A Vision-based Facial Expression Recognition and Adaptation System from Video Stream

Nurul Ahad Tawhid, Nasir Uddin Laskar, and Haider Ali

***Abstract*—The aim of this paper is to develop a real time vision-based facial expression recognition and adaptation system for human-computer interaction. Major objective of this research is to detect face, to identify and recognize user's facial expression using face image in real time and to be able to adapt with new user's facial expression. It also works on mixed race expression detection. It is based upon the eigenface algorithm. Which a small set of feature vectors are used to describe the variation between expression images. It is also being able to adapt new expression image in real time. The proposed system makes major contribution in implementing facial expression recognition and adaptation in real time. The facial expression recognition task is divided into two parts: first part consists of automatic face detection from video stream and preprocessing, second part consists of a classification step that employs Principal Component Analysis (PCA) to classify the expression into one of five categories. The algorithm has been tested using both static and dynamic images. The average precision and recall rate achieved by the system is about 88% for person specific recognition.**

***Index Terms*—Facial action coding system, facial expression recognition, hidden markov model (HMM), neural network (NN), principal component analysis (PCA).**

I. Introduction

Facial expression is one of the most powerful, natural, and immediate means of human beings to convey their emotions and intentions. The extraction and recognition of facial expression by machine can contribute to transmission of facial data and user-friendly communication between user and their computer system. Facial expressions are related to one's emotion state and play an important role in smooth communication among individuals. In this way, computers in the future will be able to offer advice in response to the mood of the users. Since the facial structure is mostly same for humans of a particular race so their facial structure for an expression is also close. This is the main thing which we tried to use in this work. Eigenface method extracts the features from the training images and uses it to categorize the test images. So if a system is trained with the facial expression of a particular race, then it can identify the expressions of that race. In this paper we present results of a fully automatic system for detection of basic emotional expressions from 2D facial images for a particular race. The system automatically categorizes facial expression from frontal faces to five dimensions: Neutral, Happy, Sadness, Surprise and Disgust.

Manuscript received July 18, 2012; revised September 10, 2012. This work was supported in part by University of Information Technology and Sciences (UITS), Dhaka, Bangladesh.

Md. Nurul Ahad Tawhid is with the dept. of IIT, University of Dhaka, Bangladesh (e-mail: tawhid_du@yahoo.com).

MD. Nasir Uddin Laskar is with the dept. of CSE and IT, UITS, Dahaka, on study leave. He is now pursuing MS in Kyung Hee University (e-mail: nasir_csedu@yahoo.com).

Md. Haider Ali is with the dep. of CSE, University of Dhaka (e-mail: haider@cse.univdhaka.edu).

II. Related Works

All facial expression recognition methods can be classified into two broad-based categories: feature based approach and probabilistic approach. The feature-based method utilizes the Facial Action Coding System (FACS) designed by Ekman and Friser [1]. In FACS, the motions of the face are divided into 44 action units (AU), and their combinations may describe any facial expression. More than 7,000 combinations of AU have been observed [2]. The probabilistic-based method does not give preference to facial features such as eyes and mouth. Instead, the feature vector can be the random distribution of image intensities and these vectors may differ from each emotion. The vectors are calculated per emotion and classification algorithms like HMM, Neural Network (NN) or a hybrid approach (HMM and NN) [3] are applied. There are some other techniques like MPEG-4 Facial Animation Parameter (FAP), measurement of facial motion through optic flow. Turk and Pentland [7] proposed Eigenfaces employed PCA which is an unsupervised learning method and treats samples of the different classes in the same way. Nowadays, various researchers reported the model-based methods [4,5] for feature extraction, such as active appearance model [6], point distribution model and labeled graphs. But those methods require heavy computation or manually detected feature nodes to construct the model, which can hardly be implemented in real-time automatic facial expression recognition (FER).

