Feature curve-net-based three-dimensional garment customization

Jituo Li¹, Guodong Lu¹, Zheng Liu¹, Jiongzhou Liu¹ and Xiaoyan Wang²

Abstract
Garment customization means that clients can provide their individual information for creating garments that fit them well. It has become a trend in garment industry. Three-dimensional (3D) methods have been emerging and have proved to be intuitive and effective ways for garment customization. In this paper, we propose a novel 3D garment customization approach based on a feature curve-net. A feature curve-net is weaved by feature curves that are initially extracted from 3D human models and then fitted onto 3D garment models. Three-dimensional garment models are locally parameterized on the feature curve-nets by applying bicubic Coons surface technology. Feature curve-nets created from different human models have the same topology connectivity. Thus, garment models on a reference human model can be transferred onto a target human model by reconstructing the garment models from the feature curve-net on the target human model. The shapes of the customized garment models can be further conveniently altered by interactively editing the feature curve-net. Our method supports both the 3D garment resizing and 3D garment editing, while most existing 3D garment customization methods can only support 3D garment resizing. Our method is flexible and can be useful in garment customization.

Keywords
three-dimensional garment customization, feature curve-net, bicubic Coons surface, local parameterization

Off-the-shelf garments are mostly designed based on standardized human models, which are based on average population sizes. However, people have a much larger variety of shapes and sizes than the standardized ones. Almost everyone has met such a problem that, when she/he enters a big shopping mall, she/he cannot find a garment fitting her/him well. Customers are willing to pay a premium price for products that fit them better¹ and they desire to have more control over the fit and design of the garments. Garment customization provides a way to satisfy such customers’ desires.

Garment customization aims at using the same production resources to design and produce a variety of similar garments, while satisfying individual demands, at a cost near that of mass production. This is customer rather than production oriented and has been seen as a trend in the garment industry. It was pointed out that there are three varieties in the garment customization, namely personalization, fit and design,² among which “fit” is considered the most critical issue.³ “Fit” means that the customers can provide their individual dimensional information for creating garments that fit them well. Our work in this paper also concentrates on garment fit.

A garment is assembled from several two-dimensional (2D) patterns. Therefore, it is straightforward to customize a garment by grading its 2D patterns. Such a strategy has been adopted in a number of software packages.⁴–⁷ Two-dimensional patterns are graded by

¹Institute of Engineering and Computer Graphics, Zhejiang University, China
²Zhejiang University of Technology, China

Corresponding author:
Jituo Li, Zhejiang University, Room 416, No. 1 Teaching Building, No. 38, Zheda Road, Hangzhou 310027, China.
Email: jituo_li@zju.edu.cn
using empirical 2D grading rules, which involves fuzzy and uncertain factors that cannot be straightforwardly transferred to a computational procedure. Before getting a fitted garment, it usually requires an experienced pattern maker to interactively alter the patterns several times, which is tedious and time consuming.

Since the ultimate goal of pattern grading is to design well-fitted individualized garments, we believe that three-dimensional (3D) customization is the most intuitive and rational approach for this goal. Very recently, a number of works have been carried out from this viewpoint. In Wang et al., for each point on a garment model, its distances and orientations relative to a set of vicinal triangles on the human model are recorded. Thus, the garment model could be updated accordingly when the shape of the human model changes. Later, the authors used radial basis functions to set up the mapping between the vicinal spaces of two humans; then, garments on one human model could be transferred onto another with this space mapping. Very recently, the authors extended the method in Want et al. to preserve the user-defined features on the garment model. In Li and Lu, to customize garments on a reference human for a target human model, the authors tetrahedralized the dressed reference human model first. Then, the tetrahedral mesh was reshaped by fitting the reference human model onto the target human model, and the customized garment model was recovered from the deformed tetrahedral mesh.

Essentially, the above 3D customization methods are based on reusing the ease allowance. Ease allowance generally means the gap or vacant space between the garment and human body. The direct reuse of ease allowance implicitly follows an assumption that for a garment with a certain style, its customization results for different humans should have the same ease-allowance distribution. However, such an assumption contradicts the common sense that obese people should have a larger ease allowance than thin ones. Unfortunately, how to adapt the ease allowance for humans in a variety of shapes and sizes is still uncertain in the garment industry. Although there exist methods for estimating or optimizing the ease allowance for different human bodies, they are mostly suitable for some certain types of garments, but hard to apply to all types of garments. Therefore, it is desirable to provide an approach supporting both the ease allowance reusing and the garment shape editing, which is also the goal of our work in this paper.

