2010 International Symposium on Nonlinear Theory and its Applications
NOLTA2010, Krakow, Poland, September 5-8, 2010

Facial Expression Recognition Using a Simplified Head Model and RBF Networks

Koichi Takahashi[†], Hironobu Fukai[‡] and Yasue Mitsukura[†]

†Tokyo University of Agriculture and Technology
2-24-16 Naka-cho, Koganei-shi, Tokyo, Japan
‡Ritsumeikan University
1-1-1 Noji-higashi, Kusatsu-shi, Shiga, Japan
Email: 50010401217@st.tuat.ac.jp

Abstract—In this paper, we propose a facial expression estimation method using radial basis function (RBF) networks. The advantage of an RBF network is that only the arbitrary area of each expression is able to be learned as a classifier. Therefore, we consider whether RBF network is suitable for learning one arbitrary partial expression. The facial expression recognition experiments we conducted demonstrate that the new method estimates the expressions more accurately than the previous method using the Mahalanobis distance.

1. Introduction

In the field of human-computer interaction and computer vision, facial expression estimation is a fundamental challenge. Facial expression estimation is expected to be applied to a variety of fields such as expression mirroring for web chat technology and psychological profiling.

Recently, many facial expression recognition methods have been presented [1]–[3]. Methods for image sequences using a Support Vector Machine (SVM) are proposed [1], [2]. The SVM has the advantage that one can classify the learned expressions accurately. In these methods, high recognition accuracy is achieved using the experimental results. Pose-invariant method proposed in [3] tracks the user's head pose and estimates the state of facial expressions simultaneously by using particle filtering.

In order to apply facial expression recognition to our lives, the method should distinguish whether the expressions are already learned or not yet learned because human facial expressions vary widely given a person's emotional state. Classification using SVM is only able to describe the boundary between the expression classes. However, this classifier is not able to recognize whether the input data belong to the class or lies outside of all classes. Of course, by using a Dynamic Bayesian Network, the recognition results will be unsuitable if unknown expression is input.

In an attempt to solve these problems, we have already presented a simplified head model [4]. The simplified head model is based on the architecture in the reference [5] and is able to track a user's head position, pose and facial deformation using particle filtering [6]. Although the simplified head model can represent a user's arbitrary facial actions, we have only employed the Mahalanobis distance in the facial deformation feature space to classify the expressions. Perhaps there are more effective methods to be utilized to determine facial expressions. In this paper, we propose a facial expression recognition method using RBF networks. By application of RBF networks for facial expression classification, it is expected that only the arbitrary area of each expression will be able to be learned.

The importance of this study is to demonstrate the effectiveness of RBF networks for facial expression recognition and to propose a method for learning arbitrary expressions. In this paper, we present the details of the proposed method and confirm that the proposed method works as thought.

2. Methods

2.1. Simplified Head Model

In this study, the simplified head model [4] is utilized for the head state tracking. Here, the head state refers to the head position, pose and facial deformation.

The strategy of the simplified head model method is shown in Figure 1. The simplified head model is generated from only one baseline image by approximating the coordinates of a user's facial parts by a cylinder. The baseline image is the user's frontal facial image which describes a neutral face. The simplified head model uses 8 feature points: inner and outer corners of both eyes, inner corners of both eyebrows and the outer corners of the mouth. In order to define these feature points, we select them manually for now. At this time, baseline templates $\boldsymbol{T}$ are also created based on the feature points from the baseline image.

The simplified head model is a deformable model which has 3 degrees of freedom d_{mx}, d_{my}, d_{ey}. The variables indicate the horizontal elastic movement of the mouth, the up-down movement of the mouth and the up-down movement of the eyebrows, respectively. Therefore, the simplified head model can represent the user's facial action, and facial expressions are recognized by analyzing of these deformation parameters when the user displays corresponding expressions.

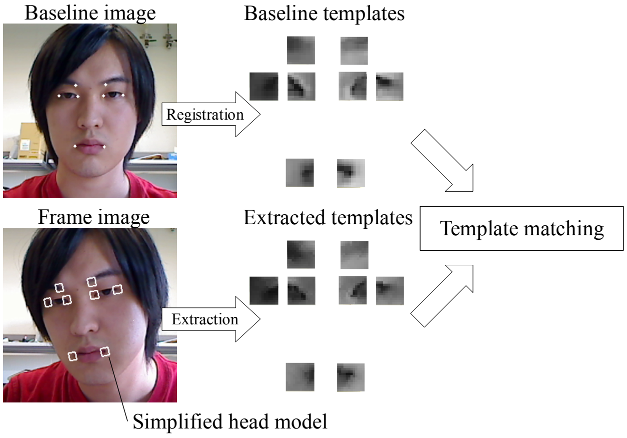


Figure 1: The simplified head model approach.