III. Methodology

In the eigenface method, the test image is compared against the training set images to find the best match of the expression. To pass through the eigenface method the test image is preprocessed, so that the features of the face, which are important to expression detection, are get sharper. Face detection from the image and the pre-processing steps are as follows:

A. Face Detection

In this paper we will attempt to detect face from an image using a template matching technology provided by the [FaceVACS]. To locate the face, an image pyramid is formed from the original image. An image pyramid is a set of copies of the original image at different scales, thus representing a

set of different resolutions. A mask is moved pixel wise over each image in the pyramid, and at each position, the image section under the mask is passed to a function that assesses the similarity of the image section to a face. If the similarity value is high enough, the presence of a face at that position and resolution is assumed. From that position and resolution, the position and size of the face in the original image can be calculated. From the position of the face, a first estimate of the eye positions can be derived. In a neighborhood around these estimated positions, a search for the exact eye positions is started. This search is very similar to the search for the face position, the main difference being that the resolution of the images in the pyramid is higher than the resolution at which the face was found before. The positions yielding the highest similarity values are taken as final estimates of the eye positions.

B. Image Normalization

The face image is normalized before passing to the facial expression recognition module. Sequences of image pre-processing techniques are applied so that the image is light and noise invariant and the facial feature points become sharper. Then it needs to apply some standard expression recognition pre-requisite such as grey image conversion and scaling into a suitable sized image.

1) Conversion to Grey Image and Scaling

Detected face is converted to grayscale using (1) and scaled to 132×132 pixels using (2) and saved as a gray bmp image. Linear interpolation technique was employed to determine the scaled output image.

$$Gr_i = \frac{R_i + G_i + B_i}{3}, \quad i = 1,2,...,M \quad x\ N \tag{1}$$

where, Gr_i is the gray level value of i^{th} pixel of the gray image. R_i, G_i, B_i corresponds to red, green, blue value of the i^{th} pixel in the color image.

$$Q(x^q, y^q) = P\left(\frac{x^p}{132}x^q, \frac{y^p}{132}y^q\right) \tag{2}$$

where, we want to re-scale image $P[(0,0)-(x^p, y^p)]$ to image $Q[(0,0)-(132\times132)]$

2) Contrast Stretching

Frequently, an image brightness values do not make full use of the available dynamic range. The following formula is used in stretching the histogram over the available dynamic range:

$$b[m,n] = \begin{cases} 0 & a[m,n] \le p_{low}\% \\ (2^B-1)\cdot\dfrac{a[m,n]-p_{low}\%}{p_{high}\%-p_{low}\%} & p_{low}\% < a[m,n] < p_{high}\% \\ (2^B-1) & a[m,n] \ge p_{high}\% \end{cases} \tag{3}$$

Here, we might choose the 1% and 99% values for $p_{\text{low}}\%$ and $p_{\text{high}}\%$, respectively, instead of the 0% and 100% values represented by the equation. It is also possible to apply the contrast-stretching operation on a regional basis using the histogram from a region to determine the appropriate limits for the algorithm.

3) Histogram Equalization

To compare two or more images on a specific basis, such as texture, it is common to first normalize their histograms to a 'standard' histogram. The most common histogram normalization technique is *histogram equalization* where one attempts to change the histogram through the use of a function $b = F(a)$ into a histogram that is constant for all brightness values.

For a 'suitable' function $F(*)$ the relation between the input probability density function, the output probability density function, and the function $F(*)$ is given by:

$$p_b(b)db = p_a(a)da \quad \Rightarrow \quad df = \frac{p_a(a)da}{p_b(b)} \tag{4}$$

From (4) we see that 'suitable' means $F(*)$ is differentiable and $dF/da >= 0$. For histogram equalization we desire $p_b(b) = constant$ and this means that:

$$f(a) = (2^B - 1)\cdot P(a) \tag{5}$$

where, $P(a)$ is the probability *distribution* function. In other words, the *quantized* probability distribution function normalized from 0 to 2^B*-1* is the look-up table required for histogram equalization.

