Robust Facial Expressions Recognition Using 3D Average Face and Ameliorated AdaBoost

Jinhui Chen
ianchen@me.cs.scitec.kobe-u.ac.jp

Yasuo Ariki
ariki@kobe-u.ac.jp

Tetsuya Takiguchi
takigu@kobe-u.ac.jp

Graduate School of System Informatics, Kobe University
1-1 Rokkodai, Nada, Kobe, Hyogo, 657-8501 Japan

ABSTRACT

One of the most crucial techniques associated with Computer Vision is technology that deals with facial recognition, especially, the automatic estimation of facial expressions. However, in real-time facial expression recognition, when a face turns sideways, the expressional feature extraction becomes difficult as the view of camera changes and recognition accuracy degrades significantly. Therefore, quite many conventional methods are proposed, which are based on static images or limited to situations in which the face is viewed from the front. In this paper, a method that uses Look-Up-Table (LUT) AdaBoost combining with the three-dimensional average face is proposed to solve the problem mentioned above. In order to evaluate the proposed method, the experiment compared with the conventional method was executed. These approaches show promising results and very good success rates. This paper covers several methods that can improve results by making the system more robust.

Categories and Subject Descriptors

I.3.5 [**Computational Geometry and Object Modeling**]: Curve, surface, solid, and object representations; I.5.4 [**Pattern Recognition**]: Applications—*Computer vision, Signal processing*

General Terms

Algorithms, Experimentation

Keywords

facial expressions recognition, 3D average face, AdaBoost

1. INTRODUCTION

One of the most crucial techniques associated with Computer Vision is technology that deals with facial recognition, especially, the automatic estimation of facial expressions. The need for this kind of technology in various different areas keeps pushing the research forward every year, and lots of achievements have been obtained due to its great potentials in real-life applications, such as human-computer interaction (HCI) [9], automatic recommendations analysis based on video contents [5] and cognitive and interactive abilities of robot. But lots of difficulties, there still exist, because of the variation and complexity of the facial expression across human population and even the same individual. So facial expressions recognition is an interesting research topic, and many researchers attach great importance to this field [6].

However many of the conventional methods above are focused on the single static picture, so that they are short of practical application. Under the condition of real-time recognition, when a face turns sideways, the expressional feature extraction becomes difficult or the view of a camera fixed in front of the user changes, and the recognition accuracy degrades significantly. Thus, many of the conventional approaches are limited to situations in which the face is viewed from the front.

Therefore, there is a great need to solve the problem that the face cannot be moved freely. Thus, in this paper, a method is proposed to estimate human emotions based on facial expression. We use a Look-Up-Table (LUT) for AdaBoost to be trained the multi-class features efficiently. Furthermore, we use three-dimensional average face, which is recovered from the original image based on Procrustes Analysis. With these approaches, we can solve the problem that many of the conventional facial expression recognition methods are limited to situations in which the face cannot be moved freely. In order to evaluate the proposed method, the experiment compared with conventional method was executed. The experimental results show that the proposed method provides better performance in comparison with the conventional approaches.

2. PROPOSED METHOD

In this section, the method to estimate facial expression is proposed. Fig. 1 shows a processing flow of the proposed method. First, the facial area is detected based on AdaBoost, using Haar-like features on the input images. Next, if the feature of facial expressions can be extracted, the facial expression on this facial area will be estimated by LUT AdaBoost; otherwise, we use 3D average face to estimate the facial expressions, and the average face model is reconstructed based on Procrustes Analysis [2] to obtain the shape of face including effective expressional features.

MM'13, October 21–25, 2013, Barcelona, Spain.

