3D Paper-Cut Modeling and Animation

Yan Li
State Key Lab of CAD&CG
Zhejiang University
yli@cad.zju.edu.cn

Jinhui Yu
State Key Lab of CAD&CG
Zhejiang University
jhyu@cad.zju.edu.cn

Kwan-liu Ma
Department of Computer Science
University of California, Davis
ma@cs.ucdavis.edu

Jiaoying Shi
State Key Lab of CAD&CG
Zhejiang University
jyshi@cad.zju.edu.cn

ABSTRACT

Paper-cut is one of the most characteristic Chinese folk arts, often used during festivals and celebrations. Chinese artists have used paper-cut to make animations. Typical paper-cut artwork is made with 2D illustrations on paper, and making many frames necessary for an entire animation can be tedious and expensive. We present a system that allows a designer to directly annotate a 3D model with paper-cut patterns, while still allowing for adding an artistic touch to the design. We have designed special motifs coupled with templates, resulting in a parameterized set of decorative paper-cut patterns which give the artist flexible control and allow editing of the pattern size, orientation, and shape. The artist chooses a pattern, and places the pattern on the model in the desired position. The system determines actual surface coverage and trims the object's surface geometry to simulate the cut-out effect we observe in traditional 2D paper-cut art. We demonstrate that our system allows faster and easier addition of paper-cut decoration to 3D models compared to general purpose modeling tools, such as Maya. Animations made with the 3D paper-cut models escape from the limitations of traditional 2D paper-cut animation on the movement in perspective, further-more, our system allows for easy pattern animations on 3D models, which is very powerful but hard to do with traditional paper-cut animation.

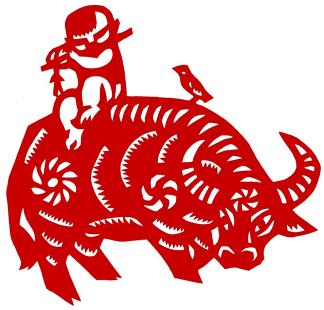
Categories and Subject Descriptors

I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism; I.3.8 [**Computer Graphics**]: Applications; J.5 [**Arts and Humanities**]: Miscellaneous

Keywords

paper-cut, pattern, modeling, animation

1. INTRODUCTION



(a) *Boy and Buffalo* by Ximing Lin

(b) *Door God* by Zhao Yingcao

Figure 1: Examples of Chinese Paper-cut

Paper-cut is one of China's most popular and characteristic folk art, it imitates nature forms by depicting characters and symbols via delicate, beautiful patterns, as shown by the examples in Figure 1. Paper-cut is popular in China during folk festivals and celebrations. Fascinated by this medium, Chinese artists even began to use paper-cut to make animation film early in 1958. Mixing paper-cut with elements of other traditional art in animation, such as Chinese ink painting, is regarded as "Chinese school" worldwide. Unfortunately very few such animation films were produced because they were too expensive.

There are several factors that prohibit hand-crafted paper-cut animations to be made, say, for TV commercial animations with hundreds episodes:

(1) Making such animations is tedious and labor intensive. The artist first decomposes a character or animal into pieces including at least the head, the trunk, arms, hands, legs and feet, and then cut some patterns inside each piece to depict features or textures, finally joins them together to form a paper-cut figure. In the phase of making animation, the movement of each piece is manipulated under great care of the artist.

(2) A notable limitation with hand-crafted paper-cut animation is that the fluid movement, particularly in perspective, is not easy to achieve with flat puppets. As a result, most actions of characters are animated either in front view or side view.

(3) Some animations are avoided in hand-crafted paper-cut animation. Take flower blooming for instance, the artist has to cut out many flower patterns changing over time, it will bring an unrealistic burden of labor.

We address these issues by designing a set of parameterized paper-cut patterns and intuitive tools that allow the artist to easily arrange and edit patterns on the existing 3D objects. With 3D models of paper-cut effects we are able to produce more natural actions in perspective and view them in any direction, furthermore, parameterized paper-cut patterns allow us to make animations like flower blooming with great ease.

