Facial Expression Morphing and Animation with Local Warping Methods*

Daw-Tung Lin and Han Huang
Department of Computer Science and Information Engineering
Chung Hua University
30 Tung-shiang, Hsin-chu, Taiwan 30067
Email: dalton@chu.edu.tw
FAX:011-886-3-5373771

Abstract

In this paper, we applied local warping techniques to construct a frame work of image-based morphing for facial expressions animation with low complexity. From the experience we obtained in various spatial transformations, the local warping methods are more adequate for facial expressions morphing. We utilized two transformations that can deal with local deformations: piecewise polynomial transformation and radial basis function transformation. The simulation results are satisfactory. The facial warping techniques could be applied to various applications, such as face recognition, authentication, dynamic imagery, speech (action) semantic animation, multimedia and virtual reality construction, low-bandwidth video conference image transmission, and intelligent man-machine interface.

1 Introduction

Facial expression warping is usually applied in animation and it plays an important role in computer graphics. Several models has been proposed to simulate actions of facial expressions. The direct way to is to construct a 3D facial model. However, it is difficult to build appropriate 3D facial model by using single 2D face image. Researchers advocate that the information of a 2D gray-scale image is plenty enough for facial expressions processing[7, 1]. They regard 2D facial image as a 2D object model, and then warp the facial expressions directly in 2D space. We adopt their methodologies and develop this work.

In order to get a good approximation to actual facial expressions with computer, it is necessary to analyze the structure of the human face. A widely used scheme for describing facial expressions was developed by Ekman and his colleagues in 1977. The system named FACS (Facial Action Coding System) describes the set of all possible basic actions (Action Units) performable on a human face and their effects on facial expressions. Each facial expression is expressed as a set of action units. Parke proposed another well-known model in facial animation: key-frame system and parameterized facial model[9]. There are also other models such as Platt model based on underlying facial structure[10], and the Nahas model which is based on the B-spline[8].

We turned to find 2D spatial transformations and used them for facial expressions warping [3, 11]. We regard the input 2D facial image as a 2D object, and used spatial transformations on the image plane to obtain the facial expressions results. Warping defines a mapping function that establishes a spatial correspondence between all points in a source image and its distorted counterpart (destination image). The schematic diagram is shown in Figure 1. The mapping function $f(x, y)$ is obtained according to some reference points in source image (indicated in the left-hand side in Figure 1) and counterpart of reference points in destination image (right-hand side in Figure 1). Global transformations that impose a single mapping function upon the whole image often do not account for local geometric distortions. We tried to find methodologies of local transformation to implement the facial expressions application. Intuitively, we have to find out these subregions of the facial image. Distinct control points are the points whose mapping is predetermined. We used two local transformations: one is linear piecewise polynomial transformation, and the other is radial basis functions transformation. Through computer simulations we concluded that the local warping methods are more suitable for facial expressions morphing. Contrast to Parke's idea, We first specify the appropriate set of parameter values and then the expressions in-between are produced in real-time with warping techniques according to those new anchor points in-between, where these new anchor points are calculated by interpolation.

*This work was supported in part by National Science Council. Taiwan, R.O.C. under grant NSC 87-2213-E216-006.

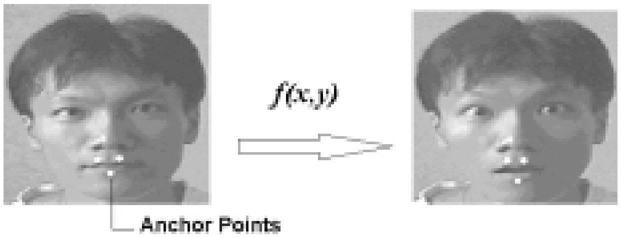


Figure 1. The schematic diagram of warping process.

2 Linear Piecewise Polynomial Mapping

Linear Piecewise Polynomial (LPP) Transformation is often applied to image registration problems [4]. Image registration is the process of overlaying two images of the same scene. Control points are uniquely identifiable points whose mapping is predetermined, like the reference points (anchor points) that described before. In piecewise polynomial transformation, control points are some of the special reference points such as line intersections, center of gravity of closed-boundary regions, or high curvature points, in an image. The mapping process can be conveniently posed as the surface fitting problem that finds a surface function to pass through control points. The idea of LPP transformation is, rather than using one global mapping function to register the whole image, but using a number of local mapping functions, each tuned to map well in local neighborhoods. Then by gathering these piecewise local mapping functions, a global mapping function is obtained.