We propose a feature curve-net-based garment customization method in this paper. The feature curve-net is woven by two types of feature curves, that is, girth-wise feature (GF) curves and lengthwise feature (LF) curves, and is used as the wireframe of a garment prototype. To transfer a garment model $G$ dressed on a reference human model $H_R$ onto a target human model $H_O$, we firstly create the initial feature curve-nets $C_R$ and $C_O$ from these two human models, as shown in Figures 1(a) and (b), respectively. The initial feature curves are determined by the human models and their control points are not necessary on the garment surface. Then $C_R$ is fit onto $G$ by mapping the control points of $C_R$ onto $G$ to get its updated state $C_R'$, and $G$ is encoded on $C_R'$ by applying bicubic Coons surface technology, as shown in Figure 1(c). We copy the shape difference between $C_R'$ and $C_R$ onto $C_O$ to get its updated state $C_O'$, as shown in Figure 1(d). The customized $G$ is then reconstructed from $C_O'$, as shown in Figure 1(e). The customized garment can be further edited by adjusting the feature curve-net, as shown in Figure 1(f), which is achieved by translating the girth feature curves on the waist and hip downside first, and then editing the shape of each curve by dragging its control points inside the plane containing the curve.

Our curve-net-based approach has the following merits:

1. the garments dressed on a reference human model can be transferred onto a target human model by reconstructing the garment geometry on the feature curve-net of the target human model;
2. the garments on a dressed human model can be interactively edited by altering the feature curve-net;
3. our approach supports directly fitting 3D garment models (without reference human model) onto 3D individual human models.

Intrinsically, our work uses analytic curve and surface technologies to customize and edit the discrete polygonal mesh-represented garment models, which provide an intuitive interactive tool for customizing 3D garment models.

The rest of the paper is organized as follows. After reviewing the related work in the following section, a method for feature curve-net construction is provided in the third section. A novel method for locally parameterizing garment model on the feature curve-net is proposed in the fourth section. Experimental results and analysis are given in the fifth section. The whole paper is concluded in the final section.

Related works

In the garment industry, 3D garment models are mostly built through 2D-to-3D schemes. Pieces of 2D patterns are first designed and exported from pattern design software packages, then these patterns are sewed together to form a garment on a 3D human
model with the assistance of physics-based techniques. Generally, the 2D patterns can only be designed by experienced pattern makers. It has been shown that it takes years for a person to become an expert pattern maker. The design result can be customized by grading the 2D patterns. However, the 2D grading techniques applied in the garment industry are heavily dependent on the specific style of a garment. A general method for automatically grading 2D patterns is still lacking.

Because of the limitation of 2D computer-aided design (CAD) technologies, the garment industry has turned to look into 3D CAD technologies. Template-based methods have been proposed; in these methods, the shape of a 3D garment can be directly modified by editing its 3D feature curves. Such approaches are more intuitive than 2D-to-3D schemes. However, they are less flexible in garment design, since for every new garment style, a new garment template should be constructed, which is a heavy task for the designers. More flexible 3D garment modeling methods, such as sketching-based methods, have emerged. These methods rely on curves drawn in multi-view directions to construct the 3D garment models. Then the 3D garments can be trimmed and flattened into planar patterns. Unfortunately, although sketch-based methods are efficient in the modeling of simple garments, they become too complex to create complicated garment models.

Although 3D technology for garment modeling has been rapidly developed, the 2D-to-3D technology remains the most powerful tool for 3D garment modeling. However, in 2D-to-3D schemes, a 3D garment is usually composed of a number of discrete patches, which makes it very difficult to interactively modify the 3D garments and keep the connectivity between the patches in interactive editing. Our approach customizes and edits the 3D garments through analytic and parametric curve and surface technologies; thus, the garment shape could be intuitively and interactively edited, and the connectivity between different patches of the garment could be effectively preserved.

