Learning a Shared Deformation Space for Efficient Design-Preserving Garment Transfer

Min Shi[a,*], Yukun Wei[a], Lan Chen[b,c], Dengming Zhu[d], Tianlu Mao[d], Zhaoqi Wang[d]

[a]*School of Control and Computer Engineering, North China Electric Power University, Beijing, China*
[b]*Institute of Automation, Chinese Academy of Sciences, Beijing, China*
[c]*School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China*
[d]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China*

Abstract

Garment transfer from a source mannequin to a shape-varying individual is a vital technique in computer graphics. Existing garment transfer methods are either time consuming or lack designed details especially for clothing with complex styles. In this paper, we propose a data-driven approach to efficiently transfer garments between two distinctive bodies while preserving the source design. Given two sets of simulated garments on a source body and a target body, we utilize the deformation gradients as the representation. Since garments in our dataset are with various topologies, we embed cloth deformation to the body. For garment transfer, the deformation is decomposed into two aspects, typically style and shape. An encoder-decoder network is proposed to learn a shared space which is invariant to garment style but related to the deformation of human bodies. For a new garment in a different style worn by the source human, our method can efficiently transfer it to the target body with the shared shape deformation, meanwhile preserving the designed details. We qualitatively and quantitatively evaluate our method on a diverse set of 3D garments that showcase rich wrinkling patterns. Experiments show that the transferred garments can preserve the source design even if the target body is quite different from the source one.

Keywords: Garment Transfer, Cloth Deformation, Shape Analysis

1. Introduction

A garment usually has different sizes to fit customers of various body shapes. Most ready-to-wear garments are designed with reference to mannequins with standard body shapes. However, for customers with body shapes of different proportions, especially

---

*Corresponding author

Email addresses: `shi_min@ncepu.edu.cn` (Min Shi), `wyk@ncepu.edu.cn` (Yukun Wei), `chenlan2016@ia.ac.cn` (Lan Chen), `mdzhu@ict.ac.cn` (Dengming Zhu), `ltm@ict.ac.cn` (Tianlu Mao), `zqwang@ict.ac.cn` (Zhaoqi Wang)
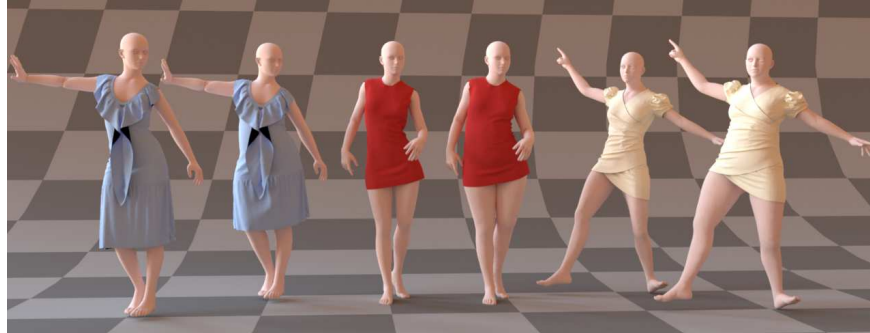
Figure 1: Given a base-size sewing pattern and an instance of its corresponding physics-simulated garment, our efficient solution can transfer the garment from a standard body to mannequins with distinctive body shapes, preserving the original garment design. Each pair of garments shows the transfer result from the source (left) to the target (right) body.

those who have distinctive somatotype characteristics, a base-size garment fails to provide superior fit and cannot preserve the original design. When choosing which garment to buy, the customer's decision largely depends on the design of sample (source) garment, and the customers expect the draped garments onto their bodies keep exactly the same design as the source garment. Generating 3D virtual garments according to the target body shape and then perform virtual try-on not only enables customers to preview the fitting effect before the garments are produced but also assist in the development of garments suitable for customers with distinctive body shapes, which can reduce the design cost but increase customer satisfaction. In addition, virtual fitting technology has attracted the interest of the entertainment industry [1, 2], since it is a significant part of movies, video games, virtual- and augmented-reality applications, etc.

A few techniques [3, 4] have been proposed to automatically perform design-preserving garment transfer from source to target body. However, [3] is sensitive to modeling setup and [4] will struggle in a case when the target body is very different from the source. Furthermore, both of them are computationally expensive, making them not suitable for online applications. Other established workflows[5, 6, 7, 8, 9, 10] mainly focus on how to dress a given garment onto target body with a fixed pose or shape, regardless of whether the source design is preserved. Moreover, existing methods either support garment retargeting for a garment with a fixed style [5, 7, 8] which has limited use, or implement garment remeshing using body topology [9, 10] which is hard to be used for representing garments with complex styles or garments that are different from body topology. What is lacking is a garment transfer workflow that is design-preserving, capable of distinctive body-shapes and new garment type, efficient in time performance, and supports garment meshes with different topologies.

To that end, we address the problem of efficient design-preserving garment transfer for mannequins with distinctive body shapes (see Fig. 1). In this paper, we select banana-shaped body as source body and pear-shaped body as target body to perform garment transfer. We first stitch sewing patterns onto the source and target body using

a physical simulator, and we use deformation gradients to build cloth deformation from the source to target garment. Then, we define garment transfer as a decomposition of style-dependent term and shape-dependent term from cloth deformation (see Sect. 3). To handle garments with different topologies, we propose embedding deformation gradients into the body (see Sect. 4). By learning a shared deformation space using an encoder-decoder network, we separate shape-dependent deformation from embedded cloth deformation (see Sect. 5). Finally, by simply applying the learned shape-related cloth deformation to the source garment, our transfer method can deform the garment from the source to the target body within a short time period, while preserving source design. We qualitatively and quantitatively evaluate our method from different aspects in Sect. 6. In summary, the key contributions of our approach are as follows:

· *Problem Formulation.* We propose factoring cloth deformation into shape-dependent deformation and style-dependent deformation. Then, we define garment transfer as deforming source garment using shape-dependent deformation to generate garment on target body shape. Once the shape-related cloth deformation is learned, we can use it to deform new garments with arbitrary topology.

· *Feature Description.* By learning a shared deformation space, we separate shape-dependent deformation from embedded cloth deformation. Our simple but efficient garment transfer workflow is suitable for characters with distinctive body shapes and garments that showcase very rich wrinkle details.

· *Data Representation.* A network usually takes fixed-size data as input, and unaligned 3D mesh would most probably struggle in this case. Remeshing garments with different topologies is difficult. We propose embedding deformation gradients into the body, providing a dimensional consistent deformation representation, which enables unaligned 3D mesh data for learning tasks (e.g. shape analysis, learning latent representation).

2. Related work

2.1. Physics-based cloth simulation

Although various professional cloth simulation software is available [11, 12], physics-based cloth simulation is still a popular research topic in the field of computer graphics. To simulate the movement of real cloth as well as possible, physics-based simulation (PBS) models different types of clothing deformation. Based on force analysis of particles, the mass-spring model [13, 14] resolved cloth deformation with a low calculation complexity. The yarn-based method [15, 16] was used to simulate woven clothes. Considering that the simulation of high-precision clothes would bring a heavy computational burden, adaptive cloth simulation [17, 18] dynamically adjusts the accuracy of cloth to increase the calculation speed. In addition, many researchers have aimed to replace elastic forces using position-based constraints [19]. To eliminate collisions during cloth simulation, many collision handling strategies [20, 21] have been proposed. As the most traditional method in clothing animation, PBS can obtain realistic and physics-compliant cloth dynamics, but it requires intensive computation [22, 23], making it difficult for it to guarantee real-time performance.

