A Product Form Innovation Method Based on 3D Morphing for Industrial Designer

Chunfa Sha¹, Dongmin Wu², Rui Li¹

¹ Graphics Technology Institute, Jiangsu University, Zhenjiang 212013, China
² Dean’s Office, Chongqing Normal University, Chongqing 401331, China.

Abstract

3D morphing methods can create many new shapes that merge characteristics of the two given models. These new shapes could be used to stimulate designers’ inspiration or to assist in creative design for product form. For industrial designers who have no special knowledge about morphing technologies, a 3D morphing method needing no complex manual interventions is proposed. Firstly, two 3D product shapes stored as stereo lithography (STL) files are embedded onto the unit sphere based on the spherical harmonic map. Secondly, feature edges are extracted with the analysis of dihedral angles. Thirdly, in order to determine the spatial relationships of the two given products, the two embedding results and also the feature edges are merged based on triangular decomposition. Lastly, new product shapes are generated with linear interpolation. Ten industrial designers were asked to use the prototype system of this method to do two morphing tasks and to evaluate the usability of the system. The results show that they are satisfied with the method, as the system can generate many new, interesting and useful shapes. The operation is easy to perform, and the time spent on any particular task is acceptable.

Keywords: 3D morphing, visual feature, product form innovation, industrial designer, usability

1. INTRODUCTION

Computer aided design is very important for many kinds of designers, such as motion desinger (Poredos et al., 2015), product designer (Guoet al., 2016), industrial designer (He and Ji, 2016), struct desiner (Moregaet al., 2016) and evennumerical simulation (Li et al., 2015; Rafiee and Sadeghiazad, 2016). For industrial designer, computer aided design is a neckesray tool for innovation. But, traditional computer aided industrial designsoftwareare just tools for industrial designers to present their ideas during the process of product form design, similar to pens or rulers. The purpose of this software, such as Rhino, 3DS Max, Alias, etc., is to improve the design efficiency and decrease the design difficulty for visual expression, using methods such as modeling and rendering. But there are no functions that can assist the designer in the creative aspect of design for product form. 3D morphingis a technique that merges the characteristics of the given models. It creates a large number of new shapes which retain the visual features of both the initial object and the target object. These new intermediate objects can be used as references to inspire designers’ inspiration or to generate new shapes directly for product form innovation. This technology has been applied in the field of film, television animation and advertising, as well as in the field of product form design.

Different from traditional computer aided industrial design software, the core objective of these methods is innovation, and not simply to present designers’ ideas in the way a pen does. New shapes are generated with a feature-based morphing method and the images of the morphed shapes are analyzed using a modified gray theory with Fourier residual
correction, which helps speed up the process of designing the required product form (Hsiao and Liu, 2002). The control points of local feature lines are matched to obtain new product shapes (Yuan and Xue, 2011). The surface features with interesting semantic of a car are used to make new car shapes by surface deformation (Cheng and Liu, 2012 and 2014). New shapes are created using a method of morphing freeform surfaces that requires manual interventions (Sanchez and Fryazinov, 2013; Yang and Jüttler, 2007). The spherical harmonic of triangulated models are found to obtain the topology of the intermediate objects for product form design (Su and Wu, 2014).

But for practical application, something more must be done because industrial designers have particular requirements that are different from other designers such as animators. Firstly, shape morphing precision is not the highest priority, but the results should retain the visual features of the given models to preserve the visual effect. Secondly, the operation should be easily performed, so complex manual interventions that require professional knowledge of 3D morphing are unfavorable. Thirdly, the size of new shapes should be suitable for further operations, such as rendering. Lastly, time spent on a morphing task should be within the range of the designers’ limits. For the purpose, we have already proposed a 3D morphing technique based on triangular decomposition and made some application examples (Sha and Lu, 2009, 2015). But some important shape features are lost or some complex manual interventions are still needed in this method, which are not suitable for real application. In this paper, the method is improved and two morphing tasks are conducted to test the usability of its prototype system.

