芸術科学会論文誌 Vol. 7 No. 1 pp. 14 - 21

A Synthesis of 3-D Kabuki Face from Ancient 2-D Images Using Multilevel Radial Basis Function

Xin Yin,* Weiwei Xu,** Ryo Akama,* and Hiromi T.Tanaka*
*Ritsumeikan University; **Microsoft Research Asia

Abstract

A technique to synthesize a 3-D Kabuki face model from ancient images is proposed. Kabuki is a traditional Japanese theater art. Kabuki researchers can know some ancient information of Kabuki from the ancient 2-D images such as Ukiyo-e (Japanese drawing) and Kumadori images (Kabuki makeup). Compared with the real face images in photographs and paintings, the face images in Ukiyo-e and Kumadori images were distorted a little from the real face for art representation. Hence, it is necessary to synthesize a new 3-D face model between the 3-D real face original model and the 2-D images. A deformation technique is proposed for transforming 3-D real face original models according to the 2-D images, and a mapping algorithm is given to map the images onto the 3-D face shape model. To ensure balance between precision and smoothing of 3-D model deformations and mappings, the radial basis function (RBF) is developed and a multilevel RBF is proposed. This multilevel RBF solver can be used to deform the 3-D real face model as well as map the 2-D images onto the 3-D face model. Some experimental results are given to demonstrate the effect of this technique.

Key words: face modeling, 3-D model deformation, mapping, radial basis function, humanity and computer graphics

1. Introduction

Kabuki is a traditional form of Japanese theater. It was founded early in the 17th century and, over the next 300 years, has developed into a sophisticated, highly stylized form of theater. There are abundant culture legacies to describe the long history of this outstanding art, such as beautiful images of Kabuki actors drawn in Ukiyo-e (a type of Japanese drawing) and the images of Kumadori (Kabuki makeup).

Ukiyo-e is a traditional Japanese drawing of life scenes that include the Kabuki. Examples of Ukiyo-e are shown in Figure 1 (a). In this type of Ukiyo-e, the faces of Kabuki actor are drawn. The Ukiyo-e will be used to synthesize the 3-D face model. Kumadori is the name of a special face makeup style in Kabuki, which is usually viewed as the most distinctive feature associated with Kabuki. Some Kumadori are represented as 2-D images. One type of Kumadori image is the Kumadori drawing shown in Figure 1 (b). Another type of Kumadori image is the Oshiguma, which is obtained by copying the face makeup of the Kabuki actor onto a paper or cloth. This type of Kumadori image is shown in Figure 1 (c). The two types of Kumadori images will be used to map the 3-D face model.

However, these 2-D images are all traditional media and lack interactivity-i.e., one can only view the static images. The aim of this research is to make use of such legacies to synthesize 3-D Kabuki face models in the computer by computer graphics (CG) techniques. It is a new way to preserve cultural heritage, and we believe that it also provides an interesting way for people to learn about Kabuki art.

1.1. Previous work

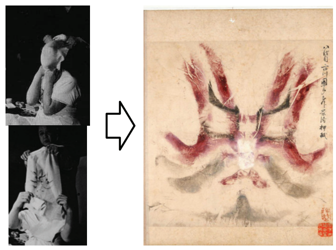
This work is mainly related to 3-D face shape modeling and texture mapping techniques. Previous works on these two techniques are introduced. The 3-D face model used is a polygon mesh model, which is used to represent the surface of 3-D objects widely in CG.

Abundant research work on face modeling exists. Sampling the 3-D coordinates of the points on the face is one generic approach to construct the 3-D face model. This approach can be found in early

** Weiwei Xu has done the main basic work of this study as a researcher in Ritsumeikan University from April 2004 to August 2005. He is co-first author of this paper.

(a) Ukiyo-e

(b) Kumadori drawing



(c) Oshiguma

Fig. 1. Ukiyo-e and Kumadori images.

studies of the face model, which was constructed by sampling the surface of the face at a number of points and connecting these points to form a skin of polygons (Parke [1]). The range data are an important source for sampling the 3-D coordinates of the points on the face (Lee [2], Tanaka [3], Sakaguchi [4]). The range data can be obtained by using a laser scanner or a special camera that can capture the distance image. Usually, the range data are reconstructed to a polygon mesh to present the 3-D model.

A different generic approach of sampling the 3-D coordinates is computing 3-D these from multiple photographs or videos. The correspondence between feature points, which are obtained interactively or automatically from the different photos, can be used to compute the 3-D coordinates to construct a 3-D face model (Akimoto [5], Pighin [6]). Similar to the multiple photos, the 3-D coordinates can also be obtained from videos (Guenter [7], Liu [8]). These techniques can construct 3-D face models from multi-view information of the same object from multiple photos or videos. It is difficult to construct the 3-D model from a single 2-D image such as the case of Ukiyo-e using these techniques.

