Zhang CX, Chen XW, Wu HY *et al.* Modeling garment seam from a single image. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 33(3): 463–474 May 2018. DOI 10.1007/s11390-018-1831-6

Modeling Garment Seam from a Single Image

Chen-Xu Zhang, Xiao-Wu Chen*, *Member, CCF, ACM, IEEE*, Hong-Yu Wu
and Bin Zhou, *Member, CCF, ACM, IEEE*

State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering Beihang University, Beijing 100191, China

E-mail: {zhangchenxu528, chen, whyvrlab, zhoubin}@buaa.edu.cn

Received January 4, 2018; revised March 5, 2018.

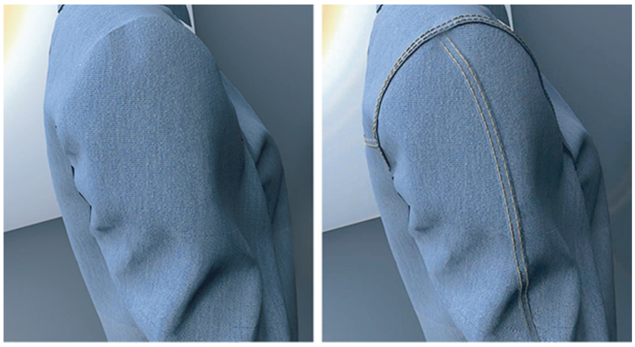
Abstract We propose an automatic garment seam modeling framework to create a garment model with the seam structure from a single image. In order to achieve this, a marked seam image database and parametric seam models have been set up. Given a real seam image, we first identify the type of the seam image based on our marked seam image database and the seam parameters are parsed automatically by our sewing thread estimation method. Second the seam initial model is generated through the pre-defined parametric seam models. A garment model with the seam structure is finally obtained based on the seam position information which users have marked on the garment. Moreover, we verify the effectiveness of our method with numerous experiments.

Keywords image-based 3D modeling, garment modeling, garment seam

1 Introduction

Garment modeling is widely used in computer games, film production and virtual dressing, etc. This modeling method has also been extensively studied in computer graphics, such as garment design and editing, image-based garment modeling, and garment simulation.

The commercial softwares such as Marvelous Designer① and VSticher② have already been used in industrial production. In clothing industry, a garment is made by stitching block patterns together. The seam between block patterns is briefly called "seam" in this paper, which is shown in Fig.1. The shadow and geometry details of the seam are crucial to the garment appearance, and make a 3D (3-dimensional) garment more photorealistic. To the best of our knowledge, the current garment modeling methods or softwares only use the 2D texture to model the garment seam, while the 3D geometry of the seam is not carefully considered.



(a) (b)

Fig.1. (a) Original garment model. (b) Garment model with the seam structure created by our garment seam modeling framework.

In this paper, we propose a garment seam modeling method based on a single seam image. A garment seam database is established and contains garment seam images and parametric seam models to guide garment seam modeling. We locate the exact position of the seam from the seam image, conduct the training

Regular Paper
Special Section of CVM 2018
This work was partially supported by the National Natural Science Foundation of China under Grant Nos. 61532003 and 61421003.
*Corresponding Author
①Marvelous Designer. https://www.marvelousdesigner.com, Mar. 2018.
②VSticher. https://browzwear.com/products/v-stitcher/, Mar. 2018.

method to classify the seam image and automatically generate seam parameters. Then the seam structure is modeled using the parametric model formula. Our model is a parametric model which users can easily edit. In addition, we also connect the seam model and the garment model to generate the garment model having the seam structure. Therefore, we successfully improve garment modeling methods such as the garment model design tool and the garment model generation based on images and generate a garment model with the seam structure based on our methods. Moreover, we have done numerous experiments to verify the effectiveness of our method.

Our system is useful for both graphic applications and industrial manufacturing. With the improvement of rendering technology and people's increasing demands on the game experience, the applications of dressing up virtual characters in films and games have more and more realistic rendering quality. Most garment applications use texture maps to generate seam visual effects, but the realistic rendering of garment could not be achieved. Seam structure can present different visual effects under different lighting or different perspective conditions. Our model can produce light and visual effects for 3D rendering. In addition, our seam model and the sewing thread model are parametric, which has guiding significance for the sewing process of industrial garments. The seam cross-sectional model can give the instruction to the size of seam structure during industrial sewing. The thread model can guide the industrial garment production, such as the width of the seam thread and sewing needle spacing. Hence, our seam systems can help graphics rendering applications for realistic rendering and guide industrial garment sewing processes.

The main contributions of this paper are as follows.

1) A garment seam modeling framework is proposed to produce garment models with the seam structure. To the best of our knowledge, it is the first method to model the garment seam.

2) In our seam modeling framework, we first propose a parametrical garment seam modeling method based on a single seam image and stitch the seam model with garment model.

3) In order to implement our automatical seam modeling framework, a seam database with labeled seam images and parametric seam models is built to model the seam structure. We also propose a method to customize seam model beyond our database to enhance the usability of our system.

The remainder of this paper is organized as follows. In Section 2 we briefly review the related work. Section 3 presents our garment seam modeling framework. Section 4 describes our seam image database and parametric seam models, and Section 5 details seam analysis from a single image. Section 6 presents the method to model garment with seam structure. Section 7 demonstrates some experiments and Section 8 draws some conclusions.

2 Related Work

3D garment has been extensively studied in computer graphics. Liu *et al.*[1] presented a comprehensive review and detailed analysis about CAD methods in 3D garment design. The detailed garment simulation and applications can be found in [2]. In this section, we briefly review the related work in the following three aspects.

Garment Modeling. Most work of garment modeling concentrated on the applications of dressing up virtual characters in films or games. Fontana *et al.*[3] presented a CAD system that allows users to interactively assign sewing and assembly rules between panels to model a complex 3D garment. 3D new garment models are created by Li and Lu[4] on individual human models through compositing 3D parts from garment examples rather than 2D patterns. Liu *et al.*[5] proposed a continuous style description to synthesize the 3D garments with various styles. In comparison, the 2D sketches were also widely used in garment design. Hence some work focused on modeling the garment from user sketches. The methods presented by Berthouzoz *et al.*[6] and Zhong *et al.*[7] can automatically parse existing sewing patterns and convert them into 3D garment models. Wang *et al.*[8] and Mok *et al.*[9] modeled a 3D garment from the sketches around the 3D human model. Moreover the 3D garments can be also modeled from the outline and seam-lines of the front or back of the garment[10-13]. Based on a set of observations about key factors that affect the shape of garments, a context-aware garment modeling method[14] was proposed to obtain more realistic modeling results. However, although the above mentioned methods can achieve good modeling results, they require strong background in the related domain such as garment design, which is very difficult for non-experts.

