is inadequate to only compare the Euclidean distance of vertices of the



Fig. 5. The first row shows the garments generated by our network with different design patterns. The second row shows the most similar garments in the training data. Our model is capable of generating new garments.

original and reconstructed garment meshes. Assume we have mesh $M_1=\{V_1, E_1, F_1\}$ and mesh $M_2=\{V_2, E_2, F_2\}$, we define a mesh-based reconstruction error as the average distance from each point in V_1 to M_2 , and each point in V_2 to M_1 , shown as follows:

$$d_m = \frac{\sum_{p_1 \in V_1} dist(p_1, M_2) + \sum_{p_2 \in V_2} dist(p_2, M_1)}{\|V_1\| + \|V_2\|}$$

where $dist(p, M)$ is the smallest distance from point p to the surface of mesh M . We randomly sampled 6,000 garment instances from all the 260,000 garment instances in our training dataset, calculated the reconstruction error for each sample and computed the error distribution. The average percentage error is less than 1%, with the largest being less than 1.4%. The error distribution is shown in Appendix 6. Our method is robust to all garment topologies, materials, body poses, and shapes.

6.3 Clothing Style Generation

In this section, we demonstrate the generalization ability of our method. Specifically, we did the experiments in the following steps. First, we fed new label images and body information not included in the training data to the network, and obtained the image representation result. The output further went through the reconstruction algorithm and the post-processing and was finally transformed to the refined 3D garment. The generated image representation was further searched for its nearest neighbor in the training data using L1 distance. We retrieved the original mesh of the nearest neighbor for comparison.

In Fig. 5, we show the generated garments in the first row with different topologies or patterns. There are cases including a single-shoulder dress (the first column) and a backless dress (the last column), showing that our model is able to generate garments of varying topology. The second row shows the nearest neighbors in the training dataset. The geometric differences between the generated meshes and the nearest neighbors are significant, which means that our network can generalize to unseen topologies.



Fig. 6. Interpolation results between two specific cases. As shown in the figure, the garment changes smoothly from the leftmost style to the rightmost style, showing that our learned latent space is smooth and compact.

6.4 Interpolation Results

We did the interpolation experiment to show the effectiveness of our method. In the experiment, we chose two garments, generated the intermediate label images and fed them into our method. We show the interpolation results between two specific cases in Fig. 6. As shown in the figure, the garment changes smoothly from the leftmost style to the rightmost style, showing that our learned latent space is smooth and compact.

6.5 Garment Retargeting

Ease of retargeting is an important property in garment generation. In this experiment, we first generated a garment model with a specific body shape, then retargeted the generated garment to different body shapes. We show some of the retargeting results in Fig. 7, which are qualitatively as good as the results of Wang *et al.* [33]. We found that both algorithms can retain the appearance of the original garment retargeted onto bodies of different shapes and sizes. However, in their method, an additional Siamese network needs to be trained to achieve the retargeting goal [33], while our method can retarget the garment directly from the generated image representation and the new body shape – requiring less computation and demonstrating greater ease. Our method can also naturally ensure the consistency of the garment style by the definition of our image presentation. Other works, such as Brouet *et al.* [7], which is based on an optimization framework, or Guan *et al.* [14], which uses a learning-based architecture, are more computationally expensive than our method in the retargeting process.

6.6 Garment Generation Methods Comparison

There are methods that can generate garments through sketches, *e.g.*, Huang *et al.* [16] and Wang *et al.* [33]. Thanks to the information contained in the sketches, Huang *et al.*'s method can generate textures of garments, and Wang *et al.*'s method can generate garments with realistic wrinkles. However, our method only needs label images instead of full sketches. Also, our method can generate garments with different topologies given our image representation of garments, while these methods can only support at most three types of topologies.