Generating 3D paper-cut effects involves object modeling, pattern design, patterns arrangement on the object surface, pattern shape editing, and pattern trimming. In Chinese paper-cut, objects are usually exaggerated in shape as shown in Figure 1. We therefore let 3D artists model exaggerated objects using typical commercial tools such as Maya. Our work focuses on the other tasks involved in making 3D paper-cut effects for animation.

In principle, 3D artist can cut out patterns directly on the object surface using Maya to simulate paper-cut effects. However, this is a time consuming process as the artist has to trim patterns by specifying each individual piece of the pattern on the mesh surface. Also, the artist has to frequently rotate and zoom the object model and edit the evolving pattern until they look pleasing during the trimming process.

In this paper we describe a system for generating paper-cut effects on 3D surface models, and for making paper-cut animations. Our goal is to provide artists with intuitive tools for easily and quickly generating 3D paper-cut effects and, when necessary, animating paper-cut patterns with little users intervention. There are two main research challenges: (1) Construction of paper-cut patterns with models that capture the features of hand crafted paper-cut patterns. These models should allow artists to parametrically edit patterns either for fine tuning pattern to desired shape or animating patterns. (2) Arrangement of the paper-cut pattern on the object surface is a creative decision of the artist; the artist may use a variety of patterns and arrange them in different ways according to their personal aesthetic. We thus provide tools that allow artists to directly arrange patterns on the object model, and let the system carry out "non-creative" operations, such as mapping the patterns to surface geometry and trimming the underlying model to simulate the cut-out effect. We also implement an intuitive UI which provides the artists flexible control and allows editing of paper-cut patterns on the mesh.

In the next two subsections, we review some previous work in two categories related to our work: artistic effects to 3D models and mesh augmentation.

1.1 Artistic effects to 3D models

Many computer graphics researchers have attempted to render 3D objects with a variety of artistic effects, such as pen-and-ink illustrations[26][27], impressionist[18], watercolor[5], graphite pencil[24], Chinese painting[25], Islamic star patterns[10], and the style of Dr. Seuss[12]. Hertzmann and Zorin[8] generate a tone and orientation field from a 3D model. Klein *et al.* develop a system for non-photorealistic rendering of virtual environments[11]. Baxter *et al.*[1], Kalnins et al.[9] and Chu and Tai[3] present 3D painting systems which provide varying visual effects. Up to date, very few efforts have been made for the synthesis of paper-cut effects. Liu *et al.*[16] propose a system to identify the dihedral symmetry group in the outer-ring of the pattern, the inner bilateral symmetry as well as the asymmetry region on the 2D circular paper-cut image. The "fold-then-cut" plan is then generated by symmetry group indexing of each sub-region of the paper-cut pattern. New paper-cut patterns with the same motif as the input, but with different symmetry groups, are finally synthesized.

1.2 Mesh augmentation

In recent years there is a growing interest in mesh augmentation such as segmentation[20], intelligent scissoring[13] and mesh editing[15]. These mesh processing operations can improve the result of some advanced techniques such as automatic texture atlas generation[14], geometry images[7] and making papercraft toys from meshes[19]. Olsen *et al.* combine sketch modeling with mesh augmentation[21].

We attempt to generate paper-cut effects on 3D object models. We begin from the introduction of patterns used in Chinese animal and human figure paper-cuts, in part because they are popular themes in Chinese paper-cut and also because they are the main elements in many animations.

2. PAPER-CUT PATTERNS

Patterns for animals and human figures can be classified into following types: face feature patterns (FFPs) depicting the eye, nose and mouth (as show by the left-most two, middle two and right two in Figure 2, respectively), circular patterns (CPs) representing joints between the trunk and legs, and saw-toothed patterns (STPs) depicting animal fur (Figure 1(a)). Some additional lines and other specific patterns may be added for aesthetic purpose, such as the scale-like patterns on the armor of the Door God as shown in Figure 1(b).

Figure 2: Face Feature Patterns

Careful observation of Figures 1 and 2 reveals that those patterns are composed of smaller shapes which we called as motifs, such as *circles*, *crescents*, and *triangles*. These motifs can be arranged according to some simple structures to form different patterns. We have developed a set of tools for exploring the design space of paper-cut patterns. In our approach, a tiling is used to guide the placement of motifs.