This process divides the whole image into some triangular regions (planes) that refer to the input of control points, said "triangulation"[6]. Every triangular patch is independent with on overlapping. Triangulation is the process that tessellates the convex hull of a set of N distinct points into triangular regions. It connects neighbor control points to form a planar graph that does not have crossing line segments. Although many configurations are possible, we are interested in finding an optimal triangulation that have the optimal locality. The optimal locality is in the sense that points inside a triangular patch are closer to its three peak vertices than to that of any other triangular patches. Lawson[5] had proposed three criteria to optimize an arbitrary triangulation.

Given n distinct corresponding control points in two images of the same scene, $[(u_i, v_i), (x_i, y_i)]$, $i = 1, \ldots, n$. (u_i, v_i) are control points in the source image, and (x_i, y_i) are corresponding ones in the destination image. After optimal triangulation is proceeded, the source image is divided into several independent triangular patches. Then our work is to determine piecewise linear mapping function for every triangular patch: $x = f(u, v), andy = g(u, v)$, where $[u, v]$ is the pixel of the input image corresponding to warped pixel $[x, y]$. f and g represent the x-coordinate mapping function and y-coordinate mapping function, respectively.

The problem can be posed as finding a surface that passes through (u_i, v_i, x_i), and another surface that passes through (u_i, v_i, y_i). We now consider the case of fitting the triangular patches with linear interpolation, *i.e.*, a plane. Each triangular region have three peak points that are control points. For X-component coordinate, according these three given control points, the equation of a plane that pass through these three points(u_1, v_1, x_1), (u_2, v_2, x_2), and (u_3, v_3, x_3) is given by

$$Au + Bv + Cx + D = 0$$

where

$$A = \begin{vmatrix} v_1 & x_1 & 1 \\ v_2 & x_2 & 1 \\ v_3 & x_3 & 1 \end{vmatrix}; B = -\begin{vmatrix} u_1 & x_1 & 1 \\ u_2 & x_2 & 1 \\ u_3 & x_3 & 1 \end{vmatrix};$$
$$C = \begin{vmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ u_3 & v_3 & 1 \end{vmatrix}; D = -\begin{vmatrix} u_1 & v_1 & x_1 \\ u_2 & v_2 & x_2 \\ u_3 & v_3 & x_3 \end{vmatrix}$$

Then, we can determine the X-component coordinate of the other points that inside the triangular patch. And the Y-component coordinate is determined similarly.

3 Radial Basis Functions Transformation

In this section, we introduce another local warping method using radial basis functions. Radial basis functions (RBF) warping regards the whole image as a 2D object that consisting of some radial basis functions that are constructed by the anchor points, and those functions may affect each other. RBFs are proposed usefully in $R^d \to R$ dimensional transformation. As the input image is a two dimensional object, and have n pixels, we want to find a $R^2 \to R^2$ dimensional transformation: $T(\bar{u}_i) = \bar{x}_i$, for $i = 1, 2, ..., n$, where $\bar{u}_i, \bar{x}_i$ are denoted as the input vector and output vector respectively, $\bar{u}_i, \bar{x}_i \in R^2$. We need a pair of transformation $T = (T_x(u, v),\, T_y(u, v))$ to achieve $R^2 \to R^2$ transformation. The most simple format of the radial basis functions transformation is the pure radial sum: $R(\bar{u}) = \sum_{i=1}^{n} a_i g(||\bar{u} - \bar{u}^i||)$, where g is an univariate radial basis function, and $||\cdot||$ denotes the Euclidean norm on R^d. The feature of radial basis function is that, for each data point $\bar{u}^i$ will have the same effect on all points of equal distance to it. Bookstein[2] has proposed the thin-plate splines which is a subclass of radial basis functions (RBF) for image registration, and suggested that RBF provides an attractive framework for image warping. But thin-plate splines has global effect that does not suitable for facial expressions warping. Arad[7] used Gaussian function with local effect to handle local warping of facial expressions. This is more suitable for our locality requirement that pixels leave away from the anchor point must have non effect from the anchor point. Moreover, Gaussian function have a locality parameter σ that can adjust the local influence. Therefore, we use Gaussian-like radial functions to implement our facial expressions warping.