3

2.2. Data-driven cloth deformation

Data-driven cloth deformation methods are designed to reuse cloth deformation statistics. Compared with the physics-based method, a data-driven approach better ensures efficiency. By determining potential collision areas between garments and bodies, Cordier et al. [24] produced cloth deformation effect that is visually satisfactory. The technique DrivenShape [25] exploits the known correspondence between two sets of shape deformations to drive the deformation of the secondary object. Zhou et al. [26] proposed an image-based virtual fitting method that can synthesize the motion and deformation of a garment model by capturing the skeletal structure of the character. As a garment generation method for various body shapes and postures, the DRAPE model [5] can quickly put a garment onto mannequins with specified body shapes and postures with the help of the SCAPE [27] model. Given a specified pose and precomputed garment shape examples as input, Xu et al. [28] presented a sensitivity-based method to construct a pose-dependent rigging solution, which can synthesize real-time cloth deformation. The learning-based garment animation pipeline with deep neural networks [8] enables virtual fitting for characters with different body shapes and poses, producing realistic dynamics and wrinkle details. By learning a motion-invariant encoding network, Wang et al. [29] learned intrinsic properties that are independent of body motion, providing a semi-automatic solution for authoring garment animation. Recently, with the rise of geometric deep learning [30], data-driven cloth deformation technology will usher in a new opportunity. ACAP [31] enables large-scale mesh deformation representation with both accuracy and efficiency. Tan et al. [32] proposed mesh-based autoencoder for localized deformation component analysis. Mesh variational autoencoder [33, 34] provides a new tool for analyzing deforming 3D meshes, which is widely used for tasks such as deformation transfer [35], shape generation [36], etc.

2.3. Garment retargeting

The easiest way to retarget a garment from a source body to a target body is to simply apply PBS (middle in Fig. 2) or perform direct deformation transfer using deformation gradients between source and target body (right in Fig. 2). However, these solutions usually fail to preserve the garment design and wrinkle details, making it look like a person wearing a wrong size garment. For this reason, many researchers [37, 38, 39] have focused on how to automatically adjust garment patterns. Design-preserving garment transfer [3] can transfer garment models onto mannequins whose body shapes and proportions are obviously different, but this approach needs to set parameters according to the garment types and body shapes. Direct garment editing [40] enables users who have no experience in garment design to mix existing garment patterns in an interactive 3D editor, and then, automatically computed 2D sewing patterns that match the desired 3D form are generated. In addition, image-based virtual try-on networks [41, 42, 43] have absorbed the attention of many researchers because they allow garment recovery and transfer from 2D images. Wang [4] regarded garment pattern adjustment as a nonlinear optimization problem and directly minimized the object function that evaluates the fitting quality. Although the system performed sewing pattern adjustment with efficiency and precision, it could not modify garment patterns properly if the difference between the body shapes was significant.
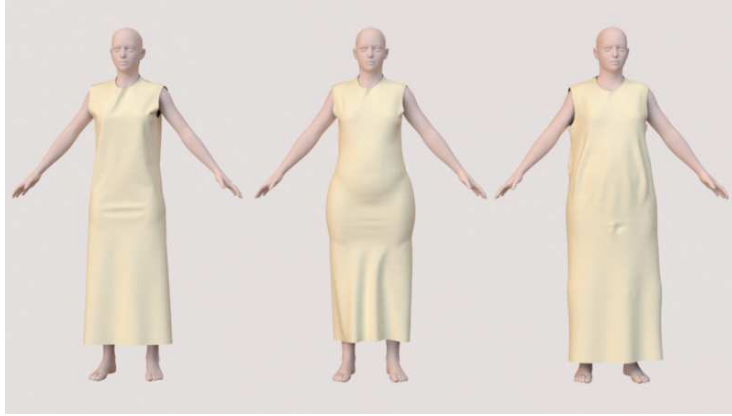
4

Figure 2: Basic fitting strategies. Left: simulated garment on the source body; middle: simulated garment on the target body; right: direct deformation transfer from the source to target body using deformation gradients between source and target body.

3. Overview

3.1. Problem formulation

Considering that the garment style of different sewing patterns varies markedly, when stitched onto the human body, garments will present folds or wrinkles details of various forms. In addition, garments will also present the overall drape commonly caused by variations in body shape. Fig. 3 illustrates the cloth deformation of garment instances from the source to the target body. Regarding the four garments shown in Fig. 3, due to the change in body shape, the abdomen and hip of each garment undergoes significant deformation (encoded in hot colors), while the other parts remain largely in the original shape. Our method starts from an assumption: cloth deformation is composed of two components: high-frequency details (e.g. folds and wrinkles), which varies with different type of garments, is called *style-dependent deformation*; low-frequency characteristics (e.g. the overall drape of the garment), which is shared by different kind of garments, is called *shape-dependent deformation*.

Following the formulation and the notion proposed in DRAPE, we use *deformation gradients* [44, 27, 5] to represent deformations between garment meshes. This allows our model to decouple deformations due to body shape and deformations induced by garment style.

Deformation gradients are linear transformations that align triangular faces between a source garment G_{Src} and its simulated mesh under the target body $\tilde{G}_{Tar}$ sharing the same topology. Suppose that G_{Src} is a mesh with T triangles, $(G_{Src}, \tilde{G}_{Tar})$ can be written as:

$$\begin{cases} G_{Src} = \bigcup_{t=1}^{T}(\vec{x}_{t,1}, \vec{x}_{t,2}, \vec{x}_{t,3}) \\ \tilde{G}_{Tar} = \bigcup_{t=1}^{T}(\vec{y}_{t,1}, \vec{y}_{t,2}, \vec{y}_{t,3}), \end{cases} \tag{1}$$
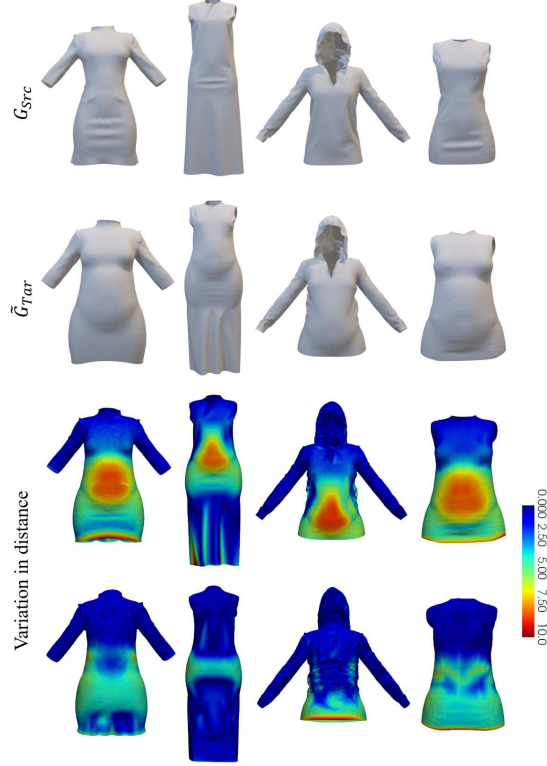
Figure 3: Visualizing cloth deformation from the source to the target body. First row: simulated garment on the source body (noted as G_{Src}). Second row: simulated garment on the target body (noted as $\tilde{G}_{Tar}$). Last two rows: front and back view of the garment examples. Per-vertex deformation are illustrated with hot/cold colors, representing large/small distance variations.