Garment Capturing. Capturing the 3D garment from a single image is useful for Internet images. Zhou *et al.*[15] captured the garment geometry details using the shading on a single garment image. Recently, Daněřek *et al.*[16] have recovered the garment wrinkle and the global shape of dynamic 3D garment from a single image using convolutional neural network. Jeong *et al.*[17] captured the garment type and primary body size parameters from a single image and Yang *et al.*[18] further estimated the accurate garment and human shape simultaneously from a single image using a statistic model. In some studies[19-20], the garment model can also be reconstructed through stitching components which are acquired by parsing the garment from the RBGD images or front and back images of a garment. The 3D scanners were often used to capture the 3D garment. Pons-Moll *et al.*[21] used a new multi-part 3D model of clothed body which can automatically segment each piece of clothing and estimate the minimally clothed body shape from 3D scans and Frâncu *et al.*[22] also proposed a scanning system for garment. Although these methods are relative simple and easy to execute for non-experts, the modeling results are often unsatisfactory since they reflect limited details and need to be further enhanced.

Garment Editing. The garment designers usually need to change the style and size of a 3D garment. In consequence, the garment editing can be quite vital. Wang *et al.*[23] proposed the garment editing method for fitting a garment to different body shapes while preserving the garment pattern grading design. Umetani *et al.*[24] presented an interactive tool for interactive bidirectional editing between 2D patterns and 3D high-fidelity simulated draped forms. A new framework was also developed, which allows designers to directly edit the garment in 3D space and create the 2D patterns via a physical draping simulation[25]. In addition, a computer aided design (CAD) solution[26], which can be applied to virtual try-on, fitting evaluation and style editing, was proposed in order to speed up the clothing design process. Spahiu *et al.*[27] presented a review about CAD systems used by the fashion designer.

To sum up, those methods study the garment modeling, capturing and editing from different aspects. However, none of them focus on the 3D structure of garment seam which is exactly the core content of our study. Hence in Section 4 and Section 5, we will describe our garment seam modeling process in details.

3 Garment Seam Modeling Framework

The pipeline of our garment seam modeling framework is shown in Fig.2. The input of our method is a single seam image and there are four main steps in our modeling approach: seam image database setup, seam image positioning and seam parameters analysis, seam parametric model design, and the generation of garment model with the seam structure.

Given a single seam image, we first use the location detector to get the exact seam location in the image, which is useful to seam type and parameters analysis because of removing unrelated factors. As shown in Fig.2, the seam classifier is used to identify the type of the seam. Then our sewing thread estimation method could generate seam parameters automatically. The seam formula set has been established according to our seam image database. When we get seam image type and parameters, the parametric model of the seam is generated by the seam modeling formula. In addition, users can mark the seam position on the garment generated by existing methods and finally obtain a garment model with the seam structure.

During the establishment stage of the seam image database, we divide the seam image database into 10

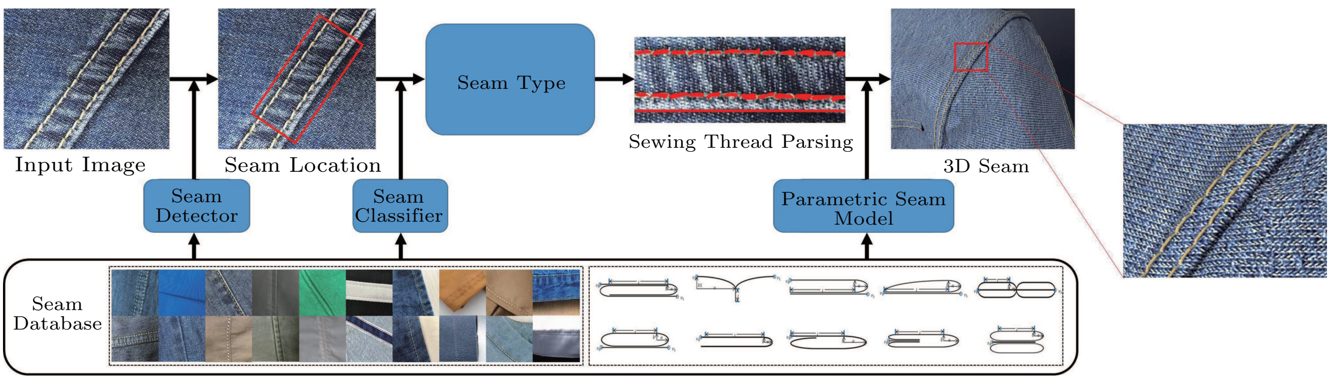


Fig.2. Pipeline of our garment seam modeling framework. Given a seam image, we first use the seam detector to locate the exact position of the seam, and the seam classifier to identify the type of the seam and automatically generate seam parameters. Then the garment model with the seam structure is produced by combining the seam model and the garment model.

categories, each of which includes hundreds of seam images with different textures and sizes. There are totally more than 2 000 seam images in our seam image database.

The exact location of the seam in the image is labeled and a location detector is trained based on the position of each seam in the database to analyze the exact location information of the seam in a single image. The seam classifier is trained using the exact position of the seam image to analyze the seam category. We have established a sewing thread estimation method for automatically analyzing the parameters of the seam model based on a single seam image. Hence, all parameters of the seam model are generated from a seam image and the seam model is also established by using these parameters.

We unfold the existing garment model in the two-dimensional plane, mark the location of the seam section, and segment the garment model. Then the seam model is connected with the garment model in the marked position by our connection method and the garment model with the seam structure can be generated ultimately.

4 Garment Seam Modeling Database

4.1 Seam Image Database

Since most of the garment is produced in the industrial process, our seam image database contains 10 common types of seams which correspond to our seam models one by one as shown in Fig.3.