However, the pure radial sum is the collection of all radial basis functions. It therefore can't reproduce polynomials. We add an polynomial component to it:

$$T(\bar{u}) = \sum_{i=1}^{n} a_i g(\|\bar{u} - \bar{u}^i\|) + p_m(\bar{u}), \quad p_m(\bar{u}) \in \Pi_m \quad (1)$$

where Π_m is the space of all algebraic polynomials of degree at most m on R^d. In our application of facial expression warping, one of the natural scene distortion is "affine" that includes scaling, rotation, shearing, and translation in the whole image. Another distortion in facial expression warping is local distortion on facial organs, and can be realized by radial component with Gaussian-like mapping functions. Therefore, we combine the affine mapping function into polynomial component. Thus, our radial basis functions transformation(RBFT) looks like:

$$T(\bar{u}) = R(\bar{u}) + A(\bar{u}) \quad (2)$$

where $R(\bar{u})$ is the radial part of transformation, and $A(\bar{u})$ is the affine part of transformation. We need two pair of functions to achieve $R^2 \rightarrow R^2$ transformation. Thus

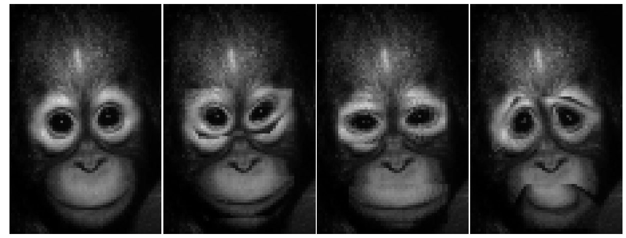
$$\begin{aligned} T &= (T_x(u,v), T_y(u,v)) \quad (3) \\ &= (\alpha_1 + \alpha_2 u + \alpha_3 v + \sum_{i=1}^{n} a_i g_i(u,v), \beta_1 + \beta_2 u + \\ &\quad \beta_3 v + \sum_{i=1}^{n} b_i g_i(u,v)) \end{aligned}$$

Our goal is to calculate $\alpha_1,\ \alpha_2,\ \alpha_3,\ \beta_1,\ \beta_2,\ \beta_3$ of the affine part, and $a_i,\ b_i$ that are coefficients of the radial part. We illustrated how to solve these coefficients in below.

Given n radial anchor points $P_1 = (u_1, v_1),\ P_2 = (u_2, v_2),\ \cdots,\ P_n = (u_n, v_n)$ in the Euclidean plane, and defined first three anchor points $P_1 = (u_1, v_1),\ P_2 = (u_2, v_2),\ P_3 = (u_3, v_3)$ are affine anchor points, and the others are radial ones. From these input information, we have to solve affine coefficients $\alpha_i,\ \beta_i$ first. Then we can obtain the effect of affine part $A(\bar{u}_i)$. The effect of radial part is derived from Equation 2 that

$$R(\bar{u}_i) = \bar{x}_i - A(\bar{u}_i),\ i = 1, 2, ..., n \quad (4)$$

After the affine part $A(\bar{x})$ is obtained, we turn to compute the radial part $R(\bar{x})$. The radial anchor points are $P_4 = (u_4, v_4),\ P_5 = (u_5, v_5), \cdots, P_n = (u_n, v_n)$ in our previous example, and their corresponding destination positions are $P'_4 = (x_4, y_4),\ P'_5 = (x_5, y_5),\ \cdots,\ P'_n = (x_n, y_n)$. Let $r_{ij} = |P_i - P_j|$ be the distance between anchor point i and j. From Equation 4, we get:



(a) (b) (c) (d)

Figure 2. Synthesis images after various perspective warping.

