

# Image-Based Virtual Fitting System with Garment Image Reshaping

Hiroki Yamada, Masaki Hirose, Yoshihiro Kanamori, Jun Mitani and Yukio Fukui

Department of Computer Science

University of Tsukuba

Email: {yamada,hirose}@npal.cs.tsukuba.ac.jp, {kanamori,mitani,fukui}@cs.tsukuba.ac.jp

**Abstract**—We propose an image-based virtual fitting system to reproduce the appearance of fitting during online shopping for garments. Inputs are whole-body images of a fashion model and the customer. We create a garment image by cutting out the garment portion from the image of the fashion model. If the garment images were naively superimposed, the fitting result would look strange mainly because the shape of the garment will not match the body shape of the customer. In this paper, we therefore propose a method of reshaping the garment image based on human body shapes of the fashion model and the customer to make the fitting result more realistic. The body shape is automatically estimated from the contours of the human body, and can easily be retouched if necessary. The fitting result is refined further by automatic color correction with reference to the facial regions and a method of retouching parts that protrude from the rear of the garment image. We verified the effectiveness of our system through a user test.

**Keywords**-image-based virtual fitting; image warping; color constancy;

## I. INTRODUCTION

The number of users shopping online for garments has tended to increase recently, and this is expanding widely as means of enjoying shopping comfortably from customers' own homes. However, there is a problem that we cannot try on clothes because we cannot handle real garments, unlike shopping in a real bricks-and-mortar store. Since fit is such an important criterion when buying garments, this is one reason impeding the spread of online shopping. To address this problem, there have been proposals for virtual fitting systems that implement fitting online in a virtual manner. For example, the online virtual fitting service *Awaseba* [1] composites a whole-body image of a registered customer (hereafter called a “*customer image*”) with an image of a garment from a catalog (hereafter called a “*garment image*”), and presents an image of the customer apparently wearing the garment. However, typical online services just provide virtual fitting by naïve compositing.

Virtual fitting by naïve compositing often produces unnatural images (Fig. 1, right). Whereas this is partially because of the differences in brightness between the images, the large cause is that the shape of the garment image does not match the body shape of the customer. For example, in Fig. 1, the garment is neither matched to the positions of the arms nor the width of the trunk.



Figure 1. Virtual fitting by a naïve superimposing. From left, the garment image, the customer image, and the fitting result. Naïve superimposition results in an unnatural image mainly because the garment does not match the body shape.

We therefore introduce an image-based virtual fitting system that suppresses the unnaturalness of the virtual fitting image. In our system, the garment image to be fitted is cut out from an image of it being worn by a fashion model, and then is warped according to a warping function with which the body shape of the fashion model matches that of the customer. This can be considered as warping in the literature of *deformation transfer* [2], where deformation is transferred from a source 3D mesh onto another target mesh. The body shapes are automatically estimated from the contours of each human body and can be retouched if necessary. The brightness differences between the garment image and the customer image are adjusted automatically by color correction based on facial colors. Finally, our system automatically retouches protrusion from the rear of the composite garment image.

Our contribution lies in the entire design of a novel virtual fitting system with the following features:

- 1) garment image reshaping based on wearers’ body shapes,
- 2) automatic color correction using facial colors, and
- 3) automatic retouching of protrusion behind garments.

We confirm the effectiveness of our system by conducting a user test.

## II. RELATED WORK

Existing virtual fitting approaches can be divided into two main groups, i.e., 3D model-based and 2D image-based methods.