IV. Facial Expression Recognition

A. Background

Much of the previous work on automated facial expression recognition has ignored the issue of just what aspects of the face stimulus are important for expression recognition. In the language of information theory, the relevant information in a face image is extracted, encoded as efficiently as possible, and then compared with a database of models encoded similarly. A simple approach to extracting the information contained in an image of a facial expression is to somehow capture the variation in a collection of facial expression images, independent of any judgment of features, and use this information to encode and compare individual facial expression images to detect expression. In mathematical terms, the principal components of the distribution of facial expression, or the eigenvectors of the covariance matrix of the set of facial expression images, treating an image as point (or vector) in a very high dimensional space is sought. Turk and Pentland [7, 8] proposed Eigenfaces employed principal component analysis (PCA). PCA is an unsupervised learning method, which treats samples of the different classes in the same way. The eigenvectors are ordered, each one accounting for a different amount of the variation among the facial expression images. These eigenvectors can be thought of as a set of features that together characterize the variation between expression images. Each image location contributes more or less to each eigenvector, so that it is possible to display these eigenvectors as a sort of ghostly face image which is called an 'eigenface' [7].

Each individual facial expression can be represented exactly in terms of a linear combination of the eigenfaces. Each facial expression can also be approximated using only the 'best' eigenfaces, those that have the largest eigenvalues, and which therefore account for the most variance within the set of expression images. The best M eigenfaces span an M-dimensional subspace which we call the 'facial expression

space' of all possible images.

B. Calculating Eigenfaces

Let a face image *I(x,y)* be a two-dimensional $N \times N$ array of 8-bit intensity values. An image may also be considered as a vector of dimension N^2, so that a typical image of size 256 × 256 becomes a vector of dimension 65,536, or equivalently a point in 65,536-dimensional space. An ensemble of images, then, maps to a collection of points in this huge space.

These vectors define the subspace of face images, which we call 'face space'. Each vector is of length N^2, describes an $N \times N$ image, and is a linear combination of the original face images. Because these vectors are the eigenvectors of the covariance matrix corresponding to the original face images, and because they are face-like in appearance, we refer to them as 'eigenfaces'. Examples of eigenfaces of Fig. 1 are shown in Fig. 2.

Let the training set of face images be $\Gamma_1, \Gamma_2, \ldots, \Gamma_M$ then the average of the set is defined by

$$\Psi = \frac{1}{M} \sum_{n=1}^{M} \Gamma_n \quad (6)$$

Each face differs from the average by the vector

$$\Phi_i = \Gamma_i - \Psi \quad (7)$$

An example training set is shown in Fig. 1. This set of very large vectors is then subject to principal component analysis, which seeks a set of M ortho-normal vectors u_n which best describes the distribution of the data. The k'th vector, u_k, is chosen such that

$$\lambda_k = \frac{1}{M} \sum_{n=1}^{M} \left(u_k^T \Phi_n \right)^2 \quad (8)$$

is a maximum, subject to

$$u_l^T u_k = \delta_{lk} = \begin{cases} 1, & \text{if } l = k \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

The vectors u_k and scalars λ_k are the eigenvectors and eigenvalues, respectively of the covariance matrix

$$C = \frac{1}{M} \sum_{n=1}^{M} \Phi_n \Phi_n^T = AA^T \quad (10)$$

where, the matrix $A = [\Phi_1 \Phi_2 \ldots \Phi_M]$ the covariance matrix C, however is $N^2 \times N^2$ real symmetric matrix, and determining the N^2 eigenvectors and eigenvalues is an intractable task for typical image sizes. We need a computationally feasible method to find these eigenvectors.

If the number of data points in the image space is less than the dimension of the space ($M < N^2$), there will be only $M - 1$, rather than N^2, meaningful eigenvectors. The remaining eigenvectors will have associated eigenvalues of zero. We can solve for the N^2 dimensional eigenvectors in this case by first solving the eigenvectors of an $M \times M$ matrix such as solving 16 × 16 matrix rather than a 16,384 × 16,384 matrix and then, taking appropriate linear combinations of the face images Φ_i.