In addition, a recent work Tex2Shape [1] can generate the combined body and garment mesh from photographs. However, it can only reconstruct the entire body mesh

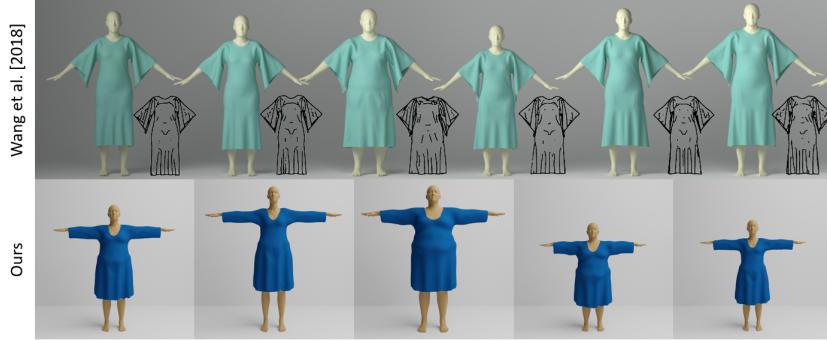


Fig. 7. Retargeting results for different body shapes and sizes, compared with Wang *et al.* [33]. The retargeting qualities are nearly the same qualitatively, *i.e.* both algorithms can retain the appearance of the original garment retargeted onto bodies of different shapes and sizes. However, an additional Siamese network is needed in their retargeting process, while our method retargets the cloth directly from the image representation, thereby requiring less computation than [33].

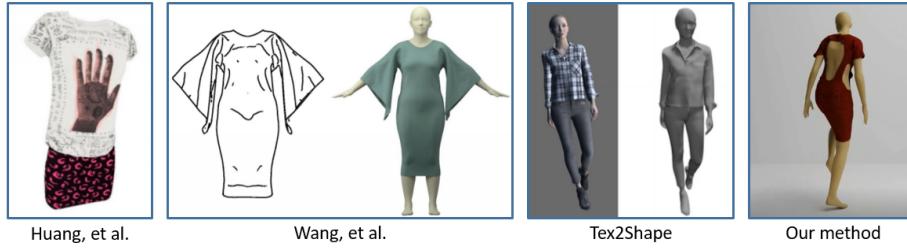


Fig. 8. Output comparison. Huang *et al.* [16] generate garment model with texture. Wang *et al.* [33] generate garments with realistic wrinkles as the sketch. Tex2Shape [1] generates combined body and garment models. Our method generates garments with various topologies.

with garments as a whole and is not able to separate the garment apart, while our method generates a stand-alone garment mesh. Moreover, Tex2Shape reconstructs the result with the same topology as the body mesh, so it can only handle body-like garments. In contrast, our method uses an extra label image to provide sewing information to the network, and reconstructs the garment mesh by training the network to assemble and stitch different pieces together, thereby applicable to generate garments of varying topologies.

We show the outputs of the three methods mentioned above and our method in Fig.8. Huang *et al.* [16] generate garment model with texture. Wang *et al.* [33] generate garments with realistic wrinkles as the sketch. Tex2Shape [1] generates combined body and garment model. Our method can generate garments with various topologies. Also, we show the different characteristics of different methods in Table 1. Because different methods have different characteristics and focus on different aspects, specific inputs would require different methods.

Table 1. Characteristic comparisons of different methods

Characteristics	Huang <i>et al.</i> [16]	Wang <i>et al.</i> [33]	Tex2Shape [1]	ours
input sketch	YES	YES	NO	NO
input photograph	NO	NO	YES	NO
input body pose or shape	NO	NO	NO	YES
input garment sewing pattern	NO	NO	NO	YES
use geometric representation	YES	NO	NO	NO
use GAN	NO	YES	YES	YES
use body UV map	NO	NO	YES	YES
infer body pose or shape	NO	YES	YES	NO
generate texture	YES	NO	NO	NO
generate wrinkles	NO	YES	NO	NO
generate body model	NO	NO	YES	NO
topology supported	Limited	Limited	Limited	Various

6.7 Performance

Our network inference (Sec. 5) takes about 369 msec on average, which is around 16.4% of the entire process. Garment reconstruction (Sec. 4.2) takes about 1,303 msec on average, around 57.9%. Post-processing refinement takes the last 25.7%, nearly 576 msec on average. Overall, our method takes 2,248 msec on average. Since the image resolution in our method is fixed to 512*512, the variation in image processing time is insignificant. It is possible to further accelerate the performance of our algorithm. Please refer to Appendix 8.

