

000 GAN-based Garment Generation Using Sewing 001 Pattern Images

004 Anonymous ECCV submission

005 Paper ID 2959

009 **Abstract.** The generation of realistic apparel model has become in-
010 creasingly popular as a result of the rapid pace of change in fashion trends
011 and the growing need for garment models in various applications such as
012 virtual try-on. For such application requirements, it is important to have
013 a general cloth model that can represent a diverse set of garments. Pre-
014 vious studies often make certain assumptions about the garment, such
015 as the topology or body shape. We propose a unified method us-
016 ing the generative network. Our model is applicable to different garment
017 topologies with different sewing patterns and fabric materials. We also
018 develop a novel image representation of garment models, and a reliable
019 mapping algorithm between the general garment model and the image
020 representation that can regularize the data representation of the cloth.
021 Using this special intermediate image representation, the generated gar-
022 ment model can be easily retargeted to another body, enabling garment
023 customization. In addition, a large garment appearance dataset is pro-
024 vided for use in garment reconstruction, garment capturing, and other
025 applications. We demonstrate that our generative model has high recon-
026 struction accuracy and can provide rich variations of virtual garments.

027 1 Introduction

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

038 However, there are many challenges in automatically generating garment
039 models. First, garments usually have different topologies, especially for fashion
040 apparel, that make it difficult to design a universal generation pipeline. Moreover,
041 it is often not straightforward for the general garments design to be retargeted
042 onto another body shape, making it difficult for customization. Some previous
043 work started to address this problem using either user-assisted input [12] or
044 cloths with fixed topology such as a T-shirt or a skirt [25].

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 [12, 25] 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. This approach takes one or more sketches as input and generates the garment model. Turquin *et al.* [22] and Decaudin *et al.* [8] developed some of the early work in this area. They used grid and geometric methods to generate garment models with sketches. However, the garment models generated by these methods have limited visual quality. Later, Robson *et al.* [21] 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. These models are, however, fixed to a given body shape. Jung *et al.* [14] proposed a method to model 3D developable surfaces with a multi-view sketch input. Recently, Huang *et al.* [12] proposed a realistic 3D garment generation algorithm based on front and back image sketches, but it cannot retarget the generated garments to other body shapes easily. Wang *et al.* [25] proposed an algorithm that can achieve retargeting conveniently, but is limited to very few topology, namely T-shirts or skirts.

In addition, a common limitation using the sketch-based algorithm is that they require domain knowledge on garment sketching. Our method, in contrast, 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.* [4] researched early on markerless image-based garment modeling using multi-view images. Later, Zhou *et al.* [27] proposed a single-view image approach. In their work, a human shape was estimated from the image and the garment model was reconstructed with the garment outline. Jeong *et al.* [13] created the garment model with a single photograph by detecting the landmark points of the garment. Yang [26] made a full use of garment and human body databases to generate the garment models from a single-view image. Daněřek *et al.*'s [7] method can estimate the 3D garment shape from a single image using deep neural networks. Tex2Shape [1] is an image-to-image translation model for detailed full-body geometry reconstruction. MGN [3] predicts body shape and clothing, layered on top of the SMPL [17] model from a few (typically 1 - 8) frames of a video. Depth information can also be useful. Chen *et al.* [6] 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.* [5] introduced a fully automatic method for design-preserving transfer of garments among characters with different body shapes.

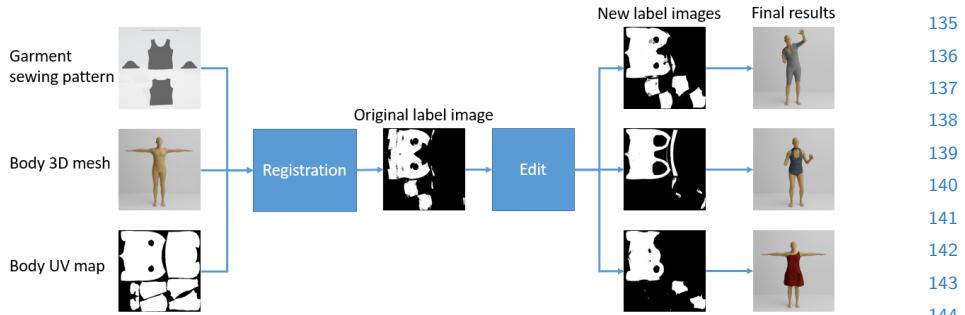


Fig. 1. Label image generation process. We first generate the label image with the pattern configuration registered on the body mesh and mapped to the body UV map. Then we can edit the original label image to new, different label images, which will lead to different garment topologies in the final results.