Consider the eigenvectors v_i of a A^TA such that

$$A^TAV_l = \mu_l v_l \quad (11)$$

From which we see that Av_i are the eigenvectors of $C = AA^T$

Following these analysis, we construct the $M \times M$ matrix $L = A^TA$ where $L_{mn} = \Phi_m^T \Phi_n$ and find the M eigenvectors, v_l, of L. These vectors determine linear combinations of the M training set face images to form the eigenfaces u_l.

$$u_l = \sum_{k=1}^{M} v_{lk} \Phi_k \qquad l = 1, 2, \ldots M \quad (12)$$

With this analysis, the calculations are greatly reduced, from the order of the number of pixels in the images (N^2) to the order of the number of images in the training set (M).

Fig. 1. Example of training face images [9].

Fig. 2. Eigenfaces with highest eigenvalues

C. Using Eigenfaces to Classify a Facial Expression

'Accurate reconstruction of the image is not a requirement'- based on this idea, the proposed expression recognition system lets the user specify the number of eigenfaces *(M')* that is going to be used in the recognition. For maximum accuracy, the number of eigenfaces should be equal to the number of images in the training set. But, it was observed that, for a training set of fourteen face images, seven eigenfaces were enough for a sufficient description of the training set members.

In this framework, identification becomes a pattern recognition task. The eigenfaces span an M' dimensional subspace of the original N^2 image space. The M' significant eigenvectors of the L matrix are chosen as those with the largest associated eigenvalues. A new face image (Γ) is transformed into its eigenface components (projected onto "face space") by the operation,

$$\omega_k = u_k^T \left(\Gamma - \Psi \right) \quad (13)$$

For $k = 1,...,M'$. This describes a set of point by point image multiplications and summations, operations performed at approximately frame rate on current image processing hardware.

Classification is performed by comparing the feature vectors of the face library members with the feature vector of the input images (mouth and eye). This comparison is based on the Euclidean distance between the two members. If the comparison falls within the user defined threshold, then the image is classified as 'known' expression for that, otherwise it is classified as 'unknown' and can be added to expression library with its feature vector for later use.

V. Architecture and Experimental Results

A. System Organization

Fig. 3 gives a high level description of the proposed system. The facial expression recognition module uses eigenspace-based principal component analysis approach to recognize known expressions. If an expression cannot be recognized as known, then the new person's expression adaptation module will be activated. The adaptation module in this system captures an image of the unknown person's expression and re-computes its person specific feature vectors using the eigenface method described earlier. The proposed facial expression recognition system passes through three main phases during an expression recognition process. They are:

1) Face Library Formation Phase

In this phase, face images are stored in a face library in the system. Every action such as training set or eigenface formation is performed on this face library. In order to start the expression recognition process, this initially empty face library has to be filled with facial expression images. After acquisition and pre-processing, face images under consideration is added to the face library. Weight vectors of the face library members are empty until a training set is chosen and eigenfaces formed.

2) Training Phase

After choosing the training set, eigenfaces are formed and stored for later use. Eigenfaces are calculated from the training set, keeping only the M images that correspond to the highest eigenvalues. These M eigenfaces define the M-dimensional "face space". As new faces are experienced, the eigenfaces can be updated or recalculated. Accordingly the corresponding weight vector of each face library member has been updated which were initially empty.

3) Recognition and Learning Phase

After obtaining the weight vector, it is compared with the weight vector of every face library member within a user defined "threshold". If there exists at least one face library member that is similar to the acquired image within that threshold then, the face image is classified as 'known'. Otherwise, a miss has occurred and the face image is classified as 'unknown'. After being classified as unknown with certainty, this new face image can be added to the face library with its corresponding weight vector for later use (learning to recognize).