$$\begin{bmatrix} x_4 - A(u_4) \\ x_5 - A(u_5) \\ \vdots \\ x_n - A(u_n) \end{bmatrix} = \begin{bmatrix} g(0) & \cdots & g(r_{4n}) \\ g(r_{51}) & \cdots & g(r_{5n}) \\ \vdots & \ddots & \vdots \\ g(r_{n4}) & \cdots & g(0) \end{bmatrix} \begin{bmatrix} a_4 \\ a_5 \\ \vdots \\ a_n \end{bmatrix} \quad (5)$$

for x-coordinate, similarly for y-coordinate. Then, with Equation 2 we build the radial basis functions transformation where all coefficients are obtain from Equation 5. The affine part of radial basis functions transformation is used to overcome changes of viewpoint and scaling. The radial part is used to handle local distortions of facial expressions.

4 Simulation Results and Animation

4.1 Warping with Global Transformation

We have implemented the perspective, bilinear, and two-degree polynomial transformations to facial expressions warping. Figure 2 presents the results of perspective transformation on monkey face images. As we have discussed in Section 1 that, the perspective transformation is a global transformation method. However the facial expression change possesses local effects, we therefore cut the main deformation regions (eyes and mouth) of face image and perform perspective warping separately with various pairs of mapping points (anchor points). Figure 2(a) is the whole original image, and Figure 2(b)(c)(d) are the synthesis images which replace the corresponding eyes regions and mouth regions with various warping results. The overall presentation was not satisfactory in the sense that the boundaries of deformation regions are apparent.

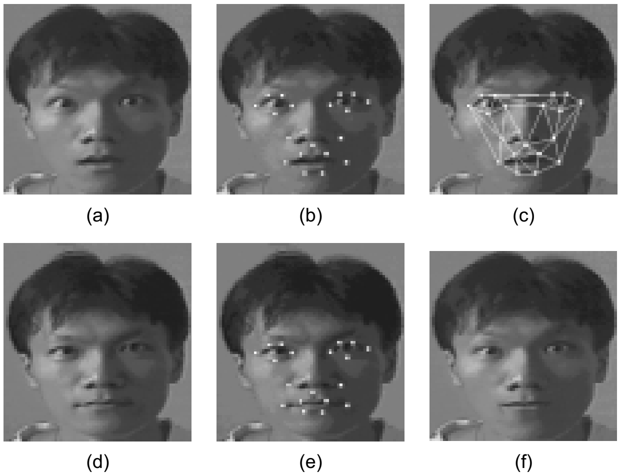


Figure 3. Fear expression to normal expression warping with Linear Piecewise Polynomial transformation.

4.2 Warping with Linear Piecewise Polynomial Transformation

We have implemented the LPP method to facial expression warping with the procedure of triangulation and mapping as illustrated in Section 2. We used a recursive algorithm to determine the optimal triangulation efficiently[6]. This algorithm applies the technique of ”divide-and-conquer”. It recursively splits control points into halves using their x coordinate values, until each subset contains only three or two points. Then merge these subsets into larger subsets using Lawson’s criteria and the convex hull algorithm. Then, a linear mapping (affine mapping) is performed on each triangle patch.

We used a man’s facial image in this demonstration. Our goal is to warp a fear expression to the normal expression. Figure 3(a) is the source image of fear expression, and Figure 3(b) is the source image with its control points. Figure 3(d) is the desired reference image of normal expression, and Figure 3(e) is the image with its desired control points that correspond to the control points in the source image Figure 3(b). Figure 3(c) is the result of optimal triangulation from the source image Figure 3(b). Finally we got the resulting image as shown in Figure 3(f). In order to observe the local effects of LPP warping, we calculated the difference between the original image and the warped image, and the difference image is plotted in Figure 4(c), where Figure 4(a) is the source image (re-plotted from Figure 3(a)), and Figure 4(b) is the resulting warped image (re-plotted from Figure 3(f)). From Figure 4(c) we can see that the only changes are located in the triangular patches, while the rest of the image is not affected.