7 Conclusion

We presented a learning-based parametric generative model, which is the first garment generative model that can support any type of garment material, body shape, and most garment topologies. To offer this capability, we propose a special image representation of the garment model. Our method also makes garment retargeting much easier. In addition, a large garment dataset will be made available for further research in this area. **Limitation and Future Work:** Currently our method does not automatically generate fabric textures. In addition, due to the intermediate image representation of the garment, our method cannot generate multi-layer garment models, e.g., multi-layer lace skirts. This problem offers new research challenges. Our network can be further used as an extension of existing garment datasets because of its applicability and generalizability to unseen topologies. The generated 3D garments can also be used in user-driven fashion design and apparel prototyping.

Acknowledgment: This work is supported in part by Elizabeth Stevenson Iribe Professorship and National Science Foundation. We would like to thank the Research Groups led by Prof. Eitan Grinspun (Columbia University), Prof. Alla Sheffer (University of British Columbia), and Prof. Huamin Wang (Ohio State University) for sharing their design patterns datasets for the benchmarking and demonstration in this paper.

References

1. Alldieck, T., Pons-Moll, G., Theobalt, C., Magnor, M.: Tex2shape: Detailed full human body geometry from a single image pp. 2293–2303 (2019)
2. Amberg, B., Romdhani, S., Vetter, T.: Optimal step nonrigid icp algorithms for surface registration. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2007)
3. Bartle, A., Sheffer, A., Kim, V.G., Kaufman, D.M., Vining, N., Berthouzoz, F.: Physics-driven pattern adjustment for direct 3d garment editing. ACM Trans. Graph. **35**(4), 50–1 (2016)
4. Bernardini, F., Mittleman, J., Rushmeier, H., Silva, C., Taubin, G.: The ball-pivoting algorithm for surface reconstruction. IEEE transactions on visualization and computer graphics **5**(4), 349–359 (1999)
5. Bhatnagar, B.L., Tiwari, G., Theobalt, C., Pons-Moll, G.: Multi-garment net: Learning to dress 3d people from images. In: IEEE International Conference on Computer Vision (ICCV). IEEE (oct 2019)
6. Bradley, D., Popa, T., Sheffer, A., Heidrich, W., Boubekeur, T.: Markerless garment capture. ACM Trans. Graph. **27**(3), 99 (2008), <https://doi.org/10.1145/1360612.1360698>
7. Brouet, R., Sheffer, A., Boissieux, L., Cani, M.: Design preserving garment transfer. ACM Trans. Graph. **31**(4), 36:1–36:11 (2012), <https://doi.org/10.1145/2185520.2185532>
8. Chen, X., Zhou, B., Lu, F., Wang, L., Bi, L., Tan, P.: Garment modeling with a depth camera. ACM Trans. Graph. **34**(6), 203:1–203:12 (2015), <https://doi.org/10.1145/2816795.2818059>
9. Danerek, R., Dibra, E., Öztireli, A.C., Ziegler, R., Gross, M.H.: Deepgarment : 3d garment shape estimation from a single image. Comput. Graph. Forum **36**(2), 269–280 (2017), <https://doi.org/10.1111/cgf.13125>
10. Decaudin, P., Julius, D., Wither, J., Boissieux, L., Sheffer, A., Cani, M.: Virtual garments: A fully geometric approach for clothing design. Comput. Graph. Forum **25**(3), 625–634 (2006), <https://doi.org/10.1111/j.1467-8659.2006.00982.x>
11. Doersch, C.: Tutorial on variational autoencoders. CoRR **abs/1606.05908** (2016), <http://arxiv.org/abs/1606.05908>
12. Gabeur, V., Franco, J.S., Martin, X., Schmid, C., Rogez, G.: Moulding humans: Non-parametric 3d human shape estimation from single images. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2232–2241 (2019)
13. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. pp. 2672–2680 (2014), <http://papers.nips.cc/paper/5423-generative-adversarial-nets>
14. Guan, P., Reiss, L., Hirshberg, D.A., Weiss, A., Black, M.J.: DRAPE: dressing any person. ACM Trans. Graph. **31**(4), 35:1–35:10 (2012), <https://doi.org/10.1145/2185520.2185531>
15. Gundogdu, E., Constantin, V., Seifoddini, A., Dang, M., Salzmann, M., Fua, P.: Garnet: A two-stream network for fast and accurate 3d cloth draping. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 8739–8748 (2019)
16. Huang, P., Yao, J., Zhao, H.: Automatic realistic 3d garment generation based on two images. 2016 International Conference on Virtual Reality and Visualization (ICVRV) (2016)