In contrast, Guan *et al.* [11] used a learning-based architecture to replace the expensive simulation process and present retargeting examples.

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 Adversarial Network (GAN) [10] and Variational Auto-Encoder (VAE) [9]. With the development of the neural network research, new variants of generative networks have been proposed, such as Pix2PixHD [23] based on GAN or VQ-VAE [20] based on VAE. In our algorithm, we design the network architecture based on the Pix2PixHD network architecture due 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

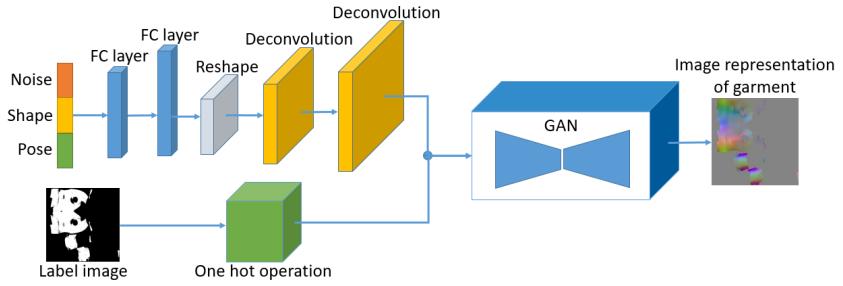


Fig. 2. Our network architecture. We first encode one dimensional input to match the sizes to the label image (upper branch). It is then concatenated with one hot labelled image (bottom branch) and fed into the GAN network. Finally, the network outputs the image representation of the garment (right).

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 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).

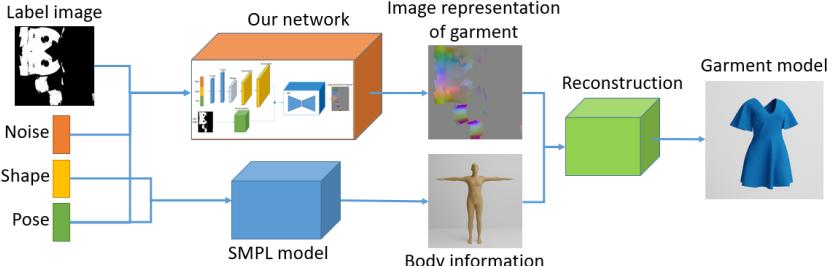


Fig. 3. Our inference pipeline. The upper branch generates the image representation of the garment, while the bottom branch generates the body mesh. Finally, we recover the garment mesh by decoding the image representation of the garment given the body mesh.

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, such as Lähner *et al.* [15] and Bhatnagar *et al.* [3]. 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 to register the cloth surface to the body surface. We penalize the scaling of the cloth more than the translation to ensure the cloth can retain its size as much as possible. The non-rigidity is also carefully enforced so that the cloth is

more evenly distributed on the body surface. Note that after the 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 of the surfaces between the cloth and the body.

We first create the correspondence between the faces of the cloth mesh and the vertices of the body mesh according to the Euclidean distance. Specifically, we cut the registered cloth mesh using the 3D Voronoi diagram of the body surface. If a face of the cloth mesh belongs to multiple Voronoi regions (*i.e.*its nodes have different closest vertices of the body), we subdivide the face using the Voronoi plane. 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 formed by the perpendicular bisector lines of each of its adjacent faces. 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. 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 the fact that the pixels on the edge of the cuts are aligned to the points on the garment edge. This ensures 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 preprocessing: 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 wrinkles. 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.

315 5 Latent Space Learning

316
317 We apply a GAN-based model to learn the latent space of the representation
318 image. Our network structure is shown in Fig. 2.