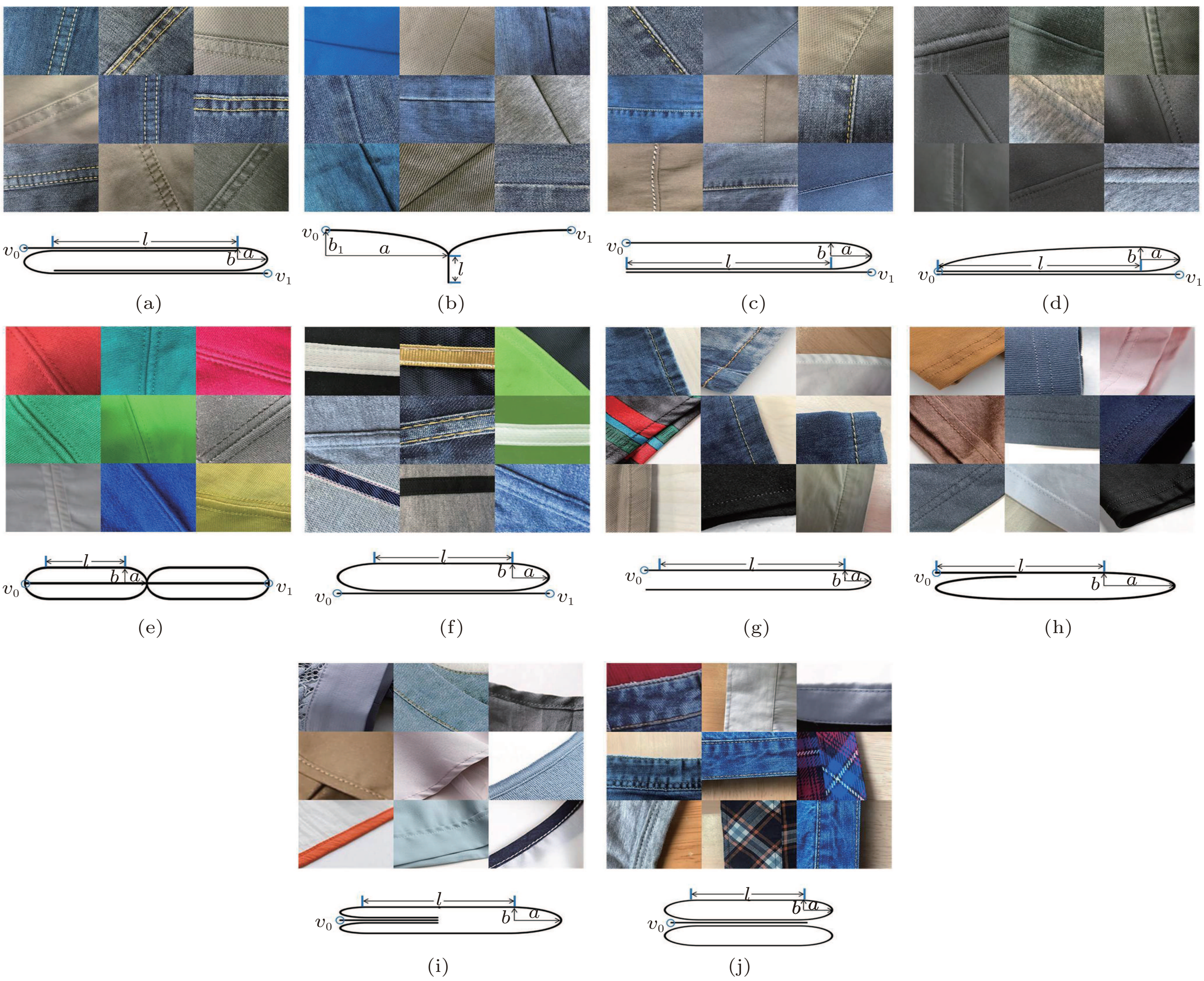


Fig.3. Seam models. (a) Flat-fell seam. (b) Plain seam. (c) Welt seam. (d) French seam. (e) Double-stitching seam. (f) Piped seam. (g) Serged seam. (h) Curling seam. (i) Stuffy seam. (j) Lapped seam.

We collect more than one hundred different seam images for each type of seam under different lighting and viewing environments. Our seam image database includes a total of 2 039 seam images as shown in Table 1.

We mark the exact position of the seam in the database with a bounding box. The most representative information of the seam is cut out in consequence of reducing the interference information.

4.2 Parametric Seam Models Database

Through the observation of seam images in our seam image database, we find that a complete seam model could be obtained by designing a seam cross-section model. Therefore, the parametric formula is established for each type of seam cross-section model in this paper.

For the flat-fell seam model shown in Fig.3(a), v_0 is the starting position of the seam cross-section, and v_1 is the end position of the seam cross-section. They are connected to different fabrics. a and b are halves of the two axises in the ellipse, which represent the degree of bending of the fabric at the sewing position. l is the distance between the two seam threads, and the two ends of l are the positions of the sewing. The rest of seam cross-section models are also shown in Fig.3, and v_0, v_1, a, b and l have the similar meaning to the previous seam model.

The seam cross-section models from Fig.3(a) to Fig.3(f) are used for the sewing method between two pieces of fabric. While the seam cross-section models from Fig.3(g) to Fig.3(j) represent the sewing method at the edge of the fabric.

We also model the sewing thread above the seam section. As shown in Fig.4, l_1 indicates the outside length of sewing thread, and a_1 and b_1 represent the curvature of the thread at the sewing position, while l_2 indicates the length of the inside of the sewing thread. By now the garment seam modeling has been completed.

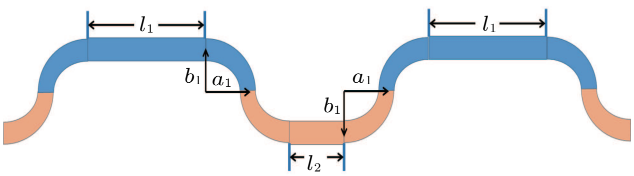


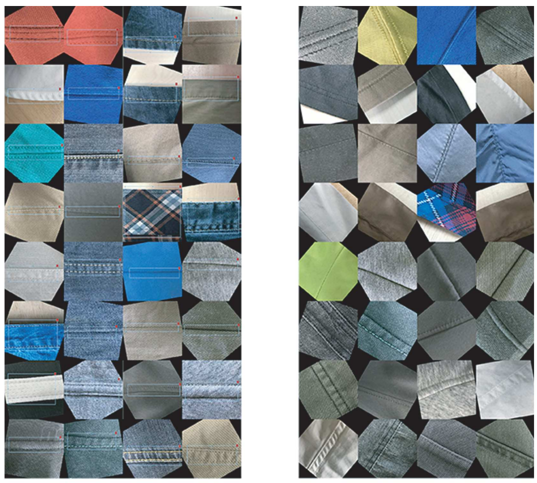
Fig.4. Sewing thread model above the seam section.

5 Seam Model Analysis Based on Seam Database

Here we need to train two detectors: one is based on a deformable-part-model (DPM)[28-29] as a seam position detector, and the other is based on ResNet-101[30] seam type detector.

5.1 Seam Detection and Classification

The DPM detector has a very successful application in target location and segmentation disciplines; hence it is used for precisely locating the seam. The DPM detector requires seam images and seam position labeling information in our seam image database. Since the DPM detector could only be used to detect the horizontal position of the bounding box shown in Fig.5, we rotate the original seam image to find out the most horizontal position of the bounding box and randomly select a seam image which is not in a horizontal position as the negative sample. Eventually we manage to train the DPM detector.