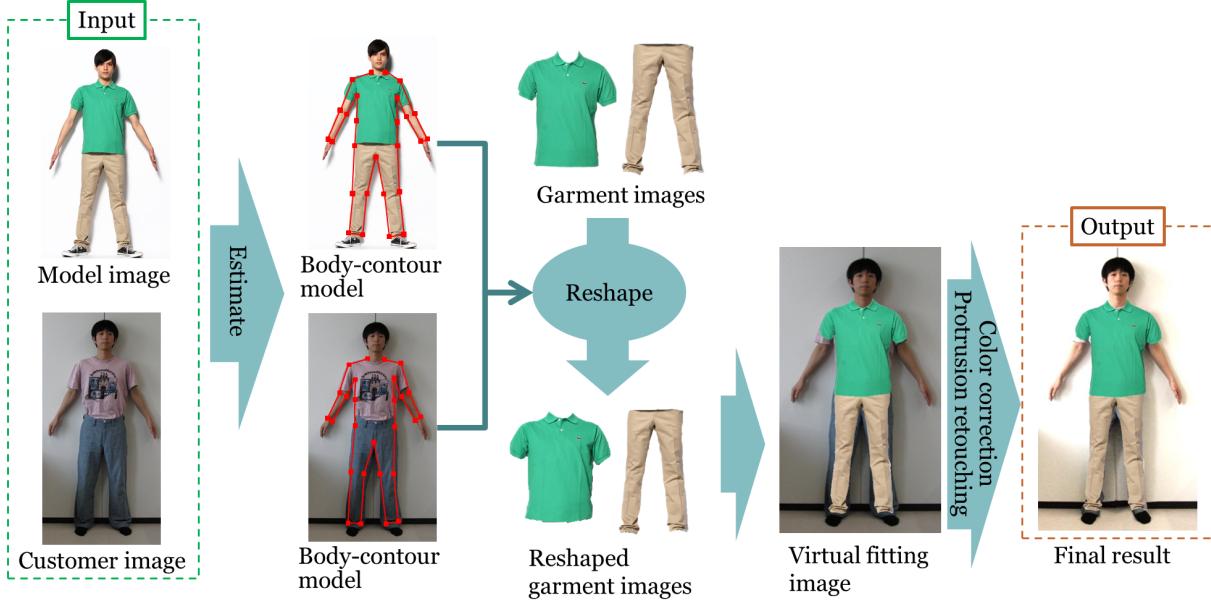


Figure 2. Overview of our system. Our system first estimates the body-contour models both for the fashion model and the customer, and then warps the garment cut out from the fashion model’s image so that the garment matches to the customer’s body shape. The fitting result is refined further by automatic color correction and automatic protrusion retouching.

There have been proposed 3D virtual try-on systems [3]–[5] that can produce compelling fitting results using measured 3D human models and cloth simulation based on physical parameters of garments. Recent dressing simulations [6]–[9] can reproduce detailed drapes or folds of garments fitted on various different body shapes. Whereas these simulations often use a 3D avatar as a 3D human model, customer’s own 3D model can be generated using depth camera, e.g., Microsoft Kinect, for whole-body scan [10], [11].

Despite these progress, providing an experience as if the customer him- or herself wore a specific garment is still difficult because photorealistic rendering of 3D human models as well as garments is not handy even now. Additionally, preparing a huge number of detailed 3D models for garments in online shops is currently impractical.

Compared to 3D model-based approaches, 2D image-based methods have advantages that collecting data and photorealistic rendering are relatively easy. As such, there are example-based approaches that use a pre-recorded database and find the best matching dataset to provide a virtual fitting result. The system by Ehara and Saito [12] used a database of marker-attached T-shirt images in different poses. Zhou et al. [13] prepared a database of animating garments in different poses and achieved a real-time virtual fitting by superimposing a garment fitted onto a person captured using Kinect. Hilsmann et al. [14] reproduced the shapes of garments with detailed folds by interpolating images of garments worn in different poses. These example-

based approaches, however, are hardly applicable in real apparel websites because of the costly database; for example, for an apparel website [15], photographs of more than ten thousand items are taken everyday. In contrast, we reduce the load of preparation beforehand, by inputting only one 2D image for creating data for one garment.

### III. PROPOSED SYSTEM

The flow of our system is shown in Fig. 2. The inputs to our system are a whole-body image of a fashion model wearing the garment to be used in the fitting (hereafter called a “*model image*”) and a customer image. We first estimate the body-contour models from contours of the human bodies in the both input images. The estimated body-contour models can be retouched by user input if necessary. We then determine how the garment is to be reshaped from the body-contour model of the fashion model and that of the customer. The customer adjusts the position of the garment image to be composited and a virtual fitting image is output. Finally, a more natural virtual fitting result is obtained by correcting the brightness of the customer image and retouching protrusions.

#### A. Body-contour model

In our system, contours of human bodies are used to reshape garments according to the differences of physique shapes between the fashion model and the customer. Ideally, we should know the underlying naked body shapes hidden by clothes, as done by Guan et al. [16]. Their method trains a

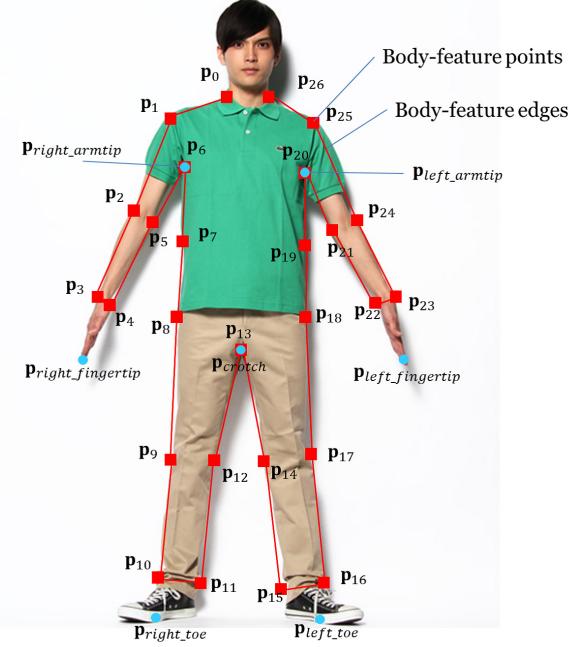


Figure 3. Body-contour model. It is a 2D shape that consists of 27 body-feature points (from  $p_0$  to  $p_{26}$ , red squares) and body-feature edges that connect the points. The blue circles indicate the reference points used to determine body-feature points.