The strategy of our approach is inspired by multi-resolution techniques in which a model is decomposed into a smooth base mesh and several levels of geometric details. The deformed shape is obtained by adding back the geometric details onto the modified base mesh. Our approach distinguishes the general multi-resolution techniques in model editing. In most multi-resolution methods, the base mesh is created by simplifying the fine mesh, and model editing methods are discrete mesh based. In our approach, a parametric

![Figure 1. Process of three-dimensional garment customization: (a) initialize the feature curve-net on the dressed reference human model; (b) initialize the feature curve-net on the target human model; (c) fit the feature curve-net onto the garment on the reference human model; (d) transfer the shape difference between the initial feature curve-net and the fitted feature curve-net of the dressed reference human model onto the target human model; (e) reconstruct the garment model by the feature curve-net on the target human model; (f) interactively edit the customized garment by adjusting the feature curve-net on the target human model.](image-url)
surface spanned by a feature curve-net is used as the base model. Thus, our approach is based on analytic and continuous surface technology, which is more flexible in interactive model editing.

**Feature curve-net**

Although the styles and shapes of garments for a human model are unlimited, the garment prototype can be simply regarded as a surface that has similar topology and geometry of the human model. Thus, the feature curves on the garment prototype can be created by altering the feature curves on the human model. These feature curves weave a feature curve-net.

As shown in Figure 2(a), GF curves are created by intersecting a number of planes against the human model along the human skeleton; the minimal convex hull is computed from the intersecting curves. The minimal convex hulls are parametrically fit by using cubic Spline technology, and are used as GF curves. The human skeleton can be extracted from a human surface model automatically. Each GF curve reflects the cross-section shape of the human model on a certain part. To make the correspondence among neighboring GF curves, the control points on each GF curve are positioned properly, similarly to the method in Li and Wang. Firstly, each minimal convex hull is partitioned with two orthogonal lines to get four intersecting points, which are shown in red in Figure 2(a). These four intersection points partition the minimal convex hull into four sub-curves. Secondly, more control points are extracted out equidistantly on each sub-curve, as shown in green points in Figure 2(a). The LF curves are created by linking the corresponding control points on neighboring GF curves. The LF curves reflect the human body shape change along the human skeleton.

The feature curves of the garment prototype are created by altering the feature curves on the human model based on the garment design rules. A garment surface

---

**Figure 2.** Feature curve-net: (a) initialize the feature curve-net on a human model; (b) initialize the feature curve-net of a garment prototype; (c) initialize the garment on the human model; (d) segment the dressed human model into several semantic parts; (e) fit the feature curve-net onto the garment model; (f) skin the feature curve-net to get a garment prototype; (g) triangulate the garment prototype; (h) encode the garment model by the feature curve-net. (Color online only.)
can be partitioned into two types of regions: loading regions, which are defined as being in direct contact with the body surface and designed to counter the gravity of the garment, and fashion regions, which are usually not in direct contact with the body but are draped freely to enhance aesthetic appearance. The loading regions are distributed on the upper parts of the arms, shoulders and chest. The feature curves on these regions are created by offsetting the corresponding human feature curves with a small distance. The feature curves on the fashion regions are created by simulating the draping phenomenon, for example, the GF curves below the chest are initialized by draping the GF curve from the chest with collision detection against the human model on a certain body part (e.g. the waist and hip) to ensure the GF curves are always outside the human model. The GF curve on the waist can be further shrunk to get a tunic prototype, as shown in Figure 2(b). Because the feature curves on the loading regions are close to the human surface, to reduce the interpenetration between the garment prototype and human model, the feature curves in the loading regions are distributed more densely than in the fashion regions, as shown in Figure 2(b). In experiments, we found 44 feature curves are enough for customizing an upper garment, among which 24 curves are GF curves, as shown in Figure 2(b).

The feature curve-nets created from different human models have the same connectivity topology, which can be used to set up the spatial mapping between feature curve-nets for different human models. For a certain garment dressed on a human model, as shown in Figure 2(c), we fit the initial feature curve-net onto the garment model by repositioning the control points on GF curves, as shown in Figure 2(e). As each GF curve is a planar curve, for a GF curve $P(u)$, the normal on a control point $P(\hat{u}) \mathbf{n}_u$ can be computed as

\[ \mathbf{n}_u = \frac{\partial P(\hat{u})}{\partial u} \times \mathbf{N} \]

where $\mathbf{N}$ is the normal of the plane containing $P(\hat{u})$ and $\times$ is the cross product between two vectors. We use a ray from point $P(\hat{u})$ in the direction of $\mathbf{n}_u$ to intersect the garment model, and use the nearest intersection point to reposition the control point. To avoid unreasonable mapping, such as fitting the chest feature curve onto the sleeve, before repositioning, we employ the skeleton-based garment model segmentation method in Li and Lu [12] to segment the garment model into several semantic parts. This segmentation method achieves the segmentation in an iterative optimization process and can robustly tackle various types of garments. The segmentation result is shown in Figure 2(d), where each part is rendered in a certain color. Thus, feature curves on a certain part would only fit onto the corresponding part on the garment model.