(a) (b)

Fig.5. Part of the training data for the DPM detector. (a) Positive samples with marked bounding boxes. (b) Negative samples.

For the input of the single seam image, we believe that the bounding box could represent the seam position information, which would be used in the seam classification.

We take the ResNet-101 as the target model to fine-tuning. The last fully connected layer whose output is a 1 000-dimensional vector is replaced by a newly initialized layer with the output of 10 dimensions and then followed by a softmax layer. We initialize the newly added fully connected layer by means of "xavier"[31] and take 1 727 samples which are randomly selected from our image database to fine-tune the model. Horizontally flipping is applied to those training data so as to obtain more samples. We set the base learning rate to 0.000 1 and weight decay to 0.000 2. The base learning will be timed with 0.1 every 10 epochs. We fine-tune the model for 12 epochs and then take the fine-tuned model as the final classifier.

5.2 Customized Seam Modeling

To enhance the practicality, our seam database contains the common seam models and these seam models already include most seam structures in daily life scenes. If the user encounters a seam model which is not in our seam database, we provide a practical solution that the user could create a seam model by drawing the special seam cross-section. Hence, our system can automatically generate other seam models by user interaction.

For the seam cross-section model drawn by a user, we first denoise the model using differential filtering function (DIFF) to get the initial curve of seam cross-section (see Fig.6).

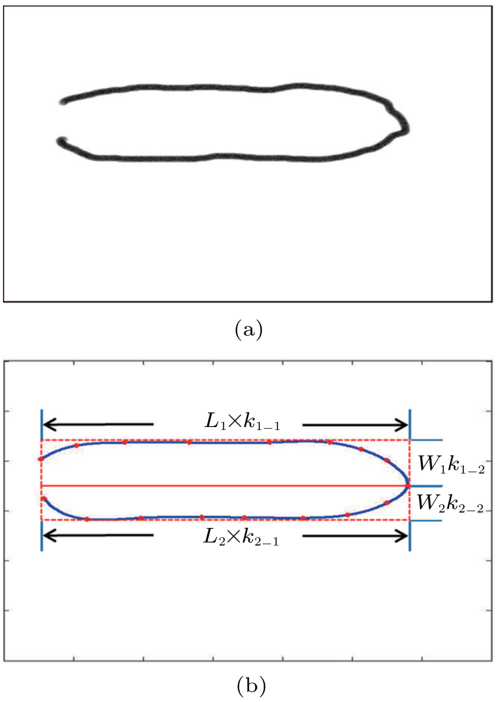


Fig.6. (a) Original image drawn by the user. (b) Seam cross-section model generated by our system. L and W represent the length and width of the original curve segment.

According to the characteristics of the seam model, each curve is divided at the point with the largest absolute value of the tangent to generate multi-segment curves, and key points are obtained on each curve sequence of the initial curve. In order to meet characteristics of seam model, we set the two parameters k_{i-1}, k_{i-2} for each curve, and the users can adjust each curve according to their own needs, as shown in Fig.7. Finally, we use the b-spline function to fit our sequence of key points to generate a complete suture cross-sectional model in (1).

$$\begin{cases} L(x) = \sum_{i=0}^{n-1} \operatorname{sign}((x-x_i)(x_{i+1}-x))S_i(x), \\ S_i(x) = f_0(\tilde{x}, k_{i_2}\tilde{y}) + f_1(\tilde{x}, k_{i_2}\tilde{y})Q + \\ \qquad f_2(\tilde{x}, k_{i_2}\tilde{y})Q^2 + f_3(\tilde{x}, k_{i_2}\tilde{y})Q^3, \\ Q \ \ = x - x_0 - k_{i_1}(x_i - x_0), \\ \text{s.t.} \ \ \tilde{x} \in [x_i, x_{i+1}], \end{cases} \qquad (1)$$

where $L(x)$ is the curve line of our seam model and i is the number of multi-segment curves. x_i indicates the end point of curve i ($curve_i$) and $(\tilde{x}, \tilde{y})$ is the point selected by the DIFF function in $curve_i$. f_0, f_1, f_2, f_3 are the parametric functions of the b-spline method which could be solved by solving equations in [32]. k_{i_1} and k_{i_2} are the horizontal and the vertical adjustment parameters of $curve_i$ respectively.

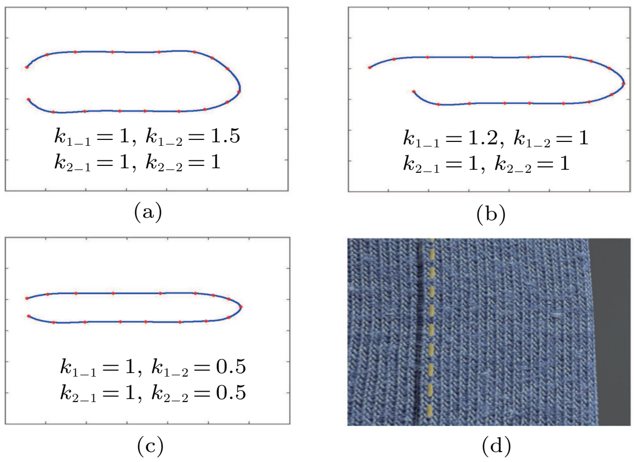


Fig.7. (a) (b) (c) Results of parametric adjustment based on seam cross-section model in Fig.6(b). (d) Rendering result of the seam model in (c).

5.3 Sewing Thread Estimation

The seam image generated by the DPM detector is divided into patches $\mathcal{O}$[33], which is useful to analyze the sewing position and the seam thread parameters (see Figs.8(a) and 8(c)).

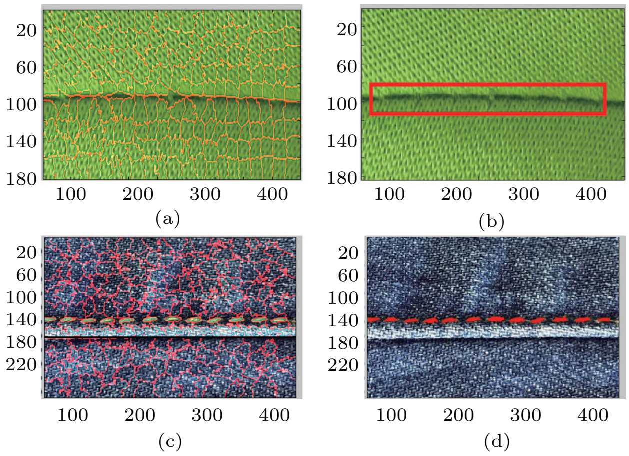


Fig.8. (a) (c) Seam images divided into patches. (b) Result of the seam gap position. (d) Result of the location of the sewing thread.