The techniques to construct the 3-D face model from single image (the photo or the painting) were also developed (Blanz [9], Blanz [10]). These techniques can construct the face model by comparing the image and a 3-D face models database. This technique is related to the deformation techniques that can be used to deform the 3-D face model to match the 2-D image. The mesh deform techniques based on the radial basis function (RBF) and the Laplacian coordinates are proposed to make an animation of the 3-D mesh models (Sorkine [11], Zhou [12]). The inspiration of this study comes from these deformation techniques. These techniques can synthesize the 3-D mesh face model from real face images such as photos and images. It is very difficult to synthesize the real 3-D face model from the 2-D images such as the Ukiyo-e and the images of Kumadori directly because the faces in the Ukiyo-e and those of Kumadori are distorted a little for art representation. This means that deforming the 2-D image is necessarily done simultaneously. For CG techniques, deforming of 2-D images mentioned here means mapping the constrained texture onto the surface of 3-D polygon mesh. We propose a deformation technique that can be applied to the 3-D face polygon model and 2-D image mapping simultaneously and that can reduce errors as well.

As mentioned above, this research is also related to texture mapping. Basically, there are two kinds of texture mapping in CG: 2-D texture mapping and 3-D texture mapping. The 2-D texture mapping is mapping 2-D texture image onto a 3-D geometry model to enhance the visual effect, so it needs 2-D parameterization of a 3-D geometry model. 2-D parameterization can be solved as an optimization problem or a scattered interpolation problem (Besl [13], Levy [14], Tang [15], Lee [16]). To map a 2-D texture onto a 3-D face model, some point constraints, such as eyes, mouth, and so on, should be considered. Levy [14] presents a global optimization method to compute the texture coordinates for face texture mapping problem. Tang [15] proposes to use single-level RBF with regulation term to map face texture onto a 3-D face model. Lee [16] presented a multilevel B-splines method to do scattered data interpolation. They built a multilevel method by increasing the density of knots. We enhance the precision of single-level RBF with multilevel RBF based on the adjustment of the length of radii. So, the precise alignment of feature points can be guaranteed, and the quality of mapping result is then enhanced.

Some studies about the face in the Ukiyo-e have also been carried out. Yamada [17] analyzed the features of the face in the Ukiyo-e and the relationship made to the artists. The 3-D face model was made using a commercial software manually also. However, this synthesis is mainly based on the experience of the software user. There is a need to improve the precision of the 3-D face model using automatic or semiautomatic techniques. How to improve the precision of 3-D face model is the subject of this study.

1.2. Overview

It is difficult to get enough 3-D information from a single distorted 2-D image to synthesize the 3-D face model of a Kabuki actor directly. Deforming the original 3-D face model according to the 2-D image is one way to solve this problem. Since the 2-D texture is distorted from the real face, mapping the constrained texture to the 3-D face model needs to be carried out simultaneously. A multilevel RBF is proposed to deform a 3-D face model and mapping texture simultaneously and to control precision and smoothing of the synthesized result. To provide the user with the ability to deform the shape of the eyes, mouth, and other features on the face, we adopt all the feature points defined in MPEG4 face animation standard (details to be introduced in section 3). Figure 2 illustrates the basic flowchart of this face modeling system. It consists of two parts: 3-D face model deformation and 2-D texture mapping. In deformation, after the user inputs an Ukiyo-e and moves the feature points to the target position, the corresponding part of the 3-D face model will be deformed according to multilevel RBF. In texture mapping, the user needs to define the corresponding feature points, for example, the eyes on the 3-D face model correspond to the eyes on the texture image, to specify how to map the texture to the 3-D face model.

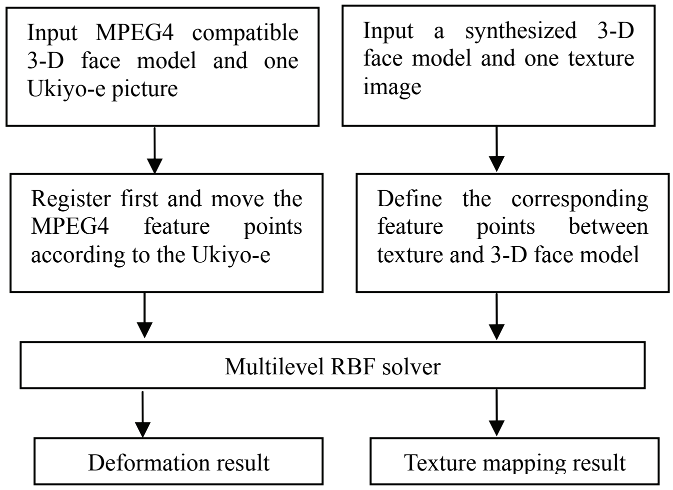


Fig. 2. Basic system flowchart.

The remainder of this paper is organized as follows. Section 2 will describe the principle of multilevel RBF method. In section 3, we will show how to synthesize the 3-D face model from the Ukiyo-e. Kumadori mapping is described in section 4. Experimental results are described in section 5 and we conclude and discuss future work in section 6.

2. Multilevel RBF

The RBF has been widely used in scattered interpolation, deformation, texture mapping, and so on. Figure 3 shows a RBF network. In the RBF theory, a regulation term is usually used to guarantee the smoothness of the synthesized surface (Orr [18]). However, this also leads to imprecision at the data points. To solve this problem, a multilevel RBF approach is proposed to achieve precision and smoothness at the same time.

