Radial Basis Network for Facial Expression Synthesis*

I. King H. T. Hou
{king,hthou}@cs.cuhk.edu.hk

Department of Computer Science & Engineering
The Chinese University of Hong Kong
Shatin, New Territories, Hong Kong

***Abstract*— Many multimedia applications require the synthesis of facial expressions. We demonstrate the synthesis of different degrees of various 2D grayscale facial expressions using the Radial Basis Function (RBF) neural network. The RBF network is used to generate spatial displacement of a set of facial feature points as in multidimensional density interpolation. We have implemented two RBF networks for synthesizing facial expressions. One network generates the spatial displacement of facial feature points for different degrees of expressions and the other generates the spatial displacement of facial feature points of mixed facial expressions. The predicted facial landmark displacement information is then fed into our image warping algorithm along with the expressionless facial image to produce the final synthesized facial image. We discuss the method used and demonstrate the results.**

1 Introduction

Facial expressions play an important role in non-verbal communications. Furthermore, we use facial gestures to convey our mood and express our feeling. Moreover, to make an intelligent, friendly, and effective machine-human interface we need to synthesize facial expressions for a variety of applications, e.g., graphics, animation, security, teleconference, and facial data compression.

Suppose we are given a 2D grayscale expressionless face image of a person, how can we synthesize different expressions of that person? One of the ways to synthesize facial expressions is to find the "approximate" displacement of prominent facial feature points. This problem is similar to non-parametric multivariate density estimation since we do not know the underlying multidimensional facial landmark displacement function. Moreover, we often cannot obtain accurate facial landmark displacement information due to inherently inaccurate input data. This is because: (1) it is hard to generate a set of standardized expressions, e.g., each person may "smile" differently, (2) it is hard to produce accurately the precise degree of a particular expression, e.g., how to generate a "20% smile"?, and (3) it is difficult to mix various facial expressions, e.g., how to gesture a "happy and sad" face?

Current approaches in synthesizing facial expressions include texture mapping approach to 3D facial image synthesis [11] and the use of 3D model of facial muscles and tissues [4]. These methods proves to be tedious in determining the actual parameter values for synthesizing and animating facial expressions. An alternative approach has been investigated by Nur Arad et al. [9], which demonstrated the use of Radial Basis Function (RBF) in interpolating the anchor points for 2D image warping, which can be applied to synthesize facial expressions. However, it does not provide a mechanism to determine the appropriate destination of the anchor points for each particular facial expression.

We have obtained a set of prominent facial landmarks which have greater potentials in revealing changes in displaying a particular facial expression. The spatial displacement of these landmarks for the facial expressions is used to generate control points in the image warping procedure for synthesizing various facial expressions.

We construct the RBF neural network that maps the relationship between necessary patterns of movements of these landmarks and the six universal facial expressions described in [3]. The distinctive difference between this method and that of [9] is that, we use the RBF in the network description for finding the necessary changes of the landmarks, rather than in interpolating the anchor points in image warping.

The next section will briefly introduce the facial expression recognition process which is crucial in understanding the facial expression synthesis process. We will then formulate the RBF network in Section 3. Lastly, we will demonstrate our results and end with some discussions.

2 Reverse of Facial Expression Recognition

Before we deal with the synthesis of facial expressions, we briefly summarize the process of facial expression recognition since each process is the inverse of the other.

*This work is supported in part by RGC Earmark Grant # 221500620, Direct Grant 220500910, and Direct Grant 220500720. A preliminary version was presented in ICNN'96 [8].

2.1 Facial Expression Recognition

In [7], Kobuyashi & Hara presented a method of classifying the six universal facial expressions using neural network. A set of 30 facial landmarks located near the eye-brows, eyes and the mouth are defined as the Facial Characteristic Points(FCPs) as shown in Figure 1(a). These points are extracted semi-automatically from a 2D grayscale expressionless face image as shown in Figure 2(a)-(c). These FCPs are selected since they have the largest variance among facial expressions; hence, they are the best candidate in revealing the changes in facial expressions.

The basic idea is to find out the spatial differences between the FCPs of the normal face and that of the expressive face. Thus, the differences of those 30 pairs of position information will constitute the 60 inputs to the two-layered neural network as shown in Figure 1(b). The number of output layer unit is six, the position of which corresponds to each of the six emotion labels in the order `HAPPY`, `SAD`, `ANGRY`, `FEAR`, `SURPRISED` and `DISGUSTED`.

