

Modeling Expressions of Peking Opera Facial Make-Ups

Xinjie Zhang
State Key Lab of CAD&CG
Zhejiang University
Hangzhou, China
Email: funnyzj@qq.com

Jinhui Yu
State Key Lab of CAD&CG
Zhejiang University
Hangzhou, China
Email: jhyu@cad.zju.edu.cn

Abstract

This paper presents a framework for modeling expressions of Peking Opera facial make-ups. We first abstract patterns in the facial make-ups by use of flood filling and 8-connected boundary tracing algorithms, and then pick up feature points from abstracted pattern boundaries with the Douglas-Peucker algorithm. A pattern bank is constructed with those feature points corresponding to the forehead patterns, eyebrow patterns, orbit patterns and mouth patterns. During the synthesis phase, users may pick up appropriate patterns from the pattern bank to compose new facial make-ups. Our system reconstructs the original patterns by interpolating corresponding feature points with the spline. Expressions of facial make-ups can be obtained by deforming pattern with the Radial basis function interpolation on either predefined or interactively specified facial feature points. Examples of both synthesized facial make-ups and facial expressions of facial make-ups are given in the paper.

1. Introduction

Peking Opera (PKO) is one type of traditional Chinese opera, which combines music, vocal performance, mime, dance and acrobatics. It has a more than 200-year history and is popular among Chinese and people of other nationalities alike for its dramatic presentation of Chinese culture and history. On November 16, 2010, the United Nations Educational, Scientific and Cultural Organization (UNESCO) approved the inclusion of Peking Opera in its Representative List of the Intangible Cultural Heritage of Humanity.

One of the notable components in PKO is its facial make-ups (FMUPs), which is painted with bright colors and exaggerated yet fine stroke patterns to produce a vivid and graceful effect. It expresses the temperament as well as the traditional appearance of different characters, and enables audiences to grasp the personality of a character portrayed and the character's social status at a glance. Figure 1 shows three examples of FMUPs in PKO.

As one of symbols of Chinese traditional culture, PKO FMUPs reflects Chinese people's way of thinking, gives people an enjoyable visual experience and enriches their

spiritual life. In addition to being painted on characters faces during the performance of PKO, FMUPs of PKO has also been applied in diverse fields such as traditional crafts making, decorative design, fashion design, packaging, advertisement, digital entertainment and education of PKO.



Figure 1. Examples of Peking Opera facial make-ups

In this paper we propose a framework for modeling and animation of PKO FMUPs. An overview of our system is given in Figure 2. First, we abstract patterns of FMUPs and construct a pattern bank. Next, we select patterns from the bank to compose new FMUPs, and fill appropriate colors in the patterns chosen. Finally, we deform patterns of FMUPs by the Radial Basis Function (RBF) interpolation on either predefined or interactively specified facial feature points to obtain varying facial expressions of PKO FMUPs.

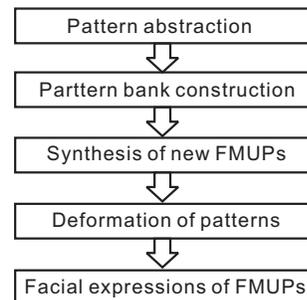


Figure 2. System framework overview

2. Related work

Modeling of traditional ethnic patterns is relatively a lightly explored area. In 1975 Alexander [1] described a

Fortran program for generating the 17 ornamental patterns. 23 years later, Wong et al [2] provided a computer-generated application with the idea of adaptive clip art, for traditional floral ornamental design. This method can be used to generate patterns that are tailored to fit a particularly shaped region of the plane.

Islamic star pattern is one famous example of traditional patterns, which has been studied by artists and historians for centuries. Recently, several algorithms and methods have been discussed for computer-generated star patterns design. Karam and Nakajima [3] carried out a deep and thorough application of group theory to the study of symmetric star patterns, which analyze the structure and organizational forms of the Islamic star patterns with symmetry group. Kaplan [4] presented a simple method for rendering Islamic star patterns based on Hankin’s ”polygons-in-contact” technique. In other work, Gulati and Katyay [5] developed a parametric modeler for designing Islamic star patterns, innovating with new interpretations of star patterns for latticed screens.