2.1. RBF

As shown in Figure 3, x is the input and y is the output. The RBF network is defined as

$$y = \sum_i (w_i h_i) x \,, \tag{1}$$

where w_i is weight coefficient and h_i is the kernel function. The

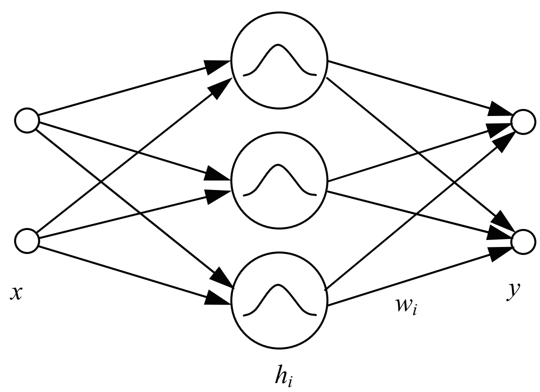


Fig. 3. RBF network.

kernel function is usually determined by a center C_i and radius r_i. The kernel function is a Gaussian-like function.

Given known pairs $(x,y)_i$, we need to compute coefficients w_i so that the following least square function is minimum:

$$g = \sum_i \left(y_i - \sum_j w_j h_j(x_i) \right)^2 + \lambda \sum_i w_j^2 \cdot \qquad (2)$$

Taking derivative to w_i, we get

$$\left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)\mathbf{w} = \mathbf{H}^T\mathbf{y}\,, \qquad (3)$$

where $\boldsymbol{H_{ij}} = h_j(x_i)$ and $\boldsymbol{w}$, $\boldsymbol{y}$ are column vectors of w_i, y_i, respectively. To select a good parameter λ, we adopt the global-ridge algorithm and the generalized cross-validation as error criterion (Golub [19]).

The second term in Equation 2 is called the regulation term. It is used to limit the values of coefficients so that the resulting RBF network does not fluctuate too much and generates a smooth curve or surface. This is suitable for machine learning or to generate a smooth approximation from noisy input. However, in this case, we need the results from the RBF network to pass the data points precisely. That means we need the RBF network for interpolation, not approximation, in this application. Concretely, corresponding feature points between 2-D image and 3-D model are first set up. The parameters of the RBF network such as w_i, etc. are computed from the position of the feature points. Then, the corresponding relationship of all pixels in the 2-D image and points in the 3-D model mesh are computed using these parameters of RBF. Details will be explained in sections 3 and 4.

2.2. Multilevel RBF method

Center c and radius r of the RBF kernel function are the parameters to control the behavior of the RBF network. In most applications, the center is obvious. So, the radius r is the parameter that we can resort to control the RBF network. We will analyze the influence of the radius to the RBF network first, and then explain why we choose the multilevel approach.

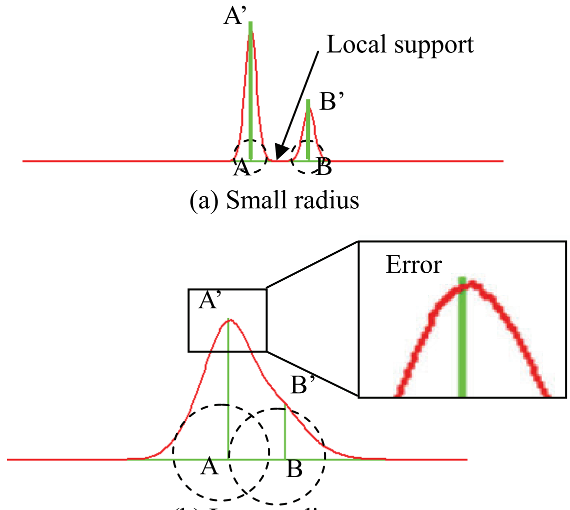
As illustrated in Figure 4, two points on a straight line will be moved to new positions. Point A is moved to the position of point A' and point B is of the point B'. As shown in Figure 4(a), the small radius is selected first, and the result from RBF shows that the middle regions between these two points are not affected. We need to point out that this is not suitable for interpolation applications, and we cannot use a small radius to do texture mapping. In Figure 4(b), we choose a large radius instead. It is clear that the radius is large enough to let two points influence each other. At this time, the middle region is influenced and interpolation is much better than in the previous case. However, there is still an approximation error, which is 1.9% at point A'. That means the resulting curve cannot model data points precisely.