Here, we can assume that the head state tracking problem is determining the best match for the simplified head model in each frame image. By using the simplified head model, we can extract each template of each feature point $\boldsymbol{T}'$ by using the Affine transform and the nearest neighbor algorithm. Hence, if the baseline templates $\boldsymbol{T}$ and extracted templates $\boldsymbol{T}'$ match by applying template matching, the simplified head model is placed according to the required parameters. Therefore, 3 dimensional position x, y, z, 3 dimensional pose $\theta_x, \theta_y, \theta_z$, and facial deformation parameter d_{mx}, d_{my}, d_{ey} have to be tracked. As a result, this becomes the 9 state tracking problem. For the tracking method, particle filtering [6] can be employed by using the similarity of template matching as a likelihood function.

Although template matching has a drawback of high computing costs in general, the simplified head model utilizes local template matching only around the feature points. Therefore, this strategy leads to low computing costs.

2.2. RBF Network for Expression Classification

In this study, we regard the facial expression recognition problem as accurate classification of the positive, negative and neutral expressions. Here, positive and negative expressions indicate a smiling face and a disgusted face, respectively.

Up to now, we have conducted facial deformation measuring experiments where the subjects display each facial expression using the simplified head model in the reference [4]. These experiments were conducted on five subjects aged 22 to 24 consisting of four males and one female. The results are shown in Figure 4(a). In this figure, each scale of axis is determined based on the distance between the center of the eyes which is set at 80. Each expression has 4103, 4352 and 4742 data points. These results indicate that each expression is densely distributed, where the positive state varies toward the d_{mx} and d_{my} axes, and the negative state varies toward the d_{ey} axis.

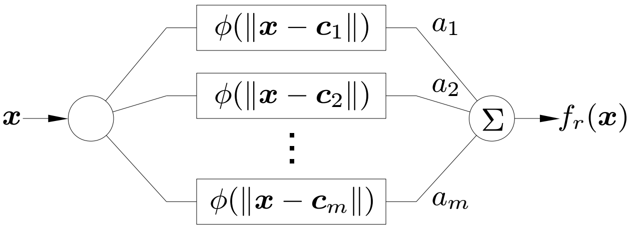

$\boldsymbol{x}$
$\phi(\|\boldsymbol{x}-\boldsymbol{c}_1\|)$
$\phi(\|\boldsymbol{x}-\boldsymbol{c}_2\|)$
$\phi(\|\boldsymbol{x}-\boldsymbol{c}_m\|)$
a_1
a_2
a_m
Σ
$f_r(\boldsymbol{x})$

Figure 2: The structure of RBF network.

For this facial expression recognition method we employ the RBF network instead of the Mahalanobis distance used in the previous method. The RBF network is utilized for the interpretation problem in general. A schematic of the RBF network with n inputs and a scalar output is described in Figure 2. This network implements a mapping $f_r : \Re^n \rightarrow \Re$ according to

$$f_r(\boldsymbol{x}) = \sum\nolimits_{i=1}^{m} a_i \phi\left(\|\boldsymbol{x} - \boldsymbol{c}_i\|\right) \tag{1}$$

where $\boldsymbol{x} \in \Re^n$ is the input vector, $\phi(\cdot)$ is an RBF, a_i, $1 \leq i \leq m$ are the weights, $\boldsymbol{c} \in \Re^n$, $1 \leq i \leq m$ are RBF centers and m is the number of centers. In this study, the Gaussian function

$$\phi(r) = \exp(-r^2/\sigma^2) \tag{2}$$

where σ is a real constant, is defined as RBF.

The challenging in this method for facial expression classifying is determining which output has the highest value between the three RBF networks corresponding to each facial expression. It is therefore required that the RBF network outputs close to 1 when the corresponding expression is displayed, and the other networks output close to 0. At this point, the considerable problem is how to select centers $\boldsymbol{c}_i$ and define constant σ. In practice, the centers are normally chosen from the data set $\{\boldsymbol{x}(t)\}_{t=1}^{N}$. Chen [7] presented a method where the set of M candidate regressors $\{\boldsymbol{x}'(t)\}_{t=1}^{M}$ is first selected randomly. Subsequently, the centers are selected based on the orthogonal least squares (OLS) algorithm.