Paper-cutting, a traditional folk art emerged soon after the paper was invented in China, is made by cutting some stylized patterns on the paper to depict various objects. Paper-cutting has been widely used for decoration and crafts to express people’s filling or wishes during festivals and celebrations in China. Today paper-cutting has been part of the folk art traditions of cultures all over the world. To simulate this beautiful medium, Liu et al [6] proposed a platform for symmetry-based computer analysis and synthesis of papercut-patterns, which can identify the sub-patterns and recreate new papercut-patterns. Xu, Kaplan and Mi [7] regarded paper-cutting as a method of composing bi-level images under a set of geometric connectivity constraints, thus, a brand new elements can be used to compose digital paper-cut designs. Li et al [8] provided a system allowing users to annotate a 3D model with paper-cut patterns, add artistic touches to the design and finally make 3D paper-cutting animations.

FMUPs of PKO are composed of some stylized patterns with varying colors, which have been remained almost unchanged for the known characters in PKO. Thus, FMUPs of PKO become the symbolic representation of existing known characters in PKO. In some applications such as advertisements and packaging design, it is desirable to create FMUPs which are not identical to existing ones, but have different patterns and colors while still look similar to those traditional ones. Cai et al [9] carried out a method for synthesis of PKO FMUPs, which can generate images with a pattern bank constructed by Bzier curve interpolation. Later on Cai and Yu [10] proposed a vector-based hierarchical model for modeling expressions of PKO FMUPs. They applied free form deformation method on local face feature patterns to obtain different expressions of FMUPs. The drawback of this approach is that other patterns drawn in the make-ups remain unchanged, besides, local deformations on

face features and the resultant expressions look somewhat rigid and mechanical.

3. Pattern abstraction and construction

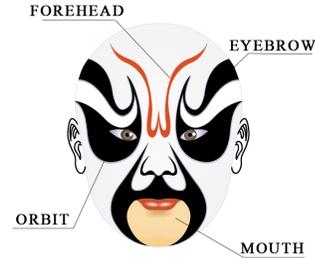


Figure 3. Patterns of Peking opera facial make-ups

Patterns in FMUPs of PKO are traditionally named according to their positions on the face. In order to simplify the classification, patterns can be roughly divided into four parts, such as forehead patterns, eyebrow patterns, orbit patterns and mouth patterns, as illustrated in Figure 3. It is fit for either simple or complex types of FMUPs. Those patterns are not only painted with diverse shapes, but also with different colors, as shown in both Figure 1 and 3. Our pattern abstraction scheme involves pattern region detection by flood filling over the pattern region, region boundary detection by use of 8-connected boundary tracing algorithm, and boundary point reduction with the Douglas-Peucker algorithm. Reconstruction of original patterns can be achieved by interpolating reduced boundary points (pattern feature points) with the spline.

3.1. Pattern region detection

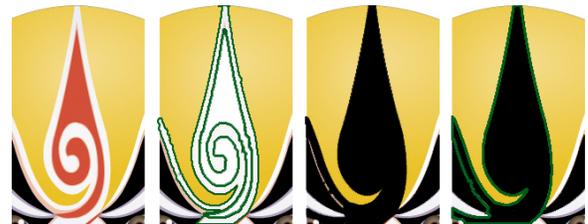


Figure 4. A forehead pattern with a smaller internal pattern

Most of patterns in PKO FMUPs are painted with constant colors. Intuitively, one may consider to detect pattern regions by use of colors painted inside them. However, in FMUPs of PKO, some smaller patterns may be painted inside other patterns, as shown by an example of forehead pattern in the left of Figure 4. Using color information we may detect pattern regions with holes of complex shapes formed by

small patterns inside them, as shown by the second figure in 4, such pattern region would prohibit us from synthesis of new FMUPs, say, by replacing smaller patterns with different shapes inside the pattern.

When painting FMUPs, the makeup artists start from bigger patterns, and then add smaller patterns inside them until a FMUP is completed. Our pattern abstraction is a reverse process of painting FMUPs by hand. We first paint small patterns black with simple flood filling, and then paint the region left inside the bigger pattern black to finally obtain a region of bigger patterns without smaller patterns inside. Briefly, the flood filling algorithm can be described with the following procedure:

1. Choose any point P inside the region as the seed pixel, and set P to the empty queue.
2. While queue is not empty:
3. Pop the first element Q of queue.
4. If the color of Q is equal to the target-color (pattern color), set Q as the handled point in the region.
5. If the color of the node to the west of Q is not handled, push the node into the queue.
6. Apply step 5 to the node in the east, north and south of Q .
7. Turn to step 2.
8. Return.