Since each 2D grayscale input face image is different, we must perform several pre-processing steps. To compensate for the differences in the size, orientation and position of the faces in the image as well as the size of the face components, we have to transform the coordinates of the FCPs so that they are comparable across the set of individual faces. Therefore, the absolute coordinates of the FCPs have to undergo the following four transformations:

Translation - It is employed to translate the origin of coordinate system to the nose top of the individual as the absolute pixel coordinates of the FCPs are obtained relative to the lower left corner of the image. A quantity called *base* is introduced, which should not be varied for each of the facial expressions,

$$base = \sqrt{(xb_2 - xb_1)^2 + (yb_2 - yb1)^2}$$

where (xb_1, yb_1) and (xb_2, yb_2) are the pixel position of inner corners of left eye and right eye respectively. The mid-point, (x_0, y_0), between (xb_1, yb_1) and (xb_2, yb_2) is also calculated using the mid-point formula, $x_0 = (xb_1 + xb_2)/2$ and $y_0 = (yb_1 + yb_2)/2$. The origin of the new coordinate system $(origin_x, origin_y)$ is calculated as $(x_0 - base * \sin\theta, y_0 - base * \cos\theta)$.

Rotation - It is employed to correct the inclination of the face so that the coordinates are expressed with respect to the vertical axis of the face in the new coordinate system. The inclination of the face with respect to the horizontal line, θ, is defined as

$$\tan^{-1}\frac{(yb_2 - yb1)}{(xb_2 - xb_1)}.$$

Normalization - It is introduced to compensate the distance effect between the client's face and the camera. The landmarks of the normal and expressive face after rotation are divided by the value *base*. These normalized values of the expressive face are subtracted from those of the normal one.

Standardization - As those subtracted values are indeed the absolute displacements of the FCPs from their normal position, thus these magnitudes are subject to individual variations. Standardization is needed to find out the relative displacement of the landmarks from their normal position. From the normal face, we determine the standard values as follows: openness of eyes: $((yn_7 - yn_5) + (yn_8 - yn_6))/2$, width of eyes: $((xn_1 - xn_3) + (xn_4 - xn_2))/2$ height of eyebrows: $((yn_{19} - yn_1) + (yn_{20} - yn_2))/2$ openness of mouth: $(yn_{26} - yn_{25})$ width of mouth: $(xn_{24} - xn_{23})$ where (xn_i, yn_i) is the *i-th* FCP of the normal face after normalization.

After the above pre-processing steps are performed, the filtered data is ready for both facial expression classification and synthesis.

3 Synthesis as a Reverse Process

The recognition of facial expressions is a classification process. The reverse of it is a multidimensional interpolation problem.

3.1 The Radial Basis Function Network

The basic principle of synthesizing facial expressions is to find out the necessary relative spatial displacement of the FCPs for each facial expression. This is similar to a nonparametric multidimensional density estimation problem. The RBF network is ideal for interpolation since it uses a radial basis function, e.g., Gaussian function, for smoothing out and predict missing and inaccurate inputs.

Given a set of n-dimensional training data, $(\vec{x}_i, \vec{d}_i), \vec{x}_i \in R^n, \vec{d}_i \in R^{n'}$ for $i = 1, \cdots, m$, find a function $F : R^n \mapsto R^{n'}$ which satisfies the interpolation conditions

$$F_k(\vec{x}_i) = d_{ik}, \qquad i = 1, \cdots, m; k = 1, \cdots, n' \text{ with } n' < m. \tag{1}$$

We would consider interpolating functions of the form

$$F_k(\vec{x}) = \sum_{j=1}^{m} w_{jk}\ g(\parallel \vec{x} - \vec{\mu}_j \parallel), \qquad \vec{x} \in R^n, k = 1, \cdots, n'$$

where $\|\cdot\|$ denotes the usual Euclidean norm on R^n and $\vec{\mu}_j \in R^{n'}, j = 1, 2, \cdots, m$ denotes the *centers* of the radial-basis functions which are given as the known data points.

Often, the $g(\cdot)$ is the normalized Gaussian activation function defined as

$$g(\vec{x}) = \frac{\exp[-(\vec{x} - \vec{\mu}_j)^2/2\sigma_j^2]}{\sum_k \exp[-(\vec{x} - \vec{\mu}_k)^2/2\sigma_k^2]}$$

where x is the input vector, μ is a set of weights and σ is the width of the RBF.