In the existing literature, the work in[10] is most close to ours in regarding to the pattern construction and tiling, the main differences between the two are:

(1) Motifs in [10] are Islamic symmetric stars and rosettes bounded by regular n-gons, while our motifs vary from circles to crescents, triangles and free lines.

(2) Tiling in [10] involves choosing the template to fix the designs *overall layout* on the plane and selecting motifs to fill the template's tiles. The resultant star patterns are joined motifs covering the design plane. While the templates used in our work are for placing similar or different motifs to construct *individual* paper-cut patterns. Furthermore, our templates are rich in structures, including n-parallel bars, n-fold stars, the comb like structure and many more.

In the following section, we first describe how to define a crescent motif (CM) and templates of the crescent-based patterns (section 3.1), and then proceed to describe how to construct the FFP (section 3.2), STP (section 3.3) and free line pattern (section 3.4).

3. CONSTRUCTION OF PAPER-CUT PATTERNS

3.1 Crescent bases patterns

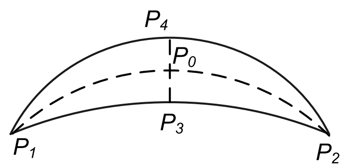


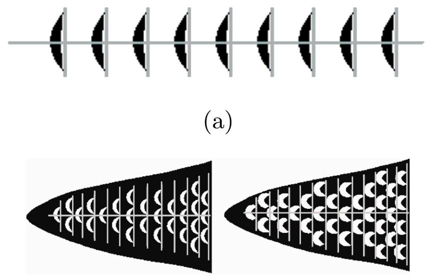
Figure 3: A Crescent Motif

To model patterns contain the motif of repeated crescents, we start form the construction of a CM.

The CM is defined as follows. First, we take an isosceles triangle with a base l_{cs} and height h_{cs} as a reference and interpolate its vertices $p1, p0, p2$ with a spline to get a light curved skeleton (long dash line in Figure 3). The camber of the crescent can be controlled by alternating the ratio of l_{cs} to h_{cs}. Next, we put a short line with length l_w across $p0$ on the curved skeleton to get two end points $p3$ and $p4$, and interpolate $p1, p3, p2$ and $p1, p4, p2$ respectively with the spline to get two curves that define the CM shape (solid lines in Figure 3).

Grouped crescent patterns include ridge patterns on the buffalo and scale-like patterns on the armor of Door God, which can be constructed by tiling CMs with a template of a n-parallel bars structure.

For the regular tiling area(Figure 4(a)), the template is defined by simply placing a number of parallel bars of constant length across a central line, as shown by grey lines in the Figure. The orientation θ_s, spacing d_s and length l_s of bars are all adjustable. We attach the CMs to the template by mapping the line P_1P_2 (Figure 3) of the CM onto each bar on the template to form the grouped crescent patterns, as shown by the white shapes in Figure 4(a).



(a)

(b)

Figure 4: Template for Grouped Crescents

For the irregular region such as black area (Figure 4(b), which is selected manually, see section 4.2), the lengths of bars in the template are no longer constant but extended to touch the border of the region for placing more CMs on bars. Positions of CM on each bar are determined by subdividing the bar length with l_{cs} of the CM. An offset is added to CMs position every other bar to simulate the arrangement of CM on the armor of Door God.

The density of the CM can be adjusted by varying the number of bars as well as l_{cs}, as shown in Figure 4(b). Note that shapes are changed by varying crescent width and camber in those two examples. This demonstrates the power of our parameterized motif and templates.

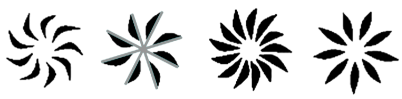


Figure 5: Variants of the CP Generated

Circular patterns. A CP is composed of a number of crescents surrounding a center point. We first define an n-fold rotational axis (grey lines in Figure 5) to guide the placement of 'on-axis' CMs, and each CM is moved a distance dc from the center. Some example CP variants with different n and crescent camber are shown in Figure 5.