Feature curve-net-based garment parameterization

The feature curve-net has a quadrilateral mesh structure. Each quadrilateral element is formed by two neighboring GF curves and two neighboring LF curves, and can be transformed into a 3D patch by applying parametric surface technologies. Thus, we can use the feature curve-net to span a parametric surface-represented garment prototype. The garment prototype roughly represents the general shape of the garment model. We further locally parameterize the geometric details of the garment model on the garment prototype. Thus, we can customize and alter the garment model by interactively editing the feature curves.

Garment prototype spanned by a feature curve-net

Coons, Bézier and NURBS are the most popular parametric surface technologies, each of which can be used to skin the quadrilateral elements. Among these parametric surfaces, a Coons surface is strictly interpolated on its boundary control curves, and its shape can be controlled just with its boundary control curves, that is, the four boundaries of a quadrilateral element in our case, which is very suitable for our application. To keep the $C^1$ continuity between the neighboring patches, we apply a bicubic Coons surface to skin the quadrilateral elements. As shown in Figure 2(f), the patches are rendered in different colors. The combination of these patches forms a garment prototype with $C^1$ continuity, as shown in Figure 2(g).

For each quadrilateral element, we assume that its four boundary curves are parametrically represented as $P(u,0)$, $P(u,1)$, $P(0,v)$, $P(1,v)$ with $u, v \in [0, 1]$, as shown in Figure 3.
in Figure 3, then the corresponding 3D patch can be parametrically represented as

$$\mathbf{P}(u, v) = \mathbf{F}(u)^T \mathbf{B} \mathbf{F}(v) \quad u, v \in [0, 1]$$

(1)

where

$$\mathbf{F}(u) = \begin{bmatrix} 2u^3 - 3u^2 + 1 \\ -2u^3 + 3u^2 \\ u^3 - 2u^2 + u \end{bmatrix}, \quad \mathbf{F}(v) = \begin{bmatrix} 2v^3 - 3v^2 + 1 \\ -2v^3 + 3v^2 \\ v^3 - 2v^2 + v \end{bmatrix}$$

and \( \mathbf{F}(u) \) are vectors containing Hermite blending functions. In matrix \( \mathbf{B} \), the sub-matrices

$$\begin{bmatrix} \mathbf{P}_u(0, 0) & \mathbf{P}_u(0, 1) \\ \mathbf{P}_u(1, 0) & \mathbf{P}_u(1, 1) \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathbf{P}_v(0, 0) & \mathbf{P}_v(0, 1) \\ \mathbf{P}_v(1, 0) & \mathbf{P}_v(1, 1) \end{bmatrix}$$

represent the coordinates and the partial first derivatives along the \( u \) direction and \( v \) direction on the four corners of the quadrilateral element, respectively; sub-matrix

$$\begin{bmatrix} \mathbf{P}_{uv}(0, 0) & \mathbf{P}_{uv}(0, 1) \\ \mathbf{P}_{uv}(1, 0) & \mathbf{P}_{uv}(1, 1) \end{bmatrix}$$

denotes the blending partial derivatives along the \( u \) and \( v \) directions, which are also known as twist vectors. The former three sub-matrices can be easily determined by the four parametric curve boundaries of the quadrilateral element. To accurately compute the twist vectors on the four corners, the parametric surface around the four corners should be given, which, however, is to be determined by Equation (1). To jump out of this endless loop, we use the twist vectors of the bilinear Coons surface to approximate the twist vectors on the four corners, as follows:

$$\mathbf{P}_{uv}(u, v) = -\mathbf{P}_u(0, v) + \mathbf{P}_v(1, v) - \mathbf{P}_u(0, u) + \mathbf{P}_u(u, 1)$$

$$- \mathbf{C}, \quad \mathbf{C} = \mathbf{P}(0, 0) - \mathbf{P}(0, 1) - \mathbf{P}(1, 0) + \mathbf{P}(1, 1)$$

(2)

With Equations (1) and (2), the 3D patch for each quadrilateral element can be determined by the coordinates and the first derivatives on its four corners, which can be easily calculated.