One of the most important features for the seam model is the seam gap position, which is the position between two pieces of fabric (see Fig.8(b)). It is obvious to find that the seam gap position is located in the border of the patch. In consequence, we design the following formula to analyze the seam gap position.

$$\begin{aligned} &\min_{i} \sum_{j=0}^{k} f(i,j) \\ &\text{s.t. } k = image.width, \\ &\quad 0 < i < image.length, \end{aligned} \tag{2}$$

where $f(i,j)$ in (2) indicates the shortest distance from point(i,j) to the patch boundary.

Another significant feature of the seam model is the location and size of the sewing thread (see Fig.8(d)). There are three factors that affect the analysis of the sewing thread, which are the color prominence, the size of the patch, and the distribution of patches in the horizontal direction. We infer the location and size of the sewing thread by (3).

$$\begin{aligned} &\max_{i} \sum_{j=0}^{k} \sum_{I=1}^{\frac{360}{bin}} \frac{(H(I)-\bar{H}(I))^2}{\bar{H}(I)} + \frac{\lambda}{e^{|\beta-\bar{\beta}|}} \\ &\text{s.t. } (i,\ j) \in patch\ \mathcal{O}, \\ &\quad k = image.width, \\ &\quad 0 < i < image.length, \end{aligned} \tag{3}$$

where H is the histogram of HSV color value in $\mathcal{O}$ and $\bar{H}$ is the histogram of average HSV color value in all patches. bin is the interval length of histogram and Chi-square distance function is used to calculate color distance. β is the size of $\mathcal{O}$. $\bar{\beta}$ is empirically set to half of the average patch size. λ controls the balance of the first term and the second term.

After obtaining the exact position of the sewing thread, the center of each patch can be easily calculated. Through the horizontal translation, patches are overlapped together (see Fig.9(b)). The area of high coincidence is used for the final parameter estimation. We set the 50% coincidence rate as the threshold based on experience and get effective results. Our sewing thread model is used for position matching to have the maximum coincidence rate with the patch block (see Fig.9(c)). The values of l_1, b_1 and a_1 have been generated from position matching. The value of l_2 can be obtained by subtracting 2@a_1 from the average distance between patches.

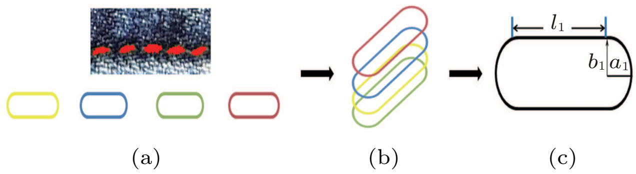


Fig.9. (a) Analytical results for the position of the sewing thread. (b) Center position of the sewing thread overlapped together by translation. (c) Seam thread parameter analysis.

For the seam structure parameters, b is equal to the thickness of fabric as a default value. As shown in Figs.3(a), 3(e), 3(f) and 3(j), the value of l is the vertical distance of the adjacent sewing thread patch and the value of a is the vertical distance between the gap and the thread patch (see Fig.10(a)). In other models, the value of a could be obtained from the average length of all patches connected to the gap position. The value of l can be obtained by subtracting a from the vertical distance between the gap and the thread patches (see Fig.10(b)). By now we have generated all the parameters of the seam models.

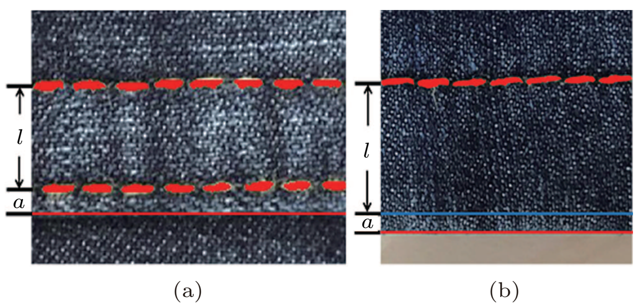
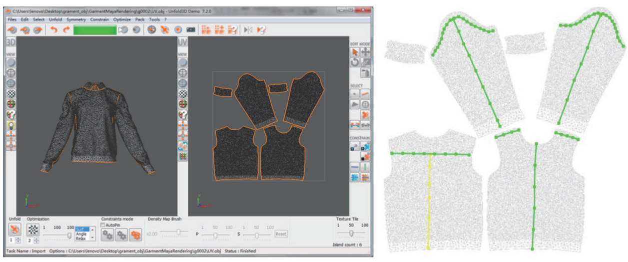


Fig.10. (a) (b) Results of two different seam models.

6 Stitching Seam with Garment

6.1 Marking Seam Positions

First, for the existing garment model (the design model with professional tool or the image-based garment model), we use the unfold3D tool③ to expand it in the 2D plane. Marking the seam of the three-dimensional model cannot accurately deal with garment wrinkles and the shelter of garment sewing position or other issues. This is why it cannot accurately mark the true position of the seam. Hence we first expand the garment model in the 2D plane. Then users mark the seam position as shown in Fig.11.



(a) (b)

Fig.11. (a) Garment model expanded in 2D plane. (b) Seam position marked by users.

6.2 Cutting and Sewing Garment Model

A series of discrete coordinate points on the 2D plane marked by users are mapped to a 3D garment model (our 2D plane points correspond to the points on the 3D model). A smooth seam curve is generated by the spline interpolation function[32].

The newly generated spline curve may not fit well with the garment model, but it will not be far from the surface of the garment. We project the curve through the shortest distance function to the surface of the garment.

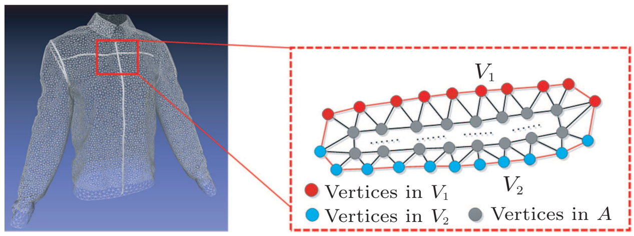
$$\begin{aligned} &\min_{x_2,y_2,z_2} \|(x_2-x_1, y_2-y_1, z_2-z_1)\|_2^2, \\ &\text{s.t.} \ (x_2,y_2,z_2) \in \mathbb{C}^+, \\ &(x_1,y_1,z_1) \in \mathbb{C}^-, \end{aligned} \tag{4}$$

where $\mathbb{C}^+$ includes all points of faces which belong to the garment model, and $\mathbb{C}^-$ includes all points the spine function generates. We use the minimum result of the point (x_2, y_2, z_2) to replace the point (x_1, y_1, z_1) in (4).