where $(\vec{x}_{t,1}, \vec{x}_{t,2}, \vec{x}_{t,3})$ represents the face of a given triangle t in G_{Src}, $\vec{x}_{t,k}(k = 1, 2, 3)$ are the vertices of triangle t, $(\vec{y}_{t,1}, \vec{y}_{t,2}, \vec{y}_{t,3})$ represents the face of triangle t in $\tilde{G}_{Tar}$ and $\vec{y}_{t,k}(k = 1, 2, 3)$ are the corresponding vertices. Our goal is to solve the following equation:

$$[\Delta \vec{y}_{t,2}, \Delta \vec{y}_{t,3}, \Delta \vec{y}_{t,4}] = Q_t[\Delta \vec{x}_{t,2}, \Delta \vec{x}_{t,3}, \Delta \vec{x}_{t,4}], \tag{2}$$

where Q_t is a 3×3 linear transformation of triangle t and $\Delta \vec{x}_{t,k}(k = 2, 3, 4)$ is:

$$\begin{cases} \Delta \vec{x}_{t,k} = \vec{x}_{t,k} - \vec{x}_{t,1}, k = 2, 3 \\ \Delta \vec{x}_{t,4} = \frac{(\vec{x}_{t,2} - \vec{x}_{t,1}) \times (\vec{x}_{t,3} - \vec{x}_{t,1})}{\sqrt{|\vec{x}_{t,2} - \vec{x}_{t,1}| \times |\vec{x}_{t,3} - \vec{x}_{t,1}|}} \end{cases} . \tag{3}$$

The fact that Q_t is applied to the edge vectors makes it translation-invariant and each linear transformation encodes the change in orientation, scale and skew of triangle

t. The introduced virtual edge [44], $\Delta \vec{x}_{t,4}$ adds the directional information of the triangular faces, making the problem well constrained.

The key idea of our method is to learn a common deformation space that is independent of the garment style but shared by different type of garments. To do so, we define Q_t as combination of linear transformations, each corresponding to different aspects of the model. We factor Q_t into style-dependent deformation and shape-dependent deformation:

$$Q_t = W_t \cdot S_t, \tag{4}$$

W_t is the style-dependent term, which is garment type specific. S_t is the shape-dependent term, which is shared by different garments.

Our goal is to separate shape-dependent deformation from cloth deformation, so W_t is identity and for a given mesh we can use $Q_t = S_t$ to generate a 3D draped garment that fits the target body but retains the original wrinkles. Since the deformation is represented as per-triangle transformations, triangles may separate after applying cloth deformation onto the G_{Src}. We solve for the vertex coordinates that best match the deformed triangles in a least squares sense to assure a consistent mesh:

$$\underset{\vec{y}_1,\ldots,\vec{y}_V}{\arg\min} \sum_{t=1}^{T} \sum_{k=2,3} ||W_t \cdot S_t \cdot \Delta \vec{x}_{t,k} - \Delta \vec{y}_{t,k}||^2. \tag{5}$$

3.2. Technical framework

Fig. 4 demonstrates our technical architecture, and each part is explained as follows:

a) Data Generation: We select banana-shaped body as source body and pear-shaped body as target body to perform garment transfer (see source and target body in Fig. 4). Taking the source body as the base size, we make various garment patterns following the industrial garment-making process and we use a physical simulator to stitch the garment patterns onto the source and target bodies, obtaining draped garment instances for the source and target body (see source and target garment in Fig. 4). Fig. 6 shows our constructed garment data set. Then, we use deformation gradients to represent cloth deformation from G_{Src} to $\tilde{G}_{Tar}$. Since different garments do not share the same topology, we propose embedding cloth deformation into the body to obtain a dimensional consistent deformation representation, which is achieved by matching the shortest distance between garment triangles and body triangles. The embedded cloth deformation is then fed into the garment transfer network for learning. For more details please refer to Sect. 4

b) Garment Transfer Network: We separate the shape-dependent deformation from embedded cloth deformation using an encoder-decoder network. Given the embedded cloth deformation as input, the encoder tries to generate a compressed representation of the cloth deformation. In this process, we expect the network to encode shape-dependent deformation only. Then, the decoder reconstruct cloth deformation from the
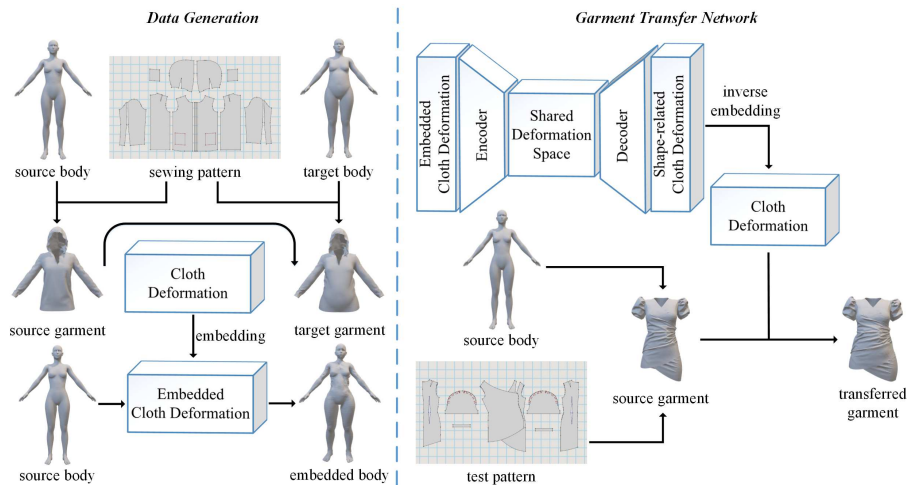
7

Figure 4: Given a source body and its corresponding simulated garment, we present a method to transfer source garment to target body with distinctive body shape, preserving source design. To handle garments with different topologies, we propose embedding cloth deformation into the body. At the heart of our method is a decomposition of style-dependent term and shape-dependent term from embedded cloth deformation using an encoder-decoder network. By learning a shared deformation space, we eliminate style-dependent deformation from embedded cloth deformation so that the reconstructed cloth deformation is shape-related only. At the garment transfer phase, we use the learned shape-related cloth deformation to generate garment that fits the target body but still preserves source design. Once our shape-related deformation is learned, our method can transfer new garments with arbitrary topologies.

shared deformation space to obtain shape-related cloth deformation. For details about how this is achieved please refer to Sect. 5. Once the shape-related cloth deformation is learned, we can use it to deform a new source garment with arbitrary topology. The transferred garment both fits the target body shape and preserves the source design. In Sect. 6, we qualitatively and quantitatively evaluated our method.