Based on the above discussion, we are able to design a multilevel RBF approach. First, large radii are selected to get a coarse approximation and the approximation errors remaining at the data points are computed. Then, the length of the radii are gradually reduced to produce a fine approximation of the data points. The multilevel RBF can be written as

$$y = \sum_j \sum_i w_{ij} h_{ij}(x) + \lambda_j \sum_i w_{ij}^2 \,, \qquad r_{ij_1} > r_{ij_2},\ if\ j_1 > j_2 \cdot \qquad (4)$$

Thus, the multilevel RBF algorithm can be described as follows:

- Step 1: Input data points and an error threshold.
- Step 2: Calculate initial radii for each data point according to the distribution of data points. (We adopt the following rule to compute the initial radius: First, for each data point, compute the maximum distance between this point and other data points, second, use half of the maximum distance as the initial radius for this data point to achieve global support effect).
- Step 3: Levels = 0.
- Loop.
- Step 4: Compute coefficients w_i for current-level RBF network (with regulation term) and store the coefficients for current level.
- Step 5: Compute the sum of the approximation error at the data points. If the sum is less than the error threshold, stop the algorithm.
- Step 6: Reduce the length of radii with formula: $r_{ij+1} = a * r_{ij}$ $(a < 1)$ (j = levels) (in this application, a is usually chosen to be 0.5).
- Step 7: Levels = Levels + 1, go to loop.



(a) Small radius

(b) Large radius

Fig. 4. Influence of radius.

Figure 5 shows the result of this approach for the problem above. Now, the resulting curve is better. Comparing with the result of a single RBF, the error is decreased and is nearly zero.

2.3. Comparing single-level and multilevel RBF

We would like to point out that this multilevel RBF approach is suitable for both local support and global support RBF. We compare the RBF and multilevel RBF in the 2-D case. As shown in Figure 6, the black points represent the feature points (total of 10 in this example) and dash lines stand for the correspondence. Points A', B', C,' and D' in Figures 6(b) and 6(c) correspond to the points A, B, C, and D in Figure 6(a), respectively. Correspondence in the feature points is irregular.

RBF can be used to determine the correspondence relationship of the total region in Figures 6(b) and 6(c). First, the coefficient such as w_i, etc. can be computed from the position of the correspondence

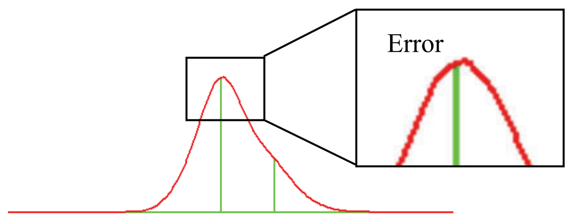
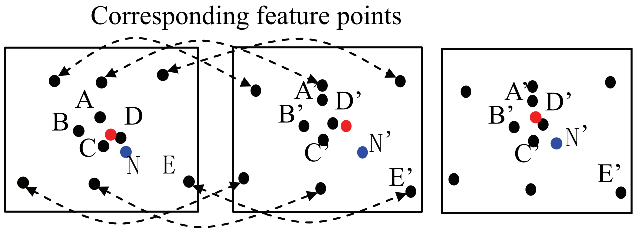


Fig. 5. Interpolation with multilevel RBF.



(a) Feature points (b) Single-level RBF (c) Multi-level RBF

Fig. 6. Comparison of the single-level RBF and the multilevel RBF in the 2-D case.

feature points (10 black points shown in Figure 6.) using the equations mentioned above. Then, the correspondence positions of all pixels included in the image are computed using the coefficient w_i. As shown in Figure 6, two sample points are chosen to compare the single-level RBF and the multilevel RBF. One is the red one inside the feature points ABCD and another is the blue one outside the feature points ABCD. The results of the single-level RBF method and the multilevel RBF are shown in Figures 6(b) and 6(c), respectively. Because the single-level RBF mainly considers global optimization, the error on the local is big. The red sample point is inside ABCD, but it is outside of A'B'C'D.' Using the multilevel RBF can decrease this error. The position of the red sample point is inside the A'B'C'D' as shown in Figure 6(c). For the blue sample point N, the ratio of the distance CN and NE is used to evaluate the error when computing the position of blue point N using single-level or multilevel RBF. The original ratio of the CN and NE is 0.397 in Figure 6(a). The ratio of D'N' to N'E' is 0.731 when computed by single-level RBF. It is 0.324 when computed by multilevel RBF. It is clear that the ratio obtained by multilevel RBF is nearer the original one compared with that obtained by single-level RBF. This means that the multilevel RBF is better than the single-level RBF in decreasing the error. In the next section, we will use multilevel RBF to synthesize the 3-D face model and do texture mapping.

3. Synthesizing the 3-D face model from Ukiyo-e

The 3-D face synthesis procedure is actually a registration and deformation procedure. It will generate a 3-D face model that matches the face in Ukiyo-e picture very precisely. The final step is to project the picture, which is calculated from the texture coordinates, onto the synthesized 3-D face model.