In order to connect the seam to the garment model, we retrieve all the vertices of the garment model, find all the vertices (recorded as A) from the garment model by the nearest neighbor method with the seam line and cut the triangular patches where the vertices are located. All the vertices connected to A at the upside of the seam are denoted as V_1 (in red), and the vertex connected to A at the downside of the seam is denoted V_2 (in blue) as shown in Fig.12.

$$\begin{aligned} &\forall\, v \in V_1,\ (v-a) \times \boldsymbol{a.t} \cdot \boldsymbol{a.n} \geqslant 0, \\ &\forall\, v \in V_2,\ (v-a) \times \boldsymbol{a.t} \cdot \boldsymbol{a.n} < 0, \end{aligned} \tag{5}$$

where $\boldsymbol{a.t}$ indicates the tangent vector of A, $\boldsymbol{a.n}$ indicates the normal vector of A, and v indicates all of vertices connected to A.



(a) (b)

Fig.12. (a) Garment model. (b) Categorization results for adjacent vertices.

All points from sets V_1 and V_2 in (5) are shown in Fig.12 with the red line between V_1 set and V_2 set. Through the red line connection relationship, we sort all points from sets V_1 and V_2, which is conducive to patch production. Specific methods refer to Algorithm 1.

Algorithm 1 . Sorting All Points from V_1 and V_2 Sets

Require: n_1 is the number of $V(1,...,n_1)$ in set V_1, and n_2 is the number of $V^*(1,...,n_2)$ in set V_2; we use V_{new} to indicate the new sort of V_1 and V_2
  for $i = 1$ to n_1 **do**
    for $j = 1$ to n_2 **do**
      if V_i is connected to V_j^* **then**
        Set V_j^* is the first V in our sort and V_i is the second V in our sort
        $V_{\text{new}}(1) = V_j^*$
        $V_{\text{new}}(2) = V_i$
        break
      end if
    end for
  end for
  $count = 2$
  while $count < n_1 + n_2$ **do**
    for each $v \in V$ or V^* **do**
      if v is connected to $V_{\text{new}}(count)$ and $v! = V_{\text{new}}(count-1)$ **then**
        $V_{\text{new}}(count+1) = v$ and $count++$
        break
      end if
    end for
  end while
  return V_{new}

③Unfold 3D tool. http://www.polygonal-design.fr/, Mar. 2018.

V_1 is connected to the upside of the seam by the nearest neighbor method, and V_2 is connected to the downside of the seam by the nearest neighbor method. Since there are obvious folds on the directly connected model, we insert smooth points between the connection points, thus creating a smooth garment model with the seam structure.

7 Experiments

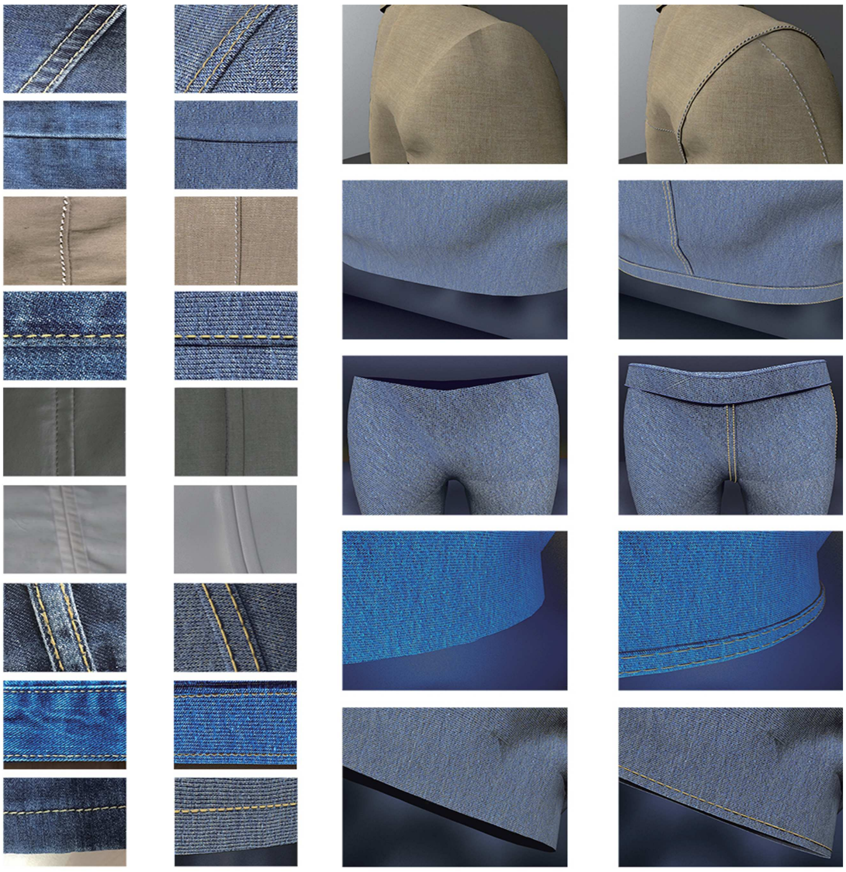
In order to verify the effectiveness of our image-based garment seam modeling method, we use our method to do numerous experiments. We perform all of our experiments on an Intel® Core i7 3.40 GHz CPU machine with 12 GB memory and 1080p GPU. All of our rendering results are done using Mitsuba④.

The seam type detector is generated based on our seam image database. 1 727 seam images are selected as the training set randomly and the test set has 312 seam images. The result is shown in Table 1. The accuracy of each type of seam images and the overall accuracy can be found in the table.

Table 1. Classification Accuracy of Seam Models

Type	Training	Testing	Precision (%)
Flat-fell seam	199	38	92.0
Plain seam	257	40	90.0
Welt seam	183	30	90.0
French seam	242	35	88.5
Double-stitching seam	137	30	90.0
Piped seam	120	29	89.6
Serged seam	228	32	90.6
Curling seam	101	24	87.5
Stuffy seam	168	30	86.6
Lapped seam	102	24	91.6
Total	1 727	312	89.7

Based on our seam modeling framework, we enter a seam image, as shown in Fig.13(a). Our method can automatically generate seam model, as shown in Fig.13(b). From the modeling result, we can see that our modeling system can create many kinds of seam model from a single image.



(a) (b) (c) (d)

Fig.13. (a) Input seam images. (b) Corresponding modeling results. (c) Garment models generated by [19]. (d) Corresponding examples of the garment model with the seam structure.