4. Deformation gradient embedding

In Sect. 3.1 we use deformation gradients to represent deformation from G_{Src} to $\tilde{G}_{Tar}$. Let the number of garment patterns in the training set be N; then, the training set G_{Sample} can be written as:

$$G_{Sample} = \bigcup_{i=1}^{N} (G_{Src}^{(i)}, \tilde{G}_{Tar}^{(i)}), \qquad (6)$$

where $(G_{Src}^{(i)}, \tilde{G}_{Tar}^{(i)})$ represents the i-th group of garment instances sharing the same topology. Suppose that $(G_{Src}^{(i)}, \tilde{G}_{Tar}^{(i)})$ has T_i triangles, the corresponding cloth deformation is a $3 \times 3 \times T_i$-dimensional matrix, making it not suitable for learning tasks because the network usually takes a fixed-size data as input. One possible solution is to use aligned 3D garment meshes to get a dimensional-consistent data space, but this

8

approach will limit the versatility of model because most of garment meshes are not homogeneous in an actual application scenarios.

To handle garments with different topologies, we propose *embedding deformation gradients into the body*. This is achieved by matching the shortest distance between body triangles and garment triangles. Let $|M|$ be the number of triangles of body, then, for a given garment G, we build two maps of triangle indices from cloth to body and body to cloth respectively:

$$
\begin{aligned}
BodyToCloth &= \bigcup_{\alpha=1}^{|M|} \arg\min_{\alpha,\beta} |\overrightarrow{P_{M,\alpha}P_{G,\beta}}| \\
ClothToBody &= \bigcup_{\beta=1}^{T} \arg\min_{\beta,\alpha} |\overrightarrow{P_{G,\beta}P_{M,\alpha}}|
\end{aligned} \tag{7}
$$

where $BodyToCloth$ records the triangle indices from body to cloth for garment instance group $(G_{Src}, \tilde{G}_{Tar})$, and it is used for building embedded cloth deformation; $ClothToBody$ records the triangle indices from cloth to body for garment instance group $(G_{Src}, \tilde{G}_{Tar})$, and it is used for building the recovered cloth deformation; $P_{M,\alpha}$ is the centroid coordinates of triangle α in M; $P_{G,\beta}$ is the centroid coordinates of triangle β in G. Algorithm 1 introduces the steps of deformation gradient embedding.

Algorithm 1: Deformation Gradient Embedding

Input: source body M, a garment instance group $(G_{Src}, \tilde{G}_{Tar})$.
Output: embedded cloth deformation, noted as B.
Compute cloth deformation matrix $C \in \mathbb{R}^{T \times 3 \times 3}$ from G_{Src} to $\tilde{G}_{Tar}$;
Initialize an $|M| \times 3 \times 3$ dimensional matrix B as the embedded cloth deformation ;
Compute $BodyToCloth$ using M and G_{Src} ;
foreach *body_triangle in M* **do**
    Search $(body\_triangle, cloth\_triangle)$ from $BodyToCloth$;
    Update $B[body\_triangle]$ with $C[cloth\_triangle]$;
end

For each garment type in G_{Sample}, we perform deformation gradient embedding. The embedded cloth deformation B is a $|M| \times 3 \times 3$-dimensional matrix for all triangles, which is then concatenated into a single column vector $\gamma_i \in \mathbb{R}^{|M| \cdot 3 \cdot 3 \times 1}$. Finally, all $\gamma_i(i = 1, ..., n)$ are collected into a matrix $\Gamma = [\gamma_1, ..., \gamma_N]$ as the network input.

5. Shape feature encoding

5.1. Feature representation

In Sect. 3.1 we factor cloth deformation into style-dependent deformation term W_t (considered as high-frequency deformations), and shape-dependent deformation term

---

Algorithm 2: Inverse Embedding

Input: source body M, learned cloth deformation $\tilde{B} \in \mathbb{R}^{|M| \times 3 \times 3}$, an arbitrary source garment with T triangles.

Output: deformed garment mesh, noted as G_{Tar}.

Initialize a $T \times 3 \times 3$ dimensional matrix $\tilde{C}$ as the recovered cloth deformation ;

Compute $ClothToBody$ using M and G_{Src};

foreach $cloth\_triangle$ in T **do**

    Search $(cloth\_triangle, body\_triangle)$ from $ClothToBody$;

    Update $\tilde{C}[cloth\_triangle]$ with $\tilde{B}[body\_triangle]$;

end

Make linear transformations using $\tilde{C}$ and solve for the vertex coordinates with Eq.5.

S_t (considered as low-frequency deformations). The key idea of our model is that high-frequency deformations are garment type-specific but low-frequency deformations are
215 shared by different garments, and we expect the reconstructed cloth deformation to contain low-frequency deformations only. We implicitly eliminate W_t from $Q_t = W_t \cdot S_t$ so that $Q_t = S_t$. This is achieved by learning a shared deformation space using a encoder-decoder network.

The network aims to learn a function set $f_{W,b}(\Gamma) = \tilde{\Gamma} \approx \Gamma$ so that $\tilde{\Gamma}$ constantly
220 approximates Γ, where Γ is the original cloth deformation and $\tilde{\Gamma}$ is the shape-related cloth deformation. The task of the encoder is to learn a compressed representation of the cloth deformation, building a shared feature space of body shapes. Then, the decoder is trained to replicate the original cloth deformation from the latent space. We impose constraints on the size of latent space so that the autoencoder cannot reconstruct
225 all the deformation, more specifically, we expect it to lose high-frequency details. The loss function of our network is a simple mean-squared error (MSE) term:

$$||\Gamma - \tilde{\Gamma}||_F^2, \tag{8}$$

where $\tilde{\Gamma}$ is the learned shape-related cloth deformation.

As mentioned above, by learning $f_{W,b}(\Gamma) = \tilde{\Gamma} \approx \Gamma$, our garment transfer network tries to reconstruct cloth deformation with low-frequency deformations only. The re-
230 constructed cloth deformation, $\tilde{\Gamma}$, is a $|M| \cdot 3 \cdot 3 \times N$-dimensional matrix. At the garment transfer phase, each $\tilde{\gamma}_i(i = 1, ..., N)$ in $\tilde{\Gamma}$ is reshaped back to a $|M| \times 3 \times 3$-dimensional matrix $\tilde{B}_i$, which is then applied onto G_{Src} to generate a new garment G_{Tar} that fits the target body but remains original high-frequency details. This step involves transmitting embedded cloth deformation back to a garment-specific space.
235 Algorithm 2 describes the inverse embedding process.