3.1. MPEG-4 face animation standard

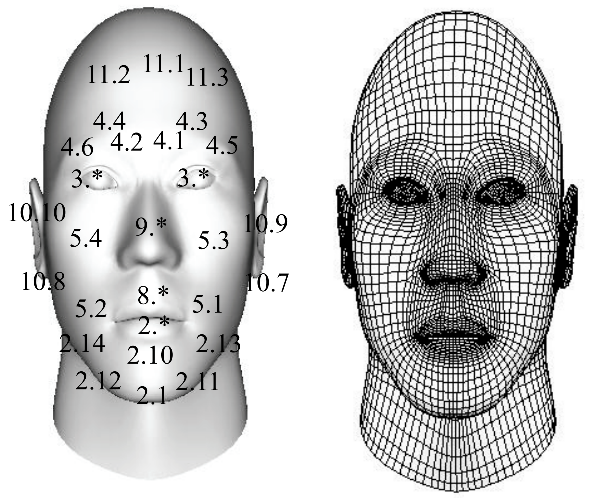
We use some feature points on the 3-D face model to control the transformation of the model. There are some existing techniques to determine the feature points on the 3-D face model. The MPEG-4 face animation standard is one of these techniques (Pandzic [20]). This technique specifies a number of face animation parameters (FAPs) that describe how a certain part of the face should be deformed and a number of facial definition parameters (FDPs) that make it possible to apply the FAPs on arbitrary face models and hence perform the same animation on different models. The total feature points of the MPEG-4 face animation standard are 84 and the FAPs are 68, which are divided into 10 groups shown in Table 1. Figure 7(a) shows some feature points defined on the 3-D face model. This technique defines the feature points on the head, eye, nose, mouth, tongue, and so on. For example, point 5.4 is the feature point on the left face in Figure 7(a). Because there is not enough space to tag all the feature points, a logogram is used. The 3.* means feature points 3.1~3.14 and are the feature points on the eyes. Controlling these feature points can generate an expression of the person. More detailed information about the MPEG-4 face animation standard can be found in Pandzic [20]. Each feature point is associated with an influence region. Thus, it is easy for us to get the parameters for the RBF network. In the next subsection, the feature points are used to transform the 3-D face model according to the 2-D Ukiyo-e. In section 5, the feature points are also used to generate some expressions of the Ukiyo-e actor.

3.2. Transformation of the 3-D face model

To synthesize the 3-D face model of Danjuro the eighth, the user inputs an Ukiyo-e of Danjuro the eighth and an original 3-D face model as shown in Figure 8. There are no photos or statues of Danjuro the eighth; the original 3-D face model is obtained from the 3-D range date of the face statue of Danjuro the ninth, who is the brother of Danjuro the eighth. At first, front image and side image are obtained from 3-D range data. Then, the rough 3-D face model is synthesized from these two images using commercial software "FaceGen Modeller," which can generate types of 3-D face models easily. This rough 3-D face model shown in Figure 7(b) is used as the original 3-D face model to synthesize a precise

Table 1. FAP groups and number of FAPs per group

Group	Number of Faps
1. Visemes and expressions	2
2. Jaw, chin, inner lowerlip, corner lips, mid lip	16
3. Eyeballs, pupils, eyelids	12
4. Eyebrow	8
5. Cheeks	4
6. Tongue	5
7. Head rotation	3
8. Outer-lip positions	10
9. Nose	4
10. Ears	4



(a) Feature points (b) Original face model

Fig. 7. Feature points defined in MPEG4.

(a) Define feature points (b) Registration result

Fig. 8. Register original face model to the picture.

3-D face model of Danjuro the eighth.

This system starts with registering the original face model to the Ukiyo-e. The purpose of registration is to get a rough match between Ukiyo-e and the 3-D face model. At first, the 3-D face model is rotated manually, and then the parameters of translation and scale are estimated at the same time. Equation 5 is the objective function we choose to register the 3-D face model to the 2-D Ukiyo-e picture.

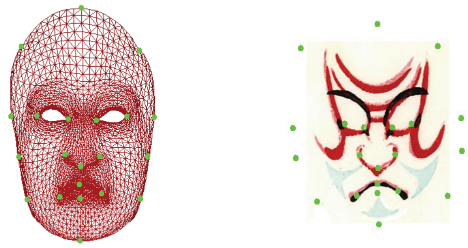
$$\rho(\mathbf{A},\mathbf{B}) = \sum_i \left((X_i BA[1,0,0,0]^T - u_i)^2 + (X_i BA[0,1,0,0]^T - v_i)^2 \right), (5)$$

where X_i is the 3-D feature point on the face model, (u_i, v_i) is the 2-D feature points on the Ukiyo-e picture, A is the homogeneous transformation matrix controlled by Euler angles, and B is the scale matrix. **A** and **B** are calculated by trying to minimize Equation 5. We select feature points manually and choose a traditional optimization algorithm, conjugating the gradient, to solve this problem. Figure. 8(b) illustrates the rough registration result.

Multilevel RBF is used to deform the 3-D face model to match the outline, eyes, nose, and mouth of the face. A user can also specify arbitrary feature points in the region when the original MPEG4 feature points are not enough to control the shape of the 3-D face model. Figures 9(a) and 9(b) illustrate the deformation result. The red points are feature points for the user to move and the system will deform the 3-D face model accordingly. In Figure 9(b), we can see how the registered face model is deformed to match the face features in Ukiyo-e (nose and mouth). Final synthesis results after texture mapping are shown in Figure 9(c). The shape of nose and mouth are different from the original face model, and the eye position and head shape change a little also.