parametric 2D contour model using a database of parametric 3D human models with different proportions and different poses [17], which requires highly-costly preparation. Our system instead uses a simple heuristics to estimate a simple 2D shape model while restricting the input poses in which the arms and feet are slightly spread to avoid occlusions of body contours (Fig. 3). Our 2D shape model, called a *body-contour model*, is specific to our system and can be estimated with less calculation. Our estimation is of course error-prone, but the user can modify the estimated model easily and quickly, as demonstrated in our user test.

The body-contour model consists of body-feature points and body-feature edges, as shown in Fig. 3. The number of body-feature points was determined through experiments regarding estimation easiness and quality of image reshaping. Consequently, we selected a total of 27 points at the neck, shoulder, arm, wrist, armpits, trunk, hips, leg, and ankle on the left and right sides plus one at the crotch, as shown by from  $p_0$  to  $p_{26}$  in Fig. 3.

1) *Heuristic estimation of body-contour model*: The positions of body-feature points are estimated under an assumption that proportional lengths of the body parts are common to human bodies even though there are individual differences. For example, the position of the arm joint can be predicted as being at a position about one-third of the length from shoulder to fingertips (Fig. 4, left). Our heuristics is thus to find noticeable points such as necks and shoulders

Right side			
<i>P</i>	<i>A</i>	<i>B</i>	<i>PA : PB</i>
Arm $p_2$	Shoulder $p_1$	Fingertip $p_{right\_fingertip}$	1 : 2
Wrist $p_3$	Arm $p_2$	Fingertip $p_{right\_fingertip}$	2 : 1
Hip $p_8$	Armpit $p_6$	Toe $p_{right\_toe}$	1 : 2
Trunk $p_7$	Armpit $p_6$	Hip $p_8$	1 : 1
Leg $p_9$	Hip $p_8$	Toe $p_{right\_toe}$	2 : 1
Ankle $p_{10}$	Leg $p_9$	Toe $p_{right\_toe}$	2 : 1

Left side			
<i>P</i>	<i>A</i>	<i>B</i>	<i>PA : PB</i>
Arm $p_{24}$	Shoulder $p_{25}$	Fingertip $p_{left\_fingertip}$	1 : 2
Wrist $p_{23}$	Arm $p_{24}$	Fingertip $p_{left\_fingertip}$	2 : 1
Hip $p_{18}$	Armpit $p_{20}$	Toe $p_{left\_toe}$	1 : 2
Trunk $p_{19}$	Armpit $p_{20}$	Hip $p_{18}$	1 : 1
Leg $p_{17}$	Hip $p_{18}$	Toe $p_{right\_toe}$	2 : 1
Ankle $p_{16}$	Leg $p_{17}$	Toe $p_{right\_toe}$	2 : 1

Figure 4. Ratios used to determine body-feature points. body-feature point  $P$  is determined by the ratio of lengths  $PA$  and  $PB$  along the contour. The left image shows an example for body-feature point  $p_2$  for the right arm. In the left image, the squares are body-feature points and the circle is a reference point.

first, and then apply pre-defined proportions to determine remaining points.

We first extract a person’s contour by tracing the boundary of a pre-defined mask, and sample the contour at  $M$  equidistant points (we use  $M = 200$ ). Of the points on this contour, we take the uppermost one (with the maximum  $y$ -coordinate) to be the point  $v_0$  at the top of the head and number the rest counterclockwise as  $v_i$  ( $i = 0, 1, \dots, M-1$ ). We also take the angle subtended by the vertices  $v_{i-1}$ ,  $v_i$  and  $v_{i+1}$  to be  $\theta_i \in [0, 2\pi]$  (counterclockwise is positive). We select body-feature points from  $p_0$  to  $p_{26}$  among the  $M$  contour points.

*Search for neck and shoulder points.* : We extract the neck and shoulder points as the points that give the greatest and smallest values of the angle  $\theta_i$  when counting along a given number of contour points from the top-of-the-head point  $v_0$ . The neck point  $p_0$  on the left side is the point at which the angle  $\theta_i$  is smallest within the range of  $l$  points (we use  $l = 10$ ) in sequence counterclockwise from  $v_0$ . Next, the shoulder point  $p_1$  on the left side is the point at which the angle  $\theta_i$  is greatest within the range of  $m$  points (we use  $m = 10$ ) counterclockwise from  $p_0$ . We search in a similar manner clockwise from  $v_0$  for the neck and shoulder points  $p_{26}$  and  $p_{25}$  on the right side.