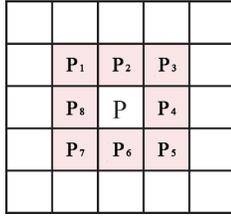


Figure 5. The Moore neighborhood in the algorithm

Once a pattern region is painted black, we apply the 8-connected boundary tracing (which is also called Moore-Neighbor Tracing) algorithm over the black pattern region. The main idea of the 8-connected boundary tracing is to use the 8-neighbors (Moore neighborhood) of a pixel, p , as illustrated in Figure 5, to trace boundary points of the region with a procedure described as follows:

1. Define a sequence $S[]$ of boundary pixels and let p denote the current tracing pixel. Note that the starting point Q is randomly selected.
2. Set $S[]$ to be empty.
3. Let $S[0] = p = Q$. The count number $k = 0$.
4. While $S[k] \neq S[0]$
5. Set $S[k] = p$ and name the Moore neighborhood pixels of p as $p_1, p_2, p_3, p_4, p_5, p_6, p_7$ and p_8 .
6. Let $k=k+1$.
7. Start from the left point p_8 , find out the first pixel p_i inside the region, with counterclockwise search. Set $p = S[k] = p_i$.

8. Return.

When the boundary tracing is finished, we put $S[]$ into corresponding point set $P_{i,j}$ to describe the boundary shapes of different patterns, where the subscript i indicates the pattern type (forehead, eyebrow, orbit and mouth pattern), and j is the point index in the point set.

3.2. Pattern construction

In order to reduce the data size of $P_{i,j}$, we adopt the Douglas-Peucker algorithm [11], [12], [13] to pick up pattern feature points $PF P_{i,k}$ from $P_{i,j}$, where k is the point index in the feature point sets. Since the Douglas-Peucker algorithm works on non-close curves, we divide each closed boundary $P_{i,j}$ with $j=1, \dots, N$ into two segments by:

```

if ( N is divisible by 2 ) {
    DouglasPeucker( 0, N/2 );
    DouglasPeucker( N/2, N-1 ); }
else {
    DouglasPeucker( 0, (N-1)/2 );
    DouglasPeucker( (N+1)/2, N-1 ); }
}

```

where DouglasPeucker() is the function applied to each segment separately to pick up pattern feature points, and it works as follows:

Given a segment in $P_{i,j}$ as shown by red dots in Figure 6 (the other segment is shown in pink dots), we calculate the distance between a point on the segment and the line formed by the two end points of the segment, and then pick up the biggest distance value d_m (as indicated with the blue points P_1 in Figure 6). Next, we set a threshold $D = 0.8$ by experiments and apply the following procedure on the segment:

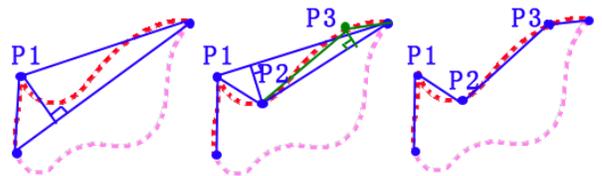


Figure 6. Process of Douglas-Peucker algorithm

If $d_m < D$, all points between two end points are regarded as non-feature points thus no feature point is picked up;

Else $d_m \geq D$, the point with d_m on the segment is picked up as a feature point p_k which is then put into $PF P_{i,k}$.

With p_k we can divide the current segment into two smaller segments on which we repeat the above comparison procedure until all feature points satisfying the condition $d_m < D$ are picked up.

Figure 6 shows the process of Douglas-Peucker algorithm in our system. The pattern feature point set $PF P_{i,k}$ is finally put into a pattern bank. During the synthesis phase, we can

compose new FMUPs by choosing appropriate $PPF_{i,k}$ from the bank and place them on the target FMUPs, the final pattern boundaries can be drawn by interpolating $PPF_{i,k}$ with the spline.

4. Pattern deformation and animation

Note that patterns in the pattern bank are abstracted from FMUPs with no expressions. In order to obtain varying expressions of PKO FMUPs, we need to deform patterns that compose FMUPs. In order to overcome the drawback of rigid and mechanical facial expressions derived by the free form deformation applied on individual patterns in FMUPs, we adopt the local warping technique, Radial Basis Function (RBF) interpolation [14], [15], [16], to deform patterns. Since facial expressions are dominated by some feature points near eyes and the mouth, we define facial feature points FFP_j , $j = 1, \dots, N1$, as shown in Figure 7 in which red dots indicate the initial facial feature points corresponding to those in FMUPs with no expressions, and green dots indicate the target facial feature points in the deformed FMUPs. Note that some points along the face contour are indicated by green dots overlaid on the bigger red dots, those initial and target points remain fixed before and after deformations are made.