4. Kumadori mapping

As mentioned before, the Kumadori images are used to map onto the 3-D face model. The Kumadori drawing will be used to introduce the mapping method. Some experimental results will be given using the Kumadori drawing and the Oshiguma in section 5.



(a) Face part of the 3-D face model (b) Kumadori drawing

Fig. 10. Feature points for Kumadori mapping.

Mapping Kumadori drawing to the 3-D face model also starts with defining the corresponding feature points. As shown in Figure 10(a), the 3-D face model surface is projected to a cylinder surface, then the 2-D mesh of the face can be obtained. The user only needs to define feature points at the eyes, mouth, nose, and some other feature points between the 2-D mesh and the Kumadori drawing. Green points in Figure 10 are the corresponding feature points. To use multilevel RBF to solve the texture mapping problem, we only need to treat the feature points on the synthesized 3-D face model as x_i and their corresponding feature points on the Kumadori drawing as y_i in Equation 4. Then, multilevel RBF can be used to compute texture coordinates for every 3-D point on the face part of the synthesized 3-D face model.

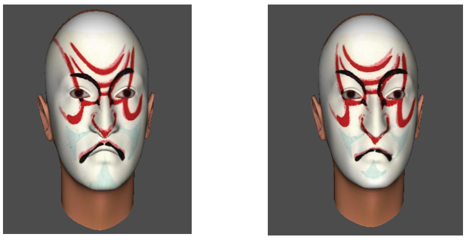
Tang [15] also adopts the RBF to solve the texture mapping problem, and he points out that RBF network with regulation term is suitable to texture mapping. However, this mapping problem is more difficult since we need to preserve the curving shape in the Kumadori drawing, and the correspondence of the feature points is not obvious and sometimes irregular. Furthermore, the multilevel RBF can enhance the precision of the RBF network, which is very important to texture mapping, as it emphasizes precise alignment of feature points. Figure 11 shows that the multilevel RBF can generate a precise match and the mapping result is much better. Single-level RBF failed to match the features for two reasons: one is that it cannot guarantee a precise match, the other is that there exists irregular correspondence in this case. Please note the correspondence between the mouths. It is hard to avoid artifacts by only increasing the feature points.

5. Results

The Kabuki face modeling system is running on a PC platform with 3.00GHz Intel Pentium-4 CPU and 1G memory. The operating system is Windows XP. It takes 1-2 s for multilevel RBF algorithm to solve the deformation and texture mapping problem. To make a 3-D Kabuki face, it needs about 15 min to do the interaction.

(a) MPEG4 feature points (b) Deformation result (c) Synthesis result seen from multi-viewpoints.

Fig. 9. Deformation and synthesis result.



(a) Single-level RBF mapping result

(b) Multilevel RBF mapping result

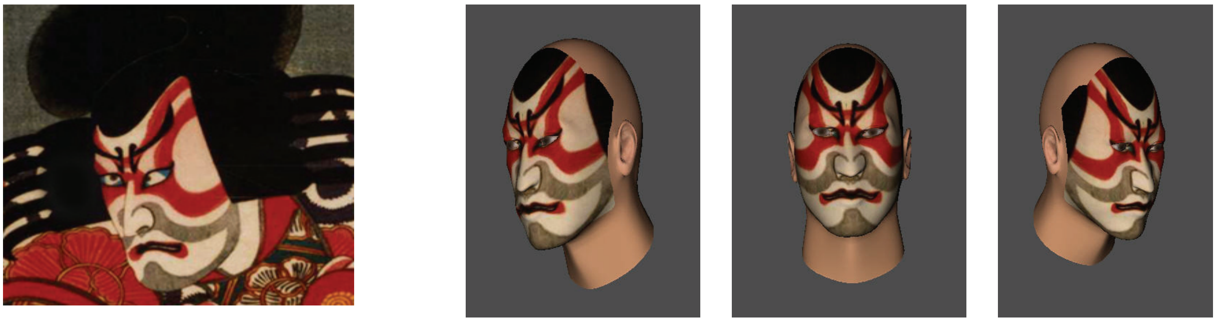
Fig. 11. Comparing mapping results.

In addition to the synthesis from the front view, we can also apply the synthesis method to the side view of the Ukiyo-e picture using the symmetry constraints in the face model (see Figure 12). The feature points defined in the MPEG-4 face animation standard facilitate the identification of symmetry constraints: for example, feature point 4.3 should be symmetric to feature point 4.4 (see Figure 7). Pandzic [20] lists constraints among MPEG-4 feature points. They are also the constraints we consider in the synthesis.

Figure 13 illustrates the Kumadori mapping result from a different Kumadori drawing. To demonstrate the ability of this texture mapping algorithm, we also applied it to some special makeup image named Lianpu in Peking opera onto the 3-D face model. Figure 14 illustrates this result. The pattern around the eye and mouth match the 3-D face model well in all of these cases because multilevel RBF is applied.