319 Since the pixel values in the representation image are related to the human
320 body pose and shape, we add them as the conditional input in the network.
321 Additionally, we provide a label map that indicates the overall topology of the
322 garment to further constrain the generated image. The noise vector here serves
323 mostly as the encoded detailed appearance, such as wrinkles and tightness of
324 the cloth. We re-format the label image to one-hot version, and concatenate
325 it with the encoded features of the other 1D input. Currently we only have
326 binary information for the garment label map, but we can also support labels of
327 different garment parts, as long as the corresponding data is provided. We use
328 Pix2PixHD [24] as our backbone network, but other state-of-the-art methods
329 can also work in practice.

330 5.1 Loss Functions and Training Process

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

$$336 \quad \mathcal{L} = \mathcal{L}_{GAN} + \lambda_0 \mathcal{L}_{feat} \quad (1)$$

$$341 \quad \mathcal{L}_{GAN} = \|D(I_{real}) - 1\|_1 + \|D(I_{fake})\|_1 \\ 342 \quad + \|D(G(I_{fake})) - 1\|_1 \quad (2)$$

$$345 \quad \mathcal{L}_{feat} = \|D^*(I_{real}) - D^*(I_{fake})\|_1 \\ 346 \quad + \|VGG^*(I_{real}) - VGG^*(I_{fake})\|_1 \quad (3)$$

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

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

$$356 \quad \mathcal{L} = \mathcal{L}_{GAN} + \lambda_1 \mathcal{L}_{smooth} \quad (4)$$

$$358 \quad \mathcal{L}_{smooth} = \left\| \frac{\partial I_{fake}}{\partial x} \right\|_1 + \left\| \frac{\partial I_{fake}}{\partial y} \right\|_1 \quad (5)$$

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

365 366 5.2 Data Preparation

367 To learn the network model with high accuracy and variety, a large dataset
 368 depicting the joint distribution between the garment geometry and the human
 369 body is required. Previous datasets such as Bhatnagar *et al.* [3] or Liang *et
 370 al.* [16] have limited garment styles and body motions and are thus not suitable
 371 for our needs. Therefore, we propose a physics-based simulated dataset to rep-
 372 resent most common garment types, human motions, and cloth materials. We
 373 sample different human motions and body shapes using the Moshed CMU Mo-
 374 Cap dataset [18]. Our garments are obtained from various online sources, which
 375 we will make public with the dataset. We initialize the human to a T-pose and
 376 dress the body with each of the garments. Then we use the cloth simulator [19] to
 377 generate the cloth motion along with the body motion. We notice that the cloth
 378 material of the garment can significantly alter the appearance, so we also vary
 379 the material parameters during data generation. For quantitative details, please
 380 refer to Sec. 6.3. We show examples of different garment data in Appendix 2.

382 383 6 Experimental Analysis

384 In this section, we will first introduce the implementation details of our method.
 385 Next, we show the effectiveness and performance of the key parts of our method
 386 by various experiments, including garment reconstruction, clothing style gener-
 387 ation, and garment retargeting.

390 391 6.1 Implementation details

392 We collected 104 types of garment models, each with 10 materials, and chose
 393 one random body motion sequence out of the 10 most commonly seen sequences.
 394 Then we dressed the garment on the body and simulated it using a cloth sim-
 395 ulator [19] to generate a series of garment meshes with different poses, thereby
 396 generating $104 \times 10 \times 250 = 260,000$ garment instances. After that, we applied the
 397 representation transfer process on those garment instances and generated the
 398 image representation as well as the label mask. Next, we fed the images together
 399 with body shapes, poses, and the label images to the network for training. In
 400 practice, we randomly chose 2 materials in each epoch, to reduce the training
 401 time while making full use of the whole dataset.

