DeepCloth: Neural Garment Representation for Shape and Style Editing

Zhaoqi Su[1], Tao Yu[1], Yangang Wang[2], Yipeng Li[1], Yebin Liu[1],
[1]Tsinghua University, Beijing, China [2]Southeast University, Nanjing, China

Abstract

Garment representation, animation and editing is a challenging topic in the area of computer vision and graphics. Existing methods cannot perform smooth and reasonable garment transition under different shape styles and topologies. In this work, we introduce a novel method, termed as DeepCloth, to establish a unified garment representation framework enabling free and smooth garment style transition. Our key idea is to represent garment geometry by a "UV-position map with mask", which potentially allows the description of various garments with different shapes and topologies. Furthermore, we learn a continuous feature space mapped from the above UV space, enabling garment shape editing and transition by controlling the garment features. Finally, we demonstrate applications of garment animation, reconstruction and editing based on our neural garment representation and encoding method. To conclude, with the proposed DeepCloth, we move a step forward on establishing a more flexible and general 3D garment digitization framework. Experiments demonstrate that our method can achieve the state-of-the-art garment modeling results compared with the previous methods.

1. Introduction

3D garment modeling and editing has numerous applications in clothing designing, realistic 3D human generation, virtual try-on and so on. Traditionally, high-fidelity 3D garment representation and animation often relies on Physically Based Simulation (PBS) [26]. Though, such a method takes enormous labor costs and computational resources. In recent years, the rapid progress of easy-to-use garment representation and modeling has been promoted by deep learning [18, 12, 21, 15, 33, 29]. However, it is still difficult for these methods to perform the garment shape style transition among different shapes and topologies, because they either lack a garment shape parametrization framework or are restricted to representing a single type of garment with similar topologies. For example, the most recent TailorNet [22] proposes a novel garment shape parametrization, which can be utilized for 3D clothing animation of different human shapes and poses. However, TailorNet [22] represents garments with a pre-defined fixed template, in which the vertexes of the garment in TailorNet heavily rely on the fixed subset on the surface of SMPL [20] model. This method limits its ability for garment representation of different topologies, e.g., long dresses and wide front-opening garments which floats away from the human body. Besides, it can only perform a small range of garment shape transition, for example, it cannot achieve garment transition from long pants to shorts, or from front-opening T-shirts to front-closing shirts.

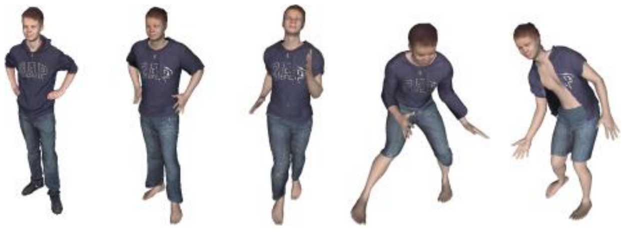


Figure 1. The demonstration of our garment shape inference, animation and 3D editing by our DeepCloth method. From left to right: (a) input 3D scan, (b) garment shape inference and reconstruction using our neural representation, (c) animation of (b) under new pose, (d, e) two garment animation with garment style and shape editing.

It is essential to perform garment representation that learns the shape and style space of 3D garments, which can enable free and smooth style transition between different garment topologies, even from open-fronting to close-fronting clothes. Such a neural garment representation will enable a large number of applications, as shown in Fig. 1. By transferring the garment representation into a neural feature space, and mapping the 3D scanned garment mesh into the same feature space, such representation can also be used to perform garment animation and 3D garment shape editing using deep neural networks.

In this paper, we propose DeepCloth, a neural garment representation framework to achieve the above goal. Our key idea is to overcome the limitations of existing methods, by assembling different garment shapes and topologies into a unified representation framework. Technically, we leverage the "UV-position map with mask" representation to encrypt both the topologies and shape details of the garments. In order to achieve plausible garment transition under different topologies, we transform the UV binary mask into a continuous distance map. By encoding the UV-position map with transformed mask into a feature space using the CNN network, we can represent garments with different shape styles in a unified network, and thus achieve flexible garment editing and continuous shape and style transition between different garments by feature interpolation and decoding. With our deep learning network trained on the large scale synthetic dataset of 3D clothed human sequences with various garment styles, i.e. CLOTH3D [5], we can parameterize clothing shape variation of front-opening T-shirt, T-shirt, shirt, pants, skirt, dress, and jumpsuits. Also, with our proposed animation and 3D-shape inference module, DeepCloth can generate dynamic garment 4D sequences (see Fig. 1(c)), or extract the clothing shape parameters from a clothed human model under arbitrary pose, which enhances its ability for 3D clothing shape editing (see Fig. 1(d, e)). The main contributions of this work are summarized as follows:

- We propose the “UV-position map with mask” garment representation for encrypting garment shapes and topologies, enabling garment shape transition between different garment styles (Sect. 4).
- We introduce a distance-transform based mask transformation for transforming the binary mask into a continuous form, which tackles the problem of smooth transition between different masks, and further between different garment topologies (Sect. 5).
- We propose ParamNet for encoding the “UV-position map with mask” representation into a feature space, which encodes the garments into a unified, neural and shape-editable framework (Sect. 5).
- We introduce the garment animation module AnimNet and 3D garment shape parameter estimation module 3DInferNet. AnimNet can generate dynamic 4D sequences for different garment styles wearing on different persons, while 3DInferNet reconstructs garment shape parameters from randomly posed garment 3D scan, and performs garment editing and animation accordingly (Sect. 6, 7).

2. Related Work

There are numerous works on garment representation, animation and reconstruction. Here, we mainly review the works that are most related to our approach.

