FACIAL ANIMATION FROM SEVERAL IMAGES

Yasuhiro MUKAIGAWA† Yuichi NAKAMURA‡ Yuichi OHTA‡

† Department of Information Technology, Faculty of Engineering, Okayama University
3-1-1 Tsushima-naka, Okayama, 700-8530 JAPAN
E-mail: mukaigaw@chino.it.okayama-u.ac.jp

‡ Institute of Information Sciences and Electronics, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573 JAPAN

Commission V, Working Group SIG

KEY WORDS: facial animation, facial expression, image-based rendering

ABSTRACT

We propose a novel method for synthesizing facial animation with 3-D pose and expression changes. On animation synthesis, one of the most important issues has been realistic face generation. Usual methods with 3-D facial model, however, have not realized natural face synthesis which represents the details and delicate changes of facial expressions.

In our method, a facial image is synthesized directly from multiple input images without explicit reconstruction of 3-D facial shape. Since this method uses the actual images, realistic facial animation which holds detailed facial features can be synthesized. The linear combination of multiple poses realizes the 3-D geometric appearance changes, and the texture blending is used for the smooth surface texture changes. Both of poses and expressions can be treated in a same framework in our method, while they are handled separately in the usual methods.

1 INTRODUCTION

A human face includes various information such as individuality and emotion. Techniques for generating facial animations have been studied for many applications, such as a man-machine interface and movies. However, a face is one of the most difficult objects for image synthesis, because we are extremely sensitive to differences between real face images and synthesized face images. In this paper, we deal with both pose and expression changes, and aim to synthesize realistic facial images which is almost indistinguishable from real images.

Model-based rendering usually have been used for this purpose. A 3-D shape model of a human head is often used and the shape is deformed according to the facial expression.

The 3-D shape model can be reconstructed from several images by structure-from-motion (Ullman S., 1979). But the reconstructed model usually includes some errors, so the synthesized image becomes unnatural. The acquisition of accurate model is difficult without special devices such as a high precision range finder (Akimoto T.,1993).

As the facial expression model, FACS (Facial Action Coding System) is often used (Ekman P.,1997). A facial expression is described as a combination of the AU (Action Unit). A facial image with an expression is synthesized by deformation defined for each AU, but it is difficult to simulate in details such as a wrinkle.

Thus, synthesized images are still far from a real face appearance as you may see in many applications. Even small modeling errors cause undesirable effect to the synthesized images.

On the other hand, there is another paradigm called *image-based rendering*. It aims to synthesize realistic images by using the textures from real images. For example, view morphing (Seitz S.M.,1996) method generates a new image easily, which generates intermediate views between two actual views.

In order to change the facial poses and expressions of an input image, Poggio, et al. proposed some methods which are related to *image-based rendering*. The *linear classes* method (Vetter T.,1995) assumes that one facial image can be represented as a linear combination of multiple facial images of other persons, and the method synthesizes a new image which has another pose and expression of the target person. However, this method requires a large number of images for reproducing the individuality. Also, the peculiarities of textures such as moles or birth-marks are not preserved. The *parallel deformation* method (Beymer D.,1993) synthesizes intermediate views of two expression images with two poses, but synthesized images are limited to intermediate views between the two views.

Therefore, to reproduce a true 3-D rotation with human individuality is still an open problem. For this purpose, we propose a new method to deal with both rotation and expression in a same framework. The linear combination is used for the 3-D geometric appearance changes, and the texture blending is used for the surface texture changes. These successfully avoid the difficult problems in 3-D reconstruction and rendering. In other words, our method realizes stable image-to-image conversion, that is from the input images to the output synthesized image, by combining *image-based* and *model-based* approach.

The principle of our framework is based on the structure-from-motion theory. Ullman, et al. (Ullman S.,1991) proved that an arbitrarily oriented object can be recognized by a linear combination of the 2-D coordinate values of feature points. We have previously applied this principle to image synthesis and showed that a facial image with arbitrary pose can be synthesized from only two images without explicit 3-D reconstruction (Mukaigawa Y.,1995). This method can be applied to facial expression synthesis.