5.2. Extension to multiple target bodies

Though our workflow supports garment transfer for different body shapes, the learned shared deformation space has to be retrained per body shape. When performing

10

garment transfer, it is often desirable to control the types of body shape to be generated. To handle a situation when there are multiple target bodies, we impose conditional constraints on input data. Since the shape types are discrete by nature, so we represent them using one-hot labels. More specifically, shape conditions are incorporated as additional input with Γ. More details are provided in Sect. 6.4.
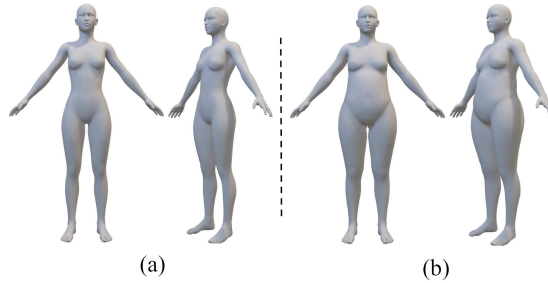

<div align="center">(a)   (b)</div>

Figure 5: We construct (a) banana-shaped body as source and (b) pear-shaped body as target for evaluation. The body is shown from a front view and a side view. Compared with the source body, the local shape (abdomen and hip) of target body has changed significantly.

Figure 6: Composition of garment dataset. Left: garments used for training; right: garments used for testing.

6. Evaluation

We propose embedding deformation gradients into the body to represent per-triangle deformation from G_{Src} to $\tilde{G}_{Tar}$ using M in Sect. 4. Then, by decoupling shape-dependent deformation and style-dependent deformation from embedded cloth deformation, our method learns a shared deformation space that is invariant to garment style in Sect. 5. We now qualitatively and quantitatively evaluate the effectiveness of our method. We performed experiments on a consumer laptop with an Intel Core i7-8750H 2.2 GHz processor, 16 GB of RAM and an NVIDIA GeForce RTX 2070 with Max-Q Design Graphics Card.

6.1. Experimental settings

We first provide the details of how the data are generated and of the network structure:

Data generation. We used DAZ Studio software [45] to generate A-pose mannequins of different body shapes. We built banana-shaped ((a) in Fig. 5) and pear-shaped ((b) in Fig. 5) as source and target body respectively. Our garment data set consists of 15 basic-style garments (left in Fig. 6) for training and 5 complex-style garments (right in Fig. 6) for evaluation. We selected different types of sewing patterns to make the training examples, which aims to cover various kinds of clothing commonly seen in daily life. All garment patterns are designed and simulated using the Marvelous Designer software [11]. The garment meshes in garment data set do not share the same topology (e.g., the number of triangles is 10 628 in a sweater and 28 044 in a shift dress).

Network architecture. We implemented deformation gradient embedding in C++, and the cloth deformation of each garment was aligned to a $|M| \cdot 9$-dimensional vector ($|M| = 37744$ in our experiment). Our shape feature encoding network is composed of linear layers with sigmoid activation function. The encoder takes an $|M| \cdot 9 \times N$-dimensional matrix as input and translates it to a $30 \times N$-dimensional latent space to represent shape feature descriptor. Then, the decoder tries to replicate the original cloth deformation from the shared deformation space. We use scaled conjugate gradient descent for the network back-propagation. It takes about 35 minutes for our network to finish training with the setting of 1000 epochs.
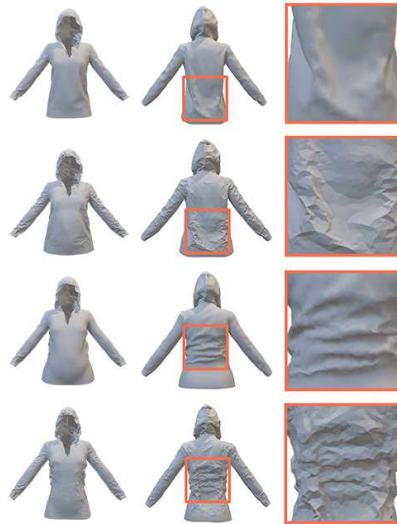

Figure 7: Garment transfer on the training set. Top row: simulated garment on the source body (G_{Src}); second row: our transfer result from the source to target body (G_{Tar}); third row: simulated garment on the target body ($\tilde{G}_{Tar}$); bottom row: transferred garment using decomposed high-frequency deformation term. In contrast to G_{Src}, $\tilde{G}_{Tar}$ revealed the outline of the target body while losing the original garment design. Our transferred results G_{Tar} not only have the somatotype characteristics of the target body but also preserve the source design.

6.2. Qualitative evaluation

Each column in $\tilde{\Gamma}$ represents the reconstructed cloth deformation of a specified garment. By simply applying the reconstructed shape-related cloth deformation to its corresponding garment mesh using Algorithm 2, our workflow can generate garments that fit the target body but preserve source design. Fig. 7 demonstrates the garment transfer result on the training set. Our method successfully decoupled style-dependent deformation and shape-dependent deformation.

One possible application of our garment transfer workflow is virtual fitting. As mentioned above, basic-style garments are used for learning the shape-related cloth deformation. Once the training process for a given body shape is finished, we can apply the learned cloth deformation to a complex-style garment to perform garment transfer. Fig. 8 shows the garment transfer results on the testing set. From top to bottom: Shift Dress, Formal Dress, Wrap Dress, Conjoined Shorts, T-shirt.

(a)   (b)   (c)

Figure 8: Garment transfer on the testing set. (a) simulated garment on the source body (G_{Src}), which showcase very rich wrinkling patterns; (b) our transfer result from the source to target body (G_{Tar}); (c) simulated garment on the target body ($\tilde{G}_{Tar}$). With our garment transfer workflow, all wrinkles on G_{Src} are correctly transferred to the target body shape without noticeable artifacts.

Now, we evaluate our shape feature encoding network. We can deform a specified source garment (see (a) in Fig. 9) onto the target body (see (b) in Fig. 9) using its

corresponding cloth deformation, and we can also deform garments of other designs (see (c)-(d) in Fig. 9). However, since the source design of different garments varies markedly, it is difficult to obtain the desired appearance by directly applying the cloth deformation of one garment onto other garments (see (c) in Fig. 9). Our deformation feature encoding network learns a shared deformation space from cloth deformation, which enables garment transfer among garments of different designs (see (d) in Fig. 9).
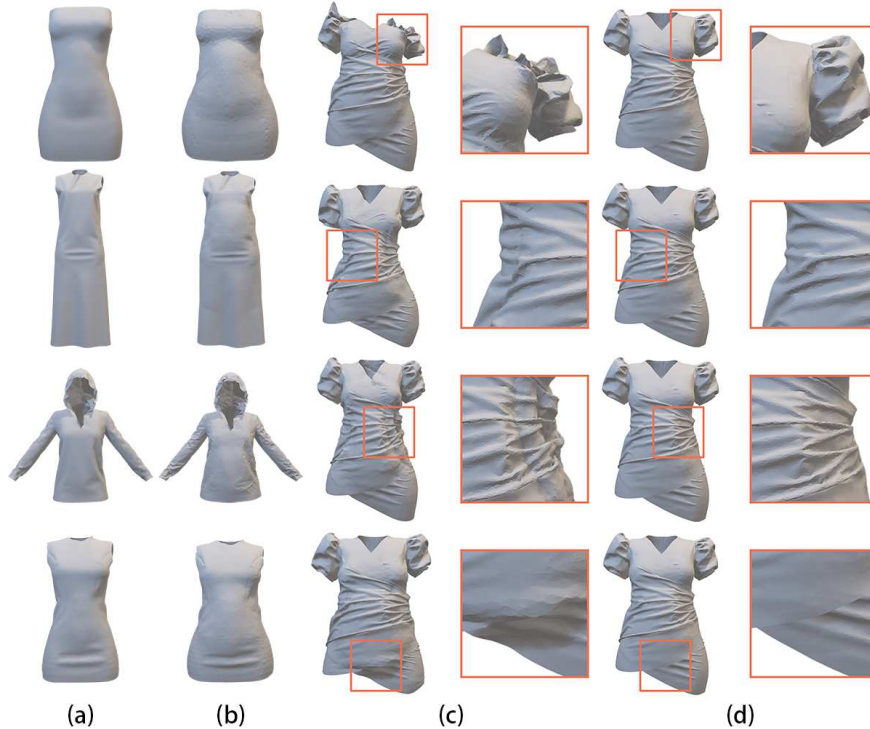