④Mitsuba. https://www.mitsuba-renderer.org/, Mar. 2018.

In addition, we connect the seam model to the garment model to produce a garment model with the seam structure. The original model of the garment is in Fig.13(c) which is generated using the method of [19] and we present the corresponding garment model with the seam structure in Fig.13(d).

Our system can be easily edited by users. The original garment model with the seam structure is generated in Fig.14(a). The seam models with different numbers of sewing threads are created according to user interaction (see Fig.14(b)). The seam type and seam thread parameters could also be changed by users (see Figs.14(c) and 14(d)).

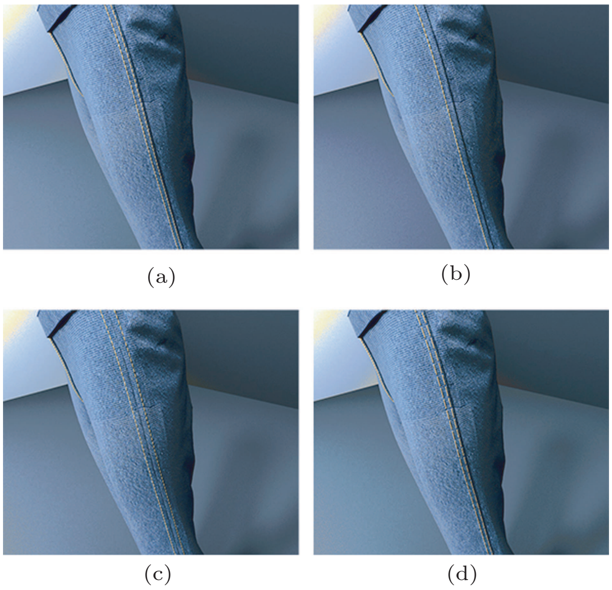


Fig.14. Editing results of our seam models. (a) Original garment model. (b) With different number of sewing threads. (c) with different seam type. (d) with different seam thread.

Seam structure will present different visual effects under different lighting or different perspective conditions. We have done two sets of comparative tests using our method and texture mapping method respectively. In two different observation perspectives and fixed lighting conditions, our seam model and the sewing thread model produce more obvious visual effects (see Fig.15(a)), while the change of texture mapping method is small in detail (see Fig.15(a)). With fixed viewing angles and changing lighting conditions, the lighting effects of our method can change obviously in Fig.15(b), but there is almost no change in the way of texture mapping method (see Fig.15(b)).

Our seam modeling method might cause problems in some situations. In the intersection of two seams, it could not produce the effect of the intersection. We show two examples of failure cases. The first example is the intersection of two seams between the fabrics, and the second example is the intersection of two seams at the edge of the fabric (see Fig.16). This is because our seam database does not have the complex structure of the intersection.

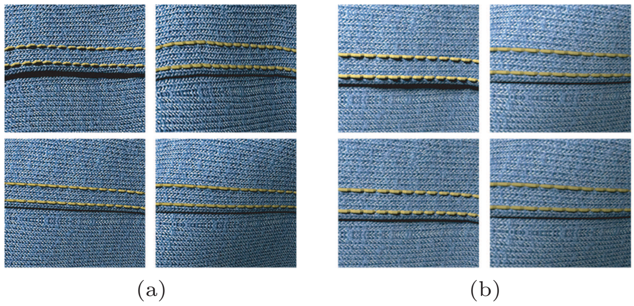


Fig.15. (a) Comparison results of our method and texture mapping method under different perspective conditions. (b) Comparison results under different lighting conditions.

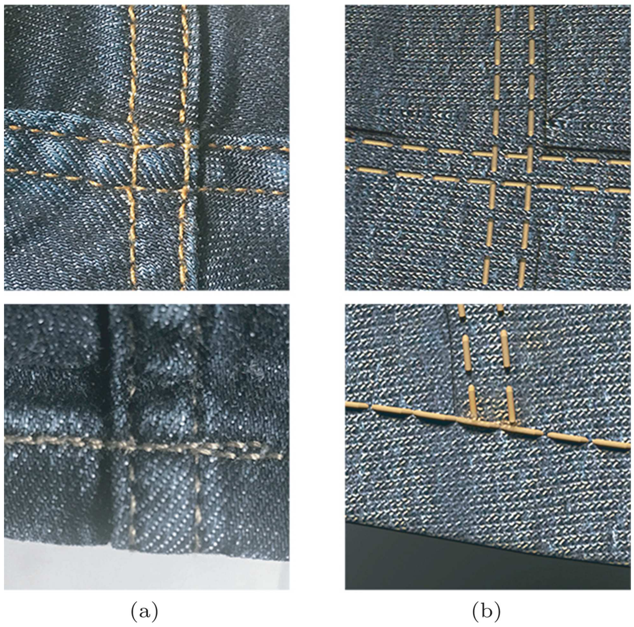


Fig.16. Failure cases. (a) Input seam images. (b) Modeling results.

8 Conclusions

In this paper, we proposed a garment seam modeling framework to generate the seam model based on a single seam image. In order to achieve our goal, we established a marked seam image database and a seam parametric formula set. We connected our seam model and the existing garment model to produce a garment model having seams. Besides, many of the experimental results verified the effectiveness of our system.

The automatical seam model generation is only available in our garment seam modeling database and our system can generate other seam models by user interaction. In addition, for the generation of a garment model with seams, we need users to mark the specific location of the seam using our system. In consequence, our existing seam modeling system can be enhanced by adding more types of images and models to our seam database in order to automatically model more seam types. Moreover, a seam position template database should be created so that users do not need to mark the seam position by selecting the seam position template provided in the database. All these will be covered in out future work.

References

[1] Liu Y J, Zhang D L, Yuen M M F. A survey on CAD methods in 3D garment design. *Computers in Industry*, 2010, 61(6): 576-593.

[2] Jiang Y, Wang R, Liu Z D. A survey of cloth simulation and applications. In *Proc. the 9th International Conf. Computer-Aided Industrial Design and Conceptual Design*, November 2008, pp.765-769.

[3] Fontana M, Carubelli A, Rizzi C, Cugini U. ClothAssembler: A CAD module for feature-based garment pattern assembly. *Computer-Aided Design and Applications*, 2005, 2(6): 795-804.

[4] Li J T, Lu G D. Modeling 3D garments by examples. *Computer-Aided Design*, 2014, 49: 28-41.

[5] Liu L, Su Z, Fu X D *et al.* A data-driven editing framework for automatic 3D garment modeling. *Multimedia Tools and Applications*, 2017, 76(10): 12597-12626.