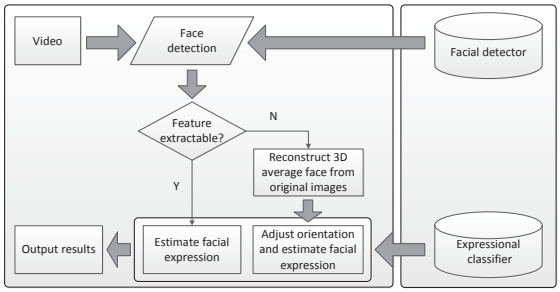

Figure 1: Processing flow of the proposed method

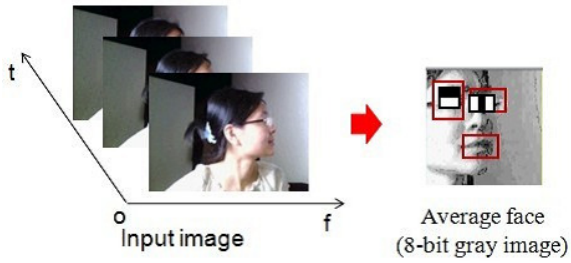

Figure 2: Example of average face creation

2.1 Average Face

Expressional features are extracted easily when the face is viewed from the front, but it is difficult to extract the feature points when the input video is lack of frontal facial frames. Therefore how to solve the problem is the gist of this subsection.

Procrustes analysis is a statistical tool for analyzing geometrical shapes. A shape (or equivalently a figure) P in R^P is represented by l landmarks. Two figures $P : l \times p$ and P': $l \times p'$ are said to have the same shape, if they are related by a special similarity transformation:

$$P' = \alpha P\Gamma + I_l \gamma^T \tag{1}$$

where the parameters of the similarity transformation are a rotation matrix Γ: $p \times p'$, $|\Gamma|$=1, a translation matrix γ: $p' \times l$, a positive scaling factor α, and I_l is a vector of ones. By using the generalized Procrustes analysis, it is possible to derive a consensus shape for a collection of figures [3], which is then used in registering new shapes into alignment with the collection by an Affine Transformation. In 3D model, the geometry of a face is defined as a shape vector $S_{3D} = (x_1, y_1, z_1, \cdots x_n, y_n, z_n) \in \mathbf{R}^3$, which contains the x, y, and z-coordinates of n vertices. And equation (1) is adjusted as follows

$$S = s_0 + \sum_{i=1}^{m} \beta_i s_i \tag{2}$$

where $\beta = (\beta_1, \beta_2, \cdots, \beta_m)^T$ is the shape parameter and m is the dimension of the shape parameter which was determined to represent the shape of 3D face model. Given the input face image indicated as $S_{2D} = (x_1, y_1, \cdots, x_n, y_n) \in \mathbf{R}^2$, the shape parameter β needs to be determined such that it minimizes the shape residual between the projected 3D facial shape generated by the shape parameter and the input 2D facial shape. The optimal shape and pose parameters (β, R_θ, T) are obtained from

$$E_r = \|P(R_\theta S_{3D} + T) - S_{2D}\|^2 \tag{3}$$

where S_{3D} is a $3 \times n$ matrix that is reshaped from the 3 $n \times 1$ model shape vector obtained using (1), P is a 2×3 orthographic projection matrix, T is a $3 \times n$ translation matrix consisting of n translation vectors $t = [t_x, t_y, t_z]^T$, and R_θ is a 3×3 rotation matrix where the yaw angle is θ. The average face creation is indicated as follows:

1. Initialization: set $\beta_0 = 0$ and $k = 1$.
2. Alignment: S_{2D} is aligned with the 2D shape obtained by projecting the frontal 3D shape (s_0) onto the $x - y$ plane.
3. Update R_θ and T with the fixed shape parameter by $min \ \ \|P(R_\theta S_{3D} + T) - S_{2D}\|^2$, and reconstruct $(S_{3D})_k$ using the shape parameter β_k.
4. Verify whether $E_r \leq \varphi$ or $k > N$, if not, go to Step 3 and $k = k + 1$.
5. Reconstruct S_{3D} using the final shape parameters.

When E_r is below a threshold (e.g. in Ref. [3], Gower suggested setting φ =10^{-4}), or the landmarks are processed over, the reconstruction would be stopped, and the consensus shape would be output.

In theory, when images are consecutive, using the shapes of current image, it is possible to estimate the shapes of front image or next image by Procrustes analysis. Since our research is under condition of real-time, namely, images are consecutive, we can use an optimal image sample as a representation of short motion process when the face is moving freely, and we call it 3D Average Face (3DAF). However the cost of processing 3D model is quite expensive, the average model is projected into 8-bit gray image from which the emotion can be estimated by classifiers (see Fig. 2).