Hence, the determination of the nonlinear map $F(\vec{x})$ has been reduced to the problem of solving the following set of linear equations for the coefficients w_j,

$$\begin{pmatrix} f_{1k} \\ \vdots \\ f_{mk} \end{pmatrix} = \begin{pmatrix} A_{11} & \cdots & A_{1m} \\ \vdots & \ddots & \vdots \\ A_{m1} & \cdots & A_{mm} \end{pmatrix} \begin{pmatrix} w_{1k} \\ \vdots \\ w_{mk} \end{pmatrix}, \quad k = 1, 2, ..., n'$$

where $A_{ij} = g(\|\vec{x}_i - \vec{\mu}_j\|),\ i, j = 1, 2, \cdots, m.$

The basic architecture of our two-layered RBF network is shown in Figure 1(b).The input layer units of the neural network is 60 since we have 30 pairs of FCP position information, and the number of output layer unit is six, the position of which corresponds to each of the six universal facial expressions in the order `HAPPY`, `SAD`, `ANGRY`, `FEAR`, `SURPRISED`, and `DISGUSTED`. The training phase of the RBF network constitutes the optimization of a fitting procedure on known spatial displacement data points of various facial expressions presented to the network in the form of input-output examples.

The x in the Gaussian function corresponds to the facial expression label for the neural network while the output of the network is a vector of movements of FCPs. The σ corresponds to the spread constants set in the training setup. The only real design decision for RBF network is to find a good value of σ, which determines the generalization ability of the RBFs. The variable σ should be large enough to allow the overlapping of the input regions of radial basis functions. This makes the network function smoother and results in better generalization for new input vectors occurring between input vectors. However, σ should not be so large that each neuron responds in essentially the same manner, i.e., any information presented to the network becomes lost. Our σ is picked manually by trial and error, within maximum and minimum of distances of input vectors. After training, 2 sets of weights, $\vec{\mu}$ and w, are obtained. This is used to produce the spatial displacement of facial landmark points.

The generalization phase is then the interpolation between the data points along the constrained surface generated during the training phase. Here the generalization will allow the user to specify various degrees of a facial expression.

We now will perform post-processing on the facial feature displacement data to synthesize facial expressions.

3.2 Mapping the Output to the Image

As the output from the RBF network is a vector that consists of relative displacements of FCPs for a facial expression, the displacement vectors have to be de-standardized, de-normalized and transformed back according to the FCPs of the normal face.

Let the output vector be in the form $(x_1, ..., x_{30}, y_1, ..., y_{30})$ where (x_i, y_i) denotes the displacement of the *i-th* FCP. To perform de-standardization, the elements of the output vector should be multiplied by their corresponding standard value: y_1 to y_{16} are multiplied by (openness of eyes), x_1 to x_{16} are multiplied by (width of eyes), y_{17} to y_{22} are multiplied by (height of eyebrows), x_{17} to x_{22} are multiplied by (height of eyebrows), y_{23} to y_{30} are multiplied by (openness of mouth), and x_{23} to x_{30} are multiplied by (width of mouth).

To de-standardize, the elements of the output vector are multiplied by their corresponding standard values as defined in [7] and then added to the normalized values of the FCPs of the normal face image:

$$x_i := x_i + normal\_x_i, \qquad y_i := y_i + normal\_y_i$$

These values are then de-normalized by the *base* value:

$$rotx_i := x_i * base, \qquad roty_i := y_i * base$$

and then rotated and translated back:

$$x_i := rotx_i * \cos\theta - roty_i * \sin\theta, \qquad y_i := rotx_i * \sin\theta + roty_i * \cos\theta$$

$$xb_i := x_i + origin_x, \qquad yb_i := y_i + origin_y$$

(xb_i, yb_i) is then the new position of the *i-th* FCP on the normal face image.

To generate an expressive face image from the normal face image, we used 2D image warping [5]. Both of the FCPs of the normal face and that of the expressive face are connected to form triangular patches as shown on Figure 1(c). Image warping is then performed by scan-converting each triangle.

4 Training by Radial Basis Network and Results

Now the remaining problem is how we obtain the displacements of the FCPs corresponding to a certain emotion. We have made use of Radial Basis Network (RBN) to carry out two sets of training: (1) training with different degrees of six universal facial expressions; (2) training with a set of six universal facial expressions.