2. METHOD DESCRIPTION

Similar to point cloud and freeform surfaces, triangular mesh is a vital data format to express 3D models. It is widely used in design, manufacturing, entertainment and other industries. Therefore, many experts have attempted to morph triangular mesh models stored as STL files. Similar methods were proposed to morph those models with centroid inside them (Kent, 1992; Alexa and Carlson, 2000). They mapped the models onto the unit sphere and then merged the mapping to obtain the topology containing the relationship of the two shapes. Methods requiring manual intervention are always a good choice to improve the morphing precision (Lee and Huang, 2003; Gregory and State, 1999). In order to align the features, it asks users to specify the feature points on the two given shapes manually (Alexa and Carlson, 2000). In the following, the improved method, needing no complex manual interventions, is shown in detail.

2.1 Key steps of the method

1) Embed the models onto the unit sphere. Using an aspherical harmonic map, the two given triangular mesh models (the original model S and the target model T) are embedded onto the unit sphere to get two models (S' and T').

2) Merge vertices. Vertices of T' are mapped onto S' and triangular decomposition is done on the related triangle to get the preliminary intermediate topology model M'.

3) Merge feature edges. Firstly, the feature edges of the target model T are obtained based on the calculation of dihedral angles, and the corresponding edges in T' are signed as E’ in order. Secondly, all the edges of E’ are mapped onto M' and triangular decomposition is done on the related triangles to update M'. At Step 2) and Step 3), the spatial relationships and coordinates of the vertices between M' and S, T are recorded.

4) Interpolate models. New shapes M are obtained based on linear interpolation.

2.2 Embedding the model onto the unit sphere
According to a certain order of the vertices in the model, the models were simplified into a tetrahedral by removing vertices gradually and then adding the corresponding vertices back to the model in the reverse order to arrive at a zero genus polyhedron model (Shapiro and Ayellet, 1998). Then, with the projection method proposed by Kent, the model could be projected onto the unit sphere. Alexausa a relaxation iteration method to deal with the self-intersections that maybe made in the projection step, but this method is time consuming.

In order to make the operation easier and to improve the usability of the technique, the spherical harmonic map was used, done in polar coordinates, in order to parameterize the surface onto the unit sphere (Brechbühler and Gerig, 1995). The two polar coordinates θ and φ are determined for all vertices of the two given models.

1) Set poles. Two vertices with maximal and minimal z coordinates in object space are selected as the north pole and the south pole respectively, and also as the first and the last vertices.

2) Calculate latitude. Latitude θ always grows smoothly from 0 at the north pole to π at the south pole, where θ is not a free variable but an unknown function. With Dirichlet conditions θnorth = 0 and θsouth = π, a corresponding continuous problem as Laplace’s equation ∇²θ = 0 (except at the poles) is formulated to determine the latitude value. This means that, in the discrete case, every node’s latitude is equal to the average of its adjacent nodes’ latitudes, except the two poles. These conditions can be expressed as linear equations Aθ = b, where A is a n x n matrix (n is the number of the nodes), θ = (θ₀, θ₁, ..., θₙ₋₁)ᵀ, b is a n constant. The border conditions create two equations θ₀ = θnorth and θₙ₋₁ = θsouth. This results in a new n x n system Aθ’ = b’, where A’ = (a₁, a₂, ..., aₙ₋₁), θ’ = (θ₁, ..., θₙ₋₁)ᵀ. And n’ = n - 2.

3) Calculate longitude. Longitude φ is a cyclic parameter increasing monotonically from 0 to 2π. The global longitude parameter runs from pole to pole as a discontinuous line with a total height of 2π. Similar to latitude, the cyclic Laplace’s equation ∇²φ = 0 (except at the poles which can be defined as 0) is formulated. Due to the cyclic boundary conditions, any φ₀ = 0 is set to be added to any row of the system.

4) Embed surface. For every node v of the surface, the corresponding latitude θ and longitude φ are obtained to define its unique mapping to the surface of a sphere, as shown in Eq. (1).

\[ v(θ, φ) = (x(θ, φ), y(θ, φ), z(θ, φ))ᵀ \]  

Where θ ∈ [0, π], φ ∈ [0, 2π] and \( x^2(θ, φ) + y^2(θ, φ) + z^2(θ, φ) = 1 \).