2 BASIC SCHEME

Fig.1 shows the flow of the face synthesis process. A set of images with different facial poses and expressions is used as input. First, 2-D coordinate values of the feature points in the synthesized image are calculated by a linear combination of the feature points detected from the input images. Then, the blended texture which is taken from the input images is mapped on to the triangular patches whose vertices are the feature points. In the following section, we will explain the method for calculating the coordinate values and the texture mapping.

3 CALCULATION OF 2-D COORDINATE VALUE

3.1 Facial poses

Let B_1 and B_2 be two input images with different poses. We assume that the feature points are located

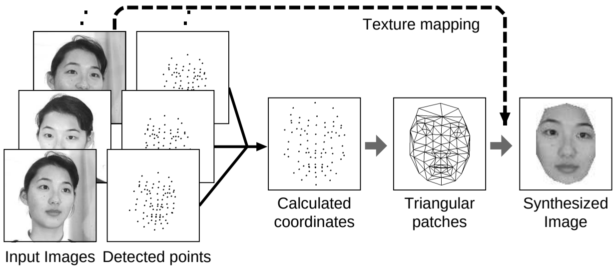


Figure 1: Flow of the face synthesis process

on the face, and that the correspondences of all feature points between two input images are known. Let (x_k^1, y_k^1) and (x_k^2, y_k^2) be 2-D coordinate values of the k-th feature point on images B_1 and B_2, respectively. These 2-D coordinate values are the result of rotating and projecting the points (X_k, Y_k, Z_k) of the 3-D space. The vectors and matrices which indicate these coordinate values are defined in the followings:

$$\boldsymbol{x}^1 = [x_1^1, x_2^1, \cdots, x_n^1] \quad (1)$$
$$\boldsymbol{y}^1 = [y_1^1, y_2^1, \cdots, y_n^1] \quad (2)$$
$$\boldsymbol{x}^2 = [x_1^2, x_2^2, \cdots, x_n^2] \quad (3)$$
$$\boldsymbol{y}^2 = [y_1^2, y_2^2, \cdots, y_n^2] \quad (4)$$

$$\boldsymbol{P} = \begin{bmatrix} X_1 & X_2 & \cdots & X_n \\ Y_1 & Y_2 & \cdots & Y_n \\ Z_1 & Z_2 & \cdots & Z_n \end{bmatrix} \quad (5)$$

For simplification, we assume rigidity, orthographic projection, and no translation. As shown in equations (6) and (7), the vectors which indicate the coordinate values of the feature points of each input image are represented as a multiplication of the 2×3 transformation matrix and the 3-D coordinate values of the feature points.

$$\begin{bmatrix} \boldsymbol{x}^1 \\ \boldsymbol{y}^1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}_x^1 \\ \boldsymbol{r}_y^1 \end{bmatrix} \boldsymbol{P} \quad (6)$$

$$\begin{bmatrix} \boldsymbol{x}^2 \\ \boldsymbol{y}^2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{r}_x^2 \\ \boldsymbol{r}_y^2 \end{bmatrix} \boldsymbol{P} \quad (7)$$

Let $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ be sets of 2-D coordinate values of the feature points in another view $\hat{B}$. Let $\hat{\boldsymbol{r}}_x$ and $\hat{\boldsymbol{r}}_y$ be the first row vector and the second row vector of the transformation matrix corresponding to $\hat{B}$. If $\boldsymbol{r}_x^1$, $\boldsymbol{r}_x^2$, $\boldsymbol{r}_y^1$ are linearly independent, a set of coefficients a_{x1}, a_{x2}, a_{x3} which satisfy equation(8) should exists, because the rank of $\hat{\boldsymbol{r}}_x$ is 3. This means that the X-coordinate values of the all feature points on $\hat{B}$ can be represented as a linear combination of 2-D coordinate values on B_1 and B_2, as shown in equation(9).

$$\hat{\boldsymbol{r}}_x = a_{x1}\boldsymbol{r}_x^1 + a_{x2}\boldsymbol{r}_x^2 + a_{x3}\boldsymbol{r}_y^1 \quad (8)$$
$$\hat{\boldsymbol{x}} = a_{x1}\boldsymbol{x}^1 + a_{x2}\boldsymbol{x}^2 + a_{x3}\boldsymbol{y}^1 \quad (9)$$