3.2 Face feature patterns (FFPs)

FFPs include eye, nose, and mouth patterns. In Figure 2 we show two eye patterns. The template for the first eye pattern can be defined by a cross with two lines of the length l_e and we respectively (grey lines in Figure 6). We place a circle with a radius r_e in the center and two CMs symmetrically above and below the circle. The second eye pattern shown in Figure 2 can be constructed using the same template as the first eye pattern, with the circle removed from the center and the template rotated by $90°$.

Figure 6: Constructed Eye Pattern

Other patterns shown in Figure 2 can be constructed using simple shapes and templates, and we omit the trivial description of their construction.

3.3 Saw-toothed like patterns (STPs)

Each saw-tooth in the STP can be simulated by a triangle with a base w_t and height h_t. The template for a STP is similar to a comb in structure, and the corresponding STP is constructed by placing a number of triangles on a crescent-shaped base skeleton. Figure 7 shows STP variants, produced by modifying both crescent camber and triangle density.

Figure 7: Variants of the STP Constructed

3.4 Free line pattern

The free line pattern is useful for depicting some feature lines on the object in the paper-cut. Since the feature lines vary from object to object in length, we allow users to first specify a few control points to guide the placement of the free line pattern. Our system then interpolates those points with the spline to get a skeleton. Two curves are added with the same distance along the two sides of the skeleton, and the width is decreased near the two ends of the line to form tips.

4. PATTERN ARRANGEMENTS

It would be desirable to automatically place the patterns on the object surface. The goal is however very difficult to achieve, as addressed below.

In the stroke based rendering of object models, extraction of features such as suggestive contours [6] and curvatures [26] [9] [17] [22] may help greatly for the automatic placement of strokes. For instance, on animal and human models, facial features can be depicted directly with strokes placed on the extracted features. Paper-cut is however a *pattern-based* annotation, each pattern has its own structure and features, which are usually *inconsistent* with those extracted from the object surface. Thus, it is inappropriate to use extracted features to guide automatic pattern placement.

Some patterns, like FFPs, have semantic meaning that must be put in correct positions on the object models. Automatic determination of FFP positions on object models is also difficult because it requires semantic recognition of some feature points on the object surface.

Pattern arrangement on the object by its nature is a *subjective* issue, because artists may arrange patterns on the object for aesthetic purpose. As a result, the pattern arrangement usually differs from artist to artist even on the same object. Mesh segmentation methods[4][23][28] usually partition the mesh surface by use of some geometric attributes associated with the surface, and the resultant segmentation may not meet the aesthetic requirements for paper-cut.

Our system presents a simple editing UI that the artist can use for pattern selection, placement, and parametric modification. We let the system determine the underlying geometry covered by the selected patterns. To this end, we offer several algorithms for determining patch sets inside the region on the object surface for mapping the different patterns.

In our UI we employ a traditional mouse for input. The input can be a point or a curve depending on the pattern type selected. We use this input to calculate the boundary of the region that the pattern may cover. The boundary of the region is used as a constraint to search for patches inside the region, as described next.

4.1 Circular region for FFP, CP and STP

We use a circle to approximate the region covered by the FFP, CP and STP on the object surface, because the circular region allows easy orientation and editing of those patterns. The eye pattern provides a good example. When the artist places an eye pattern on the object model, the patch where the central point c_e of the cross template falls is taken as the seed patch. We expand the region using a region growing algorithm, and the growing process terminates when all the neighboring patches whose distances are smaller than a threshold $T = k * l_e/2$, where l_e is the length of the eye template and the factor k is used to control the region size. The default value of k is set to be 1.5 which ensures the eye pattern to be drawn inside the region grown.

4.2 Regions for grouped crescent patterns

Grouped crescent patterns may cover regions with constant or varying width along the central line of the template on the object surface (Figure 4). To determine those regions, the user first specifies a few points on the object to indicate the central line of the template. Our system interpolates those points with the spline to get a curve Pg_i with denser samples points.