(a)　　　　(b)　　　　(c)　　　　(d)

Figure 9: Validation of the shape feature encoding network. (a) simulated garment on the source body; (b) transferred garment on the target body; (c) garment transfer before shape feature encoding; (d) garment transfer after shape feature encoding. Details in (c) and (d) are enlarged to show the difference.

We also invited a professional fashion design studio to manually transfer a garment from the source to the target body. Taking the wrap dress as reference, the pattern grader took more than ten hours to make 2D graded sewing patterns and restore the wrinkle details of the 3D garment on the target body (right in Fig. 10). We ask the pattern grader to do this because customers usually want the transferred garment to keep exactly the same design as source garment. According to the survey, most of the time was spent on handling the wrinkle elements, because the designer needed to repeatedly adjust the sewing pattern until the draped garment accorded with the desired shape. Comparatively, it took less than ten seconds for our garment transfer workflow to generate the draped garment (middle in Fig. 10). With the help of our garment transfer workflow, customers can quickly preview the fitting effects before real garments are designed and manufactured, thus saving time and design cost.
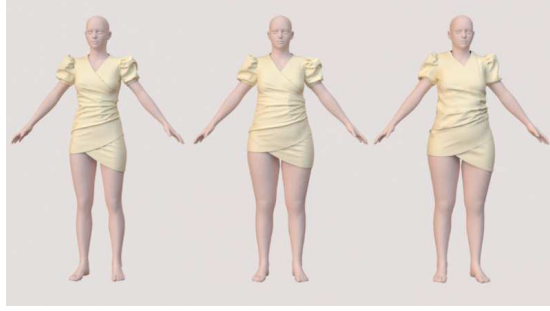
14

Figure 10: Manual vs. automatic. Left: simulated garment on the source body; middle: our transferred result; right: manually graded garment produced by a garment designer.

6.3. Quantitative evaluation

To quantitatively measure the difference in wrinkle details before and after garment transfer, we computed the mean discrete curvature of our transferred garments G_{Tar} and simulated garment $\tilde{G}_{Tar}$ for the target body. The mean discrete curvature is a tool for measuring the variation between the source and target mesh. The lower the mean discrete curvature is, the less the garment details has changed. The mean discrete curvature between G_{Src} and G_{Tar} is :

$$MDC(G_{Src}, G_{Tar}) = \frac{1}{T} \sum_{s=1}^{T} |r_{G_{Src}}(s) - r_{G_{Tar}}(s)|, \tag{9}$$

in which $r_{G_{Src}}(s)$ represents the discrete curvature of the s-th vertex in G_{Src} and $r_{G_{Tar}}(s)$ represents the discrete curvature of the s-th vertex in G_{Tar}. $r_G(s)$ can be expressed as:

$$r_G(s) = \arg\min_{i,j} \vec{n_i} \cdot \vec{n_j}$$
$$i, j \in p(s), 1 \leq i < j \leq |p(s)|, \tag{10}$$

where $p(s)$ is the triangle set adjacent to the s-th vertex. $\vec{n_i}$ and $\vec{n_j}$ represent the unit normal of the i-th and j-th triangles in $p(s)$. $|p(s)|$ is the number of triangles in $p(s)$.

Since our ultimate goal is to make the transferred garment looks similar to the source, we also made user study on visual similarity between source and transferred / simulated garment. 18 subjects were asked to score the degree of visual similarity (0-9) between source and deformed / simulated garment. Taking G_{Src} as reference, 0 represents the wrinkle details on the $\tilde{G}_{Tar}$ (or G_{Tar}) looks completely different from the reference, and 9 represents the wrinkle details on the $\tilde{G}_{Tar}$ (or G_{Tar}) looks exactly the same as the reference.

Tab. 1 summarizes the statistics of the garment examples in the test set. The mean discrete curvature of the simulated garments on the target body is much greater than ours, and our garment transfer can be finished within a short time period even on a consumer laptop, which indicates that our garment transfer workflow can deform garments onto the target body both with efficiency and can preserve the source design. In

15

the visual similarity study, $Score_{Trans}$ for each garment is much higher than that in $Score_{Sim}$, which indicates that G_{Tar} looks more similar to G_{Src}.

Example Name	#Vert	#Tri	$Score_{Trans}\uparrow$	$MDC_{Trans}\downarrow$	$Score_{Sim}\uparrow$	$MDC_{Sim}\downarrow$	$T(s)$
Shift Dress	14 149	28 004	6.0	0.020 3	2.6	0.054 1	0.764
Formal Dress	23 262	46 081	6.3	0.022 3	4.0	0.064 0	1.301
Wrap Dress	78 169	15 605 4	6.8	0.010 8	2.7	0.115 6	5.928
Conjoined Shorts	13 289	26 288	6.2	0.022 4	1.4	0.193 9	0.711
T-shirt	41 970	83 814	6.5	0.011 9	2.3	0.079 7	2.822

Table 1: Statistics of the examples. #Vert: Number of vertices in the example. #Tri: Number of triangles in the example. $Score_{Trans}$:Score of visual similarity between G_{Src} and G_{Tar}. MDC_{Trans}: Mean discrete curvature between G_{Src} and G_{Tar}. $Score_{Sim}$: Score of visual similarity between G_{Src} and $\tilde{G}_{Tar}$. MDC_{Sim}: Mean discrete curvature between G_{Src} and $\tilde{G}_{Tar}$. T: Total time for deforming G_{Src} to G_{Tar}.

Previous works like [3, 4] formulated garment transfer as a constrained optimization problem and solved it through iterative quadratic minimization, which takes hundreds of seconds. Comparatively, our learning-based garment transfer workflow only takes seconds to finish even for a complex-style garment with 156K faces (the Wrap Dress in our experiment). Besides, [3] is sensitive to the setting of tight region tolerance, which varies with input body and garment model. Once our shared deformation is learned, we can use it to generate transferred garments that fit the target body but still preserve source design without additional modeling setup. [4] is developed for human bodies with limited differences from the source body, which will struggle in a case when the target body is very different from the source body. Our workflow enables design-preserving garment transfer even for mannequins with distinctive body shapes.