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

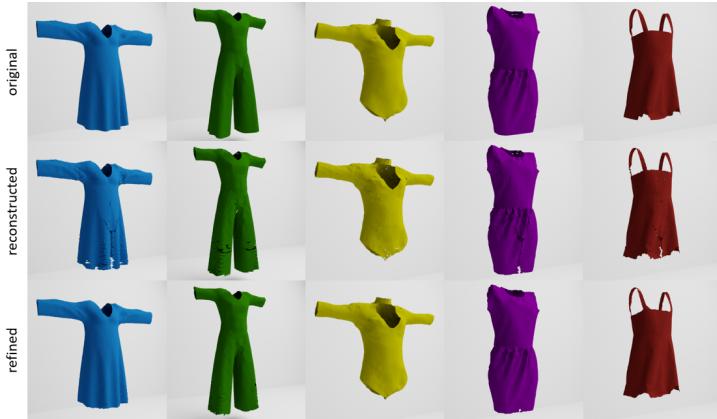


Fig. 4. Comparison between original mesh (first row), reconstructed mesh (second row), and refined mesh (third row). Our method is able to retain most of the original information, independent of the topology or the geometry of the garment mesh. The refined meshes indicate that the post-process is able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

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 [2] 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 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:

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



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.

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 4. Our method is robust to all garment topologies and materials, as well as different 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.

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



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.

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.* [25]. 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 [25], 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.* [5], which is based on an optimization framework, or Guan *et al.* [11], 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.* [12] and Wang *et al.* [25]. 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. This method can infer the body parameters, and thereby generate realistic body mesh with garments. However, it can only reconstruct the entire body mesh 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

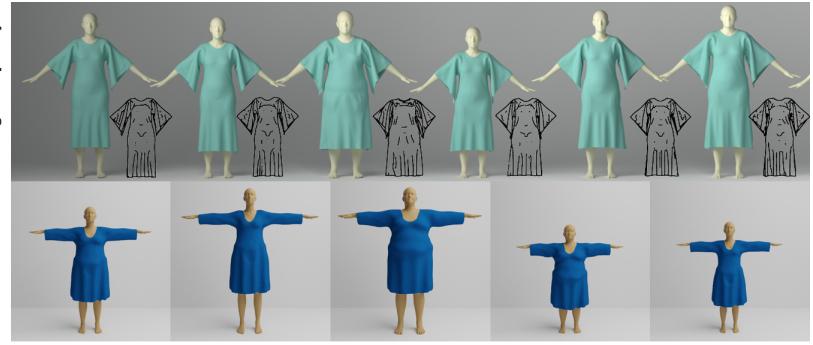


Fig. 7. Retargeting results for different body shapes and sizes, compared with Wang *et al.* [25]. The retargeting qualities are nearly the same qualitatively, *i.e.* both algorithms can retain the appearance of the original garment retargeted onto bodies of different shapes and sizes. However, an additional Siamese network is needed in their retargeting process, while our method retargets the cloth directly from the image representation, thereby requiring less computation than [25].

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.

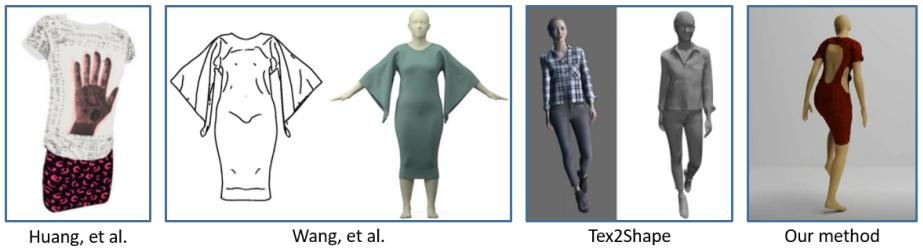
We show the outputs of the three methods mentioned above and our method in Fig.8. Huang *et al.* [12] generate garment model with texture. Wang *et al.* [25] generate garments with realistic wrinkles as the sketch. Tex2Shape [1] generates combined body and garment model. Our method can generate garments 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.

6.7 Performance

In many applications, performance is a significant consideration as it affects manufacturing efficiency. 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. Since the image resolution in our method is fixed to 512*512, the variation in image processing time is insignificant.

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



Huang, et al.

Wang, et al.

Tex2Shape

Our method

Fig. 8. Output comparison. Huang *et al.* [12] generate garment model with texture. Wang *et al.* [25] generate garments with realistic wrinkles as the sketch. Tex2Shape [1] generates combined body and garment models. Our method generates garments with various topologies.

Table 1. Characteristic comparisons of different methods

Characteristics	Huang <i>et al.</i> [12]	Wang <i>et al.</i> [25]	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

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.