In the case of the region with constant width is required, we use a number of partially overlapped circles laid on Pg_i to approximate the region. The radius of the circle is set to be $wg/2$, where wg is the width of the region. We calculate the length of the curve Pg_i, and resample Pg_i with the interval approximately to be $wg/2$ to get Cg_i as the center of the circle. We chose this interval because the region boundary can be approximated with overlapped circles in a satisfactory manner, as shown in Figure 8.

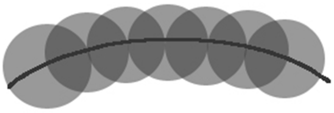


Figure 8: Region With Constant Width

Our system searches for patches within the circle one-by-one using the region-growing algorithm until all circles have been searched.

Since a free line pattern also covers a region with constant width, we can use the same procedure to determine its region.

In the case of the region with irregular shape need to be determined, we require the user to directly draw the boundary curve on the object. The user then picks a patch inside the region as the seed patch, and the system expands the region using the procedure described by the following pseudo code:

```
set a queue Q=NULL;
insert p_0 to Q, where p_0 is the seed patch;
while q is not NULL {
    set the temporary queue T_q = NULL;
    for every patch p_i of three edges e_1,e_2,e_3 in Q {
        for each e_j {
            for patch p_j which shares an edge with p_i{
                if e_j is not the boundary edge and
                    p_j are not marked {
                        mark p_j and add the marked p_j to T_q;
                    }
                }
            }
        replace Q with T_q;
    }
}
```

5. PATTERN DRAWING, EDITING AND TRIMMING

5.1 Pattern drawing

Before drawing pattern on the object surface, parameterization of the patch set representing the regions determined with different algorithms described in section 4 is necessary. There are many methods proposed for the patch set parameterization. We adopt the algorithm by Lévy *et al.* in [14] because it is efficient, and the resulting parameterization is satisfactory for our application. The main idea in [14] is the use of a least-squares approximation of the Cauchy-Riemann equations. The objective function minimizes angle deformations. Please refer to [14] for the implementation detail.

5.2 Pattern editing

Our system draws the user-selected patterns on the parameterized patch set. If the size, orientation, or shape of the surface-mapped pattern does not look satisfactory, our UI allows users to scale and rotate the pattern templates, or otherwise edit the shape of the motif by varying relevant parameters associated with the pattern shapes (section 3).

5.3 Pattern trimming

All patterns drawn on the flattened and parameterized patch sets are partitioned though Constrained Delaunay Triangulation and converted from window coordinates to mesh/world coordinates by projecting the pattern boundary into 3D. This reverse projection can be done with the viewing transformation matrix and the depth buffer of the rendering API.

We finally trim patterns on the object surface by use of a trimming algorithm proposed in [2] to achieve cut-out effects. Please refer to [2] for the implementation details.

6. EXAMPLES AND ANIMATIONS

The interface of our system is shown in Figure 9. On the left is the main window where the object model is both displayed and edited. The pattern type is selected from a menu on the bottom middle. The user first selects a pattern type, and then interactively moves the pattern to the desired position on the object model using the mouse. The pattern is overlaid on the object, and the user can then scale it to the proper size and rotate it to the desired orientation. On the right of the main window is the pattern editor, in which the user can further refine pattern shapes and templates. Patterns are finally drawn on the parameterized patch set and reverse projected onto object surface.

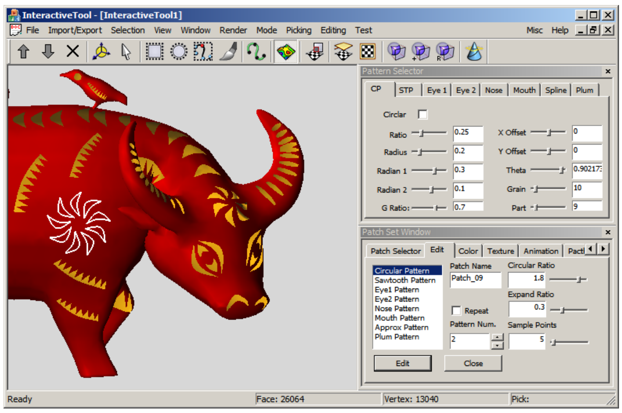


Figure 9: The Interface of the System

To demonstrate the usability of our system, we have created two 3D models so we can make a 3D version of the two 2D paper-cut works shown in Figure 1. Both models took hours for 3D artist to make using a commercial modeling tool. These 3D models also allow use to make lively animations.