2.3 Merging vertices

1) Calculate vertices’ corresponding mapping positions

As shown in Fig.1, an assumption is made that a triangular piece \( F_{S_j} \) of \( S \) is related with a vertex \( v_{T_j} \) of \( T \). Then, a vector \( o \rightarrow v_{T_j} \) can be constructed with the center-point o and point \( v_{T_j} \) which is the embedding of \( v_{T_j} \), and three triangular pieces \( Δv_{S_1}oΔv_{S_2}Δv_{S_2}oΔv_{S_3} \) can be constructed with o and three edges \( oF_{S_j} \) which is the embedding of \( oF_{S_j} \). This can be expressed as Eq. (2).
\[
\begin{align*}
\alpha &= (\mathbf{o\nu'}_{S1} \times \mathbf{o\nu'}_{S2}) \cdot \mathbf{o\nu'}_{Ti} \\
\beta &= (\mathbf{o\nu'}_{S2} \times \mathbf{o\nu'}_{S3}) \cdot \mathbf{o\nu'}_{Ti} \\
\gamma &= (\mathbf{o\nu'}_{S3} \times \mathbf{o\nu'}_{S1}) \cdot \mathbf{o\nu'}_{Ti}
\end{align*}
\] (2)

Eq. (2) and Eq. (3) must be true. In other words, only if all the angles between \( o\nu_{Ti} \) and the three triangular pieces are less than or equal to 90 degrees (using the right-hand rule), the aboved mentioned assumption holds.

\[
\begin{align*}
0 &\leq \alpha \leq 1 \\
0 &\leq \beta \leq 1 \\
0 &\leq \gamma \leq 1
\end{align*}
\] (3)

As in Fig.1, \( p \) is set as the mapping of \( \mathbf{v}_{Ti} \) onto \( F_{ij} \). So Eq. (4) is formulated to obtain three parameters \( k_1, k_2, k_3 \). There are three possible positions in \( F_{ij} \) for point \( p \): \( p \) is an interior point as shown in Fig.1(a) when \( 0 < \alpha, \beta, \gamma < 1 \); \( p \) is located on one edge, such as the edge \( \mathbf{v}_{S2} \mathbf{v}_{S3} \) as in Fig.1 (b), when \( \beta = 0 \) and \( k_1 = 0 \); \( p \) is located on one vertex, for example the vertex \( \mathbf{v}_{S2} \) as in Fig.1 (c), when \( \alpha = \beta = 0 \) and \( k_1 = k_3 = 0 \).

\[
\begin{align*}
\begin{cases}
p = o + k \mathbf{o\nu}_{Ti} \\
p = k_1 \mathbf{v}'_{S1} + k_2 \mathbf{v}'_{S2} + k_3 \mathbf{v}'_{S3} \\
(p \mathbf{v}'_{S1} \times p \mathbf{v}'_{S2}) \cdot (\mathbf{v}'_{S3} \times \mathbf{v}'_{S2}) = 0
\end{cases}
\end{align*}
\] (4)

For every point \( \mathbf{v}_{T_j} \) off', its corresponding position \( \mathbf{v}_{T_j}' \) in \( F_{T_j} \) can be calculated by Eq. (5) with the three parameters. In the same way, for every point \( \mathbf{v}_{S_j} \) of \( S' \), its corresponding position \( \mathbf{v}_{S_j}' \) in \( F_{T_j} \) can be calculated.

\[
\mathbf{v}_{T_j}' = k_{i1} \mathbf{v}_{S1} + k_{i2} \mathbf{v}_{S2} + k_{i3} \mathbf{v}_{S3}
\] (5)