630 References

- 632 1. Alldieck, T., Pons-Moll, G., Theobalt, C., Magnor, M.: Tex2shape: Detailed full
633 human body geometry from a single image pp. 2293–2303 (2019)
- 634 2. Bernardini, F., Littleman, J., Rushmeier, H., Silva, C., Taubin, G.: The ball-
635 pivoting algorithm for surface reconstruction. IEEE transactions on visualization
636 and computer graphics **5**(4), 349–359 (1999)
- 637 3. Bhatnagar, B.L., Tiwari, G., Theobalt, C., Pons-Moll, G.: Multi-garment net:
638 Learning to dress 3d people from images. In: IEEE International Conference on
639 Computer Vision (ICCV). IEEE (oct 2019)
- 640 4. Bradley, D., Popa, T., Sheffer, A., Heidrich, W., Boubekeur, T.: Markerless garment
641 capture. ACM Trans. Graph. **27**(3), 99 (2008). <https://doi.org/10.1145/1360612.1360698>,
642 <https://doi.org/10.1145/1360612.1360698>
- 643 5. Brouet, R., Sheffer, A., Boissieux, L., Cani, M.: Design
644 preserving garment transfer. ACM Trans. Graph. **31**(4),
645 36:1–36:11 (2012). <https://doi.org/10.1145/2185520.2185532>,
646 <https://doi.org/10.1145/2185520.2185532>
- 647 6. Chen, X., Zhou, B., Lu, F., Wang, L., Bi, L., Tan, P.: Gar-
648 ment modeling with a depth camera. ACM Trans. Graph.
649 **34**(6), 203:1–203:12 (2015). <https://doi.org/10.1145/2816795.2818059>,
650 <https://doi.org/10.1145/2816795.2818059>
- 651 7. Danerek, R., Dibra, E., Öztireli, A.C., Ziegler, R., Gross, M.H.: Deepgarment : 3d
652 garment shape estimation from a single image. Comput. Graph. Forum **36**(2), 269–
653 280 (2017). <https://doi.org/10.1111/cgf.13125>, <https://doi.org/10.1111/cgf.13125>
- 654 8. Decaudin, P., Julius, D., Wither, J., Boissieux, L., Sheffer, A., Cani, M.: Vir-
655 tual garments: A fully geometric approach for clothing design. Comput. Graph.
656 Forum **25**(3), 625–634 (2006). <https://doi.org/10.1111/j.1467-8659.2006.00982.x>,
657 <https://doi.org/10.1111/j.1467-8659.2006.00982.x>
- 658 9. Doersch, C.: Tutorial on variational autoencoders. CoRR **abs/1606.05908** (2016),
659 <http://arxiv.org/abs/1606.05908>
- 660 10. Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S.,
661 Courville, A.C., Bengio, Y.: Generative adversarial nets. In: Advances in Neural
662 Information Processing Systems 27: Annual Conference on Neural Information
663 Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada. pp.
664 2672–2680 (2014), <http://papers.nips.cc/paper/5423-generative-adversarial-nets>
- 665 11. Guan, P., Reiss, L., Hirshberg, D.A., Weiss, A., Black, M.J.: DRAPE: dressing any person. ACM Trans. Graph. **31**(4),
666 35:1–35:10 (2012). <https://doi.org/10.1145/2185520.2185531>,
667 <https://doi.org/10.1145/2185520.2185531>
- 668 12. Huang, P., Yao, J., Zhao, H.: Automatic realistic 3d garment generation based on
669 two images. 2016 International Conference on Virtual Reality and Visualization
670 (ICVRV) (2016)
- 671 13. Jeong, M., Han, D., Ko, H.: Garment capture from a photograph. Journal
672 of Visualization and Computer Animation **26**(3-4), 291–300 (2015).
673 <https://doi.org/10.1002/cav.1653>, <https://doi.org/10.1002/cav.1653>
- 674 14. Jung, A., Hahmann, S., Rohmer, D., Bégault, A., Boissieux, L., Cani,
675 M.: Sketching folds: Developable surfaces from non-planar silhouettes. ACM
676 Trans. Graph. **34**(5), 155:1–155:12 (2015). <https://doi.org/10.1145/2749458>,
677 <https://doi.org/10.1145/2749458>