*Search for fingertips, armpits and others.* : We then detect noticeable points such as fingertips and armpits among the contour points by reversals in the  $y$  coordinate, which we use these points as reference when obtaining the body-feature points by length ratios, and thus we call them “reference points”. Note that we do not use the top-of-the-head point as a reference point. The reference points in our system are the following seven points (Fig. 3): the fingertips, armpits, and toetips on the left and right sides, plus the

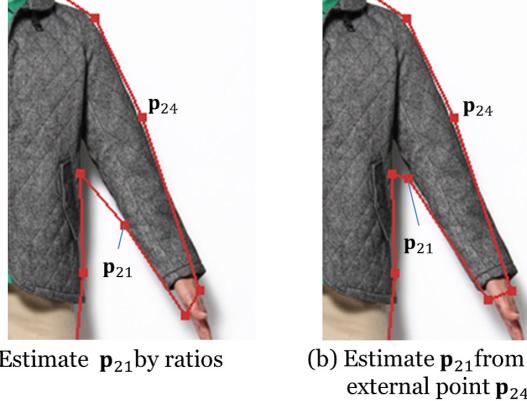


Figure 5. Comparsion of the feature-point calculation. Body-feature point  $p_{24}$  is calculated (a) by length ratio and (b) based on external point  $p_{24}$ . In (b), both  $p_{24}$  and the wrist point are calculated more accurately based on external points.

crotch. Regarding notations, the left and right fingertips are  $p_{left\_fingertip}$  and  $p_{right\_fingertip}$ , the left and right armpits  $p_{left\_armpit}$  and  $p_{right\_armpit}$ , the left and right toe tips  $p_{left\_toe}$  and  $p_{right\_toe}$ , and the crotch  $p_{crotch}$ .

The body-feature points for armpits and crotch are selected from reference points:  $p_6 = p_{right\_armpit}$ ,  $p_{20} = p_{left\_armpit}$  and  $p_{13} = p_{crotch}$ . For the remaining body-feature points for the arms and stomach, we determine the positions according to the pre-defined length ratio between each pair of certain points, as listed in Fig. 4.

Regarding the internal points at the arm and leg parts, we also tried the ratio-based approach but often failed because of accumulated errors. We thus instead find these points from the corresponding external points at opposite sides according to the contour normals (Fig. 5). The pairings of the internal and external points are shown in Table I.

The resultant body-contour models estimated automatically by our heuristics are demonstrated in the accompanying video.

Table I  
PAIRINGS OF INTERNAL AND EXTERNAL BODY-FEATURE POINTS.

Positions	Right side		Left side	
	Internal	External	Internal	External
Arms	$p_5$	$p_2$	$p_{21}$	$p_{24}$
Wrists	$p_4$	$p_3$	$p_{22}$	$p_{23}$
Legs	$p_{12}$	$p_9$	$p_{14}$	$p_{17}$
Ankles	$p_{11}$	$p_{10}$	$p_{15}$	$p_{16}$

### B. Garment image reshaping

Using the body-contour models, the garment image is reshaped to fit the customer's body shape. We used the 2D mesh warping of Weng et al. [18] for reshaping the garment image. Their method suppresses excessive distortions while trying to keep the area of the input mesh, which seems

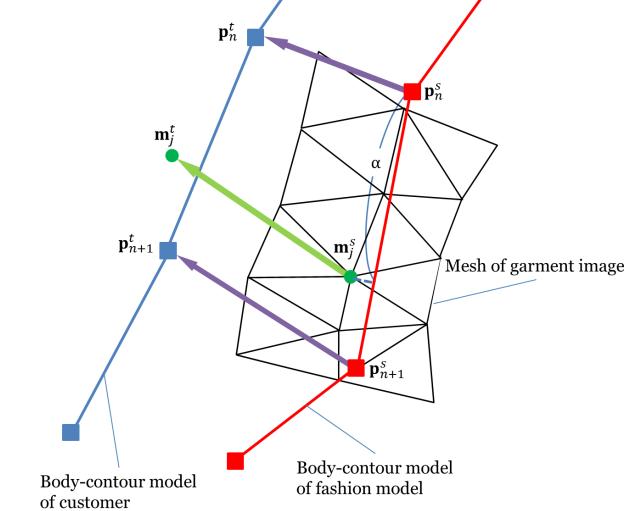


Figure 6. Illustration of the calculation of control points from the body-contour model. The red curve indicates fashion model's contour while the blue curve the customer's. Control point  $m_j^s$  is selected as the mesh vertex that is the closest from fashion model's feature edge.

beneficial for emulating the real behavior of clothes; e.g., the trunk part of a too-small shirt should become short when worn. To apply their method, we first divide a garment image into a triangular mesh using constrained Delaunay triangulation. We then automatically set control points for image warping based on body-contour models as follows.