2) Merging vertices

In order to preserve their visual features, the topologies of the two given surfaces \( S' \) and \( T' \) must be merged. So, all vertices \( \mathbf{v}_{T_j} \) off'' are inserted into \( S' \) based on the coordinate relationship, established as indicated above, and triangular decomposition is done according to the decomposition rules shown in Fig.2. If \( p_i \) is an interior point on the related face, as point \( p_1 \), the related face is decomposed into three new ones \( \Delta \mathbf{v}_4 \mathbf{v}_3 \mathbf{p}_1, \Delta \mathbf{v}_3 \mathbf{v}_2 \mathbf{p}_1 \), and \( \Delta \mathbf{v}_2 \mathbf{p}_1 \). If \( p_i \) is located on one edge of the related face, such as point \( p_2 \), then there are two adjacent faces that must be handled. \( \Delta \mathbf{v}_3 \mathbf{v}_2 \mathbf{p}_1 \) is decomposed into two triangles \( \Delta \mathbf{v}_3 \mathbf{p}_2 \mathbf{p}_1 \) and \( \Delta \mathbf{p}_2 \mathbf{p}_3 \mathbf{p}_1 \). \( \Delta \mathbf{v}_3 \mathbf{v}_1 \mathbf{v}_2 \) is decomposed into two triangles \( \Delta \mathbf{v}_3 \mathbf{v}_2 \mathbf{p}_1 \) and \( \Delta \mathbf{v}_1 \mathbf{v}_2 \mathbf{p}_2 \). If \( p_i \) is located on one vertex, such as point \( p_3 \), no face needs to be decomposed.
2.4 Merging feature edges

The preliminary intermediate topology model $M'$ is obtained as indicated in 1.3. All visual features of $S$ are retained, but not for $T$. Some features of $T$ are lost, especially the feature edges. So it is necessary to obtain the feature edges of $T$ and insert them into $M'$.

1) Extract feature edges

In sharp regional, boundary features are always seen as important visual features of models (Sunil and Pandey, 1995). In the sharp region, a dihedral angle, which is the angle between the outer normal vectors of two adjacent triangles, is greater than that in the flat region. So, the common lines of the two adjacent triangles with a dihedral angle that is great enough could be taken as feature edges.

Firstly, all the dihedral angles between all adjacent triangles of $T$ are obtained with Eq. (6).

$$\omega_i = \arccos \frac{N_{i1} \cdot N_{i2}}{|N_{i1}| \times |N_{i2}|}$$ (6)

Where $N_{i1}$ and $N_{i2}$ are the normal vectors of the two adjacent triangles respectively, and $i = 1, 2, ..., n$. $n$ is the number of the edges of $T$.

Secondly, the average dihedral angles of all dihedral angles are obtained with Eq. (7).

$$\omega_{\text{average}} = \frac{\sum_{i}^{n} \omega_i}{n}$$ (7)

Thirdly, the threshold for dihedral angles is calculated with Eq. (8).

$$\omega' = \omega_{\text{average}} + \sqrt{\frac{\sum_{i}^{n} (\omega_i - \omega_{\text{average}})^2}{n}}$$ (8)

Lastly, all edges of $T$ with two adjacent triangles that have a dihedral angle greater than $\omega'$ are selected as feature edges. The corresponding edges in $T'$ are signed as $E'$ in order.

2) Merge feature edges

In order to keep the visual feature of the target model $T'$, it is effective to insert all feature edges of $E'$ into $M'$. There are five possible kinds of spatial relationships between the two vertices of $e$ and the corresponding triangle $F_S$, as shown in Fig. 3: as shown in Fig. 3(a), one vertex is located inside the triangle and the other one is located on one edge or one vertex of the triangle; as in Fig. 3(b), the two vertices are both located inside the corresponding triangle; as in Fig. 3(c), two vertices are located on one or two edges of the triangle; as in Fig. 3(d), one vertex is located inside the triangle and the other one is located outside the triangle, but one vertex of the triangle is located on the feature edge; as in Fig. 3(e), the feature edge and one edge of the triangle intersect somewhere but not at any
vertices of the triangle.

**Figure 3.** Five kinds of possible spatial relationships between feature edges and the corresponding triangle.

For the situation as shown in Fig.3 (a) (b) (c) (d), the visual features of these edges can be preserved by the triangular decomposition operation done in 1.3. The situation in Fig. 3 (e) is the only one left to be dealt with.