Garment representation and animation. There are basically three approaches for garment animation: physics based simulation (PBS), data-driven methods, and animation based on capture.

For physics based simulation (PBS), traditional physics based garment simulation formulate the garment as mass-spring system with the force-based simulation [26, 10, 19], or other physical models based on the finite element method [7, 14], with the explicit Euler method [25] or implicit/semi-implicit Euler method [32, 4, 25]. These methods can generate realistic clothing with vivid dynamics given a designed garment shape and garment template, but mostly take much computational costs for numerous integration iterations for clothing dynamics, and cannot perform a more general shape control of the garment.

For data-driven methods, they aim to shorten the computational time for garment animation with a more flexible garment representation. Early methods such as [34, 11, 17] use nearest neighbor search or linear regression to animate clothing under human body with different poses and shapes. Recent works mainly adopt deep learning methods to perform garment animation. [35] learns a shared space for garment style variation, and can predict garment shape from a user sketch with a fixed pose. [28, 36] regress the garment shape with various human poses and shapes with MLP or RNN methods. [13, 12, 21] propose garment animation by 3D garment draping or SMPL-based garment deformation, using graph convolution network for obtaining garment shape wearing on human model with different shapes and poses. [18, 15] propose pixel-based garment 2D representation based on texture mapping on the human model, or on a template based texture space, which is similar to our representation method, but they cannot generating the shape parameters of the garments. [29] can interpolate between different garment styles, but it can only interpolate the “sewing patterns”, meaning that [29] only interpolates the area of the body covered by the garments, without a general shape parametrization framework for generating more kinds of garment shape. These methods use garment vertexes, graph based garment representation or pixel-based garment representation to perform garment animation, but lack garment shape parametrization module for flexibly and conveniently controlling the garment shape.

For garment animation based on real capture, this kind of method for garment animation focus on recovering the static or dynamic garment shape from a given picture or video. [8, 24, 30] recover garment shape from multi-view stereo, [23, 18] recover garment dynamics from 4D-sequences, [9, 37] propose a system for garment shape recovery from single RGBD camera, while [31] proposes a method for extracting the garment template shape and recovering garment dynamics from single RGB input. Recently, [6] proposes a method for extracting multiple garments from several input images and dress it on other human bodies, while [16, 1] propose CNN-based method for recovering the human shape and pose, such methods express the garment as deformation of the subset of human model, difficult to express more kinds of garments like dresses and loosened front-opening clothing. Alldieck *et al*. [3] proposes a 2D texture based human with garment shape representation method for recovering the whole shape from a single image. These methods mainly focus on recover garment shape from input images, without proposing a general garment shape representation framework.

Garment shape parametrization. Recently, a few works focus on establishing a garment shape parametrization framework. Shen *et al*. [29] shows the garment style interpolation results, but it only controls the change of covering area on the human body, not generating a general shape expression. Tiwari *et al*. [33] proposes a framework for parsing the 3D input to extract the garment shape and change the size of the garment, but in view of shape parametrization, it only controls one dimension of the garment shape. TailorNet [22] proposes a garment shape parametrization and animation framework, however, as mentioned in Sect. 1, with the "offset on template vertexes" expression, it is hard for TailorNet [22] to be generalized to more kinds of garment topology, such as front-opening garments and long dresses. Besides, it shows limited ability on performing large garment shape changes, e.g., from long trousers to shorts, or from long dresses to skirts. Also, compared to our DeepCloth, it has less capability on performing 3D garment shape inference and flexible 3D shape editing. Therefore, it does not meet the demand for establishing a general framework for garment representation enabling garment shape and style transition. Meanwhile, our DeepCloth proposes a general garment shape representation framework, which enables more general 3D garment generation, animation and editing.

3. Overview

Our goal is to perform smooth and reasonable garment style transition between different garment shapes and topologies, e.g., from long-sleeve open-fronting shirts to short-sleeve T-shirts, and represent them in a same framework. Therefore, we first propose a topology-specific UV-position map with mask representation, in which the mask denotes the topology and covering areas of the garments, while the rendered value on the UV map denotes the geometry details (see Sect. 4). Such a representation transfers the garment shape style and topology into a 2D UV-map, naturally suitable for continuously transition between different garment shapes. Then, we perform UV map encoding by introducing ParamNet, which maps both the UV map and its mask information into a feature space by using a CNN-based encoder-decoder structure (see Sect. 5). By changing and interpolating the features in the feature space, garment shape transition and editing can be performed, and can be applied to the following garment animation and shape inference module (see Sect. 6 and Sect. 7). With the learned garment space, the animation module can generate dynamic 4D clothed human sequences (Sect. 6), and the shape inference module can reconstruct garment shape from given 3D scans, which transforms the mesh or point cloud representation to our neural garment representation, and allows further garment animation or shape editing (Sect. 7).

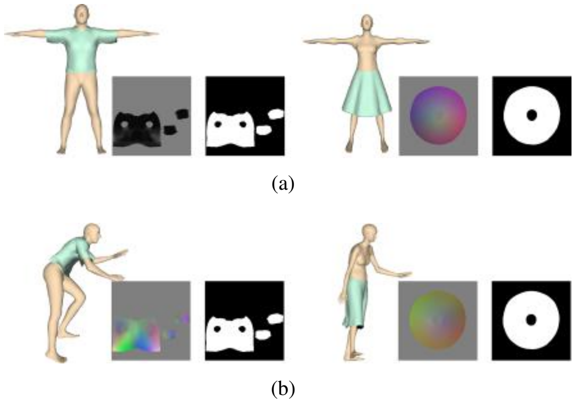


Figure 2. The demonstration on coupling UV-map representation of the garment. (a) 2D representation for T-posed garment for Sect. 4. (b) 2D representation for randomly posed garment for Sect. 6. Left: T-shirt mesh and its representation (homotopy to human body), right: skirt mesh and its representation (not homotopy to human body).