1) *Automatic setting of control points:* To apply the method of Weng et al. [18], we select control points and designate their destination positions for reshaping. Fig. 6 illustrates this schematically. Let  $p_n^s$  be fashion model's body-feature point and  $p_n^t$  be customer's. Control points  $m_j^s$  are selected for each body-feature edge, and selected among mesh vertices that lie within a certain distance from each body-feature edge. Let  $m_j^t$  be the destination position of control point  $m_j^s$ .  $m_j^t$  is calculated from the linear sum of the vectors  $w_n = p_n^t - p_n^s$  and  $w_{n+1} = p_{n+1}^t - p_{n+1}^s$  as follows:

$$m_j^t = m_j^s + (1 - \alpha)w_n + \alpha w_{n+1},$$

$$\alpha = \frac{(p_{n+1}^s - p_n^s) \cdot (m_j^s - p_n^s)}{|p_{n+1}^s - p_n^s|^2}. \quad (1)$$

Note that, if the same mesh vertex is selected as control points for multiple body-feature edges, the vertex is used as the control point only for the nearest body-feature edge.

### C. Brightness correction based on facial regions

We perform automatic color correction to suppress the brightness differences between the garment image and the customer image. This is based on *color constancy* on facial colors, i.e., an assumption that the facial colors of the fashion model and the customer are the same, and the strangeness

is caused solely by the difference of the photographic environments. Recently, Bianco et al. [19] also proposed a method for estimating illumination based on facial colors. Their method learns a database consisting of each pair of a neutral-color reference card (used to know the ground truth) and a facial color. Our method is much simpler; we just adjust the brightness only from a pair of images without using such database.

Our method works as follows. First, facial regions are extracted automatically using *Active Shape Model* [20] both from the model image and the customer image. Here we assume that the fashion model is photographed under an ideal illumination while the customer under an unknown illumination, and the facial color of the fashion model  $\mathbf{f}_m$  and that of the customer  $\mathbf{f}_c$  have the following relationship based on the von Kries model [21]:

$$\mathbf{f}_m = D_{c,m} \mathbf{f}_c, \quad (2)$$

where  $D_{c,m}$  is a diagonal matrix that converts colors under the customer's illumination into those under the ideal (i.e., fashion model's) illumination.  $D_{c,m}$  is then estimated as follows.

$$D_{c,m} = \text{diag}(\mathbf{f}_c)^{-1} \mathbf{f}_m, \quad (3)$$

where  $\text{diag}(\mathbf{x})$  represents a diagonal matrix consisting of the three components of vector  $\mathbf{x}$ . We then apply matrix  $D_{c,m}$  to the customer image. We confirmed that the Lab color space yields good results by experiments.

In case that the colors of the fashion model and the customer are largely different, our system lets the customer choose the most appropriate facial color from face examples with different colors.

#### D. Protrusion retouching

1) *Local reshaping of image to reduce protrusions:* To prevent customer's cloth from protruding from the garment image (Fig. 7, left), we warp the customer image locally (Fig. 7, right). For this, we use the method of Schaefer et al. [22] to provide faster feedback. In the following, we describe how to automatically set control points used for the image warping.

2) *Control point setting:* Control points are assigned automatically along image boundaries and customer's body contour (Fig. 10). For image boundaries, 20 control points are set equidistantly for each vertical and horizontal boundary. For customer's body contour, 200 control points are set in the same way as the contour points in Section III-A.

3) *Addition of boundary information to garment image:* Control points along customer's contour are not necessarily displaced; some parts of the garment contour, e.g., the throat and cuffs, permit protrusions of customer's cloth, and thus control points around such parts do not have to be moved. We call such edges of the garment contour as “open-margin edges”. Conversely, edges around shoulders forbid

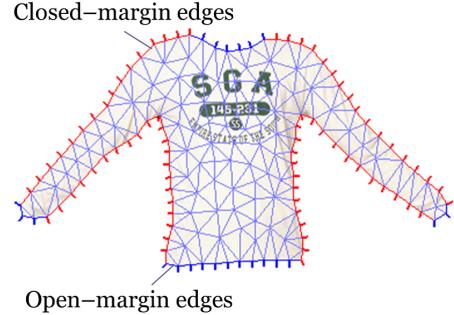


Figure 8. Open-margin edges (blue) and closed-margin edges (red) of a garment image. Edges that permit protrusions, e.g., sleeves and neck, are configured as open-margin edges.