17. Jeong, M., Han, D., Ko, H.: Garment capture from a photograph. *Journal of Visualization and Computer Animation* **26**(3-4), 291–300 (2015), <https://doi.org/10.1002/cav.1653>
18. Jung, A., Hahmann, S., Rohmer, D., Bégault, A., Boissieux, L., Cani, M.: Sketching folds: Developable surfaces from non-planar silhouettes. *ACM Trans. Graph.* **34**(5), 155:1–155:12 (2015), <https://doi.org/10.1145/2749458>
19. Lähner, Z., Cremers, D., Tung, T.: Deepwrinkles: Accurate and realistic clothing modeling. In: *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*. pp. 698–715 (2018), https://doi.org/10.1007/978-3-030-01225-0_41
20. Li, M.: FoldSketch: Enriching garments with physically reproducible folds. Ph.D. thesis, University of British Columbia (2018)
21. Liang, J., Lin, M.C.: Shape-aware human pose and shape reconstruction using multi-view images. In: *Proceedings of the IEEE International Conference on Computer Vision*. pp. 4352–4362 (2019)
22. Lin, M.C.: Efficient collision detection for animation and robotics. Ph.D. thesis, PhD thesis, Department of Electrical Engineering and Computer Science . . . (1993)
23. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: a skinned multi-person linear model. *ACM Trans. Graph.* **34**(6), 248:1–248:16 (2015), <https://doi.org/10.1145/2816795.2818013>
24. Loper, M.M., Mahmood, N., Black, M.J.: MoSh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* **33**(6), 220:1–220:13 (Nov 2014), <http://doi.acm.org/10.1145/2661229.2661273>
25. Narain, R., Samii, A., O'Brien, J.F.: Adaptive anisotropic remeshing for cloth simulation. *ACM Trans. Graph.* **31**(6), 152:1–152:10 (2012), <https://doi.org/10.1145/2366145.2366171>
26. Patel, C., Liao, Z., Pons-Moll, G.: Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 7365–7375 (2020)
27. Razavi, A., van den Oord, A., Vinyals, O.: Generating diverse high-fidelity images with vqvae-2. In: *Advances in Neural Information Processing Systems*. pp. 14866–14876 (2019)
28. Robson, C., Maharik, R., Sheffer, A., Carr, N.: Context-aware garment modeling from sketches. *Computers & Graphics* **35**(3), 604–613 (2011), <https://doi.org/10.1016/j.cag.2011.03.002>
29. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: *Proceedings of the IEEE International Conference on Computer Vision*. pp. 2304–2314 (2019)
30. Turquin, E., Cani, M., Hughes, J.F.: Sketching garments for virtual characters. In: *34. International Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2007, San Diego, California, USA, August 5-9, 2007, Courses*. p. 28 (2007), <https://doi.org/10.1145/1281500.1281539>
31. Wang, T., Liu, M., Zhu, J., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. pp. 8798–8807 (2018), http://openaccess.thecvf.com/content_cvpr_2018/html/Wang_High-Resolution_Image_Synthesis_CVPR_2018_paper.html
32. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2018)

33. Wang, T.Y., Ceylan, D., Popovic, J., Mitra, N.J.: Learning a shared shape space for multimodal garment design. CoRR **abs/1806.11335** (2018), <http://arxiv.org/abs/1806.11335>
34. Yang, S., Pan, Z., Amert, T., Wang, K., Yu, L., Berg, T., Lin, M.C.: Physics-inspired garment recovery from a single-view image. ACM Transactions on Graphics (TOG) **37**(5), 170 (2018)
35. Zheng, Z., Yu, T., Wei, Y., Dai, Q., Liu, Y.: Deephuman: 3d human reconstruction from a single image. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7739–7749 (2019)
36. Zhou, B., Chen, X., Fu, Q., Guo, K., Tan, P.: Garment modeling from a single image. Comput. Graph. Forum **32**(7), 85–91 (2013), <https://doi.org/10.1111/cgf.12215>

1 Samples of garment sewing pattern

We show garment sewing pattern samples in Fig.9, including a dress, pants, a shirt, and a skirt. Since sewing patterns offer common information about the garments, they are generally available.