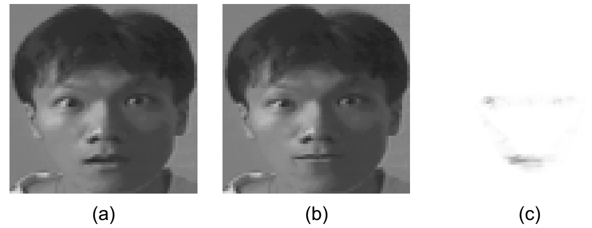


Figure 4. The difference image Linear Piecewise Polynomial transformation result and the original image.

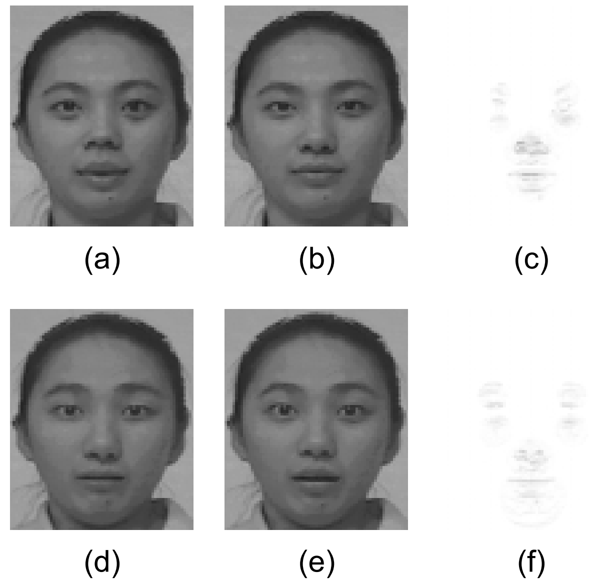


Figure 5. Difference images of RBF warping.

4.3 Warping with Radial Basis Function Transformation

We demonstrate the local influence of radial basis warping in Figure 5. Figure 5(a) is the warped image from normal expression to surprise expression, and Figure 5(b) is the source image. Figure 5(c) is the difference image of Figure 5(a) and Figure 5(b). Similarly, Figure 5(d) is the resulting warping image from surprise expression back to normal expression, and Figure 5(e) is the source image. We can see from Figure 5(c) and Figure 5(f) that radial basis functions have great locality, since the only change in these pictures are in the region of eyes and mouth where really changes in human facial expression various, while the rest of the image does not change such as hairs, neck etc.

The sigma value σ of Gaussian function plays an important role of local influence in RBF Warping. We engaged the experiments with several different sigma values on the

Figure 6. The original baby image.

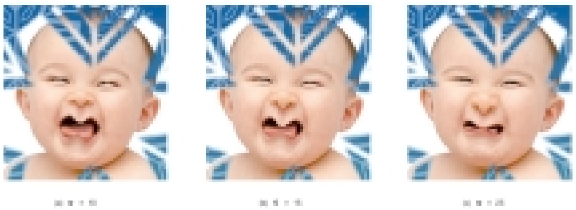


Figure 7. Different sigma values of RBF Warping. (a) $\sigma = 10$**, (b)** $\sigma = 15$**, (c)** $\sigma = 25$**.**

image shown in Figure 6. By changing the value of σ, we obtained different results. Figure 7(a), (b), (c) are the resulting warping image of σ values 10, 15, 25, respectively. We observed that the larger the value changes, the area affected became wider. The selection of sigma values σ of RBF warping are subjective. They depend on the size of facial image and the location. From the image size point of view, the sigma values in a 145×175 facial image are about 5, and ones in a 237×250 facial image are bout 10 in our experiments. From the location point of view, the affected regions of facial muscles are not the same, such as the influences of eyes are smaller than ones of the mouth. In our experiments, the local deformation of mouth is the largest. Therefore, the sigma values of anchor points that locate near mouth is greater than ones of anchor points that locate in other places of face.

It also requires some experimental techniques to define the amount of anchor points and the location of anchor points. We illustrate this phenomenon in Figure 8. Figure 8(a), (b), (c) are the input image with different amount of anchor points 12, 4, and 1 respectively. All sigma values of anchor points in Figure 8 were set to 10. Figure 8(g), (h), (i) are the warping results with 12, 4, and 1 anchor point, respectively. We prefer to use less anchor points as we could, due to the speed of computation. Large amount of anchor points will reduce the performance in real time warping implementation.