Let $\{\boldsymbol{x}(t)\}_{t=1}^{N}$ is the set of N data points corresponding to the arbitrary expression. We can regard that learning an RBF network for each expression is the least squares problem as follows:

$$\boldsymbol{y} = \boldsymbol{P}\boldsymbol{a} \tag{3}$$

where $\boldsymbol{y} \in \Re^N$ is a desired output vector whose elements are all 1, $\boldsymbol{a} \in \Re^M$ is a weight vector whose ith element is a_i and $\boldsymbol{P} = [\boldsymbol{p}_1\ \boldsymbol{p}_2 \cdots \boldsymbol{p}_M] \in \Re^{N \times M}$ is a matrix whose element p_{ji} is defined as

$$p_{ji} = \phi(\|\boldsymbol{x}(j) - \boldsymbol{x}'(i)\|). \tag{4}$$

Here, the matrix $\boldsymbol{P}$ can be decomposed into

$$\boldsymbol{P} = \boldsymbol{W}\boldsymbol{B} \tag{5}$$

where $\boldsymbol{W} \in \Re^{N \times M}$ is a matrix with orthogonal columns $\boldsymbol{w}_i$ and $\boldsymbol{B} \in \Re^{M \times M}$ is an upper triangular matrix.

```
**Step** 1 ($k = 1$)
    $\boldsymbol{w}_1^{(i)} = \boldsymbol{p}_i$
    $r_1^{(i)} = \|\boldsymbol{w}_1^{(i)}\|_1^2 / N((\boldsymbol{w}_1^{(i)})^T \boldsymbol{w}_1^{(i)})$
    $i_1 = \arg\max_{1 \leq i \leq M} r_1^{(i)}$
    $\boldsymbol{w}_1 = \boldsymbol{w}_1^{(i_1)}$
**Step** $\boldsymbol{k}$ ($k = 2, 3, \ldots, M$)
    $b_j^{(i)} = \boldsymbol{w}_j^T \boldsymbol{p}_i / (\boldsymbol{w}_j^T \boldsymbol{w}_j),\ j = 1, \ldots, k-1$
    $\boldsymbol{w}_k^{(i)} = \boldsymbol{p}_i - \sum_{j=1}^{k-1} b_j^{(i)} \boldsymbol{w}_j$
    $r_k^{(i)} = \|\boldsymbol{w}_k^{(i)}\|_1^2 \ / \ N((\boldsymbol{w}_k^{(i)})^T \boldsymbol{w}_k^{(i)})$
    $i_k = \arg\max_{1 \leq i \leq M, i \neq i_1, \ldots, i \neq i_{k-1}} r_k^{(i)}$
    $\boldsymbol{w}_k = \boldsymbol{w}_k^{(i_k)}$
```

Figure 3: RBF network learning algorithm.

Chen's method computes each column of $\boldsymbol{W}$ and orthogonalizes $\boldsymbol{P}$ simultaneously using the OLS method based on the Gram-Schmidt scheme. In the case where the elements of desired output vector $\boldsymbol{y}$ are all 1, Chen's algorithm is described as shown in Figure 3. $\|\cdot\|_1$ denotes an L^1 norm. We can consider that $r_k^{(t)}$ explains the contribution ratio of the candidate regressor $\boldsymbol{x}'(t)$ at step k. The regressor with the highest contribution ratio is therefore selected in each step. The procedure continues while

$$1 - \sum_{j=1}^{k} r_j^{(i_j)} < \rho \qquad (6)$$

where $0 < \rho < 1$ is a chosen tolerance.

3. Experimental Results and Discussions

Fortunately, each facial expression distribution is not very complicated. It is therefore important that the tolerance should be set low, but overfitting has to be avoided. In order to evaluate resulting RBF networks, we utilize the a_{max} and a_{min} which is the maximum and minimum element of the weight vector $\boldsymbol{a}$, respectively. If a_{max} is well over 1 or a_{min} is a negative value, we can consider that overfitting has occurred. For this reason, we conducted the learning 10000 times per expression due to the fact that $\boldsymbol{a}$ depends on the randomly selected regressors $\{\boldsymbol{x}'(t)\}_{t=1}^{M}$, and selected the result with the least a_{max} and nonnegative a_{min}. Moreover, we decided σ such that the minimum results σ without overfitting. ρ is set to 0.02 due to the fact that overfitting happens frequently when ρ is less than one. Figure 4(b) shows the RBF network learning results for each expression, with σ corresponding to the expression positive, negative, and neutral at 3.5, 3.0, and 2.0, respectively. In addition, the number of the center points for each expression is 14, 13, and 11, respectively.