4. T-posed Garment Representation

The first step of our DeepCloth is to represent garments with different shapes and topologies in a unified framework. Therefore, different garment shape styles of a garment type, i.e., front-opening/front-closing T-shirts with long/short sleeves, can be mapped into a same feature space. Notice that this section will only deal with T-posed garment model for garment shape encoding and transition, while garments under arbitrary poses will be handled in Sect. 6.

For dealing with different garment types, we divide different garment types into two categories, and introduce our topology specific 2D garment geometry representation. The main idea is described as follows:

Case 1: for clothing that is homotopy to human surface (e.g. T-shirts, pants), as it can be seen as a geometry structure covering the human surface, we map such clothing to a standard human model UV map [2], in order to establish the relationship between the garment vertexes and the nearby vertexes on the human model, and better represent the geometry features.

Case 2: for clothing that is not homotopy to human surface (e.g. dresses), we map them to an independent UV coordinate, to better reflect the characteristic of the garment geometry. Similar to Case 1, the boundary of the UV map demonstrates the edges and basic shape information, and the rendered values indicates the 3D positions of the garment vertexes.

Specifically, when dealing with clothing that is homotopy to human surface, our goal is to find the correspondences between the garment vertexes and the human model UV coordinates. Here we use the same UV map that was used in [2]. For each T-posed garment vertex $\vec{g}_i^T$, rays are emitted from every point on SMPL model surface with its normal direction. We find the ray nearest to $\vec{g}_i^T$, and record the barycentric coordinate of the corresponding point on its SMPL triangle face. We denote such SMPL point as the corresponding sub-vertex $\vec{v}_i^T$ of $\vec{g}_i^T$. In this way, with the pre-defined UV coordinates of each SMPL triangle face by [2], we can accordingly find the UV coordinate t_i of the sub-vertex $\vec{v}_i^T$, which serves as the corresponding UV coordinate for $\vec{g}_i^T$. After calculating the length from $\vec{v}_i^T$ to $\vec{g}_i^T$, we set the length value as the rendered value, which indicates the normal distance from $\vec{v}_i^T$. The rendered T-posed UV-map is shown in Fig. 2(a) left.

When dealing with clothing that is not homotopy to human surface (e.g. dresses), we set an independent UV coordinate accordingly. For T-posed dresses and skirts, as their geometry circles around the lower body, we leverage the cylindrical coordinate and calculate the UV coordinates as follows: for each garment vertex $\vec{g}_i'^T$, we transfer it into cylindrical coordinate: $\vec{g}_i'^T(x, y, z) \rightarrow \vec{g}_i'^T(r, y, \theta)$ where $r = \sqrt{x^2 + z^2}$ and $\theta = \arctan(z, x)$. The UV coordinate t_i' for $\vec{g}_i'^T(r, y, \theta)$ is $t_i' = ((y_0 - y)\cos(\theta) + 0.5, (y_0 - y)\sin(\theta) + 0.5)$, and the rendered value is just $(x + 0.5, y + 0.5, z + 0.5)$ to indicate the vertex positions. Here y_0 serves as the height threshold of the skirts, in practice we set $y_0 = 0.2$ above the root joint of the human model. The rendered T-posed UV-map is shown in Fig. 2(a) right.

In this way, each garment type is represented into one UV spaces. For each type of garments (upper garments, pants, dresses), the mask of its UV-map denotes its covering area of the human body, and thus contains the topology information of different shape styles. As both the mask and the values for one UV-map can be continuously transferred to another on the 2D UV-space, without limiting a garment type into a fixed template (like TailorNet [22]), the UV-based representation naturally has the potential for garment shape transition. The next step is to perform UV map encoding, so that garments with different shape styles and topologies can be then applied to a shape transition framework.

5. Learning Garment Shape and Style Space

In order to achieve garment shape and style transition, the UV-position based garment representation should be mapped into a continuous space, so that the garment shapes and topologies can be transitioned in a smooth way. By dimensional reduction and feature extraction, we map the garment representation UV-map to a low-dimensional feature vector, where both the UV-map and its mask information is encoded into a continuous feature space. Therefore, by editing, interpolation and decoding from the feature space, we can achieve our goal for continuously garment shape transition between different garment topologies and shape styles.

We introduce ParamNet, a CNN-based network for garment shape and style space learning. The main idea is to leverage a CNN encoder-decoder structure to encode the given T-posed garment UV representation generated in Sect. 4. Our UV representation actually contains two piece of information: (1) the mask, i.e., the area where the UV map has rendered values illustrates the area where the garment covers the human body (with T-shirts and pants), or the height range of the T-posed dresses; (2) the rendered value of the UV map illustrates the vertex positions of the garment. Therefore, when performing the encoding, we also make the decoder generate two maps, one for the mask, and the other for the vertex positions.

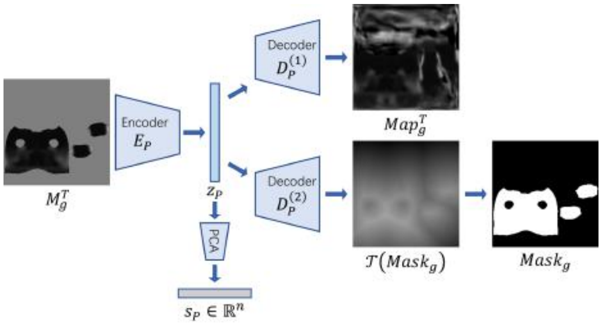


Figure 3. The basic structure of our proposed ParamNet.