The base vectors of the linear combination are not always linearly independent. In order to get sufficient base vectors stably, we apply the principal component analysis of the four base vectors ($\boldsymbol{x}^1$, $\boldsymbol{y}^1$, $\boldsymbol{x}^2$, $\boldsymbol{y}^2$), and we use the first three eigen vectors ($\boldsymbol{p}_1$, $\boldsymbol{p}_2$, $\boldsymbol{p}_3$) as the base vectors. As shown in equations (10) and (11), X- and Y-coordinate values can be stably represented as linear combinations of the linearly independent vectors.

$$\hat{\boldsymbol{x}} = a_{x1}\boldsymbol{p}_1 + a_{x2}\boldsymbol{p}_2 + a_{x3}\boldsymbol{p}_3 \qquad (10)$$

$$\hat{\boldsymbol{y}} = a_{y1}\boldsymbol{p}_1 + a_{y2}\boldsymbol{p}_2 + a_{y3}\boldsymbol{p}_3 \qquad (11)$$

As shown above, the 2-D coordinate values of the feature points can be easily calculated once the coefficients of the linear combination are obtained. In order to determine the coefficients corresponding to the synthesized image, we have two ways; specifying by a reference image and directly specifying the orientation.

Specifying by a reference image : A reference image is used for specifying the pose. We synthesize a new image whose pose is the same as this reference image. On the reference images, at least four representative points need to be detected. The coefficients are calculated so that these points of the reference image coincide with the corresponding points of the input images by using a least square method.

Directly specifying the orientation : For specifying the pose with an angle, we use representative points whose 3-D coordinates are known. We call these points control points. The control points are rotated to the requested pose, and projected on to the 2-D coordinates. By using these 2-D coordinate values, the coefficients are determined in the same way using reference images.

The coefficients of the linear combination are determined from representative points, and the 2-D coordinate values of all feature points on the facial image with arbitrary poses can be calculated.

3.2 Facial expressions

We try to synthesize a new image with arbitrary expression. Let $B_j (1 \leq j \leq m)$ be input images which have the same pose but different facial expressions. If these input images include a sufficient variety of facial expressions, the coordinate values of the feature points with any expressions can be approximated by the linear combination of those in the input images.

Let $\boldsymbol{x}_j$ and $\boldsymbol{y}_j$ be sets of coordinate values of the feature points on the input image B_j. As shown in equation (12), the vectors $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$, which indicate sets of coordinate values on a new facial expression $\hat{B}$, are represented as a linear combination. The 2-D coordinate values of the feature points are calculated with the appropriate coefficients $b_1, \cdots, b_m$.

$$\hat{\boldsymbol{x}} = \sum_{j=1}^{m} b_j \boldsymbol{x}_j \ , \qquad \hat{\boldsymbol{y}} = \sum_{j=1}^{m} b_j \boldsymbol{y}_j \qquad (12)$$

It is convenient to specify the facial expression of the synthesized image by the interpolation of the input images, because no heuristic knowledge is necessary. In this method, a variety of synthesized facial expression depends on the kind of the input images. In other words, if we can prepare a sufficient variety of facial expressions as input images, we can obtain more realistic images.

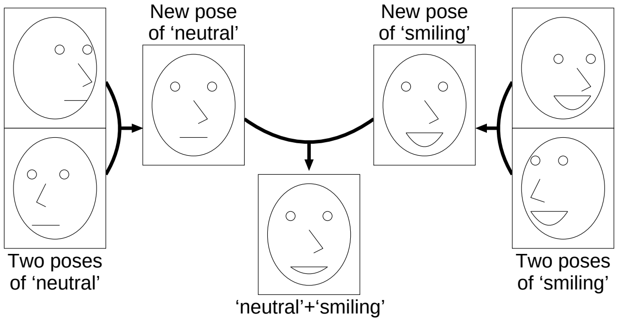


Figure 2: Integration of pose and expression generation

3.3 Integration of the poses and expressions

Since both pose and expression are represented as a linear combination of the coordinate values of the feature points, these can be easily integrated. We assume that a set of input images includes some facial expressions and has at least two different poses for each expression. As shown in Fig.2, we try to synthesize a new image with arbitrary pose and expression from the input images. First, the facial pose in the input images is adjusted to a specified pose for each expression by the method explained in section 3.1. Then, the facial expression is adjusted to a specified expression by the method explained in the section 3.2.