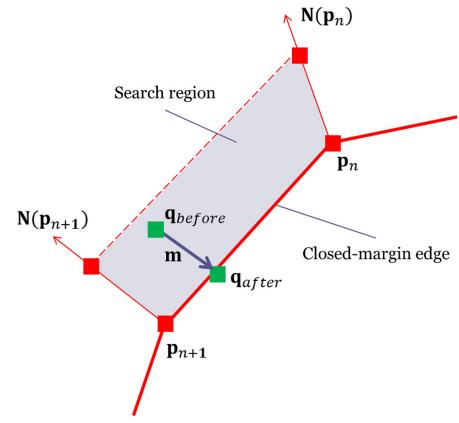


Figure 9. Illustration of control point's destination. For each closed-margin edge, control points are searched and moved onto their closed-margin edge.

protrusions and we call such edges of the garment contour as “closed-margin edges”. Fig. 8 shows an example. By default, all the edges of the garment contour are set as closed-margin, and we let the user manually mark open-margin edges.

Control points around closed-margin edges are to be displaced, and are searched within a trapezoidal region formed by extensions in contour normal directions from both endpoints of each closed-margin edge (Fig. 9). If control points are found within the search region and are outside of the garment image, they are registered as to be displaced.

4) *Displacement of control points:* Fig. 9 illustrates how to determine the destination positions of the control points. For each control point, we first find its belonging closed-margin edge. Let  $\mathbf{m}$  be the displacement direction vector of the control point, then  $\mathbf{m}$  is calculated by linear interpolation of two normals at both endpoints of the closed-margin edge:

$$\begin{aligned} \mathbf{m} &= -(1 - \alpha) \mathbf{N}(\mathbf{p}_n) - \alpha \mathbf{N}(\mathbf{p}_{n+1}) \\ \alpha &= \frac{(\mathbf{q}_{\text{before}} - \mathbf{p}_n) \times \mathbf{N}(\mathbf{p}_n)}{(\mathbf{q}_{\text{before}} - \mathbf{p}_n) \times \mathbf{N}(\mathbf{p}_n) + (\mathbf{q}_{\text{before}} - \mathbf{p}_{n+1}) \times \mathbf{N}(\mathbf{p}_{n+1})} \end{aligned} \quad (4)$$

where  $\mathbf{N}(\mathbf{p}_n)$  denotes the normal at  $\mathbf{p}_n$  and is calculated as an unit vector orthogonal to the vector  $\mathbf{p}_{n+1} - \mathbf{p}_{n-1}$ .

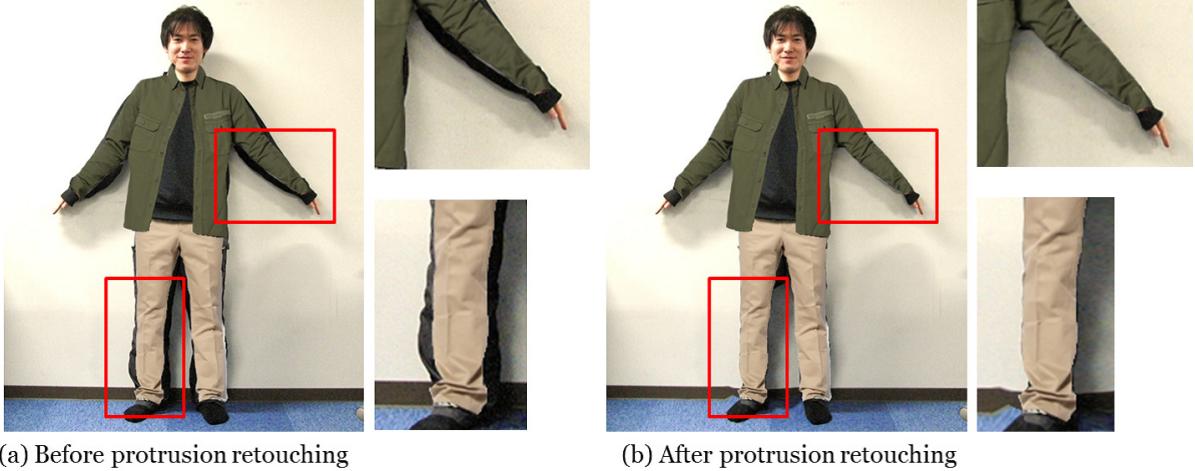


Figure 7. Example of protrusion retouching. (a) Before and (b) after the retouching. The red regions in each image are magnified. The protrusions from the backs of garments are retouched naturally.