Encoding/decoding garment geometry by a feature curve-net

Encoding with local coordinate systems. For a point on patch \( \mathbf{P}(u, v) \) with its parameter coordinate as \((\tilde{u}, \tilde{v})\), its tangent vectors in \( u \) and \( v \) directions can be respectively computed as \( \partial \mathbf{P}(u, v)/\partial u \bigg|_{\tilde{u}=\tilde{v}} \) and \( \partial \mathbf{P}(u, v)/\partial v \bigg|_{\tilde{u}=\tilde{v}} \).

These tangent vectors form a local coordinate system, with its three orthogonal axes \( \xi, \psi, \zeta \) being

$$\xi = \frac{\partial \mathbf{P}(u, v)/\partial u}{||\partial \mathbf{P}(u, v)/\partial u||_{\tilde{u}=\tilde{v}}}, \quad \psi = \frac{\partial \mathbf{P}(u, v)/\partial v}{||\partial \mathbf{P}(u, v)/\partial v||_{\tilde{u}=\tilde{v}}}, \quad \zeta = \frac{\mathbf{P}(u, v) \times \partial \mathbf{P}(u, v)/\partial v}{||\mathbf{P}(u, v) \times \partial \mathbf{P}(u, v)/\partial v||_{\tilde{u}=\tilde{v}}}$$

(3)

where the operator “\( \times \)” is the cross product between two vectors.

Let \( \mathbf{q}(x, y, z) \) be a point on the garment model near the point \( \mathbf{P}(\tilde{u}, \tilde{v}) \) on the garment prototype, then \( \mathbf{q} \) can be locally parameterized as

$$\mathbf{q} = \mathbf{P}(\tilde{u}, \tilde{v}) + \alpha \mathbf{\xi} + \beta \mathbf{\psi} + \gamma \mathbf{\zeta}$$

(4)

where \((\alpha, \beta, \gamma)\) is the local coordinate of \( \mathbf{q} \) under the local coordinate system \((\tilde{\xi}, \tilde{\psi}, \tilde{\zeta})\) centered on \( \mathbf{P}(\tilde{u}, \tilde{v}) \), and \( \alpha, \beta, \gamma \) are calculated as \( \alpha = (\mathbf{q} - \mathbf{P}(\tilde{u}, \tilde{v})) \cdot \mathbf{\xi} \), \( \beta = (\mathbf{q} - \mathbf{P}(\tilde{u}, \tilde{v})) \cdot \mathbf{\psi} \), and \( \gamma = (\mathbf{q} - \mathbf{P}(\tilde{u}, \tilde{v})) \cdot \mathbf{\zeta} \), respectively. We name the processing of parameterizing each vertex on the garment model by the feature curve-net garment encoding.

Definition of the ideal local coordinate centers. Ideally, the difference between \( \mathbf{q} \) and \( \mathbf{P}(\tilde{u}, \tilde{v}) \) should as small as possible, that is, the parameters \( \alpha, \beta, \gamma \) should be as small as possible, otherwise undesirable distortion could be caused in garment customization. We use Figures 4(a) and (b) to explain the influence of local coordinates on the garment customization results. In the figures, the local coordinate systems are centered on different positions. In each figure, \( \mathbf{\tilde{q}} = \mathbf{P}(\tilde{u}, \tilde{v}) + \alpha \mathbf{\xi} + \beta \mathbf{\psi} + \gamma \mathbf{\zeta} \) is the projective point of garment point \( \mathbf{q} \) on the tangent plane on patch point \( \mathbf{P}(\tilde{u}, \tilde{v}) \). \( \Delta ABC \) is a triangle inside the tangent plane, and the projective point \( \mathbf{\tilde{q}} \) is on edge \( AC \). Without losing the generality, supposing that patch \( \mathbf{P}(u, v) \) is compressed and \( \Delta ABC \) is deformed into \( \Delta A' B' C' \), as shown in the bottom images in Figures 4(a) and (b), then by using Equation (4), \( \mathbf{\tilde{q}} \) is updated to \( \mathbf{\tilde{q}}' \), which deviates from \( A' C' \) where \( \mathbf{\tilde{q}}' \) should be ideally located. When \( \mathbf{P}(\tilde{u}, \tilde{v}) \) and \( \mathbf{\tilde{q}}' \) become closer, that is, the parameters \( \alpha \) and \( \beta \) become smaller, the derivation would be reduced, as the comparison between Figures 4(a) and (b) shows. Small \( \alpha \) and \( \beta \) mean that \( \mathbf{q} - \mathbf{P}(\tilde{u}, \tilde{v}) \) should be nearly parallel with the surface normal on \( \mathbf{P}(\tilde{u}, \tilde{v}) \), which guides the processing of finding out a proper \( \mathbf{P}(\tilde{u}, \tilde{v}) \) for \( \mathbf{q} \).
Searching the ideal local coordinate centers. As a preprocessing of the garment encoding, the garment prototype is firstly discretized into a triangular mesh, as shown in Figure 2(g). We assume that the garment prototype $G_p$ is composed with a series of 3D patches $\{P_k(u,v)\}$, $k \in [1,n]$, with $n$ being the number of 3D patches. For each patch $P_k(u,v)$, we evenly sample $l$ and $m$ points along $u$ and $v$ axes. Then $P_k(u,v)$ is triangulated into $2(l-1)(m-1)$ triangles. In our system, $l$ and $m$ are both set to be 5. Figure 5 shows a certain triangulated patch on $G_p$.