Figure 8. RBF warping with different amounts of anchor points.

4.4 Facial Expression Warping Animation

We also developed an animation warping tool. Figure 9 and Figure 10 show the interface and the final pose of the animation, respectively. Users can define the number of frames before the processing of warping. We use the equivalent interpolation method that equalizes the difference distance of anchor points at each frame. Moreover, our animation warping is not like the morphing processing that has the target image to interpolate frames, it only warps from source image. We first specify the appropriate set of parameters and then the expressions in-between are produced in

Figure 9. The Interface of the animation tool.

Figure 10. The final pose of animation.

real-time with warping techniques according to those new anchor points in-between, where these new anchor points are calculated by interpolation. The animation results of LPP and RBF warping are gratifying. In our experiments, the computation speed of LPP warping is faster than that of RBF warping. Because RBF warping with Gaussian function consumes much time in the exponential calculation. The solution is to use look-up table of the order of the image size in order to avoid direct Gaussian calculation.

5 Conclusions

In recent years, facial expressions warping was applied some interesting visual effects on multimedia. We have constructed a frame work of facial expression animation with less efforts of preprocessing by using local warping methods. From the experience we obtained in various spatial transformations, the local warping methods are more adequate for facial expressions warping. We used two transformations that can deal with local deformations: one is ”Linear Piecewise Polynomial Transformation” and another is ”Radial Basis Function Transformation”. Due to the warping of facial expressions is local deformations that constructed by several muscles, using local transformations produces a good approximation to actual face.

The main difference of anchor points (control points) between radial basis functions warping and piecewise polynomial warping is that, it must at least three anchor points exists in piecewise polynomial warping if we have to handle local distortions (three points form a triangle), while only one anchor point is needed in radial basis warping. Furthermore, due to the computation complexity, the animation speed can achieve in real-time with piecewise polynomial method. Traditionally, warping was applied to correct geometric distortions such as the distortion of viewing geometry. In this paper, we applied warping for human facial expressions. The facial warping techniques are also required for various future applications, such as face recognition, criminal identification, authentication in secure system, dynamic imagery, speech (action) semantic animation, multimedia and virtual reality construction, low-bandwidth video conference image transmission, and intelligent man-machine interface.

References

[1] Nur Arad and Daniel Reisfeld. Image warping using few anchor points and radial functions. *Computer Graphics forum*, 14(1):35–46, 1995.

[2] F. L. Bookstein. Principal warps: Thin plate splines and the decomposition of deformations. *IEE Transactions on Pattern Analysis Machine Intelligence*, 11(6):567–585, 1989.

[3] Lin Daw-Tung. Facial expression morphing with radial basis function warping technique. *International Symposium on Multimedia Information Processing*, pages 330–336, 1998.

[4] Ardeshir Goshtasby. Piecewise linear mapping functions for image registration. *Pattern Recognition*, 19(6):459–446, 1986.

[5] C. L. Lawson. Software for c^1 surface interpolation. *Mathematical Software III. Academic Press*, pages 161–194, 1977.

[6] D. T. Lee and B. J. Schachter. Two algorithms for constructing a delaunay triangulation. *International Journal of Computer and Information Sciences*, 9:219–242, 1980.

[7] D. Reisfeld N. Arad, N. Dyn and Y. Yeshurun. Image warping by radial basis functions: application to facial expressions. *CVGIP: Graphical Models and Image Processing*, 56(2):161–172, 1994.

[8] Huitric H Nahas M and Saintourens M. Animation of a b-spline figure. *The Visual Computer*, 3(5):272–276, 1988.

[9] Frederic I. Parke. Parameterized models for facial animation. *IEEE Computer Graphics and Applications*, 2(9):61–68, 1982.

[10] S. Platt and N. I. Badler. Animating facial expression. *Proc SIGGRAPH ’81, Computer Graphics*, 15(3):245–252, 1981.

[11] George Wolberg. *Digital Image Warping*. IEEE Computer Society Press, Third Edition, 1994.