Fig. 9. Garment sewing pattern samples. We show 4 cases here, including a dress, pants, a shirt, and a skirt. Since sewing patterns offer common information about the garments, they are generally available.

2 Example Meshes from Our Garment Dataset

In Fig. 10, we show that our garment dataset consists of clothing on different human body poses/sizes/shapes and of varying garment topologies, patterns, and materials. We sample ten human motions from the CMU MoCap dataset, including motions of walking, running, climbing and dancing. As stated in the main text, we have over 100 different garment types in the dataset, including dresses, t-shirts, pants, skirts and swimsuits. We use different material parameters and material space scales to control the sizes of the garments. Given this large and diverse dataset, our network can successfully disentangle different parts of the body label to generate garments with topologies totally different with those in the training dataset, while keeping a visually plausible result.

3 One-hot Version of the Label Image

In Fig. 2, we first transfer the label image to one-hot version by the one-hot operation. One-hot version of the label image is an array of binary images, where each pixel



Fig. 10. Examples meshes from our garment dataset. The dataset includes several common garment topologies and materials, as well as various human poses. The last two columns show the same garment pattern with different materials. The wrinkle appearances of the two sequences are different.

on the i -th binary image represents whether the label ID of that pixel on the original image equals to i . We use one-hot to support different garment components (e.g., shirts+jacket+pants) in future work. One-hot version of label image can decouple different class IDs and will be easier for the network to learn. Currently, we use it to differentiate between garment and other pixels. Overall, it is an extensible data format.

4 Data Format Transfer Process

Fig. 11 shows the data format transfer process. The garment model and the image representation of the garment can transfer to each other using the body mesh and UV map, as discussed in Sec. 4.

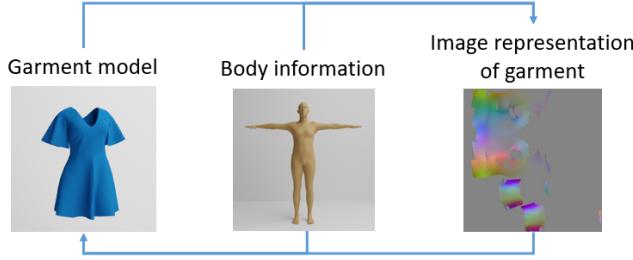


Fig. 11. Data format transfer process. The garment model and the image representation of the garment can transfer to each other using the body mesh.

5 Point Matching between Body and Garment

We show the mapping process from body surface pixels to the garment surface in Fig. 12. Within the Voronoi region of a body vertex, the ray direction of a pixel (brown) is interpolated between the vertex normal (black) and the face normal (gray), according to the barycentric coordinates.

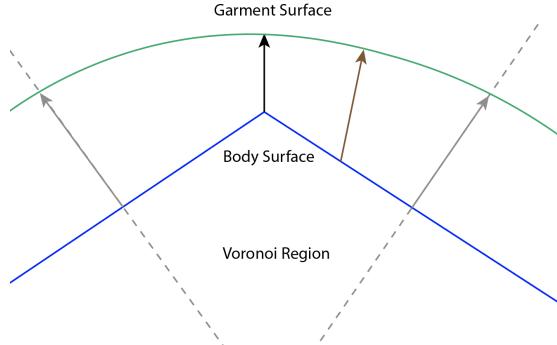


Fig. 12. Mapping from body surface pixels to the garment surface. Within the Voronoi region of a body vertex, the ray direction of a pixel (brown) is interpolated between the vertex normal (black) and the face normal (gray), according to the barycentric coordinates.