For each vertex $q_i$ on the garment model, its proper local coordinate center on $G_p$, $P_k(\tilde{u}_i, \tilde{v}_i)$ is set with the expectation that $q_i - P_k(\tilde{u}_i, \tilde{v}_i)$ is nearly parallel with the surface normal on $P_k(\tilde{u}_i, \tilde{v}_i)$ and $||q_i - P_k(\tilde{u}_i, \tilde{v}_i)||$ should be as short as possible, where $(\tilde{u}_i, \tilde{v}_i)$ is the parametric coordinate of the proper local coordinate center on patch $P_k$. $P_k(\tilde{u}_i, \tilde{v}_i)$ is obtained and optimized in an iterative way, as follows.

**Step 1.** A line paralleling with $q_i$’s normal, $N_{q_i}$, is used to intersect with the triangulated $G_p$. The intersection $P_0^i$ with its normal $N_{P_0^i}$ is served as the initial local coordinate system center of $q_i$, as shown in Figure 5.

**Step 2.** Let $W = P_0^i - (q_i - P_0^i) \cdot N_{P_0^i}$. A line passing through $W$ and paralleling with $N_{P_0^i}$ is used to intersect with the triangulated $G_p$, to get an intersection point $P_1^i$, as shown in Figure 5. Let the surface normal on $P_1^i$ be $N_{P_1^i}$; if $P_1^i$ meets the following conditions, $P_1^i$ is regarded as more suitable to be the local coordinate center for $q_i$ than $P_0^i$; otherwise, let $P_1^i = P_0^i$ and stop the iteration:

$$\left( N_{P_0^i}(q_i - P_0^i) > (N_{P_1^i}(q_i - P_1^i)) \right) \& \left( ||q_i - P_1^i|| > ||q_i - P_0^i|| \right)$$
where \((A^\top B)\) is the angle between vector \(A\) and vector \(B\).

**Step 3.** Set \(P_i^j\) and \(N_i^j\) to be \(P_i^j\) and \(N_i^j\), respectively. Go to Step 1 until \((N_i^j, N_i^j) < \varepsilon \). \(\varepsilon\) is a threshold, in our system \(\varepsilon = 2^5\).

**Parameters of encoding/decoding.** Without loss of generality, we assume that \(P_i^j\) is inside the \(k\)th patch and enclosed by a triangle \((P_k(u_j, v_j), P_k(u_{j+1}, v_{j+1}), P_k(u_{j+2}, v_{j+2}))\). Let \((\sigma, \varsigma, \tau)\) be the barycentric coordinate of \(P_i^j\) in the triangle, then the parameters \((u_j, v_j)\) of \(P_i^j\) on the patch can be computed as

\[
u_i = \sigma u_j + \varsigma u_{j+1} + \tau u_{j+2}, \quad v_i = \sigma v_j + \varsigma v_{j+1} + \tau v_{j+2},
\]

\[\quad \text{with } \sigma + \varsigma + \tau = 1\]

(5)

Thus, each \(q_i \in G\) can be encoded on the curve-net with a triple-component:

\[
<k_i, (u_i, v_i), (\alpha_i, \beta_i, \gamma_i)>
\]

where \(k_i\) is the id of the patch relative to \(q_i\), \((u_i, v_i)\) is the parametric coordinate of \(q_i\)’s local coordinate system center in the patch and \((\alpha_i, \beta_i, \gamma_i)\) is the local coordinate of \(q_i\). When the feature curve-net is modified, the new position of \(q_i\) can be decoded by Equation (4) with the local coordinate system updated from Equation (3). In the decoding process, the surface of the garment prototype is no longer needed. Thus, the garment prototype is only computed once just for garment encoding.