As shown in Fig. 3, the basic structure of ParamNet contains two parts, the encoder $E_P(M_g^T) = z_P \in \mathbb{R}^N$ encodes the T-posed garment 2D representation M_g^T to a high-dimension hidden space, and the decoder $D_P^{1,2}(z_P) = Map_g^T, Mask_g$ decodes the vector z_P in high-dimension hidden space to the corresponding map and mask.

In practice, we found that the binary masks could hardly perform smooth transition. Besides, the mask decoder tends to miss the small parts of the masks. That is because the discrete binary mask does not have natural continuity in transition. Therefore, we propose the distance-transform based method for pre-processing the masks, in order to make the transformed mask a continuous map. Specifically, with a given $Mask_g$, we first generate its "bi-distance transform" map as follows:

$$\mathcal{T}(Mask_g) = \mathcal{DT}(Mask_g) - \mathcal{DT}(\mathcal{I} - Mask_g) \quad (1)$$

Here $\mathcal{I}$ is the map with the value 1, and $\mathcal{DT}$ refers to the standard distance transform operation on the mask map. The transformed map, as shown in Fig. 3, demonstrates how far a pixel is away from the map boundary, and the continuation of the transformed mask map makes it easy to be parameterized and learned from the decoder. Therefore, the decoder becomes $D_P^{1,2}(z_P) = Map_g^T, \mathcal{T}(Mask_g)$, and the loss functions are as follows:

$$\begin{aligned}\mathcal{L}_P^{(map)} &= ||M_g^T - Map_g^T * Mask_g^{(gt)}||_1 \\ \mathcal{L}_P^{(mask)} &= ||\mathcal{T}(Mask_g) - \mathcal{T}(Mask_g^{(gt)})||_1\end{aligned} \tag{2}$$

After the training phase of ParamNet, the vectors in high-dimensional hidden space $z_P = E_P(M_g^T)$ encrypts the shape variations and characteristics of the T-posed garment shape. In order to extract the features from the high-dimensional hidden space, we compute the PCA subspace from the hidden space, and sample shape parameters $s_P = E_P(M_g^T) \in \mathbb{R}^n$ from the PCA subspace. In order to recover the T-posed garment shape from shape parameters, we reversely obtain the vector $z_P = PCA^{-1}(s_P)$ and perform the decoders to obtain the 2D representation M_g, and finally generate the garment mesh from it.

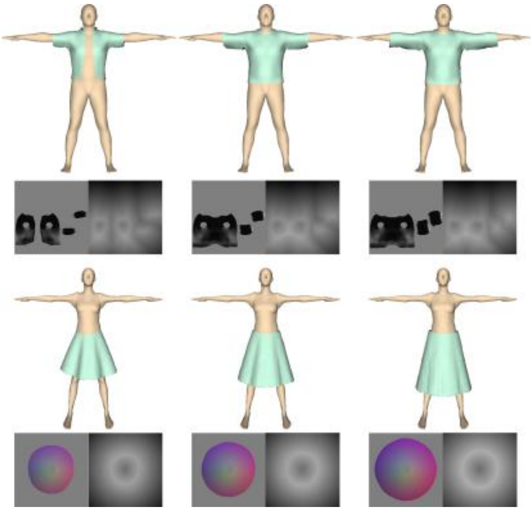


Figure 4. The demonstration of shape variation in our garment encoding pipeline. Each figure shows the T-posed mesh, corresponding 2D representation M_g^T and transformed mask $\mathcal{T}(Mask_g)$. From left to right: same type of garment with feature variation on one PCA dimension.

The demonstration of our T-posed garment shape transition is shown in Fig. 4, which shows that we can perform smooth shape transition from short-sleeve T-shirts to long-sleeve shirts, or from skirts to long dresses, by interpolating the shape parameters in the feature space. Benefiting from our UV-based representation with mask transformation, we could guarantee the continuity in transition process. The results also demonstrate the function of a general and flexible garment shape encoding and control framework.

6. Garment Animation under New Poses

As proposed in Sect. 4 and Sect. 5, we represent the garment shape with the UV-map, and encode it into a feature space. Apart from being applied to garment shape style transition, the representation and encoding module can also be used for dynamic garment animation to synthesis the clothed human into arbitrary new poses. The garment animation module in our DeepCloth takes the input garment shape parameters from Sect. 5 with human pose and shape, and generate the animated garment mesh. In order to achieve this goal, we introduce AnimNet, which is a CNN-based network for the garment animation module.

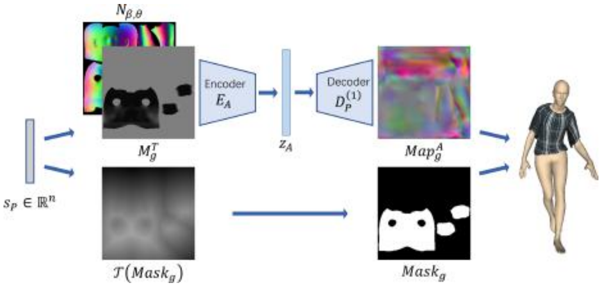


Figure 5. The basic structure of our AnimNet.

The first step is to represent garments under arbitrary poses. As shown in Fig. 2, we establish a topology-consistent coupling UV-map for T-posed garment and the same garment under arbitrary poses. As the previous steps determines the UV coordinates of each garment vertex, for garment animated on the human with other poses, we fix the UV coordinates and set the rendered value representing the animated shape. Specifically, for clothing that is homotopy to human surface, we calculate the position shift between the garment vertex $\vec{g}_i^T$ and corresponding SMPL sub-vertex $\vec{v}_i^T$, and set the position shift $(dx + 0.5, dy + 0.5, dz + 0.5)$ as the rendered value. For clothing that is not homotopy to human surface, we directly use the vertex positions to set the rendered value, as used in T-posed scenario.