6 Reconstruction Results under Different Conditions

We show in Fig. 14 that our algorithm introduced in Sec. 4 is robust to any kind of garment input. We tested our algorithm with garments of different topologies on different human bodies. Different cloth materials are expressed with different sizes and detailed

wrinkles based on the geometry. Our method can also retain the material information faithfully.

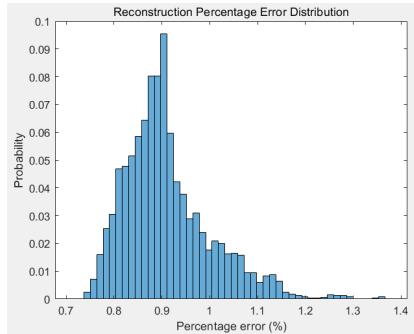


Fig. 13. Distribution of reconstruction error d_m (in percentage w.r.t. the garment height) over 6,000 randomly selected garment instances. The error is relatively small across all types of garments, with the largest being less than 1.4% and most within 1%.

Fig. 13 shows the distribution of d_m in our training dataset. The error is relatively small across all types of garments, with the largest being less than 1.4% and most of them within 1%.

7 Garment Retargeting

In Sec. 6.5, we show the garment retargeting results using only a T-pose. In Fig. 15, we show more cases with different poses. As shown in the figure, our method can retarget garments with different topologies, patterns, and materials to bodies with different shapes, sizes, and poses.

8 Performance

Our method takes about 2,248 msec on average. For garment generation, 2 seconds would be quite acceptable if the quality is good enough, while the garments designed manually usually take much longer. Also, there is room for performance improvement and parallelization of the post-processing after the network inference. More importantly, we use a resolution of 512*512 for the displacement map, so there are up to 512*512 vertices and 511*511*3 edges in our reconstructed mesh, with resolution much higher than other works, thus taking slightly longer. As needed, the implementation of our method can be much improved by reducing the resolution of the displacement map.



Fig. 14. Reconstructed mesh results under different human poses and shapes, garment topologies, sizes, and materials. Our data transfer method is able to map any 3D mesh to its 2D image representation with little information loss.

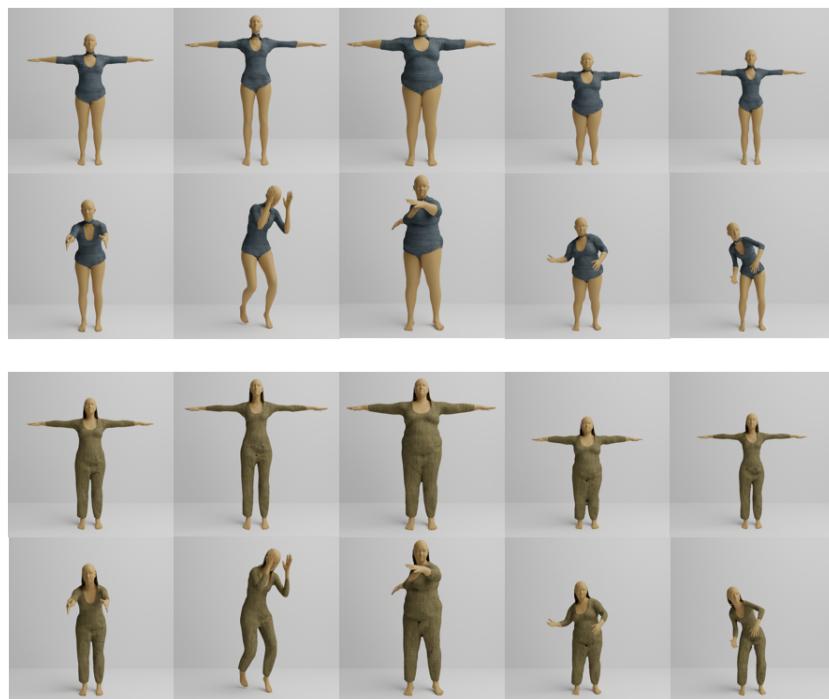


Fig. 15. Garment retargeting results. Our method can retarget garments with different topologies, patterns, and materials to bodies with different shapes, sizes, and poses.