Figure 7. Initial and target positions of face feature points

In our system we offer two options to set target facial feature points, one utilizes predefined target facial features points to achieve some typical facial expressions such as smile, angry, sad, etc, the other requires users to interactively specify the target facial feature points to achieve facial expressions they desired. With both initial and target facial feature points, the general process of RBF interpolation goes as follows.

For each feature point p_j in FFP_j , we denote its initial position by p_j^0 in the non-deformed FMUPs and target position by p_j^1 in the deformed FMUPs. The displacement d_j of p_j between p_j^0 and p_j^1 is:

$$d_j = p_j^1 - p_j^0 \quad (1)$$

Next, we need to determine the displacement dn_k of pattern feature points in $PPF_{i,k}$. Let np_k denote the points for a given pattern type i , we try to find out an interpolation function $S(p)$ to determine the displacement dn_j of np_k . For each point $p \in np_k$ the format of the interpolation function is expressed as:

$$S(p) = \sum_{j=1}^{N1} c_j g(\|p - p_j\|) + Mp + t \quad (2)$$

where c_j is the weight coefficient corresponding to p_j , and $g(\|p - p_j\|)$ is a radial symmetric function with the following form:

$$g(\|p - p_j\|) = \frac{1}{\sqrt{c_j^2 + \|p - p_j\|^2}} \quad (3)$$

where $\|\cdot\|$ denotes the Euclidean norm on R^2 .

In our warping application for expressions of FMUPs, one of the natural pattern distortions is "affine" that includes scaling, rotation, shearing, and translation in the whole image, so we integrate the affine mapping function $Mp + t$ into equation. Affine variables are M and t , where M is a 2×2 matrix and t is a 2×1 vector.

$$\begin{cases} \sum_{j=1}^{N1} c_j = 0, \\ \sum_{j=1}^{N1} c_j p_j^T = 0. \end{cases} \quad (4)$$

In order to calculate the coefficient c_j and determine the affine component M and t , we need to solve a system of linear equations as Equation 4, where p_j^T is the transposed matrix of p_j . Once the weight coefficient c_j and the affine component M and t are obtained, we can calculate the displacement dn_k of each point np_k by use of Equation 4:

$$dn_k = S(np_k) \quad (5)$$

With both d_j and dn_k we can deform patterns in FMUPs that draw the final expressions of FMUPs. Animations of facial expressions of FMUPs can be simply achieved by linearly interpolating initial and target facial feature points with time to get intermediate positions of facial feature points, with which we can deform patterns of FMUPs in a step of predefined time interval.

5. Results

In this section we present some results of synthesis of new FMUPs and facial expressions of FMUPs. Our system allows users to select different patterns from the pattern bank and compose a variety of new FMUPs by combining them in different fashion, as shown in Figure 8. Although those synthesized FMUPs are not identical to the existing



Figure 8. Synthesized FMUPs of PKO.



Figure 9. FMUPs expressions generated by predefined feature points



Figure 10. FMUPs expressions generated by interactively specified feature points.

FMUPs painted in PKO, they preserve characteristics of PKO FMUPs well and look similar to traditional FMUPs.

In Figure 9 we show three images with expression from normal to smile and then to being angry, generated by predefined target face feature points for a synthesized FMUPs. While Figure 10 shows the results of different expressions by specifying the target feature points manually. In both cases, deformations of patterns in FMUPs can be achieved in real-time, as demonstrated by the accompanying video.

In Figure 11 we present two series of facial expressions taken from animations generated by our system.

6. Limitation

For some special FMUPs, our method may do no help.

In the synthesis of FMUPs phrase, the first limitation is that it requires PKO FMUPs images with high-quality as the input of pattern abstraction stage, or we cannot abstract the



Figure 11. Animation series of FMUP expressions.



Figure 12. FMUPs expressions generated by interactively specified feature points.

patterns completely, especially for the small and sharp ones. Secondly, the patterns of some FMUPs cannot be separated by simple flood fill, since the pattern may combine with each other. Besides, if one pattern composed by too many smaller patterns, much time has to be taken to manually abstract and construct it, which makes troubles for synthesis of FMUPs.

While, for the process of modeling expressions of FMUPs, there are also a few problems happening, as shown in Figure 12. The leftmost image in Figure 12 is the synthesis image of a PKO FMUP. In the following images, we can see that displacement differences exist before and after the pattern deformation, besides, the corners may not stay smooth as before, or even the pattern distortion just occurs. Since the fitting method may be too simple to obtain smooth curves.