**Figure 4.** The intersection in 3D space.

The situation in Fig. 3 (e) can be converted into 3D space as shown in Fig. 4. According to the mapping position of $T$ obtained in 1.3, which triangle the feature edge relates to is easily determined: one vertex is located inside the triangle and the other one is located outside the triangle. $v_{S2}v_{S3}$ is set as the edge of the triangle where the feature edge intersects, and point $p$ is set as the mapping point onto the unit sphere of the intersection. This means that $op$ intersects with the feature edge $v_{T1}v_{T2}$ and the edge $v_{S2}v_{S3}$ at the same time, thus yielding Eq. (9).

The position of these five vertices of the feature edge and the related triangle are known. So, Eq. (9) is easy to solve and the coordinate of point $p$ can be found. It is clear that $0 \leq k_m, k_n, k_1, k_2 \leq 1$ is necessary, otherwise the solution is not true and the other two edges of the triangle should be taken into calculation.

\[
\begin{align*}
|op| &= 1 \\
op \cdot (ov_{T2} \times ov_{T1}) &= 0 \\
op \cdot (ov_{S2} \times ov_{S3}) &= 0 \\
k_mp &= v_{T1}' + k_1(v_{T2}' - v_{T1}') \\
k_np &= v_{S2}' + k_2(v_{S3}' - v_{S2}')
\end{align*}
\]  

(9)

If anyone of $k_1$ and $k_2$ is 1 or 0, then the feature edge passes through one vertex of the related triangle. So only when $k_1$ and $k_2$ are both not 1 and 0, the point $p$ must be inserted.
into $M'$. The related triangular is decomposed based on the rule shown in Fig. 5. In Fig. 5 (a), blue lines are the feature edges and green lines are the new edges created by merging vertices of the feature edges by the way discussed in 1.3, and red points are the intersection points. In Fig. 5(b), red lines are the new edges created to merge the feature edges into $M'$ by inserting intersection points $p$ into $M'$ and doing decomposition on related triangulates.

**Figure 5.** Feature edge merging rule.

### 2.5 Morphing calculation

Lastly, the model $M'$ which can define the topology of the new desired model $M$ is formed, and the corresponding coordinates on $S$ and $T$ of all vertices in $M'$ are recorded. So, model $M$ can be generated with linear interpolation as shown in Eq. (10).

$$ M = (1 - t)(M' \otimes S) + t(M' \otimes T) $$

(10)

Where $t$ ($t \in [0,1]$) is the controlling factor, and symbol $\otimes$ means to extract the corresponding coordinates from the related model for $M'$. On model $M$, the visual characteristics of $T$ grow stronger with an increase of $t$, but the visual characteristics of $S$ reduce.

An example is given to show the morphing process is shown in Fig. 6, where two given models $S$ and $T$ are embedded onto the unit sphere to obtain two embeddings $S'$ and $T'$. The feature edges $E$ are selected from $T$. $S'$, $T'$ and $E$ are merged to obtain the topology $M'$ of the new models $M$. The four new shapes shown in Fig. 6 are created by setting $t$ as $0.2$, $0.4$, $0.6$ and $0.8$ respectively in Eq. (10).

**Figure 6.** A morphing example.

### 3. THE USABILITY OF THE PROTOTYPE SYSTEM
Usually, industrial designers are not familiar with the technological background of 3D morphing. In fact, they pay more attention to the application system of the technique and the morphing results. It is more important for industrial designers that morphing results retain the main visual features of the two given models, and that the system is easy to operate and is not overly time consuming. So, the usability of the prototype system of this method must be analyzed.

### 3.1 The prototype system

A prototype system for product form innovation based on the 3D morphing method proposed in this paper is designed with VC++ 6.0 and Open GL.