The 2-D coordinate values corresponding to a facial pose with an expression can be represented as the linear combination of base vectors shown in equations (13) and (14). Note that a^j_{xi}, a^j_{yi}, and $\boldsymbol{p}^j_i$ $(i = 1, 2, 3)$ are the coefficients and the base vectors of the j-th expression in the equations (10) and (11), respectively.

$$\hat{\boldsymbol{x}} = \sum_{j=1}^{m} b_j (\sum_{i=1}^{3} a^j_{xi} \boldsymbol{p}^j_i) = \sum_{j=1}^{m}\sum_{i=1}^{3} b_j a^j_{xi} \boldsymbol{p}^j_i \qquad (13)$$

$$\hat{\boldsymbol{y}} = \sum_{j=1}^{m} b_j (\sum_{i=1}^{3} a^j_{yi} \boldsymbol{p}^j_i) = \sum_{j=1}^{m}\sum_{i=1}^{3} b_j a^j_{yi} \boldsymbol{p}^j_i \qquad (14)$$

4 TEXTURE MAPPING

4.1 Texture blending

Basically, the textures taken from input images which have similar poses and expressions are mapped onto the synthesized image. However, if we take all the textures from one image with the closest facial pose, the synthesized image will be warped unnaturally as the facial pose changes. This undesirable warping is caused by a drastic deformation of texture. The same can be said to the facial expression. It is obvious that the natural expression can be synthesized by directly using textures of the similar expression rather than by deforming the textures of different expressions. For example, the wrinkles that appear when smiling cannot be synthesized by warping the texture taken from a neutral expression.

We can solve this problem by texture blending. In our method, facial images are synthesized by mapping the blended texture taken from multiple input images. The weight for blending is set larger for the similar pose and expression.

Although the facial orientation of the input image is unknown, a rough orientation can be estimated by the factorization method (Tomasi C.,1992). First, several feature points which do not move by the facial expression changes, such as the top of nose, are selected. Then, the relative facial orientation of the reference image is estimated. Since this value is only used for determining the texture blending weights, small errors are not critical. The weights of the texture blending are determined to be inversely proportional to the square of the angle difference of facial orientations. The weights of the blending for facial expression are determined in proportion to the coefficients b_j in the equation (12).

4.2 Two dimensional texture mapping

The texture is mapped by using triangular patches whose vertices are the feature points. For each triangular patch, the texture is clipped from multiple input images, and deformed by affine transformation according to the 2-D coordinate values of the feature points. Then, the textures are blended by using the weights and mapped. In this step, it is checked whether each patch is obverse or reverse. The texture is not mapped on to the reverse patch. This simple judgment is enough for facial image synthesis, because complicated occlusion never occurs.

5 EXPERIMENTS

We chose 86 feature points on a face. The number is relatively small compared to the ordinary facial models used in other applications. These feature points are located not only on the facial components such as