2.2 LUT AdaBoost

In order to make AdaBoost algorithm also be suitable for classifying multi-class objects, the theory of LUT is employed. By this method AdaBoost can estimate the facial expression in real-time. First of all we have to let it be able to classify the multi-class object, then treat training classifiers with high quality. The approaches will be indicated in this subsection.

2.2.1 Multi-class AdaBoost

AdaBoost is a learning algorithm that selects a small number of weak classifiers from a large weak classifier pool or hypothesis space to construct a strong classifier. The two-class AdaBoost algorithm has been successfully used in face detection, gender classification, etc. However, many problems such as expression recognition are multi-class in nature. Fortunately, based on a measure of Hamming loss, Schapire and Singer [7] have extended AdaBoost to a multi-class multi-label version. We denote the sample space by $\mathbf{X}$ and the label set by $\mathbf{Y}'$. A sample of a multi-class multi-label problem is a pair (x, Y) , where $x \in \mathbf{X}, Y \subseteq \mathbf{Y}'$. For $Y \subseteq \mathbf{Y}'$, define $Y[l]$ for $l \in \mathbf{Y}'$ as

$$Y[l] = \begin{cases} 1 & if \quad l \in Y \\ -1 & if \quad l \notin Y \end{cases} \tag{4}$$

Given: the sample set $S = \{(x_1, y_1), \cdots, (x_m, y_m)\}$, and the size of the final strong classifier $\mathbf{T}$.
Initialize:$D_t(i, l) = \frac{1}{mk}, i = 1, \cdots, m, l = 1 \cdots, k$, where $k = |\mathbf{Y}'|$. Let $t = 0, 1, 2, \cdots T_i$

1. Under the distribution D_t, select a weak classifier h_t:$\mathbf{X} \times \mathbf{Y}' \rightarrow [-1, 1]$ from the weak classifier pool to maximize the absolute value of

$$r_t = \sum_{i,l} D_t(i,l) Y_i(l) h_t(x_i, l) \tag{5}$$

2. Let

$$\alpha_t = \frac{1}{2} ln \left(\frac{1 + r_t}{1 - r_t} \right) \tag{6}$$

3. Update the distribution:

$$D_{t+1} = \frac{D_t(i,l) exp(-\alpha_t Y_i(l) h_t(x_i, l))}{Z_t} \tag{7}$$

where Z_t is a normalization factor,and D_{t+1} will be a distribution.
4. Output the final strong classifier:

$$H(x, l) = arg \quad max \left(\sum_t \alpha_t h_t(x, l) \right) \tag{8}$$

and the confidence of H can be defined as

$$Conf_H(x) = \left| \frac{\sum_t \alpha_t h_t(x, H(x, l))}{\sum_t \alpha_t} \right| \tag{9}$$

2.2.2 *LUT for Boosting Data Mapping*

In order to train the emotional features by multi-class AdaBoost, weak classifier pool of simple features needs using to be configured. We construct a weak classifier based on the Haar-like feature, which is a kind of simple rectangle feature proposed by Viola and Jones [8], and we can calculated the value of it very fast through the integral image. For each Haar feature, one weak classifier is configured. In their paper, a threshold-type weak classifier, whose output is Boolean value, is used, that is, $h(x) = sign\left[f_{Haar}(x) - b\right]$ where f_{Haar} is the Haar feature and b is a threshold. The main disadvantage of this threshold model is that it is too simple to fit complex distributions, such as a multi-Gaussian model. For a multi-class, $\varpi_1 \cdots \varpi_k$, we use a real-valued 2D LUT type weak classifier, which is proposed by Y.Wang et al. [9]. It is indicated as follows:
1. Assuming f_{Haar} has been normalized to $[0, 1]$, the range is divided into n sub-ranges:

$$bin_j = [\frac{(j-1)}{n}, \frac{j}{n}], j = 1 \cdots, n \tag{10}$$

2. Define the weak classifier $h(x, l)$ as follows

$$if \quad f_{Harr}(x) \in bin_j \quad then \quad h(x, l) = 2P_l^j - 1 \tag{11}$$

note

$$P^{(j)}{}_l = P\left(x \in \varpi_l | f_{Haar}(x) \in bin_j\right) \tag{12}$$

3. Given the characteristic function

$$B_n^{(j,l)}(u, y) = \begin{cases} 1 & u \in [(j-1)/n, j/n] \wedge y = l \\ 0 & otherwise \end{cases} \tag{13}$$