![Figure 7. The interface of the prototype system.](image)

For real application for industrial designers to create new shapes, the system should be as easy as possible to operate. So, there are only four buttons in the system as shown in Fig. 7. One can get new shapes as follows: Firstly, click button 1 to input the original model which will be shown in the original model region in the system. Secondly, click button 2 to input the target model which will be shown in the target model region. Thirdly, click button 3 to set the value of the controlling factor, then a new shape is created and shown in the result region. Lastly, click the button 4 to save morphing results. Also, there is a statue region in the system for the designer to check the information about the morphing, for example model size, time consumption, and so on. In the system, designers can rotate, zoom in and out on models with the mouse just like in the software Rhino which is currently the most popular modeling software among industrial designers.

### 3.2 The usability of the prototype system

The usability of the prototype system is analyzed based on two morphing tasks. After listening to an introduction on how to operate the system, five female and five male graduate students whose major is industrial design from Jiangsu University were asked to complete two morphing tasks. The two morphing examples shown in Fig. 6 and Fig. 8 are set as the two tasks. Four given models of the two tasks are divided into two groups and stored in different folders respectively. The participants were asked to complete the tasks one by one alone in a general classroom setting using a laptop with XP 2010 system, quad core Celeron 2 GHz and 2 GB memory. There are three kinds of data that must be collected in the two tests in order to analyze the usability of the prototype system.
Figure 8. A morphing task.

The first kind is the models’ data as shown in Table 1, which contains the number of triangulates, vertices, edges in two given models and new shapes, and also the number of feature edges in the target models. This data is used for participants to evaluate the morphing performance, specifically for participants to evaluate whether the size of the shapes is reasonable for further processing, such as rendering and remodeling.

Table 1 The models’ data of two tasks

<table>
<thead>
<tr>
<th>Items (Models’ data)</th>
<th>Mission 1</th>
<th>Mission 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>The triangular number of S</td>
<td>15146</td>
<td>7000</td>
</tr>
<tr>
<td>The triangular number of T</td>
<td>38654</td>
<td>1088</td>
</tr>
<tr>
<td>The feature edge number of T</td>
<td>1202</td>
<td>696</td>
</tr>
<tr>
<td>The triangular number of M</td>
<td>65061</td>
<td>15308</td>
</tr>
</tbody>
</table>

The second kind is the operation data of the missions as shown in Table 2, which contains the task completion time, waiting time and error rate. The task completion time is the time that participants spend on doing a particular task. The waiting time is the time that participants spend waiting for the computer’s response for morphing calculation. The task completion time and waiting time are used for participants to estimate whether time consumption of the method is acceptable. The difference between the task completion time and the waiting time is the operation time. The error rate is the value showing how many times participants clicked the wrong button. As shown in Table 2, the task completion time is closely related to waiting time, and waiting time increases when the triangular number of given models get larger. For each task, users spend the same amount of time on waiting, because the calculation time for a task is the same and is not related to who does the task. The error data of the first task is much more than that of the second task and the operation time of the first task is longer than that of the second task. This means participants make more errors and spend more operation time in the first task, and after an actual operation, error and operation time reduce rapidly. In another words, the system is easy to learn.

Table 2 Means (standard deviation) of operation data

<table>
<thead>
<tr>
<th>Items (Objective data)</th>
<th>Task 1</th>
<th>Task 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task completion time (s)</td>
<td>306.7(34.3)</td>
<td>87.6(10.5)</td>
</tr>
<tr>
<td>Waiting time (s)</td>
<td>274.6(0)</td>
<td>71.2(0)</td>
</tr>
<tr>
<td>Error data</td>
<td>2.3(1.77)</td>
<td>0.3(0.48)</td>
</tr>
</tbody>
</table>

The third kind data is users’ satisfaction. After each task, every participant is asked to fill out a satisfaction scale in which there are five items. According to one’s own subjective feelings, a mark between 0 and 100 is given for each item. A higher mark means that the participant
agrees with the item more. As shown in Table 3, the system satisfies ten users. In particular, they think that the system is easy to perform, new shapes are creative and helpful for product form design, and the size of new shapes is suitable for further processing. In interviews after the tasks, several designers even think that there are some new shapes can be directly used as new products, such the ones corresponding with \( t = 0.6 \) and \( t = 0.5 \) in two tasks respectively. Furthermore, more designers would like to use the system to create new product forms, if time consumption could be reduced more.