There are three main advantages for our coupling UV-map representation. First, by fixing the UV coordinates, we can represent randomly posed garments more easily, e.g., floating open-fronting T-shirts or folding skirts. Second, with the same UV coordinate for every garment vertexes, the mapping between T-posed garments and its randomly posed condition can be learned more easily using a CNN-based network. Third, since the coupling UV-map has the same mask, during animation, we only need to infer the rendered value of the second map.

We denote M_g^T for T-posed standard garment UV map, and M_g^A for animated garment UV map. In AnimNet, we take M_g^T as input and generate M_g^A as output. Meanwhile, in order to encode the human pose and shape information, we find that the normal information actually guides the position map of the garment, therefore, we use the normal map $N_{\beta,\theta}$ of the human model to represent the human shape β and pose θ information. As shown in Fig. 5, we use a CNN-based encoder E_A and decoder D_A to generate the inferred

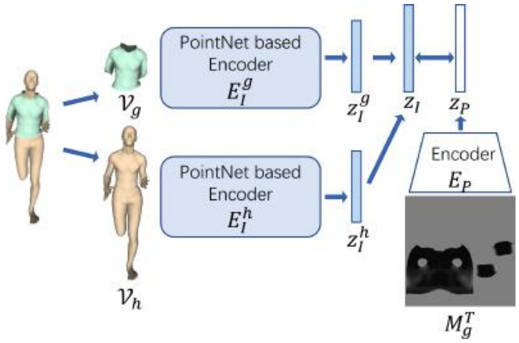


Figure 6. The basic structure of our 3DInferNet.

garment map Map_g^A. The main loss function is as follows:

$$\mathcal{L}_A^{(map)} = ||M_g^{A(gt)} - Map_g^A * Mask_g^{(gt)}||_1 \tag{3}$$

Here $Mask_g^{(gt)}$ is the ground truth mask, and we only need to constraint the generated position map to have the same value with the ground truth one inside the masked area, as the input garment shape parameters contain the mask information. The example results are shown in Fig. 7, which shows that we can animate different types of garments with various human shape, pose and garment styles.

7. Shape Inference and Shape-Style Editing

In order to reconstruct the garment shape from any give 3D raw data, and perform static-to-dynamic garment 3D animation and 3D editing, we introduce the garment shape inference module, which takes a given garment mesh under randomly posed human as input, and extracts the corresponding garment shape parameters. For previous works, SIZER [33] can only perform static garment editing. TailorNet [22] has the potential for shape extraction by un-posing the scan to a standard pose and fit to its fixed garment template, but it cannot perform large garment shape and topology change, nor can it deal with garments not fit to its templates. While our method encodes the garment shape with different styles and topologies to a feature space, enabling garment shape extraction from the encoded space, and 3D editing by shifting the parameters and performing the garment animation module. In this way, we can obtain and edit the shape from a static input garment mesh, and wear it on a moving human body, and accomplish our goal for 3D garment inference, editing and animation.

We introduce 3DInferNet to finish this task. In our garment inference module, in order to balance the different input branches (one branch for garment mesh and one for human information), we replace the input human pose and shape information with the corresponding human mesh. For a given garment mesh $\mathcal{V}_g$ and the corresponding human model mesh $\mathcal{V}_h$, our goal is to generate the corresponding garment parameters s_g. The basic structure of 3DInferNet contains a two-branch PointNet-based encoders $E_I^h(\mathcal{V}_h) = z_I^h$ and $E_I^g(\mathcal{V}_g) = z_I^g$, separately encode the input randomly posed human mesh and garment mesh to hidden space vectors. And we perform a fully-connected operator $\mathcal{F}$ for extracting features from z_I^h and z_I^g as $\mathcal{F}(z_I^h, z_I^g) = z_I$. The loss is then introduced to constraint the output feature z_I to have less deviation with the vector z_g encoding the shape parameters (see Sect. 5):

$$\mathcal{L}_I = ||z_I - z_g||_1 \tag{4}$$

After the extraction of the shape parameters by 3DInferNet, we can perform shape editing by the following steps: (1) mapping the shape parameters to the same PCA subspace calculated in Sect. 5 as $s_I = PCA^g(z_I)$, (2) edit some dimensions of shape parameters s_I as s_I', (3) obtain the vector $z_I' = PCA^{-1}(s_I')$, (4) feed z_I' into garment animation network AnimNet in Sect. 6 to generate results with edited garment shape. The results are shown in Fig. 9.

As the PointNet [27] based encoder structure does not rely on the topology or vertex numbers of the input garments, we can extract the garment shape parameters from garment meshes with any 3D input, and generate the animation results, similar to the method mentioned above, with another human pose sequence. The results are shown in Fig. 9.

8. Experiments

In our experiments, we use CLOTH3D [5] as our training and testing data, which is a large scale synthetic dataset with various garment shape styles, suitable for our pipeline. We test our garment encoding module with four garment kinds: upper garments, pants, skirts, and jumpsuits. The animation module is tested with all these kinds of garments. Besides, we take BUFF Dataset [38] as input to test our garment inference module for 3D garment animation and editing from input 3D data.

Besides the data preparation, with the NVIDIA GeForce GTX TITAN X GPU, the training procedure for ParamNet takes around 50 hours, while AnimNet and 3DInferNet takes 100 hours separately for each kinds of garments. Garment rendering results generated from the network output, together with a standard collision resolving procedure, takes around 2 seconds per human per frame with one garment.