4. Finally, the LUT weak classifier can be formally expressed as:

$$h_{LUT}(x, y) = \sum_{j=1}^{n} \sum_{l=1}^{k} \left(2P_l^j - 1\right) B_n^{(j,l)}(f_{Haar}(x), y) \tag{14}$$

Based on the approaches above, it can be designed as a classifier to extract expressional features, and the expressional samples which are mainly extracted from eyes and mouth, are used to train the classifier.

3. EXPERIMENTS

Based on the theories indicated before, here we have to note that unless it is clearly specified we are dealing with faces under real-time condition. The performance of the system with profile faces is also evaluated in our research. We do not consider the use of both views: frontal and profile, but consider the condition of free facial movement.

3.1 Experimental Dataset

Training Database Set The database of JAFFE [4] is a set of 213 images of different expressions posed by 10 Japanese females. All the face samples were normalized to 64×64-pixel patches and processed by histogram equalization, no color information was used. However, the amount of JAFFE database has only 213 images, they are too less to train the classifier. In order to enhance the generalization performance of AdaBoost learning, we dealt with the images by some transformations (mirror reflection, rotate the images etc.), finally, we got 6816 face samples for training facial expression classifier.
Testing Database Set The testing set was obtained from soap opera and the surrounding persons with the total of 22. And they existed in 22 video clips individually, which were recorded by 25 fps with the length of 30 seconds.
Parameters Setting For training the expression classifier, the database was classified into 7 sets. We set $i = 7$. Parameters for total of 7-type classifiers were set as T_0=720, T_1=T_2=T_3=T_4=T_5=T_6= 1,400. It means that 720 weak classifiers were selected to construct the strong classifer of Natural. The amount of Anger, Disgust, Fear, Happy, Sadness, and Surprise were all set as 1,400.

3.2 Experimental Results

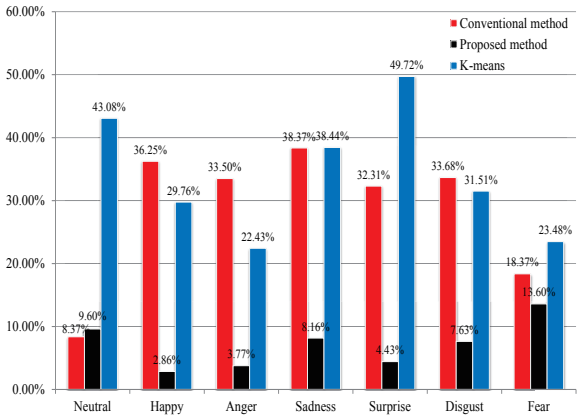

Figure 3: Experimental results

In order to evaluate the proposed method, we compared it among the conventional method which bases on LUT AdaBoost without 3DAF, and the method based on K-means [1] (k=13). Fig. 3 shows the error rate of experimental results in the facial expression estimation. The estimation maximum error rate of proposed method is 13.60%, and the

best result is 2.86%. From this result, we can confirm the validity of proposed method. The conventional method keeps the error rate approximately 28.69%, and the method of K-means is about 34.06%. Table 1 shows the average processing time by PC with CPU Core i3-330M 2.13 GHZ and 2.0 GB memory per one image with the size of 640 × 480.

Table 1: Average processing time per one image.

Method	Proposed	Conventional	K-means
Time cost	19.36 ms	13.11 ms	31.70 ms

Moreover, in order to evaluate the efficiency of the weak classifier constructed by LUT AdaBoost, we compared the efficiency among conventional AdaBoost, Multi-class AdaBoost and LUT AdaBoost. The evaluation is executed through comparing the recognition rate in classifying the Multi-class probability events among the methods mentioned above.