6.4. Multiple target shapes

Given a sewing pattern and its corresponding simulated garment on a source body, our workflow can perform garment transfer from the source body to mannequins with distinctive body shapes. The experiment results show that our pipeline works well on a single body-shape case. Now we evaluate our method on bodies with more shape variations. The body is represented using SMPL [46]. SMPL is a generative model that factors the body into shape (noted as β) and pose (noted as θ) parameters. We obtained A-pose θ from BMLmovi [47, 48] dataset. Then, we sampled $\beta_1, ..., \beta_{-4}$ from range [-4,1] to generate bodies with 6 different shape variations (noted as $M(\theta, \beta_1), ..., M(\theta, \beta_{-4})$). Shape conditions are encoded as one-hot labels from 000000 to 100000. The generated body meshes share the same topology ($|M|$=13776). We take $M(\theta, \beta_1)$ as source body and others as target to perform garment transfer. We trained network using the method proposed in Sect.5.2. Since the deformed garment reflects only the general outline of the target body, it is difficult to guarantee that the transferred garment is always collision-free. To eliminate the penetration between the garment mesh and body surface, we iteratively update the vertex position until the garment vertices lie completely outside the target body with the Marvelous Designer software [11]. Fig. 11 demonstrates fitting effect from $M(\theta, \beta_1)$ to $M(\theta, \beta_0), ..., M(\theta, \beta_{-4})$.

Figure 11: Retargeting a source garment to mannequins with different body shapes, the blue dress is simulated garment on source body ($M(\theta, \beta_1)$) and the yellow dresses are transferred garments on different target bodies (from left to right: $M(\theta, \beta_0), ..., M(\theta, \beta_{-4})$). Even if the target body is very different compared with the source body, our solution can still preserve the source design and wrinkle details.

7. Limitation and future work

In this paper, we presented an automatic, design-preserving, efficient garment transfer workflow that enables unaligned garment retargeting between characters with significant differences in body shape. This is achieved by learning a shared deformation space from the embedded cloth deformation. Unlike existing methods, our method can generate design-preserving garments that showcase very rich wrinkle details with both accuracy and efficiency. However, our system has several limitations:

(i). We embed cloth deformation into the body by matching triangles with shortest distance, which depends on the mesh quality. If the garment or body has a low-resolution, the deformation of some triangles would be lost. While we use a very high-resolution garment and body mesh, the garment transfer can be done in seconds on a consumer laptop.

(ii). Our goal is to generate design-preserving 3D draped garments for distinctive body shapes before garments are made. We do not consider inverse pattern design problem. How to accurately translate these draped garments into graded sewing patterns needs to be further studied.

8. Acknowledgements

References

[1] J. Liang, M. C. Lin, Machine learning for digital try-on: Challenges and progress, Computational Visual Media (2020) 1–9.

[2] M. Wang, X.-Q. Lyu, Y.-J. Li, F.-L. Zhang, Vr content creation and exploration with deep learning: A survey, Computational Visual Media (2020) 1–26.

[3] R. Brouet, A. Sheffer, L. Boissieux, M.-P. Cani, Design preserving garment transfer, ACM Trans. Graph. 31 (4). `doi:10.1145/2185520.2185532`.
URL `https://doi.org/10.1145/2185520.2185532`

[4] H. Wang, Rule-free sewing pattern adjustment with precision and efficiency, ACM Trans. Graph. 37 (4). `doi:10.1145/3197517.3201320`.
URL `https://doi.org/10.1145/3197517.3201320`

[5] P. Guan, L. Reiss, D. A. Hirshberg, A. Weiss, M. J. Black, Drape: Dressing any person, ACM Trans. Graph. 31 (4). `doi:10.1145/2185520.2185531`.
URL `https://doi.org/10.1145/2185520.2185531`

[6] G. Pons-Moll, S. Pujades, S. Hu, M. J. Black, Clothcap: Seamless 4d clothing capture and retargeting, ACM Trans. Graph. 36 (4). `doi:10.1145/3072959.3073711`.
URL `https://doi.org/10.1145/3072959.3073711`

[7] Z. Lahner, D. Cremers, T. Tung, Deepwrinkles: Accurate and realistic clothing modeling, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[8] I. Santesteban, M. A. Otaduy, D. Casas, Learning-based animation of clothing for virtual try-on, Computer Graphics Forum 38 (2) (2019) 355–366. `doi:10.1111/cgf.13643`.

[9] Q. Ma, J. Yang, A. Ranjan, S. Pujades, G. Pons-Moll, S. Tang, M. J. Black, Learning to dress 3d people in generative clothing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[10] C. Patel, Z. Liao, G. Pons-Moll, Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2020.

[11] Marvelous designer software (2020).
URL `https://www.marvelousdesigner.com/`

[12] Optitex fashion design software (2020).
URL `https://optitex.com/`

[13] K.-J. Choi, H.-S. Ko, Stable but responsive cloth, in: Proceedings of the 29th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '02, Association for Computing Machinery, New York, NY, USA, 2002, p. 604–611. `doi:10.1145/566570.566624`.
URL `https://doi.org/10.1145/566570.566624`

[14] T. Liu, A. W. Bargteil, J. F. O'Brien, L. Kavan, Fast simulation of mass-spring systems, ACM Trans. Graph. 32 (6). `doi:10.1145/2508363.2508406`.
URL `https://doi.org/10.1145/2508363.2508406`

18

[15] J. M. Kaldor, D. L. James, S. Marschner, Simulating knitted cloth at the yarn level, ACM Trans. Graph. 27 (3) (2008) 1–9. doi:10.1145/1360612.1360664.
URL https://doi.org/10.1145/1360612.1360664

[16] G. Cirio, J. Lopez-Moreno, D. Miraut, M. A. Otaduy, Yarn-level simulation of woven cloth, ACM Trans. Graph. 33 (6). doi:10.1145/2661229.2661279.
URL https://doi.org/10.1145/2661229.2661279

[17] R. Narain, A. Samii, J. F. O'Brien, Adaptive anisotropic remeshing for cloth simulation, ACM Trans. Graph. 31 (6). doi:10.1145/2366145.2366171.
URL https://doi.org/10.1145/2366145.2366171

[18] J. Li, G. Daviet, R. Narain, F. Bertails-Descoubes, M. Overby, G. E. Brown, L. Boissieux, An implicit frictional contact solver for adaptive cloth simulation, ACM Trans. Graph. 37 (4). doi:10.1145/3197517.3201308.
URL https://doi.org/10.1145/3197517.3201308

[19] M. Macklin, M. Müller, N. Chentanez, T.-Y. Kim, Particle physics for real-time applications, ACM Trans. Graph. 33 (4). doi:10.1145/2601097.2601152.
URL https://doi.org/10.1145/2601097.2601152

[20] M. Tang, t. wang, Z. Liu, R. Tong, D. Manocha, I-cloth: Incremental collision handling for gpu-based interactive cloth simulation, ACM Trans. Graph. 37 (6). doi:10.1145/3272127.3275005.
URL https://doi.org/10.1145/3272127.3275005

[21] L. Jiang, J. Ye, L. Sun, J. Li, Transferring and fitting fixed-sized garments onto bodies of various dimensions and postures, Computer-Aided Design 106 (2019) 30 – 42. doi:https://doi.org/10.1016/j.cad.2018.08.002.
URL http://www.sciencedirect.com/science/article/pii/S0010448518304676

[22] N. J. Weidner, K. Piddington, D. I. W. Levin, S. Sueda, Eulerian-on-lagrangian cloth simulation, ACM Trans. Graph. 37 (4). doi:10.1145/3197517.3201281.
URL https://doi.org/10.1145/3197517.3201281

[23] Y. R. Fei, C. Batty, E. Grinspun, C. Zheng, A multi-scale model for simulating liquid-fabric interactions, ACM Trans. Graph. 37 (4). doi:10.1145/3197517.3201392.
URL https://doi.org/10.1145/3197517.3201392

[24] F. Cordier, N. Magnenat-Thalmann, A data-driven approach for real-time clothes simulation, in: 12th Pacific Conference on Computer Graphics and Applications, 2004. PG 2004. Proceedings., 2004, pp. 257–266.