- 675 15. Lähner, Z., Cremers, D., Tung, T.: Deepwrinkles: Accurate and realistic
676 clothing modeling. In: Computer Vision - ECCV 2018 - 15th European Conference,
677 Munich, Germany, September 8-14, 2018, Proceedings, Part IV. pp. 698–715 (2018). https://doi.org/10.1007/978-3-030-01225-0_41
678
- 680 16. Liang, J., Lin, M.C.: Shape-aware human pose and shape reconstruction using
681 multi-view images. In: Proceedings of the IEEE International Conference on Computer Vision.
682 pp. 4352–4362 (2019)
- 684 17. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: a skinned
685 multi-person linear model. ACM Trans. Graph. **34**(6), 248:1–248:16 (2015). <https://doi.org/10.1145/2816795.2818013>
686
- 688 18. Loper, M.M., Mahmood, N., Black, M.J.: MoSh: Motion and shape capture
689 from sparse markers. ACM Transactions on Graphics, (Proc. SIGGRAPH Asia) **33**(6),
690 220:1–220:13 (Nov 2014). <https://doi.org/10.1145/2661229.2661273>
692
- 694 19. Narain, R., Samii, A., O'Brien, J.F.: Adaptive anisotropic remeshing for cloth simulation.
695 ACM Trans. Graph. **31**(6), 152:1–152:10 (2012). <https://doi.org/10.1145/2366145.2366171>
697
- 699 20. Razavi, A., van den Oord, A., Vinyals, O.: Generating diverse high-fidelity images with
700 VQ-VAE-2. CoRR **abs/1906.00446** (2019), <http://arxiv.org/abs/1906.00446>
- 702 21. Robson, C., Maharik, R., Sheffer, A., Carr, N.: Context-aware garment modeling from sketches.
704 Computers & Graphics **35**(3), 604–613 (2011). <https://doi.org/10.1016/j.cag.2011.03.002>
706
- 708 22. 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>
710
- 712 23. 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). <https://doi.org/10.1109/CVPR.2018.00917>,
714 http://openaccess.thecvf.com/content_cvpr_2018/html/Wang_High-Resolution_Image_Synthesis_CVPR_2018_paper.html
- 716 24. 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)
- 718 25. Wang, T.Y., Ceylan, D., Popovic, J., Mitra, N.J.: Learning a shared shape space for multimodal garment design. CoRR **abs/1806.11335** (2018),
720 <http://arxiv.org/abs/1806.11335>
- 722 26. 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)

- 720 27. Zhou, B., Chen, X., Fu, Q., Guo, K., Tan, P.: Garment modeling
721 from a single image. Comput. Graph. Forum **32**(7), 85–91 (2013).
722 <https://doi.org/10.1111/cgf.12215>, <https://doi.org/10.1111/cgf.12215>

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

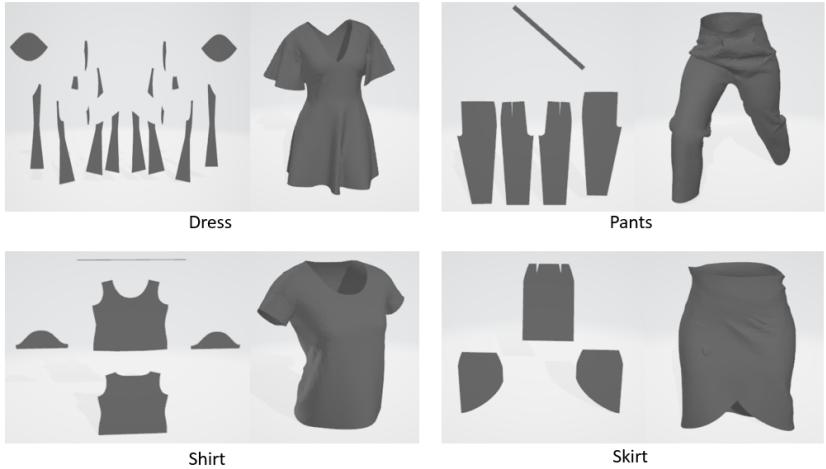
762

763

764

765 1 Samples of garment sewing pattern

766
767
768 We show garment sewing pattern samples in Fig.9, including a dress, pants, a
769 shirt, and a skirt. Since sewing patterns offer common information about the
770 garments, they are generally available.