Fig. 4 shows the processing time in training classifiers constructed by the methods mentioned above. The red line indicates the results of LUT AdaBoost, the blue is conventional AdaBoost and the green one shows the data of multi-class AdaBoost. We used the images of training set to test the processing time in training classifiers.

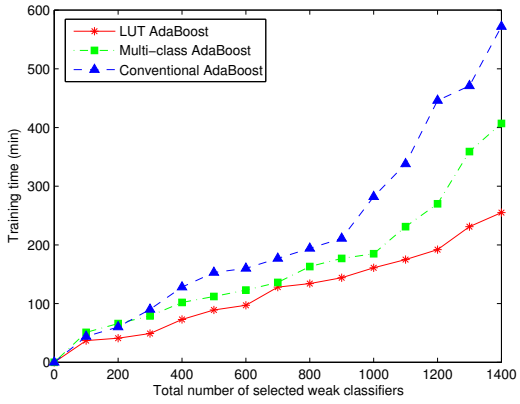

Figure 4: Processing time in training classifiers

In the experiments of classifying the Multi-class probability events, we took use of the JAFFE database to train classifiers. Each class trained by 30 images. We evaluated the efficiency of classifiers for 100 static images, which were obtained by Google Picture Searching Engine in advance. In Fig. 5, the recognition rates of conventional AdaBoost, Multi-class AdaBoost, and LUT AdaBoost are indicated by blue line, green one and the red, respectively. From the figure, we can find that LUT AdaBoost is an excellent method for classifying the multi-class probability events, especially for more than 3 classes. However we also recognized that it is not performing quit good in 2-class recognition, compared with the conventional AdaBoost.

4. CONCLUSIONS

In this paper, we proposed the method combining 3D average face and ameliorated AdaBoost to recognize facial expressions of face free moving. The experiments have shown

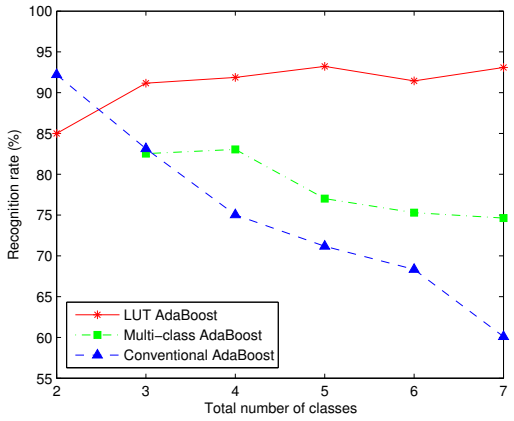

Figure 5: The recognition rate of classifiers

that our approaches obtained more effective classifiers and improved the exaction rate. As for future plan, considering a possible implementation in a real scenario, the research will be focussed on combining with speech recognition.

5. REFERENCES

[1] N. Alldrin, A. Smith, and D. Turnbull. Classifying facial expression with radial basis function networks, using gradient descent and k-means. *CSE253*, 2003.

[2] C. Goodall. Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 285–339, 1991.

[3] J. Gower. Generalized procrustes analysis. *Psychometrika*, 40(1):33–51, 1975.

[4] M. Kamachi, M. Lyons, and J. Gyoba. The japanese female facial expression (jaffe) database. *URL http://www. kasrl. org/jaffe. html*, 21, 1998.

[5] M. Miyahara, M. Aoki, T. Takiguchi, and Y. Ariki. Tagging video contents with positive/negative interest based on user's facial expression. In *Proceedings of the 14th international conference on Advances in multimedia modeling*, pages 210–219. Springer-Verlag, 2008.

[6] M. Pantic and L. Rothkrantz. Automatic analysis of facial expressions: The state of the art. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1424–1445, 2000.

[7] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3):297–336, 1999.

[8] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE, 2001.

[9] Y. Wang, H. Ai, B. Wu, and C. Huang. Real time facial expression recognition with adaboost. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 3, pages 926–929. IEEE, 2004.