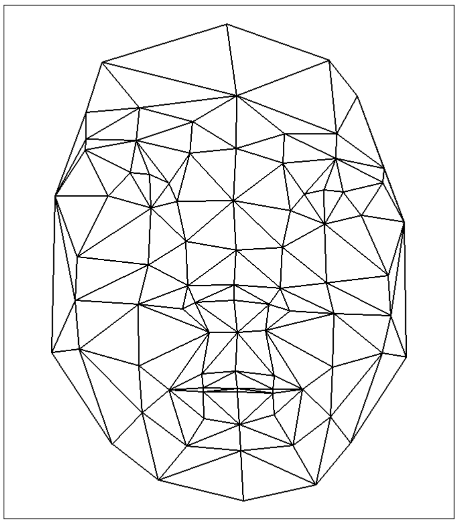
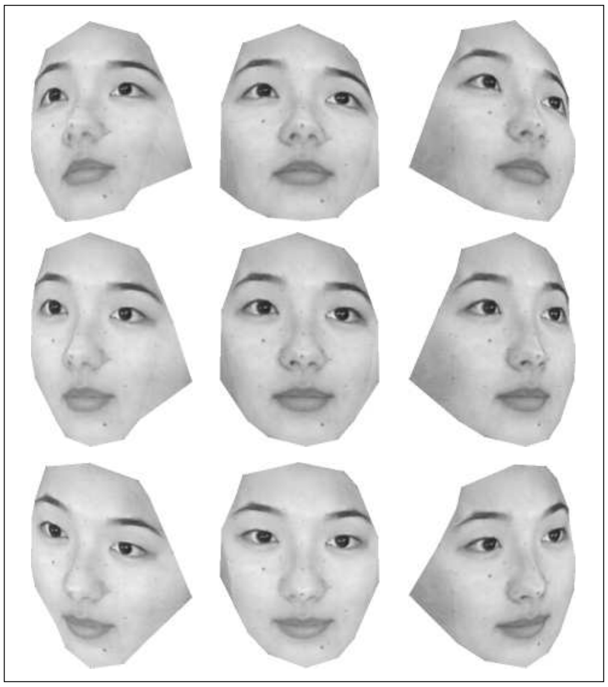


Figure 3: 2-D triangular patches

(a) Two input images



(b) Synthesized images

Figure 4: Synthesized images of various poses using two input images

(a) Reference image (b) Synthesized image

Figure 5: Reference image and synthesized image

eyes and mouth, but also on the movable parts such as cheeks. The 2-D coordinate values of the points are given manually by referring to the marks drawn on the face. The 156 triangular patches are created as shown in Fig. 3.

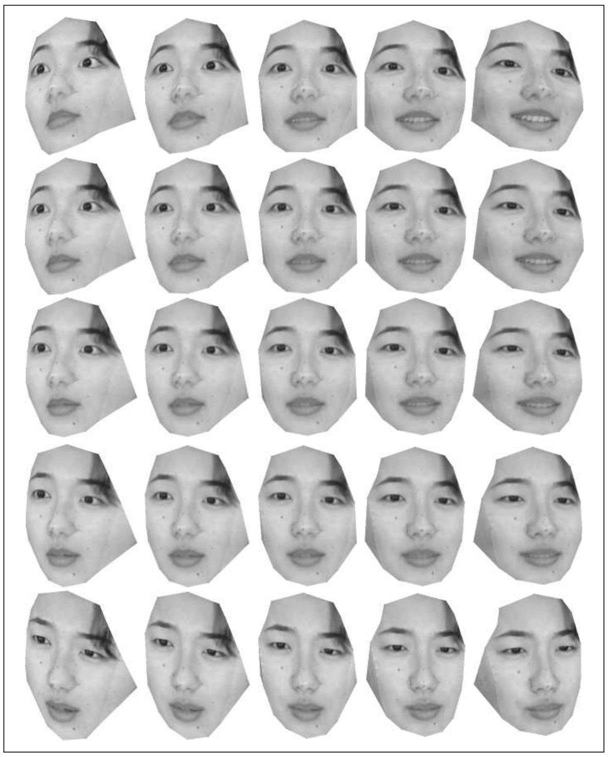
First, we show the experimental results whose facial poses are specified directly by the control points. Two input images are shown in Fig.4(a). Five points (ends of both eyes, top of nose, and bottoms of both ears) were chosen as the control points with known 3-D coordinates. The synthesized images with various poses are shown in Fig.4(b). We can see that true 3-D rotations are realized.

Next, in order to compare the synthesized image and the real image, we synthesize a new image which has the same pose as the reference image from two input images shown in Fig.4(a). A reference image was prepared as shown in Fig.5(a). On the reference image, five feature points (ends of both eyes, top of nose, and bottoms of both ears) were manually detected. A new facial image was synthesized and overwritten onto the reference image as shown in Fig.5(b). The outer region of the face such as hair, boundary, and background is copied from the reference image and the inner region of the face is overlaid by the synthesized image. As we can see in this example, the synthesized images are often indistinguishable from real images.