Figure 2(h) shows the encoding result of a coat. In the figure, the garment is displayed by several regions, each of which is encoded on a certain garment prototype patch that has the same color as the region, as shown in Figure 2(f).

**Comparison between our method and others**

Local parameterization techniques have also been adopted in existing 3D garment customization methods. In Wang et al., the garment geometry is encoded on the local human surface. As shown in Figure 4(c), if we suppose that \(\Delta A B C\) is a triangle on the human model, which is near to a garment point \(q\), then \(q\) is locally parameterized by a local coordinate system on the triangle center \(O\), with its three axes \(\xi, \psi, \zeta\) being

\[
\xi = \frac{A - O}{||A - O||}, \quad \psi = \frac{\zeta \times \xi}{||\zeta \times \xi||}, \quad \zeta = \frac{N_O}{||N_O||}
\]

where \(N_O\) is the normal of the triangle. For clarity, without losing generality, suppose that the projective point of \(q\) on the \(\xi \psi\) plane, \(\bar{q}\), coincides with the midpoint of \(AC\), and \(\Delta A B C\) is compressed into \(\Delta A'B'C'\). Subsequently, \(\xi, \psi, \zeta\) is updated to \(\xi', \psi', \zeta'\). An orientation angle, \(\delta\), exists between the two local coordinate systems \((\xi, \psi, \zeta)\) and \((\xi', \psi', \zeta')\), as shown in Figure 4(c). As a result, the projective point recovered from the updated local coordinate system, \(q'\), derivates from the ideal position, \(\bar{q}'\). Such a derivation could cause significant ramps and bumps, as pointed out by Wang et al., to blur such undesirable effects, each garment point is parameterized and weighted by 1% of the total number of triangles of a human model; for example, to encode a garment model on a human model with 10,000 triangles, for each garment point, 100 proper triangles on the human model are required, which increases the algorithm complexity. In contrast, in our approach, we use a \(C^1\) Coons surface to define the local coordinate systems. The orientations of the local coordinate systems thus change smoothly throughout garment prototyping, as described above; for each garment point, a triple-component is enough for local parameterization.

**Experimental results and analysis**

Figure 1 shows an example of customizing a garment model on a reference human model to fit a target human. In altering the customized garment model, as shown in Figure 1(f), two kinds of interactive operations are provided for editing the curve-net. One is translating a whole GF curve along the human skeleton; the other is editing the shape of GF curves by dragging their control points inside the plane containing the curve. Because the LF curves share the same set of control points with the GF curves, when a GF curve is altered, its connected LF curves are modified simultaneously. To preserve the local shape details of the garment, we use the Laplacian mesh editing method to filter the garment-altering result. In filtering, we fix the boundary vertices and their one-ring neighbors to keep the continuity between the neighboring patches.

Our approach also can customize garment models without the reference human model, which, however, is necessary in most of the existing garment customization methods. As shown in Figure 6, a client whose human model is shown in Figure 6(a) favors a suit of garments with a certain style as shown in Figure 6(b). The suit of garments is composed of an upper garment, a shirt and a necktie. For the upper garment, its body, two pockets and fasteners are represented by independent patches that are rendered in different colors in the figure for clarity. However, the current sizes of garments are not suitable for the client. To fit the garments onto the client, the pose of the human model and the shapes of the garment models have to be altered. We employ the skeleton-based deformation method to adjust the human pose, so as to align the human model with the garments, as shown in Figure 6(d).
Interpenetration between the human model and the garment models unavoidably occurs in this process. This interpenetration can be rectified with our feature curve-net-based approach. Figure 6(e) shows the initial feature curve-net of the garment models, which, however, does not well represent the general shape of the garment models and is further interactively edited, as shown in Figure 6(f). Then the garment models are locally parameterized on the feature curve-net, as shown in Figure 6(g). The garment model could be

![Figure 6](https://example.com/figure6.png)

**Figure 6.** Dressing the human bodies: (a) initialize the human model; (b) initialize the suit of garments whose initial sizes are not suitable for the human model; (c) adjust the human model pose; (d) align the human model with the garment models; (e) initialize the feature curve-net; (f) alter the feature curve-net; (g) encode the garment models; (h) customization result; (i) human bodies in a variety of shapes and sizes; (j) customization results on different human models.
further altered by interactively editing the feature curve-net to well fit onto the human model, as shown in Figure 6(h). Regarding the dressed human model shown in Figure 6(h) as a reference dressed human model, the garments can be further customized for human bodies in a variety of shapes and sizes, as shown in Figure 6(i). The customized results are shown in Figure 6(j).