[25] T.-Y. Kim, E. Vendrovsky, Drivenshape: A data-driven approach for shape deformation, in: ACM SIGGRAPH 2008 Talks, SIGGRAPH '08, Association for Computing Machinery, New York, NY, USA, 2008. doi:10.1145/1401032.1401121.
URL https://doi.org/10.1145/1401032.1401121

[26] Z. Zhou, B. Shu, S. Zhuo, X. Deng, P. Tan, S. Lin, Image-based clothes animation for virtual fitting, in: SIGGRAPH Asia 2012 Technical Briefs, SA '12, Association for Computing Machinery, New York, NY, USA, 2012. doi:10.1145/2407746.2407779.
URL https://doi.org/10.1145/2407746.2407779

[27] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, J. Davis, Scape: Shape completion and animation of people, ACM Trans. Graph. 24 (3) (2005) 408–416. doi:10.1145/1073204.1073207.
URL https://doi.org/10.1145/1073204.1073207

[28] W. Xu, N. Umentani, Q. Chao, J. Mao, X. Jin, X. Tong, Sensitivity-optimized rigging for example-based real-time clothing synthesis, ACM Trans. Graph. 33 (4). doi:10.1145/2601097.2601136.
URL https://doi.org/10.1145/2601097.2601136

[29] T. Y. Wang, T. Shao, K. Fu, N. J. Mitra, Learning an intrinsic garment space for interactive authoring of garment animation, ACM Trans. Graph. 38 (6). doi:10.1145/3355089.3356512.
URL https://doi.org/10.1145/3355089.3356512

[30] Y. Xiao, Y. Lai, F. Zhang, C. Li, L. Gao, A survey on deep geometry learning: From a representation perspective, Comput. Vis. Media 6 (2) (2020) 113–133. doi:10.1007/s41095-020-0174-8.
URL https://doi.org/10.1007/s41095-020-0174-8

[31] L. Gao, Y.-K. Lai, J. Yang, Z. Ling-Xiao, S. Xia, L. Kobbelt, Sparse data driven mesh deformation, IEEE transactions on visualization and computer graphics.

[32] Q. Tan, L. Gao, Y. Lai, J. Yang, S. Xia, Mesh-based autoencoders for localized deformation component analysis, in: AAAI Conference on Artificial Intelligence, AAAI Press, 2018, pp. 2452–2459.

[33] Q. Tan, L. Gao, Y. Lai, S. Xia, Variational autoencoders for deforming 3d mesh models, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, 2018, pp. 5841–5850.

[34] Y. Yuan, Y. Lai, J. Yang, Q. Duan, H. Fu, L. Gao, Mesh variational autoencoders with edge contraction pooling, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, 2020, pp. 1105–1112.

[35] L. Gao, J. Yang, Y.-L. Qiao, Y.-K. Lai, P. L. Rosin, W. Xu, S. Xia, Automatic unpaired shape deformation transfer, ACM Trans. Graph. 37 (6).

doi:10.1145/3272127.3275028.
URL https://doi.org/10.1145/3272127.3275028

[36] L. Gao, J. Yang, T. Wu, Y.-J. Yuan, H. Fu, Y.-K. Lai, H. Zhang, Sdm-net: Deep generative network for structured deformable mesh, ACM Trans. Graph. 38 (6). doi:10.1145/3355089.3356488.
URL https://doi.org/10.1145/3355089.3356488

[37] C. C. Wang, Y. Wang, M. M. Yuen, Design automation for customized apparel products, Computer-Aided Design 37 (7) (2005) 675 – 691. doi:https://doi.org/10.1016/j.cad.2004.08.007.
URL http://www.sciencedirect.com/science/article/pii/S0010448504001599

[38] N. Umetani, D. M. Kaufman, T. Igarashi, E. Grinspun, Sensitive couture for interactive garment modeling and editing, ACM Trans. Graph. 30 (4). doi:10.1145/2010324.1964985.
URL https://doi.org/10.1145/2010324.1964985

[39] Y. Meng, C. C. Wang, X. Jin, Flexible shape control for automatic resizing of apparel products, Computer-Aided Design 44 (1) (2012) 68 – 76, digital Human Modeling in Product Design. doi:https://doi.org/10.1016/j.cad.2010.11.008.
URL http://www.sciencedirect.com/science/article/pii/S0010448510002186

[40] A. Bartle, A. Sheffer, V. G. Kim, D. M. Kaufman, N. Vining, F. Berthouzoz, Physics-driven pattern adjustment for direct 3d garment editing, ACM Trans. Graph. 35 (4). doi:10.1145/2897824.2925896.
URL https://doi.org/10.1145/2897824.2925896

[41] T. Y. Wang, D. Ceylan, J. Popović, N. J. Mitra, Learning a shared shape space for multimodal garment design, ACM Trans. Graph. 37 (6). doi:10.1145/3272127.3275074.
URL https://doi.org/10.1145/3272127.3275074

[42] X. Han, Z. Wu, Z. Wu, R. Yu, L. S. Davis, Viton: An image-based virtual try-on network, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[43] S. Yang, Z. Pan, T. Amert, K. Wang, L. Yu, T. Berg, M. C. Lin, Physics-inspired garment recovery from a single-view image, ACM Trans. Graph. 37 (5). doi:10.1145/3026479.
URL https://doi.org/10.1145/3026479

[44] R. W. Sumner, J. Popović, Deformation transfer for triangle meshes, in: ACM SIGGRAPH 2004 Papers, SIGGRAPH '04, Association for Computing Machinery, New York, NY, USA, 2004, p. 399–405. doi:10.1145/1186562.1015736.
URL https://doi.org/10.1145/1186562.1015736

21

[45] Daz studio software (2020).
URL https://www.daz3d.com/

[46] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, M. J. Black, SMPL: A skinned multi-person linear model, ACM Trans. Graphics (Proc. SIGGRAPH Asia) 34 (6) (2015) 248:1–248:16.

[47] S. Ghorbani, K. Mahdaviani, A. Thaler, K. Kording, D. J. Cook, G. Blohm, N. F. Troje, Movi: A large multipurpose motion and video dataset (2020). arXiv:2003.01888.

[48] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, M. J. Black, AMASS: Archive of motion capture as surface shapes, in: International Conference on Computer Vision, 2019, pp. 5442–5451.