Next, we show the results of both facial poses and expressions changes. Eight input images (two poses for every four expressions) are shown in Fig.6(a). The facial poses are changed by the same method as the previous experiment. The facial expressions are also changed by the interpolation of four expression, as show in Fig.6(b). We can see that both the poses and the expressions are treated in a unified way.

Lastly, we show the results of changing facial expressions of a reference image. The input images include two different poses for every three different expressions as shown in Fig.7(a). Fig.7(b) shows two reference images which are the same person as the input images. New facial animations with different

(a) Input images with four expressions



(b) Synthesized images

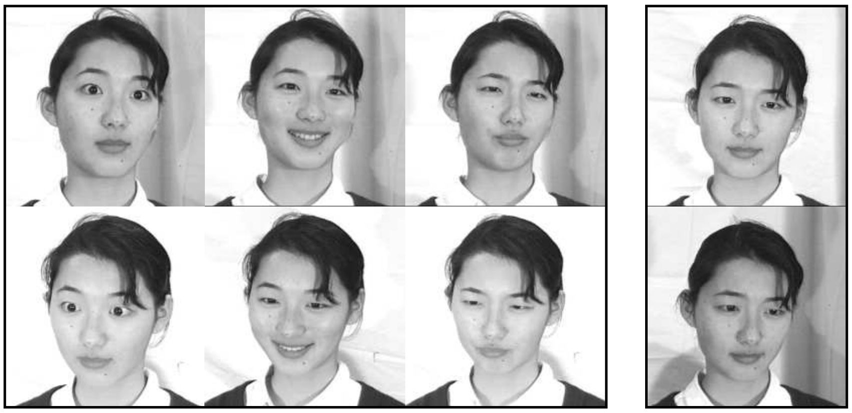
Figure 6: Synthesized images of various poses and expressions

expressions were synthesized and overlaid onto each reference image as shown in Fig.7(c). We can see that the realistic animations can be generated.

6 CONCLUSION

We have proposed a new method for synthesizing facial animations with arbitrary poses and expressions without explicitly modeling the 3-D shape and the facial expression. We have shown that both the poses and the expressions are treated in a unified way. We have implemented this method and demonstrated that the natural facial images can be synthesized.

The number of facial expressions that can be synthesized in the current system is limited, because we



(a) Input images with three expressions (b) Reference images

(c) Synthesized animations with various expressions

Figure 7: The results of changing facial expressions of reference images

used only few kinds of facial expressions. In order to synthesize *arbitrary* facial expression, we are now working on a new framework which utilizes a set of typical expressions obtained from a sequence of images.

REFERENCE

[1] Akimoto T. and Suenaga Y., 1993. Automatic Creation of 3D Facial Models. IEEE Computer Graphics and Applications, September 1993, pp.16–22.

[2] Beymer D., Shashua A. and Poggio T., 1993. Example based image analysis and synthesis. A.I.Memo No.1431, Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

[3] Ekman P. and Friesen W.V., 1997. Facial action coding system. Consulting Psychologists Press.

[4] Mukaigawa Y., Nakamura Y. and Ohta Y., 1995. Synthesis of Arbitrarily Oriented Face Views from Two Images. Proc. Asian Conference on Computer Vision (ACCV'95), Vol.3, pp.718–722.

[5] Seitz S.M. and Dyer C.R., 1996. View Morphing. Proc. SIGGRAPH'96, pp.21–30.

[6] Tomasi C. and Kanade T., 1992. The factorization method for the recovery of shape and motion from image streams. Proc. Image Understanding Workshop, pp. 459–472.

[7] Ullman S., 1979. The interpretations of visual motion. MIT Press, Cambridge, MA.

[8] Ullman S. and Basri R., 1991. Recognition by linear combinations of models. IEEE Trans. PAMI, Vol.13, No.10, pp. 992–1006.

[9] Vetter T. and Poggio T., 1995. Linear object classes and image synthesis from a single example image. A.I.Memo No.1531, Artificial Intelligence Laboratory, Massachusetts Institute of Technology. a