To generate paper-cut effects on the boy and buffalo model, we first load the model, and then select patterns from the menu and place them directly on the model. Patterns on the ridge and horns of the buffalo are placed by first indicating the central line on the model to guide the position of the patterns, and then drawing the selected grouped crescent patterns on those lines. The resultant effects are shown in Figure 10. It took less than 20 minutes to complete the whole paper-cut process on this model.

We animated the paper-cut effects of the boy and buffalo in a scene composed of 2D hand-made paper-cuts. We constructed the scene using flowers as the foreground and a

Figure 10: Paper-cut Effects on Boy&Buffalo Model

big tree image as the background using Maya. To maintain stylistic parity with the background, we projected the boy and buffalo to form a 2D paper-cut effect, and animated it using a skeleton driven technique offered in Maya. The buffalo's head moves while the buffalo walks, and the 3D information is encoded in the projected 2D paper-cut effects. This eliminates the limitations of 2D traditional animation on actions in perspective. As shown in the accompanying video, the paper-cut appearance of the boy and buffalo are indistinguishable from the hand-made paper-cuts, Figure 11 is a frame taken from the animation.

Figure 11: A Frame of Animation From Video

The second example is Chinese Door God. The model was designed by the 3D artists to simulate the image of Figure 1(b). During the phase of putting patterns on the model, we omit some trivial patterns and just put main patterns to simulate the Door God image. Traditionally, pictures or statues of the Door Gods can be found guarding the entrances to Chinese homes, temples, restaurants and stores to keep evil spirits from entering, we put patterns on Door Gods face to present an angry expression that enhances the mighty of the Door God, and resultant effects is shown in Figure 12.

The last example is flower blooming. To illustrate animating patterns, we use a 2D paper-cut image as the background, as shown in Figure 11(b). The artist first places some flowers

Figure 12: Paper-cut Effects on the Chinese Door God Model

of proper size and orientation on trees, and then uses a slider on our UI to change both length and width of all petals over time, creating a flower-blooming animation.

(a) Flower blooming process

(b) A frame from flower blooming animation

Figure 13: Flower Blooming Animation

Figure 13(a) shows the process of a flower blooming, and Figure 13(b) is a frame taken from the flower blooming animation.

7. USABILITY

We tested ease-of-use by asking users (including amateurs and art students) to annotate existing 3D models using our system. With little training or detailed instruction, all the users were able immediately to select a pattern, place it on

Pattern	t_{spsr}	t_{cb}	t_{ed}
FFP	3∼5s	-	3∼6s
CP	3∼5s	-	3∼6s
STP	3∼s	-	3∼6s
GCP1	-	15∼20s	20∼30s
GCP2	-	15∼30s	20∼30s

Table 1: Usage Statistics For Three Operations

the object model, and scale or rotate the pattern if necessary. They could also draw central curves or boundary curves on the object and place the selected pattern inside the region indicated by the central curves or boundary curves. Most users found our system to be very intuitive for both interactive operations and parameter based editing.

For art students with 3DMax or Maya experience, our system was substantially easier and faster for generating paper-cut effects. For instance, it took an art student about 30 seconds to trim a CM on the object surface with 3DMax, and about 5 minutes to trim an eight-crescent CP. While using our system, the art student was able to select, place and scale a CP on the object in only 5 seconds.

For grouped crescent patterns, such as the rightmost pattern in Figure 4(b), it would take about 18 20 minutes to trim them with 3DMax, but would take about 1 minute with our system.

Table 1 presents some usage statistics recorded while placing different patterns on the object models. The test machine contained a 2.4GHz Pentium 4 processor and 512MB of RAM. The reported timings in the table include:

t_{spsr} = time spent on selecting, placing, scaling, and rotating the pattern;
t_{cb} = time spent on drawing central and boundary curves;
t_{ed} = time spent on editing.