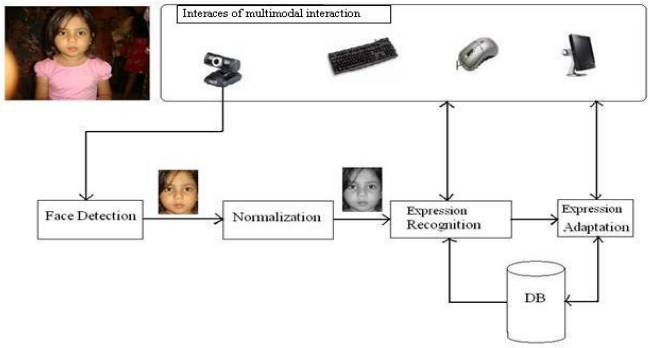

Fig. 3. System architecture for expression recognition and adaptation

B. Experimental Results and Performance Evaluation

Table I shows the confusion matrix for the results of facial expression recognition algorithm with image preprocessing and adaptation with one image per person's facial expression. The diagonal elements represent the correct recognition of corresponding expression.

TABLE I: Confusion Matrix of Expression Recognition (Person Specific)

			Detected facial expression					
		Total	Neu	Happy	Sad	Sur	Disg	Success
Input Expression	Neutral	20	19		1			95%
	Happy	20	1	18		1		90%
	Sad	20	1		17	1	1	85%
	Surprise	20		2	1	17		85%
	Disgust	20			1	2	17	85%

We need to define two parameters for the evaluation of our method's performance. Table II presents the precision (%) and recall (%) rates of facial expression recognition method. The precision (%) is defined by the ratio of the numbers of correct recognition to total numbers of recognition for each expression. The recall rate (%) is defined by the ratio of the numbers of correct expression recognition to total numbers of input expression images for each person.

TABLE II: Performance of Expression Recognition (Person Specific)

Expression	Precision (%)	Avg. Precision	Recall (%)	Avg. Recall
Neutral	90.48		95.0	
Happy	90.00		90.0	
Sad	85.00	88.17%	85.0	88%
Surprised	80.95		85.0	
Disgust	94.44		85.0	

Next we perform a variation above settings where we do not adapt the system with one image per person's expression. Instead we randomly selected 40 images from 120 images. These 40 images contain 10 images per expression independent of the persons. Table III shows the confusion matrix for the results of the system for this configuration

TABLE III: CONFUSION MATRIX OF EXPRESSION RECOGNITION (PERSON INDEPENDENT)

		Total	Detected facial expression					Success
			Neu	Happy	Sad	Sur	Disg	
Input Expression	Neutral	20	17	2	1			85%
	Happy	20	2	15	1	1	1	75%
	Sad	20	2	1	14	1	2	70%
	Surprise	20	2	2	1	13	2	65%
	Disgust	20	1	1	2	2	14	70%

TABLE IV: PERFORMANCE OF EXPRESSION RECOGNITION (PERSON INDEPENDENT)

Expression	Precision (%)	Avg. Precision	Recall (%)	Avg. Recall
Neutral	70.83		85.0	
Happy	71.43		75.0	
Sad	73.68		70.0	
Surprised	76.47	73.22%	65.0	73%
Disgust	73.68		70.0	

If the light condition is good and not constantly changing, if the user's face orientation does not undergo on major variation, the system always performs according to above success rates. As we can see in case of person specific method, the recall rate and the precision rates are quite high than the existing results, suggesting that with image preprocessing and one training image/person's expression our recognition system produces satisfactory outcome on a wide range of real time environment Fig. 4 compares the results of the above experiments.