789
790 **Fig. 9.** Garment sewing pattern samples. We show 4 cases here, including a dress,
791 pants, a shirt, and a skirt. Since sewing patterns offer common information about the
garments, they are generally available.

792 2 Example Meshes from Our Garment Dataset

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



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.

3 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.

4 Reconstruction Results under Different Conditions

We show in Fig. 13 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

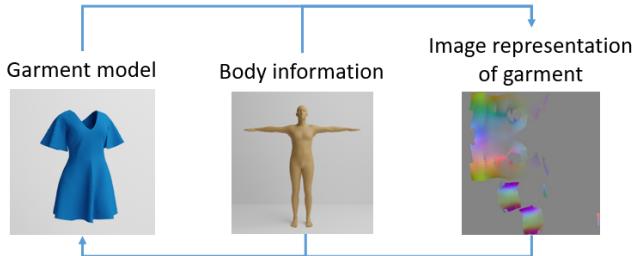


Fig. 11. Data format transfer process. The garment model and the image representation of the garment can transfer to each other using the body mesh.

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

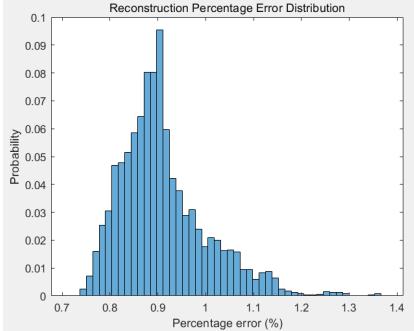


Fig. 12. Distribution of reconstruction error d_m (in percentage w.r.t. the garment height) over 6,000 randomly selected garment instances. The error is relatively small across all types of garments, with the largest being less than 1.4% and most within 1%.

Fig. 12 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%.

5 Garment Retargeting

In Sec. 6.5, we show the garment retargeting results using only a T-pose. In Fig. 14, 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.



Fig. 13. 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.

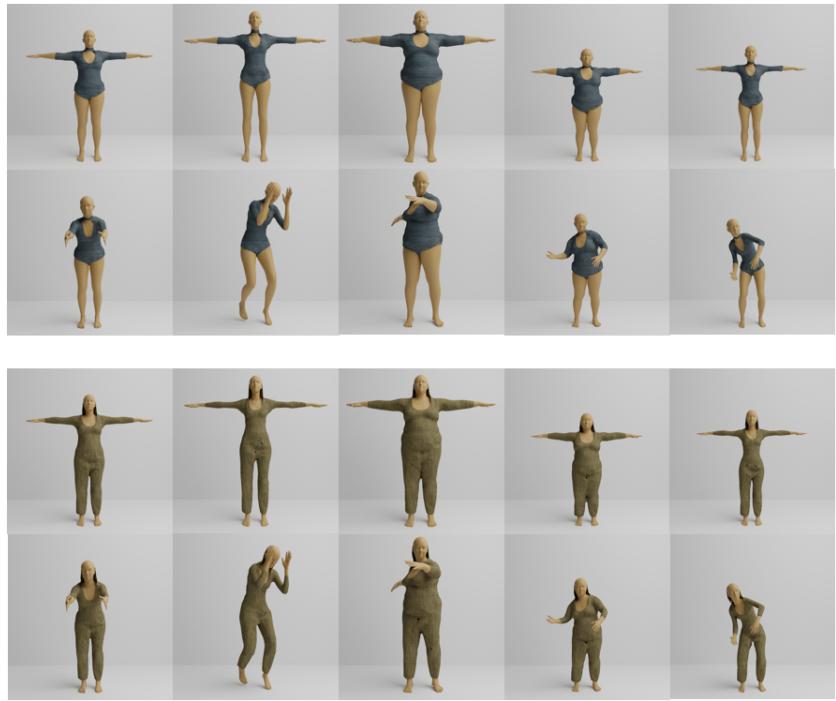


Fig. 14. Garment retargeting results. Our method can retarget garments with different topologies, patterns, and materials to bodies with different shapes, sizes, and poses.