In Table 1, GCP1 stands for the grouped crescent patterns covering the region with constant width, and GCP2 stands for the grouped crescent patterns covering the region with varying width or irregular boundary. "-" means that there are no relevant operations needed for the corresponding patterns.

We did not list exact times in Table 1, but instead listed time ranges spent on corresponding operations, because some editing operations may vary even for the same pattern. For example, consider the eye pattern. In addition to operations like selecting, placing, scaling and rotating, the user may further edit its length or width if necessary. Placing grouped crescent patterns on the object models usually takes longer time because it involves more interactions such as drawing the central and boundary curves as well as editing operations such as varying the pattern density and shapes.

8. CONCLUSIONS AND FUTURE WORK

In this work the computer is utilized as a tool to enhance the speed and ease with which a user can create paper-cut effects to existing 3D models. With a parameterized set of paper-cut patterns, artists are able to produce impressive 3D paper-cut effects of nontrivial complexity, as demonstrated in the paper and the accompanying video.

In regarding to paper cut modeling, although our motifs and templates are simple in structure, they effectively represent artistic features of hand-made paper-cuts patterns. In regarding to paper cut animation, parameterized pattern editing and grouping enable artists to create 3D paper-cut effects much faster than manual trimming using commercial tools. This may significantly reduce the cost and the time involved in making paper-cut animations. Furthermore, our system allows for pattern animations on 3D models, which is very powerful but hard to do with traditional paper-cut animation. The 3D extension of the paper-cut effects makes fluid movement in perspective possible, thus resultant animations escape from the limitations of traditional 2D paper-cut animation.

Our work suggests two areas for future investigations:
Pattern creation based on samples. In our current implementation, all patterns are constructed using parameterized motifs that are grouped with templates. In addition to FFP, CP and STP, there are many other paper-cut patterns for depicting different parts of fish, birds, flowers and trees. The construction of these patterns involves much tedious work. It would be desirable to construct these patterns based on hand-made sample via some automated mechanism.
Effects animation. Effects like fire and water can add realism and richness to the animation. In hand-crafted paper-cut animation, however, such effects animations are not generally done because they require the artist to cut out frame-coherent patterns varying over time to show movement, which demands excessive labor even for just a small part of the scene. Automatic effects animations require further study on how to make models capturing the stylized paper-cut effects patterns varying over time.

9. ACKNOWLEDGMENTS

We thank anonymous reviewers for their suggestions and comments. This work is supported by National Key Basic Research and Development Program of China (973 program) (No. 2002-CB-312101), the National Natural Science Foundation of China (No. 60673007), and National 863 High Tech. Research and Development Program of China (No. 2006AA01Z312).

10. REFERENCES

[1] B. Baxter, V. Scheib, M. Lin, and D. Manocha. Dab: Interactive haptic painting with 3d virtual brushes. In *Proc.of SIGGRAPH'01*, pages 433–438, 2001.
[2] C. D. Bruyns and S. Senger. Interactive cutting of 3d surface meshes. *Computers and Graphics*, 25(4):635–642, 2001.
[3] N. S.-H. Chu and C.-L. Tai. An efficient brush model for physically-based 3d painting. In *Proc. 10th Pacific Conference on Computer Graphics and Applications*, pages 413–422, 2002.
[4] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. *ACM Transactions on Graphics*, 23(3):905–914, 2004.
[5] C. Curtis, S. Anderson, J. Seimis, K. Fleischer, and

D. Salesin. Computer generated watercolor. In *Proc. of SIGGRAPH'97*, pages 421–430, 1997.

[6] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, and A. Santella. Suggestive contours for conveying shape. In *Proc. of SIGGRAPH'03*, pages 848–855, 2003.

[7] X. Gu, S. Gortler, and H. Hoppe. Geometry images. In *Proc. of SIGGRAPH'02*, pages 355–361, 2002.

[8] A. Hertzmann and D. Zorin. Illustrating smooth surfaces. In *Proc. of SIGGRAPH'00*, pages 517–526, 2000.