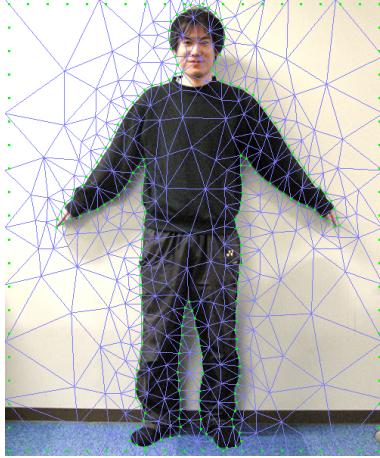


Figure 10. Control points (green) and mesh (blue) for the customer image. Control points are assigned equidistantly along the customer's contour and the image boundary.

denotes a 2D vector operation, i.e.,  $\mathbf{a} \times \mathbf{b} = (a_x, a_y) \times (b_x, b_y) = a_x b_y - a_y b_x$ .  $\mathbf{q}_{before}$  is the position of the control point before displacement. Its destination  $\mathbf{q}_{after}$  is determined by moving  $\mathbf{q}_{before}$  onto the closed-margin edge along vector  $\mathbf{m}$  (Fig. 9).

#### IV. RESULTS

Our prototype system was written in C++ using OpenGL and wxWidgets. The execution environment was a PC with an Intel Core i7-2600 CPU (3.40 Ghz), and 4 GB main memory, with an NVIDIA GeForce GTX460 GPU.

##### A. Virtual fitting results

Fig. 11 shows virtual fitting images using our system. Without garment image reshaping (Fig. 11(c)), the result

looks unnatural because the garment image does not conform to the arm and leg positions. With reshaping (Fig. 11(d)), the result becomes better. In Fig. 11(e), color correction reduces the brightness difference caused by the illumination differences, and the protrusion retouching makes the result look closer to real fitting results. Color correction is especially useful because customers often take photographs under uncalibrated lights whereas the fashion model are taken under studio-adjusted lights.

##### B. User testing

To evaluate the validity of our results, we conducted experiments with ten students majoring in computer science (six males and four females). We asked the test subjects to use images of themselves as customer images and evaluate whether or not the virtual fitting images created by using the customer-side system were natural composite images. We used a five-point Likert scale in the evaluation. We prepared four garment sets as the garment data for the virtual fitting: (A) coat, (B) T-shirt and jeans, (C) jacket and jeans, and (D) poncho and skirt. For each of the garment sets, we commissioned the creation of two types of virtual fitting image: virtual fitting images obtained by simple superimpositions and images created using our system. The male test subjects were given the three garment sets (A), (B), and (C) while the female test subjects were given all of the four sets.

Fig. 12 shows the results. The virtual fitting images created using our system were more highly evaluated than those obtained by simple superimposition, making it clear that natural composite images are produced using our system. However, there were also comments by test subjects who did not score as high as 5, “*the thinning of the clothing conforming to the body shape is unnatural*” and “*the center*

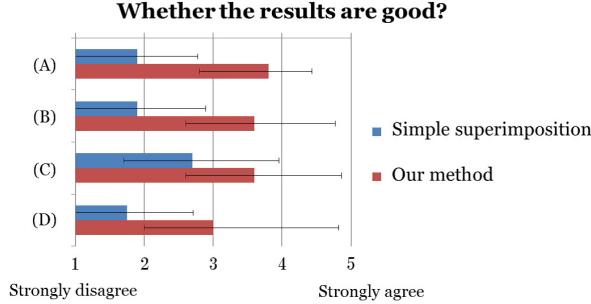


Figure 12. Result of the user test. We asked the subjects whether the virtual fitting results by our system were satisfactory or not using a five-point Likert scale. The error bars indicate standard deviations.

*lines of the body and the garment image are skew.”*

Regarding the time for modifying automatically-estimated body-contour models, test subjects took 72 seconds on average. Since this operation need only be done once for each customer, we believe the time is sufficiently short.

## V. CONCLUSIONS AND FUTURE WORK

We have introduced an image-based virtual fitting system that 1) reshapes the garment image based on wearersbody shapes, 2) automatically adjusts brightness differences between the garment image and the customer image based on facial colors, and 3) automatically retouches protrusions of customer’s cloth behind garments. For reshaping, we proposed a simple body-contour model and its heuristic estimation method. Garment images are warped so that fashion model’s body-contour model matches customer’s body-contour model. We confirmed that out virtual fitting results were sufficiently natural-looking by conducting a user test.

Future challenges include more advanced virtual try-on by fitting a parametric 3D human model [23] to the customer’s image. That will allow us handle more realistic 3D effects such as shading changes and cloth deformation.