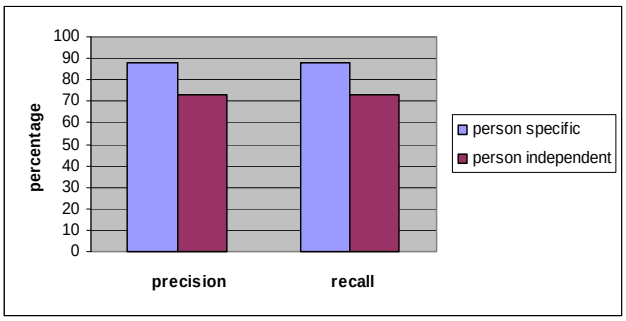

Fig. 4. Performance comparisons with training image variation

VI. CONCLUSION

Facial expression recognition in real time is at the crossroads of some critical optimization issues. This work makes major contribution in areas of facial expression recognition. There are several systems for facial expression recognition. These systems, although were successful in terms of recall and precision rates, but most of these systems could not be used in real time. The proposed system, on the other hand, can run in real time with reasonable success rate. The system achieved on average 88.17% and 88% precision and recalls rates with person specific and 73.22% and 73% precision and recall rates for person independent expression recognition. Another area of contribution of the proposed work is the new user's expression adaptation algorithm that facilitates registration of unknown persons' facial images with the system in real time. Real time expression adaptation is an impressive achievement that eases interaction between user and the system.

REFERENCES

[1] P. Ekman and W. Frisen, "Facial action coding system," Consulting Psychologists Press, 1977.
[2] P. Ekman, "Methods for measuring facial actions," *Handbook of Methods in Nonverbal Behavior Research, Cambridge*: Cambridge University, 1982, pp. 45-90.
[3] I T. Hu, L. C. D. Silva, and K. Sengupta, "A hybrid approach of nn and hmm for facial emotion classifiation," *ELSEVIER Pattern Recognition Letters Journal*, vol. 23, no. 11, 2002, pp. 1303-1310.
[4] I. A. Essa and A. P. Pentland, "Coding, analysis, interpretation, and recognition of facial expressions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no. 11, 1997, pp. 757-763.
[5] B. Fasel and J. Luettin, "Automatic facial expression analysis: a survey," *Pattern Recognition*, vol. 36, 2003, pp. 259-275.
[6] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, 2001, pp. 681-685.
[7] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," IEEE, 1991, pp. 586-591.
[8] K. M.. Sabrin, T. Zhang, S. Chen, M. Nurul Ahad Tawhid, M. Hasanuzzaman, M. Haider Ali, and H. Ueno, "An Intensity and Size Invariant Real Time Face Recognition Approach", Proc. Of ICIAR 2009, LNCS 5627, pp. 502–511 Springer-Verlag Berlin Heidelburg 2009, pp 502-511
[9] Michael J. Lyons, Shigeru Akamatsu, Miyuki Kamachi, Jiro Gyoba Proceedings, "Coding facial expressions with gabor wavelets" *Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara Japan*, IEEE Computer Society, 1998, pp. 200-205.

Md. Nurul Ahad Tawhid was born in 1985 in Bangladesh. He has completed his graduation from University of Dhaka, Dhaka, Bangladesh, in Computer Science and Engineering in 2008. He also completed his post graduation for the same university in same subject in 2010. His major field of study was Image Processing and Artificial Intelligence.

He is currently working as Lecturer in Institute of Information Technology, university of Dhaka, Dhaka, Bangladesh. Before joining here, he was worked as Software Engineer in M&H Informatics BD Ltd (An IMS Health Company), Bangladesh.

Md. Nasir Uddin Laskar received his B. Sc. Degree in Computer Science and Engineering from University of Dhaka, Bangladesh in 2008. Currently he is a Faculty member in the dept. of Computer Science and Engineering, University of Information Technology and Sciences (UITS), Dhaka, Bangladesh. At present, he is on study leave and pursuing the MS degree in Artificial Intelligence Lab, dept. of Computer Engineering, Kyung Hee University, Korea.

His current research interests include neural network, machine learning and robotics. Mr. Nasir is a member of IACSIT.