4.1 Experimental Results

We have carried out 2 sets of training using the Neural Network Toolbox of Matlab running on Sun Sparc20 and C programs on SGI INDY machines. A set of 128×128 grayscale images are used in our experiment. We set $\sigma = 0.5$ and 1 for RBN1 and RBN2 respectively. The total time for facial feature extraction, pre-processing, neural network calculation, and image warping takes less than 15 seconds.

4.1.1 Training with Different Degrees of Six Universal Facial Expressions

Facial expressions can have different strengths, e.g, the degree of happiness ranges from smile to grin to laugh. Therefore, we used five images with different degrees for each of the six universal facial expressions, thus a total of 30 images are used as the training set. For each expression, we manually arrange the images in the order of decreasing strength and assign to each of them the value 1.0, 0.9, 0.8, 0.7 and 0.6 accordingly. Thus the input vector would be in the form (0.6,0,0,0,0,0) for the weakest degree of happiness among the five images.

After training, the network is able to generate the six universal facial expressions, plus different degrees of variation for each facial expression (RBN1). Figure 3(a)-(f) illustrate how the neural network is able to capture the features of the facial expressions: for happy face, the upward movement of mouth corners is captured; for sad face, the inner eyebrows raise and mouth corners go down; for surprise the whole eyebrows raise and the mouth widely open, etc. Figure 4(a)-(e) display five degrees of happy faces generated. We find that this process is certainly not a linear one.

4.1.2 Training with a Set of Six Universal Facial Expressions

We find that the above network is unable to generate mixed expressions, i.e., if we specify the input as a mix of some degrees of sadness and happiness, the network cannot generate the output as a mixed expression. Hence, we trained another RBF network with a set of six universal expressions which we denote as RBN2. When mixed inputs such as $(1.0, 1.0, 0, 0, 0, 0)$ are applied, the output shows a dependence on the mixed input, resulting in a mixed expression. However, this network is unable to generate different degree for each expression. Figure 5(a)-(e) show the results obtained from this network.

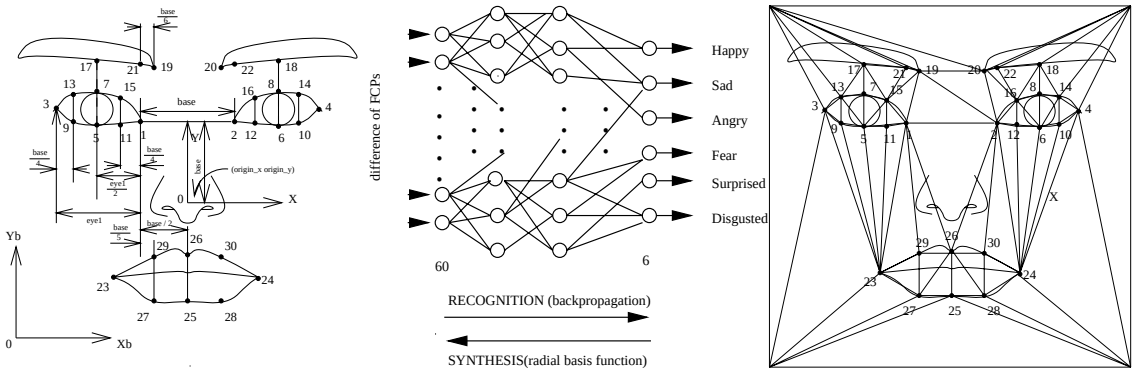
4.2 Discussions

Although the experimental results are encouraging, we briefly outline the shortcoming of this facial synthesis method using the RBF network.

1. **Hard to obtain accurate FCPs for training.** – From our experimental results, one main problem we faced is that we were unable to unify the two RBF network into one network capable in generating both mixed expressions and different degrees of each expression. This is due to the highly inaccurate and hard-to-obtain training information.
2. **The basic parameters are hand-tuned.** – We have mentioned that the input vectors of RBN1 consist of arbitrarily assigned values that represent the degree of emotion. These distinct data points are required for different degrees of variation for each expression. Therefore there is a need to adjust the inputs for the training to be successful. This is a highly subjective judgement; hence, an adaptive one would be better.
3. **Cannot generate artificial features.** – Apart from the movements and shapes of the facial components, wrinkles such as naso-labial folds or crow's-feet wrinkles on the face contribute significantly to facial expressions. However, our system only uses normal face images where the face is free from any wrinkle, and simple warping technique is employed. Special technique such as texture mapping should be adopted to add such wrinkles to the final image.
4. **Additional FCPs are needed.** – Also, more FCPs need to be added near the eyebrows and mouth region to obtain smooth, detailed, and better warped images.