## REFERENCES

- [1] A. Corporation, “Awaseba,” [www.awaseba.com/](http://www.awaseba.com/).
- [2] R. W. Sumner and J. Popović, “Deformation transfer for triangle meshes,” *ACM Trans. Graph. (Proc. of SIGGRAPH 2004)*, vol. 23, no. 3, pp. 399–405, Aug. 2004.
- [3] F. Cordier, W. Lee, H. Seo, and N. Magnenat-Thalmann, *From 2D Photos of Yourself to Virtual Try-on Dress on the Web*, Springer, 2001, pp. 31–46.
- [4] D. Protopsaltou, C. Luible, M. Arevalo-Poizat, and N. Magnenat-Thalmann, “A body and garment creation method for an internet based virtual fitting room,” in *Proc. Computer Graphics International 2002 (CGI ’02)*. Springer, 2002, pp. 105–122.
- [5] F. Cordier, H. Seo, and N. Magnenat-Thalmann, “Made-to-measure technologies for an online clothing store,” *IEEE Comput. Graph. Appl.*, vol. 23, no. 1, pp. 38–48, Jan. 2003.
- [6] H. Wang, F. Hecht, R. Ramamoorthi, and J. O’Brien, “Example-based wrinkle synthesis for clothing animation,” in *ACM SIGGRAPH 2010 Papers*, 2010, pp. 107:1–107:8.
- [7] R. Brouet, A. Sheffer, L. Boissieux, and M.-P. Cani, “Design preserving garment transfer,” *ACM Trans. Graph.*, vol. 31, no. 4, pp. 36:1–36:11, Jul. 2012.
- [8] P. Guan, L. Reiss, D. A. Hirshberg, A. Weiss, and M. J. Black, “DRAPE: DRessing Any PErsom,” *ACM Trans. Graph.*, vol. 31, no. 4, pp. 35:1–35:10, 2012.
- [9] W. Xu, N. Umentani, Q. Chao, J. Mao, X. Jin, and X. Tong, “Sensitivity-optimized rigging for example-based real-time clothing synthesis,” *ACM Trans. Graph. (Proc. of SIGGRAPH 2014)*, vol. 33, no. 4, Aug. 2014, to appear.
- [10] S. Hauswiesner, M. Straka, and G. Reitmayr, “Free viewpoint virtual try-on with commodity depth cameras,” in *Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry*, ser. VRCAI ’11, 2011, pp. 23–30.
- [11] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan, “Scanning 3D full human bodies using kinects,” *IEEE Transactions on Visualization and Computer Graphics (Proc. of IEEE Virtual Reality)*, vol. 18, no. 4, pp. 643–650, 2012.
- [12] J. Ehara and H. Saito, “Texture overlay for virtual clothing based on pca of silhouettes,” in *Proceedings of the 5th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ser. ISMAR ’06. IEEE Computer Society, 2006, pp. 139–142.
- [13] Z. Zhou, B. Shu, S. Zhuo, X. Deng, P. Tan, and S. Lin, “Image-based clothes animation for virtual fitting,” in *SIGGRAPH Asia 2012 Technical Briefs*, 2012, pp. 33:1–33:4.
- [14] A. Hilsmann, P. Fechteler, and P. Eisert, “Pose space image based rendering,” *Comput. Graph. Forum (Proc. of Eurographics)*, vol. 32, no. 2, pp. 265–274, 2013.
- [15] L. STARTTODAY CO., “ZOZOTOWN,” [zozo.jp/](http://zozo.jp/).
- [16] P. Guan, O. Freifeld, and M. J. Black, “A 2D human body model dressed in eigen clothing,” in *Proceedings of the 11th European Conference on Computer Vision: Part I*, 2010, pp. 285–298.
- [17] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, “SCAPE: Shape completion and animation of people,” in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 408–416.
- [18] Y. Weng, W. Xu, Y. Wu, K. Zhou, and B. Guo, “2D shape deformation using nonlinear least squares optimization,” *Vis. Comput.*, vol. 22, no. 9, pp. 653–660, 2006.
- [19] S. Bianco and R. Schettini, “Color constancy using faces,” in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 65–72.



Figure 11. Our virtual fitting results. (a) Fashion model images, (b) customer images, (c) naïve superimposition, (d) our virtual fitting results, and (e) results with protrusion retouching as well as color correction. While naïve superimposition in (c) cannot match garments to the positions of arms or legs, our system successfully reshapes garments as shown in (d) and yields more natural fitting results as in (e).

- [20] G. J. Edwards, C. J. Taylor, and T. F. Cootes, “Interpreting face images using active appearance models,” in *Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998. Proceedings.*, 1998, pp. 300–305.
- [21] J. von Kries, “Influence of adaptation on the effects produced by luminous stimuli,” in *Sources of Color Vision*, D. MacAdam, Ed., 1970, pp. 109–119.
- [22] S. Schaefer, T. McPhail, and J. Warren, “Image deformation using moving least squares,” in *ACM SIGGRAPH 2006 Papers*. New York, NY, USA: ACM, 2006, pp. 533–540.
- [23] S. Zhou, H. Fu, L. Liu, D. Cohen-Or, and X. Han, “Parametric reshaping of human bodies in images,” in *ACM SIGGRAPH 2010 Papers*, 2010, pp. 126:1–126:10.