Garment shape representation and animation. To evaluate our garment shape representation results, Fig. 7 demonstrates the garment shape variations controlled by different parameters, with the PCA parameter s_P varies within range of 1.0 σ. The results in Fig. 7 shows that we can perform plausible garment shape variation from long-sleeve shirts to short-sleeve T-shirts, from front-closing T-shirts to front-opening T-shirts, from long pants to shorts, and from long dresses to short skirts. Which clarifies our method for a more general garment shape representation

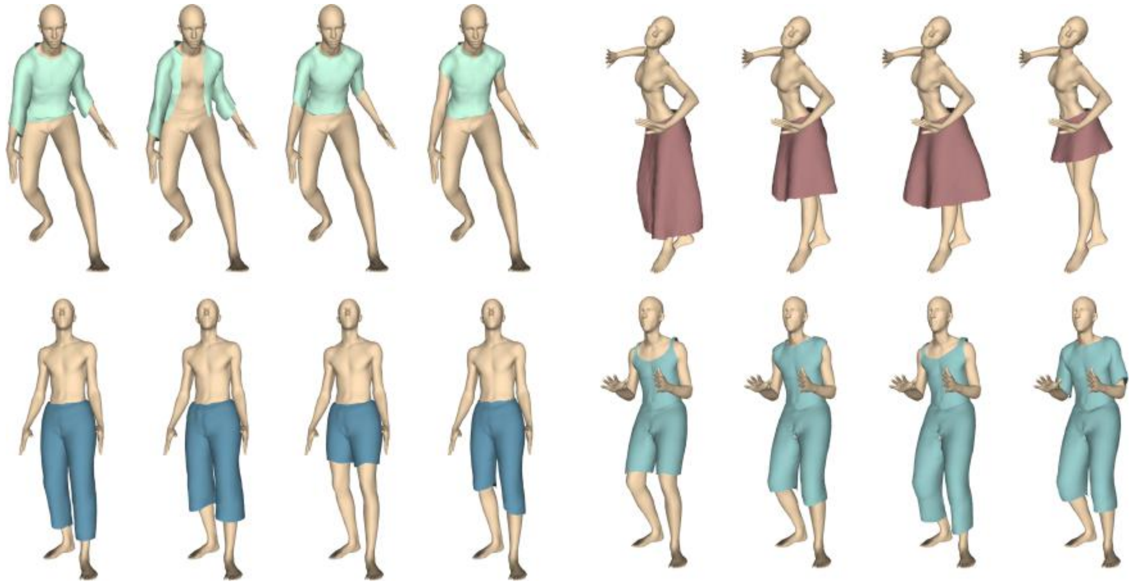


Figure 7. The demonstration of our shape variation for different kinds of garments. Each block shows the garment shape parameter variation on one type of garments.

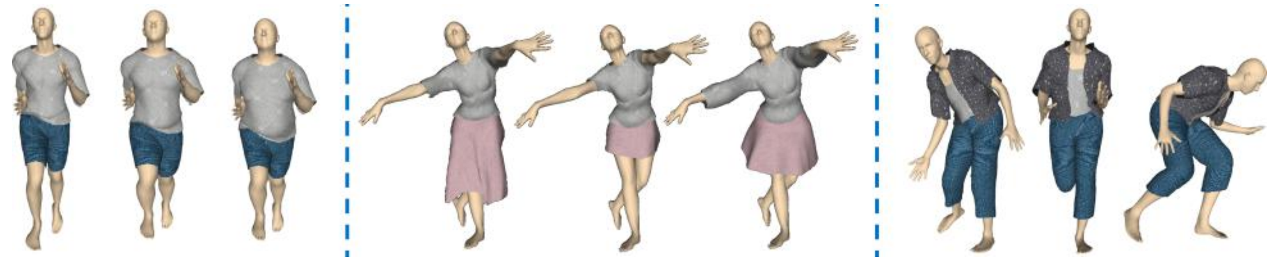


Figure 8. The demonstration of garment animation results. Left: same garments under different human shapes. Middle: different garment styles on the same person. Right: same garments under different human poses.

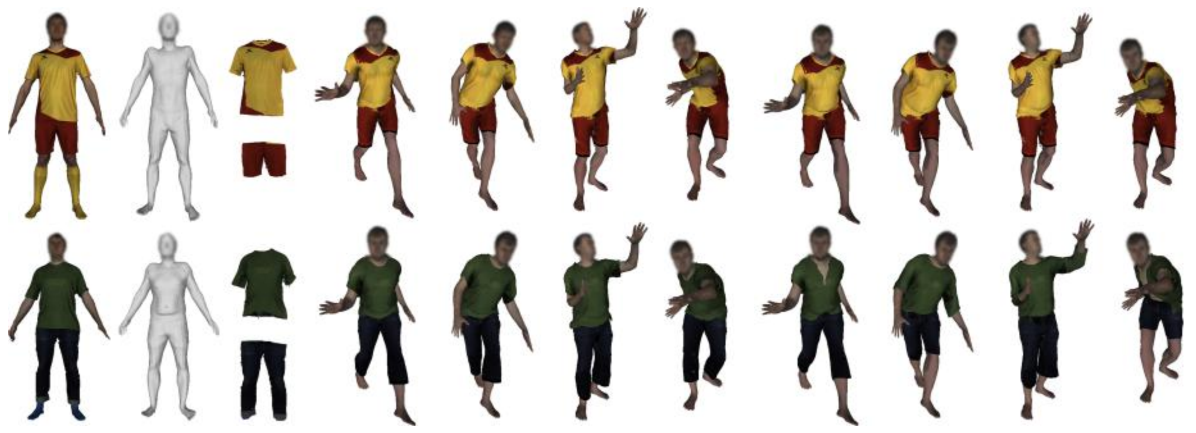
model than TailorNet [22]. Also, as demonstrated in Fig. 8, given different garment styles and different body shapes, we can generate clothed human animation results, which makes our framework capable of representing garment shape under various human shape, poses and garment styles.