In the above process, because we use the $C^1$ Coons surface technology to encode/decode the garment models, the spatial relationship between different garment models and the continuity between discrete patches on the same garment model can be well preserved, as shown in Figures 6(h) and (j).

Figure 7 shows another garment customization example. Figures 7(b) and (c) show the customization results. Three-dimensional patterns can be designed by directly drawing cutting lines on the garment surface, and then they are flattened to get 2D patterns, as shown in Figure 7(d) where the curves on the garment surface are cutting lines, and the 2D polygons are 2D patterns. The customization result can be further stylized by surface trimming with cutting lines, as shown in Figures 7(f) and (g). Figures 7(e) and (h) show the texture mapping on the customized garment models.

We have implemented our approach using Visual C++ on a computer with an Intel(R) Core(TM)2 Quad central processing unit (CPU) Q8300 @ 2.50 GHz and with 3.00 G random-access memory (RAM). The human model in Figure 1(a) contains 38,564 triangles and 19,284 vertices. The human models in Figures 1(d) and 6 are created by using the method in Hasler et al., and each model contains 12,894 triangles and 6449 vertices. The garment models in Figures 1 and 6 contain 3953 triangles, 2060 vertices and 4985 triangles, 2631 vertices, respectively. In our experiments, initializing each feature curve-net takes under 3 s, including automatically creating the human skeleton. Fitting the feature curve-net and encoding the garment models on the feature curve-net requires less than 2 s. It takes less than 0.02 s to update the garment models in each interactive alteration of the feature curve-net. Thus, our approach...
can effectively support the real time interactive editing, that is, the garment shapes can be simultaneously updated in editing the feature curve-nets, which provides a convenient way to edit the garment models.

Conclusions

How to customize existing garment design results for various human bodies is a key problem in the clothing industry. The emerging 3D garment customization technology has been seen as a potentially effective way for the reuse of garment design results. Existing 3D garment customization methods mostly concentrate on resizing the garment model by transferring the spatial garment–human relationship of a dressed reference human onto various target human models. Little can be done by the user in this process, even if the customized result is not satisfactory. In this paper, we propose a novel 3D garment customization method that supports both garment resizing and interactive editing. Our approach is based on feature curve-net, which is used as the wire frame of the garment model. The garment model can be encoded, decoded and edited via the feature curve-net by using the bicubic Coons surface technology. Our approach is user oriented and flexible in garment customization.

Our approach provides a framework of feature curve-net-based garment customization. Currently, the garment prototype spanned by the feature curve-net is relatively simple, for example, there is no special geometric information for the collar. Thus, the current version can only customize the general shape of the garment models. To customize more detailed geometries of the garment models, for example, the shape of a collar, the feature curve-net should be refined. Moreover, in this paper, we only give examples of

Figure 7. Getting two-dimensional (2D) patterns from three-dimensional (3D) customized garments: (a) skirt on the reference human model; (b) customization result A; (c) customization result B; (d) getting 2D patterns (shown in 2D polygons) on customization result A; (e) texture mapping on customization result A; (f) customization result B is further stylized by 3D surface trimming – the green regions are to be trimmed (color online only); (g) getting 2D patterns on the stylized result; (h) texture mapping on the stylized result.
upper garment models and skirts. However, our approach could also be used for customizing trousers, as long as the feature curve-net for trousers were provided. These developments are to be carried out in future work.

Our approach is geometry based, which is efficient in interactive garment customization. However, the customization results may not obey the rules of physics; for example, the shapes of the same garment on various body shapes demonstrate the same drapes/wrinkles, which is not true in the real world. This problem can be solved by applying physics-based garment draping on the customization results, which would be carried out in our future work.

**Funding**

This work was supported by the National Natural Science Foundation of China (No. 60903145, 61070105), the Beijing Natural Science Foundation (No. 4102062), the Fundamental Research Funds for the Central Universities (2012QNA4003), the Doctoral Program of the Ministry of Education of China (2010011110025) and the Research Project of Department of Education of Zhejiang Province (Y201018160).

**References**