5 Conclusion

We have built a system for synthesizing mixture and various degrees of facial expressions. Radial Basis Function networks are used to map the emotion labels to the displacements of the set of FCPs. Depending on the positions of the sample data points, the RBF approach constructs a nonlinear function space according to an arbitrary distance measure. Thus an interpolating surface which exactly passes through all the pairs of the training set can be produced so that when data points not in the training set are



(a) (b) (c)

Figure 1: (a) The 30 facial characteristic points, (b) the neural network for facial expression classification and synthesis, and (c) the FCPs are connected to form patches for image warping.

presented to the RBF network, the mapping can also be interpolated. Since facial expressions have different degrees and are often mixed with one another, the use of RBF network for high dimensional interpolation makes the synthesis of facial expressions possible.

Acknowledgments

The authors gratefully acknowledge Ms. Mary Y.Y. Leung and Ms. Yen-Hui Hung for their initial design and implementation of the image warping software program at the Neural Computing & Engineering Lab.

References

[1] P. J. Benson. Morph transformation of the face image. *Image and Vision Computing*, 12:691–696, 1994.

[2] D. Broomhead and D. Lowe. Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355, 1988.

[3] P. Ekman and W. V. Friesen. *Unmasking the Face.* Consulting Psychologists Press, Inc., 1975.

[4] F. Hara and H. Kobayashi. Computer graphics for expressing robot-artificial emotions. IEEE International Workshop on Robot and Human Communication, 1992.

[5] P. Heckbert. Graphics gems. pages 65–77, 1990.

[6] K. Hertz and Palmer. *Introduction to the theory of neural computation.* Addison Wesley, 1991.

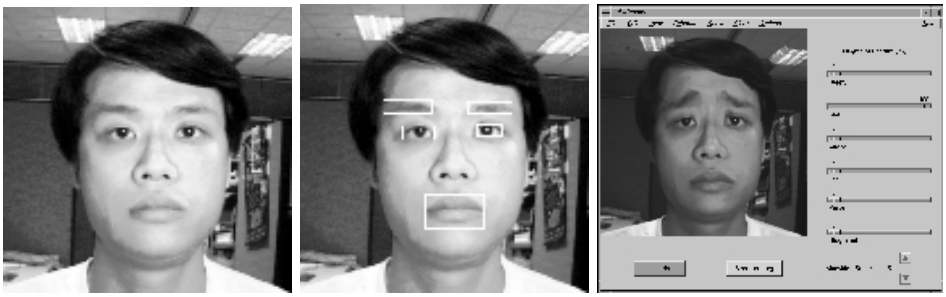
[7] H. Kobayashi and F. Hara. Recognition of six basic facial expressions and their strength by neural network. IEEE International Workshop on Robot and Human Communication, 1992.

[8] Mary Y.Y. Leung, H. Y. Hung, and I. King. Facial expression synthesis by radial basis function network and image warping. In *IEEE International Conference on Neural Networks*, volume III, pages 1400–1405, Washington D.C., 1996. IEEE Computer Society.

[9] D. R. Nur Arad, Nira Dyn and Y. Yeshurun. Image warping by radial basis functions: Application to facial expressions. *CVGIP: Graphical Models and Image Processing*, 56, No. 2:161–172, 1994.

[10] G. Wolberg. *Digital Image Warping.* IEEE Computer Society Press Monograph, 1990.

[11] J. Yau and A. Duffy. *A Texture mapping approach to 3D facial image synthesis.* 1988.



(a) (b) (c)

Figure 2: (a) The NORMAL face, (b) the FCPs are extracted semi-automatically from the expressionless face by first marking the facial feature area in white rectangles, and (c) a window view of the system.

(a) (b) (c) (d) (e) (f)

Figure 3: The six universal facial expressions generated by RBN1: (a) NORMAL, (b) HAPPY, (c) SAD, (d) ANGRY, (e) FEAR, (f) SURPRISED and (g) DISGUSTED.

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

Figure 4: Various degrees of the "HAPPY" expression increasing from left to right.

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

Figure 5: A mixture of various expressions: (a) FEAR-SURPRISED, (b) HAPPY-SAD, (c) HAPPY-SURPRISED, (d) SAD-ANGRY, and (e) SAD-DISGUSTED.