Garment shape inference and editing. In order to evaluate our garment shape inference and editing module, we use the 3D scan of BUFF dataset [38] to perform garment shape inference and editing. BUFF Dataset [38] is a high-resolution 4D scan dataset with both clothed human with pose sequences, and T-posed standard human shape without clothing.

In order to fit in our module, for the 3D scan clothed model, we first perform a pose alignment procedure with the T-posed standard human model to obtain the pose information and the inside posed human mesh $\mathcal{V}_h$, and segment the 3D scan to each garment mesh $\mathcal{V}_g$. We then perform 3DInferNet introduced in Sect. 7 to extract the shape parameters s_I for the garment, and generate posed sequences using AnimNet using garment 2D representation generated by s_I. The results are shown in Fig. 9, which shows that we can correctly recover the shape of the original garments. Besides, we can perform shape editing by shifting the shape parameters s_I as s'_I, and also perform the animation procedure. The results are shown in Fig. 9.

Qualitative and quantitative evaluation. For garment shape representation, we make a qualitative comparison with TailorNet [22]. Benefiting from our UV-position with mask representation, we can represent different garment shape styles and topologies in a same framework, while TailorNet [22] needs separate templates for each kind of garment. Please refer to our supplemental materials for more details. Meanwhile, our model can perform shape transition and reasonable interpolation between these shapes, as shown in Fig. 7 and our video demo.

For the garment animation module, we compare our UV-based garment animation method with the PointNet-base [27] method, which extract the point features of the the garment mesh and the posed human mesh, to infer the shift of garment vertexes. As CLOTH3D [5] dataset contains various garment styles, e.g., front-opening and front-closing T-shirts with long or short sleeves, the garment styles cannot be fit into a fix garment template. Therefore, traditional MLP or other methods suitable for dealing with meshes



(a) (b)

Figure 9. The experiment results of our garment inference module. (a) Garment inference with given input. from left to right: original 3D scan, aligned human model, 3D segmented T-shirts with pants, animation results. (b) Top: garment retargeting results, bottom: garment 3D editing results.

garment type	Ours	PointNet-based method
T-shirts & shirts	16.34	20.45
pants & shorts	13.51	18.63
long dresses & skirts	31.32	40.98

Table 1. Mean vertex-to-vertex error (mm) of our AnimNet method and PointNet-based method for different garment types.

with fixed number of vertexes could not be evaluated. The CLOTH3D [5] is split into 95 percent for training and 5 percent for testing. The results applied on test set is as follows, here the loss is the mean vertex-to-mesh error.

As shown in Table 1, our method outperforms the PointNet-based method. This is because the garment styles and topologies varies in a wide range in the CLOTH3D [5] dataset, while traditional PointNet-based [27] methods have some limits especially in long dress case. Notice that as our goal is to establish a general garment representation enabling garment shape and style transition, the PointNet-based method actually does not meet our demand, while our UV-based representation can handle these problems, as demonstrated in Fig. 7.

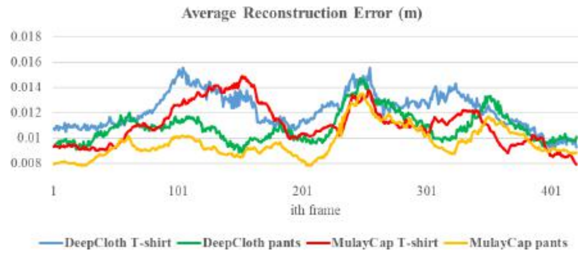


Figure 10. Quantitative comparison with MulayCap [31] using rendered 4D model and aligned SMPL poses and shape in BUFF [38] Dataset as input. The result shows quantitative comparison between two methods in one 4D sequential using per-vertex average error.

For garment shape inference and application, we compare our method with the state-of-the-art garment reconstruction method MulayCap [31], which takes a single-view RGB video as input and dynamically generates a two-layer human with garment mesh. We use a 4D sequence in BUFF [38] dataset as the input, as demonstrated in Fig. 9. We provide [31] with the aligned SMPL shape and poses for every frame, and compare the vertex-to-mesh error between the generated garments and the ground truth input. For our method, we use only the ground truth garment mesh of the first frame for garment shape inference, similar to Fig. 9, and provide garment animation with SMPL poses and shape. As demonstrated in Fig. 10, our method performs similar results with MulayCap [31]. However, note that MulayCap [31] takes all the RGB frames as input, while ours only use one garment mesh for inference and animate the inferred garment based on the SMPL body motion.

9. Conclusion

In this paper, we propose DeepCloth, a unified neural garment representation framework, which can perform garment shape and style transition by learning the shape space of 3D garments. Our method enables modeling garments under different topologies using “UV-position map with mask” representation, and can perform smooth and free garment transition by mapping such representation into a continuous feature space. By introducing AnimNet and 3DInferNet, our representation enables generating of 4D clothed human dynamic sequences, or recovering garment shape from 3D scans and performing animation and garment shape editing. We believe our DeepCloth will inspire more researches in the area of garment representation and modeling.