<table>
<thead>
<tr>
<th>Items (Users’ satisfaction)</th>
<th>Satisfaction scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>The system is easy to perform</td>
<td>93.4(4.2)</td>
</tr>
<tr>
<td>Time consuming is reasonable</td>
<td>80.2(5.6)</td>
</tr>
<tr>
<td>The size of new shapes is suitable for further processing</td>
<td>88.3(7.3)</td>
</tr>
<tr>
<td>New shapes are creative and helpful for product form design</td>
<td>89.3(4.7)</td>
</tr>
<tr>
<td>I like to use the system to create new product forms</td>
<td>82.3(7.5)</td>
</tr>
<tr>
<td>Mean data of the users’ satisfaction</td>
<td>86.7(5.86)</td>
</tr>
</tbody>
</table>

4. DISCUSSION

The core thought and main actualization process of the 3D morphing method in this paper is as the same as the morphing technique based on triangular decomposition we proposed earlier (Sha and Lu, 2009 and 2015). But the test results indicate that it is optimized with several improvement on the key steps of this method. And these improvement are very useful for industrial designers to operate the prototype system easier. It is very important to the real application for the system.

Firstly, instead of the projection method proposed by Kent and Alexa (Kent, 1992; Alexa and Carlson, 2000), Spherical Harmonic is used to embed the given models onto the unit sphere. There are no self-intersections on mapping results, so relaxation iteration that needs termination condition that is usually set manually and sometimes does not converge is no longer necessary for removing self-intersections. The embedding process could be completed automatically with no manual intervention. So, it is not important any more weather industrial designers know the basic principle of embedding or not. It also means that at least one button could be deleted from the old prototype system and designer do not have to manually set termination condition for relaxation iteration for the two given models.

Secondly, for the purpose to keep the features on the target model, it is important to extract feature edges on the target model. Instead of the average dihedral angle, an interference value which is varying with different models is added onto the average dihedral angle to get an adaptive threshold. The adaptive threshold make sure that only those important feature edges whose dihedral angle is obviously out of a range are extracted. As the results of morphing, new shapes are successfully made and themost important is that even those small key features are kept. In fig. 6, the feature edges such as boundary, decorative lines on the bonnet and the trunk are extract successfully. These features are more and more obvious with the increasing of controlling factor for 3D morphing. Similarly, the fold features on a given bottle are well kept and exhibited on the new shapes as shown in fig. 8. The feature edges exacted in the way do not lead to shaprisein the size of new shapes as shown in table 1.

It takes industrial designers’ requirements which are different from other designers as start point to construct the prototype system. Compare with the old prototype system we made for industrial designers (Sha and Lu, 2009), there are only four buttons in the new system and more information is shown. It is easy to control the four buttons even though a
designer knows nothing about the morphing technique. As discussed in section 3.2, the system can make useful new shapes without complex human intervention. Usability test results show all the ten designers think the system is easy to perform. But, due to the new embedding and feature exacting method, it takes more time to compute out the 3D morphing results. So, waiting time shown in table 2 is longer. It is the main factor decrease the designers’ satisfaction about the system.

5. CONCLUSIONS

In this paper, the previously proposed 3D morphing method based on triangular decomposition has been improved. Features of the two given models are preserved with feature edges extracting and spherical harmonic mapping. The usability of the prototype system of the method is analyzed. The results show that the system is easy to operate even if the designers have no knowledge about the mathematical background of the method. The system needs no complex manual interventions, time consumption is acceptable and the new shapes created are useful for product form design.

But, there are still some disadvantages reflected in the test. Complex models cannot be handled in the method and waiting time is a bit long. So, for the perfect actual application, the system and the method should be further improved.

6. ACKNOWLEDGMENTS

We thank the support from the Graduate Student Innovation Project for Common College in Jiangsu Province, China (No.CXLX11-0566).

7. REFERENCES