[9] R. Kalnins, L. Markosian, B. J. Meier, M. Kowalski, J. Lee, P. Davidson, M. Webb, J. Hughes, and A. Finkelstein. Wysiwyg npr: Drawing strokes directly on 3d models. In *Proc. SIGGRAPH'02*, pages 755–762, 2002.

[10] C. S. Kaplan and D. Salesin. Islamic star patterns in absolute geometry. *ACM Transactions on Graphics*, 23(2):97–119, 2004.

[11] A. Klein, W. Li, M. Kazhdan, W. Correa, A. Finkelstein, and T. Funkhouser. Non-photorealistic virtual environments. In *Proc. of SIGGRAPH'99*, pages 527–534, 1999.

[12] M. Kowalski, L. Markosian, J. Northrup, L. Bourdev, and R. Barzel. Art-based rendering of fur, grass and trees. In *Proc.of SIGGRAPH'99*, pages 433–438, 1999.

[13] Y. Lee, S. Lee, and A. Shamir. Intelligent mesh scissoring using 3d snakes. In *Proc. of Pacific Graphics'04*, pages 279–287, 2004.

[14] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. In *Proc. of SIGGRAPH'02*, pages 362–371, 2002.

[15] W.-C. Li, B. Lévy, and J. Paul. Mesh editing with an embedded network of curves. In *Proc. of Shape Modeling International conference*, pages 63–71, 2005.

[16] Y.-X. Liu, J. Hays, Y.-Q. Xu, and H.-Y. Shum. Digital paper-cutting. In *SIGGRAPH sketch*, 2005.

[17] L. Markosian, M. A. Kowalski, S. J. Trychin, L. D. Bourdev, D. Goldstein, and J. F. Hughes. Real-time nonphotorealistic rendering. In *Proc. of SIGGRAPH'97*, pages 415–420, 1997.

[18] B. Meier. Painterly rendering for animation. In *Proc.of SIGGRAPH'96*, pages 477–484, 1996.

[19] J. Mitani and H. Suzuki. Making papercraft toys from meshes using strip-based approximate unfolding. *ACM Transactions on Graphics*, 23(3):259–263, 2004.

[20] M. Mortara, G. Patane, B. S. M., and F. Rossignac. Blowing bubbles for the multiscale analysis and decomposition of triangle-meshes. *Algorithmica*, 38(1):227–248, 2003.

[21] L. Olsen, F. F. Samavati, M. C. Sousa, and J. J. A. Sketch-based mesh augmentation. In *Eurographics Workshop on Sketch-Based Interfaces and Modeling*, pages 43–52, 2005.

[22] T. Saito and T. Takahashi. Comprehensible rendering of 3-d shapes. In *Proc. of SIGGRAPH'90*, pages 197–206, 1990.

[23] P. V. Sander, Z. J. Wood, S. J. Gortler, J. Snyder, and H. Hoppe. Multi-chart geometry images. In *Proc. Symposium on Geometry Processing*, pages 146–155, 2003.

[24] M. Sousa and J. Buchanan. Computer generated graphite pencil rendering of 3d polygonal models. *Computer Graphics Forum,(Proc. Eurographics'99)*, 18(3):195–207, 1999.

[25] D.-L. Way and Z.-C. Shih. The synthesis of rock textures in chinese landscape painting. *Computer Graphics Forum,(Proc. Eurographics'01)*, 20(3):123–131, 2001.

[26] G. Winkenbach and D. Salesin. Computer generated pen-and-ink illustration. In *Proc. of SIGGRAPH'94*, pages 469–476, 1994.

[27] G. Winkenbach and D. Salesin. Rendering parametric surfaces in pen and ink. In *Proc. of SIGGRAPH'97*, pages 91–100, 1997.

[28] H. Yamauchi, S. Lee, Y. Lee, Y. Ohtake, A. Belyaev, and H.-P. Seidel. Feature sensitive mesh segmentation with mean shift. In *Proc. of Int'l Conf. on Shape Modeling and Applications*, pages 236– 243, 2005.