References

[1] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single rgb camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[2] Thiemo Alldieck, Marcus Magnor, Weipeng Xu, Christian Theobalt, and Gerard Pons-Moll. Video based reconstruction of 3d people models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3, 4

[3] Thiemo Alldieck, Gerard Pons-Moll, Christian Theobalt, and Marcus Magnor. Tex2shape: Detailed full human body geometry from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 3

[4] David Baraff and Andrew Witkin. Large steps in cloth simulation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 43–54, 1998. 2

[5] Hugo Bertiche, Meysam Madadi, and Sergio Escalera. Cloth3d: Clothed 3d humans. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020. 2, 6, 7, 8

[6] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[7] J. Bonet and R. D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, Cambridge, 1997. 2

[8] Derek Bradley, Tiberiu Popa, Alla Sheffer, Wolfgang Heidrich, and Tamy Boubekeur. Markerless garment capture. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 27(3):99, 2008. 2

[9] Xiaowu Chen, Bin Zhou, Feixiang Lu, Lin Wang, Lang Bi, and Ping Tan. Garment modeling with a depth camera. *ACM Trans. Graph.*, 34(6):203:1–203:12, Oct. 2015. 2

[10] Kwang-Jin Choi and Hyeong-Seok Ko. Stable but responsive cloth. In *ACM SIGGRAPH 2005 Courses*, page 1, 2005. 2

[11] Edilson de Aguiar, Leonid Sigal, Adrien Treuille, and Jessica K. Hodgins. Stable spaces for real-time clothing. *ACM Trans. Graph.*, 29(4), July 2010. 2

[12] E. Gundogdu, V. Constantin, S. Parashar, A. Seifoddini Banadkooki, M. Dang, M. Salzmann, and P. Fua. Garnet++: Improving fast and accurate static 3d cloth draping by curvature loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020. 1, 2

[13] Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. Garnet: A two-stream network for fast and accurate 3d cloth draping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[14] Chenfanfu Jiang, Theodore Gast, and Joseph Teran. Anisotropic elastoplasticity for cloth, knit and hair frictional contact. *ACM Trans. Graph.*, 36(4):152:1–152:14, July 2017. 2

[15] Ning Jin, Yilin Zhu, Zhenglin Geng, and Ron Fedkiw. A pixel-based framework for data-driven clothing. In *Symposium on Computer Animation (SCA)*, October 2020. 1, 2

[16] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[17] Doyub Kim, Woojong Koh, Rahul Narain, Kayvon Fatahalian, Adrien Treuille, and James F. O'Brien. Near-exhaustive precomputation of secondary cloth effects. *ACM Trans. Graph.*, 32(4), July 2013. 2

[18] Zorah Lahner, Daniel Cremers, and Tony Tung. Deepwrinkles: Accurate and realistic clothing modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 1, 2

[19] Tiantian Liu, Adam W Bargteil, James F O'Brien, and Ladislav Kavan. Fast simulation of mass-spring systems. *ACM Transactions on Graphics (TOG)*, 32(6):214, 2013. 2

[20] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 1

[21] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to dress 3d people in generative clothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[22] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 3, 4, 6, 7

[23] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael J. Black. Clothcap: Seamless 4d clothing capture and retargeting. *ACM Trans. Graph.*, 36(4):73:1–73:15, July 2017. 2

[24] Tiberiu Popa, Qingnan Zhou, Derek Bradley, Vladislav Kraevoy, Hongbo Fu, Alla Sheffer, and Wolfgang Heidrich. Wrinkling captured garments using space-time data-driven deformation. *Computer Graphics Forum (Proc. Eurographics)*, 28(2):427–435, 2009. 2

[25] William H Press. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007. 2

[26] Xavier Provot et al. Deformation constraints in a mass-spring model to describe rigid cloth behaviour. In *Graphics interface*, pages 147–147. Canadian Information Processing Society, 1995. 1, 2

[27] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 6, 7, 8

[28] Igor Santesteban, Miguel A. Otaduy, and Dan Casas. Learning-based animation of clothing for virtual try-on. *Computer Graphics Forum*, 38(2):355–366, 2019. 2

[29] Yu Shen, Junbang Liang, and Ming C. Lin. Gan-based garment generation using sewing pattern images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, August 2020. 1, 2, 3

[30] Carsten Stoll, Juergen Gall, Edilson de Aguiar, Sebastian Thrun, and Christian Theobalt. Video-based reconstruction of animatable human characters. *ACM Trans. Graph.*, 29(6):139:1–139:10, 2010. 2

[31] Z. Su, W. Wan, T. Yu, L. Liu, L. Fang, W. Wang, and Y. Liu. Mulaycap: Multi-layer human performance capture using a monocular video camera. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–1, 2020. 2, 8

[32] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. Elastically deformable models. In *ACM Siggraph Computer Graphics*, volume 21, pages 205–214, 1987. 2

[33] Garvita Tiwari, Bharat Lal Bhatnagar, Tony Tung, and Gerard Pons-Moll. Sizer: A dataset and model for parsing 3d clothing and learning size sensitive 3d clothing. In *European Conference on Computer Vision (ECCV)*. Springer, August 2020. 1, 3, 6

[34] Huamin Wang, Florian Hecht, Ravi Ramamoorthi, and James F. O'Brien. Example-based wrinkle synthesis for clothing animation. *ACM Trans. Graph.*, 29(4), July 2010. 2

[35] Tuanfeng Y. Wang, Duygu Ceylan, Jovan Popovic, and Niloy J. Mitra. Learning a shared shape space for multimodal garment design. *ACM Trans. Graph.*, 37(6):1:1–1:14, 2018. 2

[36] Jinlong Yang, Jean-Sebastien Franco, Franck Hetroy-Wheeler, and Stefanie Wuhrer. Analyzing clothing layer deformation statistics of 3d human motions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2

[37] Tao Yu, Zerong Zheng, Yuan Zhong, Jianhui Zhao, Qionghai Dai, Gerard Pons-Moll, and Yebin Liu. Simulcap : Single-view human performance capture with cloth simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 2

[38] Chao Zhang, Sergi Pujades, Michael J. Black, and Gerard Pons-Moll. Detailed, accurate, human shape estimation from clothed 3d scan sequences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 6, 7, 8