Table 1: Recognition Results.

Input	Positive[%]	Negative[%]	Neutral[%]
Positive	100.0(99.1)	0.0(0.0)	0.0(0.0)
Negative	0.0(0.0)	98.4(99.1)	1.6(0.0)
Neutral	0.2(0.8)	0.8(4.5)	98.9(94.7)

For facial expression recognition, if the RBF network outputs over 0.1 and greater than other outputs, the expression can be regarded as the corresponding expression. At this point, neutral is given preference since the false estimations from neutral into other expressions should be prevented. Figure 5 describes the results of the expression recognition using a test movie. In this movie, we defined Frames 1 to 433, 686 to 763, and 978 to 1078 as the neutral, frames 446 to 672 as the positive, and the frames 773 to 971 as the negative. The average results of 10 experiments are shown in Table 1. The values shown in parentheses denote the results using the previous classifier, the Mahalanobis distance. The total recognition accuracy shows 99.1% which is better than the 96.5% shown by the previous method. Consequently, we were able to confirm that the RBF networks effectively represent the expressions.

4. Conclusions

In this study, we adopted RBF networks for a simplified head model and utilized them for facial expression recognition. According to the experimental results, the effectiveness of our approach is confirmed, and the precision of recognition is increased. In the future, we are going to modify the system to recognize more kinds of expressions.

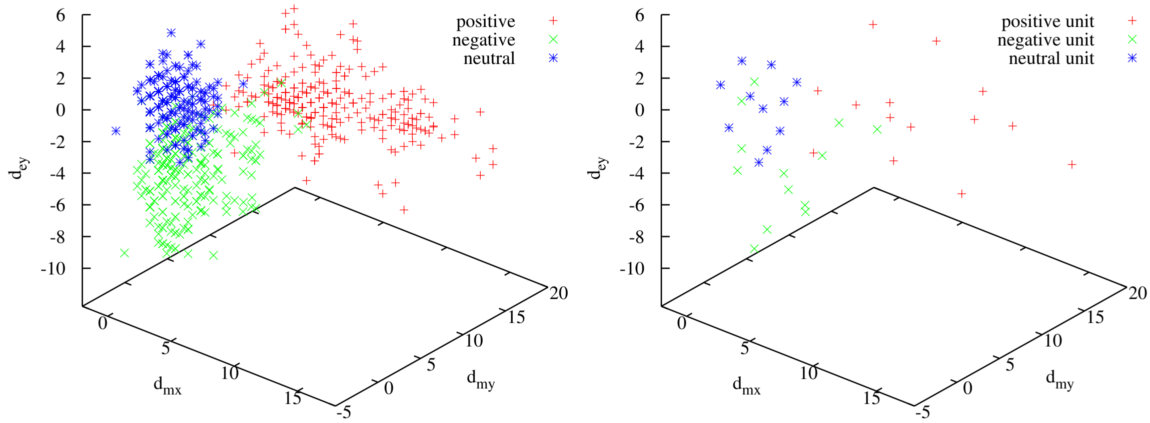
References

[1] P. Michel and R. Kaliouby, "Real time facial expression recognition in video using support vector machines," *The 5th International Conference on Multimodal Interfaces*, Vancouver, pp. 258–264, 2003.

[2] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *IEEE Transaction on Image Processing*, Vol. 16, No. 1, pp. 172–187, 2007.

[3] S. Kumano, K. Otsuka, J. Yamato, E. Maeda and Y. Sato, "Pose-invariant facial expression recognition using variable-intensity templates," *Asian Conference on Computer Vision*, pp. 324–334, 2007.

[4] K. Takahashi, H. Fukai and Y. Mitsukura, "Facial expression estimation using simplified head model based on particle filtering," *The 11th IEEE International Workshop on Advanced Motion Control*, pp.173–178, 2010.



(a) The distribution of each facial expression

(b) The position of each unit

Figure 4: The learning results of RBF network.