7. Conclusions and future work

In this paper we propose a framework for modeling and animation of PKO FMUPs, with which users can easily to compose new FMUPs and further make facial expression animations of PKO FMUPs in real-time. In the future, we

intend to extend our work to 3D PKO FMUPs, this requires fitting patterns of PKO FMUPs to the face features on 3D face models as well as exaggerated deformations on 3D face surfaces.

Acknowledgments

This work is supported by the State Key Program of National Natural Science Foundation of China (No. 60933007), the Key Technologies R&D Program of China (No. 2007BAH11B02).

References

- [1] Howard Alexander, *The computer/plotter and the 17 ornamental design types*, In SIGGRAPH '75 Proceedings of the 2nd annual conference on Computer graphics and interactive techniques, pages 160-167, ACM New York, NY, USA, 1975.
- [2] Michael T. Wong, Douglas E. Zongker, David H. Salesin, *Computer-generated floral ornament*, In SIGGRAPH '98 Proceedings of the 25th annual conference on Computer graphics

and interactive techniques, pages 423-434, ACM New York, NY, USA, 1998.

- [3] Hussein Karam, Masayuki Nakajima, *Islamic Symmetric Pattern Generation Based on Group Theory*, In Proceedings of the International Conference on Computer '99, pages 112-119, IEEE Computer Society, Canmore, Alta., Canada, 1999.
- [4] Craig S. Kaplan, *Islamic star patterns from polygons in contact*, In GI '05 Proceedings of Graphics Interface, ACM International Conference Proceeding Series 112, pages 177-186, Ontario, Canada, 2005.
- [5] Vishal Gulati, Puneet Katyal, *Parameterized Modeling of Star Patterns for Traditional Latticed Screens*, In International Journal of Computer Applications, vol.11, Issue 1, pages 28-36, 2010.
- [6] Yanxi Liu, James Hays, Ying-Qing Xu, Heung-Yeung Shum, *Digital Papercutting*, In SIGGRAPH '05 ACM SIGGRAPH 2005 Sketches, ACM New York, NY, USA, 2005.
- [7] Jie Xu, Craig S. Kaplan, Xiaofeng Mi, *Computer-Generated Papercutting*, In 15th Pacific Conference on Computer Graphics and Applications (PG '07), pages 343-350, Maui, HI, 2007.
- [8] Yan Li, Jinhui Yu, Kwan-liu Ma, Jiaoying Shi, *3D paper-cut modeling and animation*, In Computer Animation and Virtual Worlds (CASA '07) archive, vol.18, Issue 4-5, pages 395C403, John Wiley and Sons Ltd. Chichester, UK, 2007.
- [9] Cai Feilong, Yu Jinhui, Peng Ren, *Analysis and synthesis of peking opera facial make-ups*, Journal of Computer-Aided Design and Computer Graphic, vol.21, Issue 8, pages 1092-1097, China, 2009.
- [10] Cai Feilong, Yu Jinhui, *Modeling and Editing of Peking Opera Facial Makeup Expressions*, Journal of Image and Graphics, China, vol.15, Issue 6, China, 2010.
- [11] David Douglas, Thomas Peucker, *Algorithms for the reduction of the number of points required to represent a digitized line or its caricature*, Cartographica: The International Journal for Geographic Information and Geovisualization, University of Toronto Press, Canada, pages 112-122, 1973.
- [12] John Hershberger, Jack Snoeyink, *Speeding Up the Douglas-Peucker Line-Simplification Algorithm*, Proc 5th Symp on Data Handling, pages 134-143, 1992.
- [13] Urs Ramer, *An iterative procedure for the polygonal approximation of plane curves*, Computer Graphics and Image Processing, vol.1, Issue 3, Pages 244-256, 1972.
- [14] Zeng Zhaohua, Zhang Yongmei, *3D Face Modeling Based on Front and Side Photos*, Mechanical Engineering and Automation, China, 2008.2, No. 1.
- [15] Daw-Tung Lin, Han Huang, *Facial Expression Morphing and Animation with Local Warping Methods*, In Image Analysis and Processing '99, pages 594-599, Venice, Italy, 1999.
- [16] Mengdong Li, Qiuqi Ruan, *Adapting non-feature points of facial wire-frame model*, China:Journal of China Railway Society, vol.23, Issue 3, 2001.