[6] Berthouzoz F, Garg A, Kaufman D M, Grinspun E, Agrawala M. Parsing sewing patterns into 3D garments. *ACM Trans. Graphics*, 2013, 32(4): Article No. 85.

[7] Zhong Y Q, Jiang J F, Wang Z L, Liu H Y. 3D garment prototyping from 2D drawings. In *Advanced Research on Computer Science and Information Engineering*, Shen G, Huang X (eds.), Springer, 2011, pp.252-257.

[8] Wang C C L, Wang Y, Yuen M M F. Feature based 3D garment design through 2D sketches. *Computer-Aided Design*, 2003, 35(7): 659-672.

[9] Mok P Y, Xu J, Wang X X, Fan J T, Kwok Y L, Xin J H. An IGA-based design support system for realistic and practical fashion designs. *Computer-Aided Design*, 2013, 45(11): 1442-1458.

[10] Turquin E, Cani M P, Hughes J F. Sketching garments for virtual characters. In *Proc. ACM SIGGRAPH 2007 Courses*, August 2007.

[11] Xu W P, Wu Z J, Qiang X H. Interactive garment modeling based on features through sketches. In *Proc. the 10th Int. Conf. Virtual Reality Continuum and Its Applications in Industry*, December 2011, pp.391-394.

[12] Turquin E, Wither J, Boissieux L, Cani M P, Hughes J F. A sketch-based interface for clothing virtual characters. *IEEE Computer Graphics and Applications*, 2007, 27(1): 72-81.

[13] Decaudin P, Julius D, Wither J, Boissieux L, Sheffer A, Cani M P. Virtual garments: A fully geometric approach for clothing design. *Computer Graphics Forum*, 2006, 25(3): 625-634.

[14] Robson C, Maharik R, Sheffer A, Carr N. Context-aware garment modeling from sketches. *Computers & Graphics*, 2011, 35(3): 604-613.

[15] Zhou B, Chen X W, Fu Q, Guo K, Tan P. Garment modeling from a single image. *Computer Graphics Forum*, 2013, 32(7): 85-91.

[16] Daněřek R, Dibra E, Öztireli C, Ziegler R, Gross M. DeepGarment: 3D garment shape estimation from a single image. *Computer Graphics Forum*, 2017, 36(2): 269-280.

[17] Jeong M H, Han D H, Ko H S. Garment capture from a photograph. *Computer Animation & Virtual Worlds*, 2015, 26(3/4): 291-300.

[18] Yang S, Ambert T, Pan Z R, Wang K, Yu L C, Berg T, Lin M C. Detailed garment recovery from a single-view image. arXiv: 1608.01250, 2016. https://arxiv/abs/608.01250, March 2018.

[19] Chen X W, Zhou B, Lu F X, Wang L, Bi L, Tan P. Garment modeling with a depth camera. *ACM Trans. Graphics*, 2015, 34(6): Article No. 203.

[20] Huang P, Yao J F, Zhao H H. Automatic realistic 3D garment generation based on two images. In *Proc. Int. Conf. Virtual Reality and Visualization*, Sept. 2016, pp.250-257.

[21] Pons-Moll G, Pujades S, Hu S, Black M J. ClothCap: Seamless 4D clothing capture and retargeting. *ACM Trans. Graphics*, 2017, 36(4): Article No. 73.

[22] Frâncu M, Moldoveanu F. Virtual try on systems for clothes: Issues and solutions. *UPB Scientific Bulletin, Series C*, 2015, 77(4): 31-44.

[23] Wang J P, Wang R M, Zhou F. 3D virtual garments deformation method based on size-driven. In *Proc. the 5th Int. Conf. Digital Home*, November 2014, pp.192-197.

[24] Umetani N, Kaufman D M, Igarashi T, Grinspun E. Sensitive couture for interactive garment modeling and editing. *ACM Trans. Graphics*, 2011, 30(4): Article No. 90.

[25] Bartle A, Sheffer A, Kim V G, Kaufman D M, Vining N, Berthouzoz F. Physics-driven pattern adjustment for direct 3D garment editing. *ACM Trans. Graphics*, 2016, 35(4): Article No. 50.

[26] Meng Y W, Mok P Y, Jin X G. Computer aided clothing pattern design with 3D editing and pattern alteration. *Computer-Aided Design*, 2012, 44(8): 721-734.

[27] Spahiu T, Shehi E, Piperi E. Advanced CAD/CAM systems for garment design and simulation. In *Proc. the 6th Int. Conf. of Textile*, November 2014.

[28] Felzenszwalb P F, Girshick R B, McAllester D, Ramanan D. Object detection with discriminatively trained part-based models. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2010, 32(9): 1627-1645.

[29] Girshick R B, Felzenszwalb P F, McAllester D. Discriminatively trained deformable part models (Version 5). 2012. http://people.cs.uchicago.edu/∼rbg/latent-release5/, Mar. 2018.

[30] He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In *Proc. the IEEE Conf. Computer Vision and Pattern Recognition*, June 2016, pp.770-778.

[31] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In *Proc. the 13th Int. Conf. Artificial Intelligence and Statistics*, March 2010, pp.249-256.

[32] Reinsch C H. Smoothing by spline functions. *Numerische Mathematik*, 1967, 10(3): 177-183.

[33] Achanta R, Süsstrunk S. Superpixels and polygons using simple non-iterative clustering. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, July 2017, pp.4895-4904.

Chen-Xu Zhang is a Master student of School of Computer Science and Engineering, Beihang University, Beijing. He received his B.S. degree in software engineering from Beihang University, Beijing, in 2015. His research interests include computer graphics, virtual reality, and augmented reality.

Xiao-Wu Chen is a professor in Beihang University, Beijing. He received his B.S. degree in applied mathematics in 1994, and M.S. and Ph.D. degrees in computer science in 1997 and 2001 respectively, all from Beihang University, Beijing. His research interests include computer vision, computer graphics, virtual reality, and augmented reality.

Hong-Yu Wu is a research assistant in the State Key Laboratory of Virtual Reality Technology and Systems, and also the School of Computer Science and Engineering, Beihang University, Beijing. He received his B.S. degree from University of Electronic Science and Technology of China, Chengdu, in 2010, and Ph.D. degree in computer science from Beihang University, Beijing, in 2017. His research interests include computer graphics, virtual reality, augmented reality and image processing.

Bin Zhou is a lector in the State Key Laboratory of Virtual Reality Technology and Systems, and also the School of Computer Science and Engineering, Beihang University, Beijing. He received his B.S. and Ph.D. degrees in computer science from Beihang University, Beijing, in 2006 and 2014 respectively. His research interests include computer graphics and virtual reality.