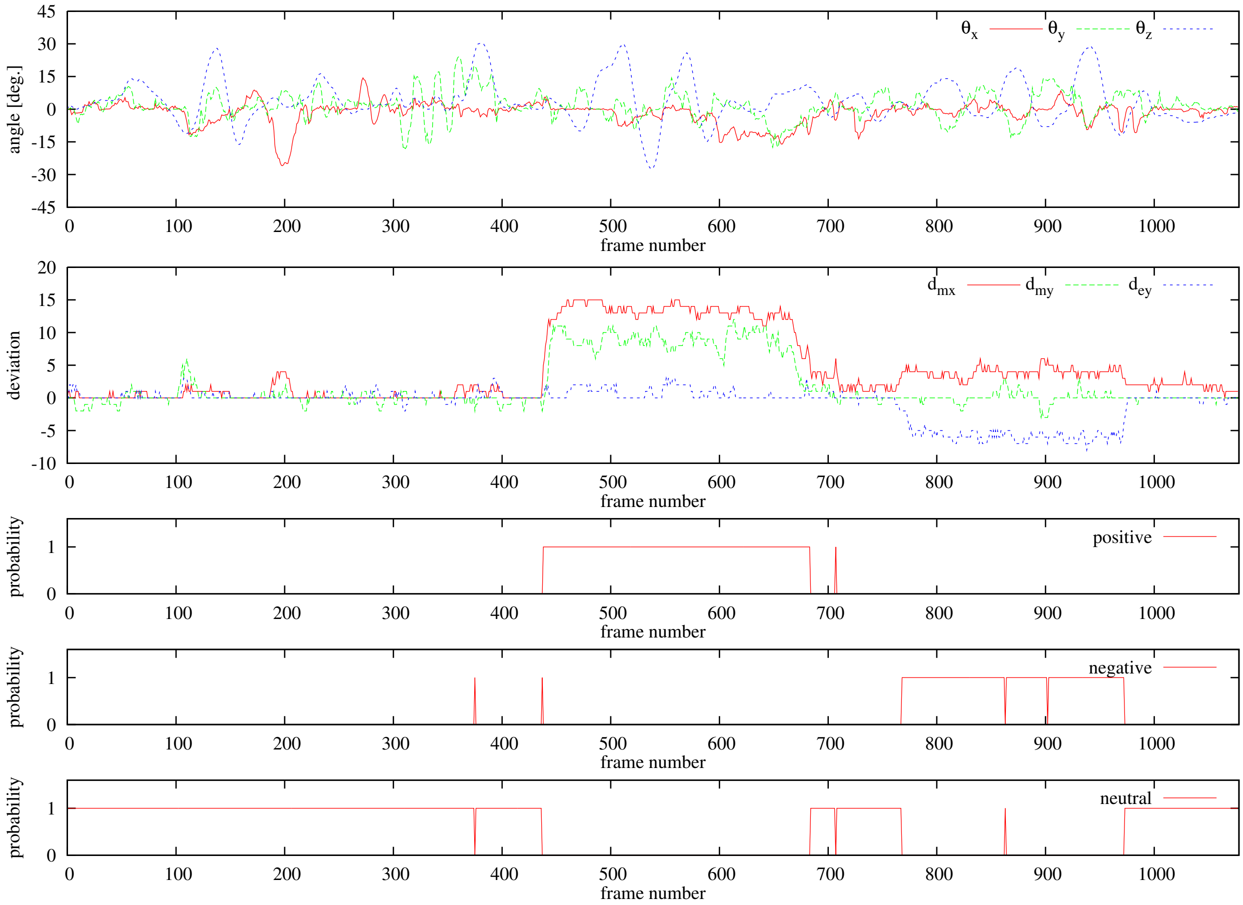


Figure 5: Simulation results using the proposed method.

[5] K. Oka, Y. Sato, Y. Nakanishi and H. Koike, “Head pose estimation system based on particle filtering with adaptive diffusion control,” *IAPR Conf. Machine Vision Applications*, pp. 586–589, 2005.

[6] M. Isard and A. Blake, “Condensation - conditional density propagation for visual tracking,” *International Journal of Computer Vision*, Vol. 29, No. 1, pp. 5–28, 1998.

[7] S. Chen, C. F. M. Cowan and P. M. Grant, “Orthogonal least squares learning algorithm for radial basis function network,” *IEEE Transaction on Neural Networks*, vol. 2, no. 2, pp. 302–309, 1991.