There are some Oshiguma acquired from the face of an ancient famous Kabuki actor, which is regarded as a treasure of Japanese culture. The picture in Figure 15(c) is one Oshiguma acquired from Danjuro the eighth who was a famous actor about 150 years ago. Mapping such kind of Oshiguma to the 3-D face model is of special meaning to the people. As shown in Figure 15(a), a 3-D face model obtained from the statue of Danjuro the ninth by a 3-D laser scanner is used to do Kumadori mapping. Since this ancient Oshiguma is obtained by putting a paper (or cloth) on the face of the actor after the performance, it is quite similar to cylindrical projection. Shown as Figure 15(b), the 3-D face model is projected to a cylindrical plane and some feature points on the Oshiguma are defined. Then, the Oshiguma is mapped onto the 3-D face model using multilevel RBF. In Figure 15, the mouth is very close to the nose. It is caused by the acquisition method of this Oshiguma. However, the result shows that this multilevel RBF approach can handle this quite well.

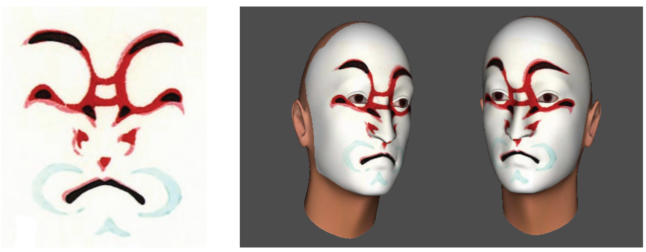
Since this technique is based on the MPEG4 face animation standard, it is easy to generate the animation of the 3-D face model. The feature points can be used to control the action of the eyes, mouth, and so on. Figure 16 shows some expression pictures of the 3-D Kabuki face model. Using this technique, the ancient Kabuki actor face model can say something in the virtual world and many subsequent applications can benefit from this, such as virtual environments, web applications, and so on.



(a) Side view of the Kabuki face in Ukiyo-e

(b) Synthesis result seen from different viewpoints

Fig. 12. Synthesis result from the side view.



(a) Kumadori drawing

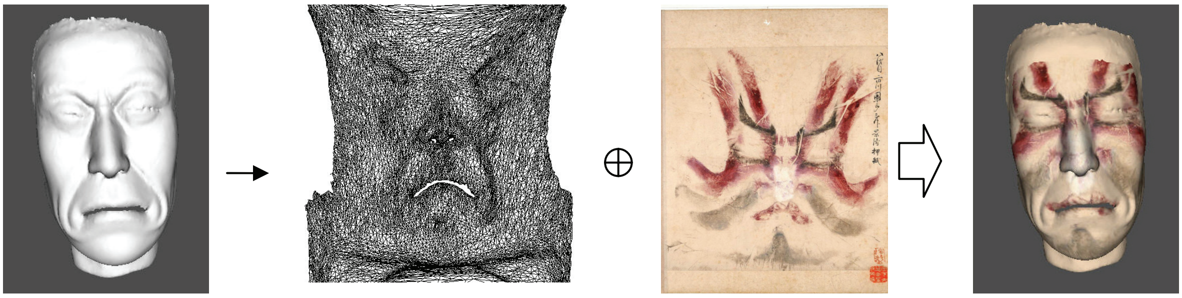
(b) Mapping result

Fig. 13. Kumadori mapping results using Kumadori drawing.

(a) Lianpu image in Peking Opera

(b) Mapping result

Fig. 14. Mapping makeup image named Lianpu in Peking Opera.



(a) A 3-D face model obtained from a statue of Danjuro the ninth

(b) 2-D mesh projected from 3-D face model

(c) An Oshiguma

(d) Mapping result

Fig. 15. Mapping an ancient Oshiguma.

(a) Happy (b) Surprise (c) Quiet (d) Fear (e) Angry

Figure. 16. Kabuki face expression images. The face model is transformed by controlling the MPEG4 points. Notice that the mouth and eye shapes change.

6. Conclusion and discussion

A 3-D synthesis method from ancient drawings of Kabuki actors has been described. The 3-D face model compatible with MPEG4 face animation standard is deformed according to the Ukiyo-e to get the 3-D geometry information, and then a texture mapping algorithm is used to map the texture to the synthesized 3-D face model. The main contributions of this study are developing the single-level RBF into a multi-level RBF and using this as an interpolation algorithm to solve CG problems such as 3-D mesh model deformation and mapping to achieve smoothness and precision simultaneously.

There is a new research hot spot on investigating how to apply the CG techniques to the field of cultural heritage. This face modeling system can be classified into this research field. The ancient Kabuki face model is synthesized from one drawing and the animation of the face can be made easily based on the MPEG4 face animation standard. As a limitation of the input (only one drawing and distortion of the drawing from the real face), it is very difficult to get the precise ancient person's face model comparable with the technique in physical anthropology, which synthesizes the head model based on the skull. The 3-D face model which we obtained can be used to other cultural heritage applications such as digital museum, entertainment, and so on.

Some problems remain to be solved. First, to synthesize the 3-D face model of an ancient Kabuki actor, the 3-D original face model obtained from the statue of Danjuro the ninth is used because they belong to the same family. For the others, the original 3-D face model of the Danjuro family cannot be used because they are far from each other. One idea is to use the 3-D average face model of a Japanese person 200 years ago as the original 3-D face model for transformation. In one future work, the 3-D ancient Japanese average face model needs to be synthesized on the basis of research results of physical anthropology. Second, the feature points of the face in the 2-D drawing are determined manually now. To facilitate the synthesis procedure, these feature points should be detected automatically and registered as MPEG4 face animation standard points on the 3-D face mesh model. To improve the visual effect, two things need to be done in future. One is adding the hair of the 3-D model of the Kabuki. Another is capturing the real expression of the Kabuki actor and applying the animation techniques to mimic the expressions in Kabuki.

Acknowledgments

This work is supported partly by the Grants-in-Aid for Scientific "Research Scientific Research(A) 17200013" and the "Encouragement of Young Scientists(B) 19700104" of the Japan Society for the Promotion of Science. This work is also part of the 21st Century Center of Excellence Program "Kyoto Art Entertainment Innovation Research" and the Global Center of Excellence Program “Digital Humanities Center for Japanese Arts and Culture."

References

[1] F. I. Parke, “Computer-generated animation of faces”, Proceedings of the ACM Annual Conference, vol. 1, pp: 451-457, 1972.

[2] Y. Lee, D. Terzopoulos, K. Walters, “Realistic modeling for facial animation”, Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’95), pp: 55-62, 1995.

[3] H. T. Tanaka, “Accuracy-based sampling and reconstruction with adaptive meshes for parallel hierarchical triangulation”, Computer Vision and Image Understanding, vol. 61, no.3, pp: 335-350, 1995.

[4] T. Sakaguchi, M.Ueno, S. Morishima, H. Harashima, “Analysis and synthesis of facial expression using high-definition wire frame model,” 2nd IEEE International Workshop on Robot and Human Communication 1993. (ROMAN93), pp. 194-199, 1993.

[5] T. Akimoto, Y. Suenaga, R. S. Wallace, “Automatic creation of 3-D facial models”, IEEE Computer Graphics and Applications, vol. 13, no. 5, pp: 16-22, 1993.

[6] F. Pighin, J. Hecker, D. Lischinski, R. Szeliski, D. H. Salesin, “Synthesizing realistic facial expressions from photographs”, Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’98), pp: 75-84, 1998.

[7] B. Guenter, C. Grimm, D. Wood, H. Malvar, F. Pighin, “Making faces”, Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’98), pp:55-66, 1998.

[8] Z. Liu, Z. Zhang, C. Jacobs, M. Cohen. “Rapid modeling of animated faces from video,” Journal of Visualization and Computer Animation, vol 12, no. 4, pp: 227-240, 2001 (Also in MSR-TR-2000-11).

[9] V. Blanz and T. Vetter, “A morphable model for the synthesis of 3D faces”, Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’99), pp: 187-194, 1999.

[10] V. Blanz, C. Basso, T. Poggio, T. Vetter, “Reanimating faces in images and videos”, Computer Graphics Forum, vol. 22, no. 3, pp: 641-650, 2003.

[11] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C.Rossl, H.P. Seidel, “Laplacian surface editing”, Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, pp: 179-188, 2004.

[12] K. Zhou, J. Huang, J. Synder, X. Liu, H. Bao, B. Guo, H. Y. Shum, “Large mesh deformation using volumetric graph laplacian”, Proceedings of the International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’05), pp: 496-503, 2005.

[13] P. J. Besl, N. D. McKay, “A method for registration of 3-D shapes”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp: 239-256, 1992.

[14] B. Levy, “Constrained texture mapping for polygonal meshes”, Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH’01), pp: 417-424, 2001.

[15] Y. Tang, J. Wang, H. Bao, Q. Peng, “RBF-based constrained texture mapping”, Computers and Graphics, vol. 27, no.3, pp: 415-422, 2003.

[16] S. Lee, G. Wolberg, S. Y. Shin, “Scattered data interpolation with multilevel B-Splines”, IEEE Transactions on Visualization and Computer Graphics, vol. 3, no. 3, pp: 228-244, 1997.

[17] S. Yamada, M. Hayakawa, K. Hanihara, "Zyouhou kagaku teki Ukiyo-e kennkyu no kokoromi", 67th research seminar report of the Data Processing Center of Kyoto University, pp. 62-69, 2001. (in Japanese).

[18] M. J. L. Orr, “Introduction to radial basis function networks”, Technical report, Center for Cognitive Science, University of Edinburgh, 1996.

[19] G.H. Golub, M. Heath, G. Wahba, “Generalized cross-validation as a method for choosing a good ridge parameter”, Technometrics, vol. 21, no. 2, pp: 215-223, 1979.

[20] I. S. Pandzic, R. Forchheimer, “MPEG-4 facial animation: the standard, experiment and applications”, John Wiley and Sons Ltd., ISBN: 0-470-84465-5, 2002.

[21] K. Balci, “Xface: MPEG-4 based open source toolkit for 3D facial animation”, Technical Report, http://xface.itc.it.