Autonomous Robot Control Using Facial Expressions

Rohin Mittal, Prateek Srivastava, Alpha George, and Asim Mukherjee

***Abstract*—In this paper, a novel real time robot control system using human facial expressions is presented. The proposed system mainly consists of three modules: face detection, facial expression recognition and robot control. The first module aims to find the user's face from the image captured from a live video through a series of steps including skin color classification, edge detection and mathematical morphology, while the second module analyses the detected face to recognize the facial expressions like happiness, sadness, surprise, anger and neutral using Principal Component Analysis (PCA) and Euclidian distance calculation. Finally, the detected facial expressions are used as controlling commands for the robot. Experiments were conducted for respective modules. 150 images under different lightening conditions and complex background were employed. In the experiment on detecting the facial region, the rate of detection was 99.3% and on recognizing the facial expression, the recognition rate was 97.3%. The robot was then controlled autonomously with an accuracy of 100%. Combining the above three rates, the overall rate of success of our system was 97.3%.**

***Index Terms*— Autonomous robot, euclidian distance, face detection, facial expression recognition, principle component analysis.**

I. Introduction

Recently, the interaction between humans and robots has become an important issue. Autonomous robot control system is a complex system that consists of a sensor, a decision-making control system and a motor drive system. The sensor can either be a visual system, a speech system, or a manual control system.

In [1], [2], numerous systems are proposed to control a robot or a wheelchair using head or face movement. Such systems involve body movement and are not suitable for people with extreme physical disabilities where head or face movement is difficult. Speech controlled systems [3] are also not suitable for people with speech disability. Thus, current research has been focused on design of systems, which can be a good solution to these problems. The best alternative is to design a system where control is derived from recognizing the user's facial expressions like happiness, sadness, surprise, anger and neutral by processing images captured from a live video. Such systems do not require any physical movement of the body and hence can be sufficient for all kinds of people.

Recognizing the facial expression mainly consists of two steps: face detection and expression recognition. Many novel methods have been proposed for face detection. A detailed review over work in the face detection domain can be found in [4], [5]. T. Liu, H. Guo, and Y. Wang [6] provided an approach combining multiple color models for stable color-based face detection. T. Lalanne and C. Lempereur [7] used a learning mechanism based on neural network for color measurement. After the face is detected, facial expression of the user is to be identified. A survey on automatic facial expression recognition can be found in [8], [9] and [10]. The survey revealed that most of the recognition systems are based on Facial Action Coding System (FACS) which describes 44 different action units (AUs), each of which is related to the contraction or relaxation of one or more facial muscle. However, more than 7000 different combinations of action units are possible due to variations in age, size and ethnicity which is a problematic issue.

In this paper, we present an autonomous robot control system for helping the physically disabled people. The main task of the system is to obtain and analyze real-time images, precisely recognize a number of human face expressions like happiness, sadness, surprise, anger and neutral representing a set of commands that the user can give to the robot. The system is able to detect the facial expressions in both indoor and outdoor lighting conditions and distinguish between the user's face and the background faces.

The organization of the remainder of this paper is as follows. Section II describes the overview of the proposed robot control system. Section III explains how to detect a face from the input image under variable lighting environments. Facial Expression recognition is described in Section IV. Hardware implementation is done is Section V. Experimental results are mentioned in Section VI. Finally, conclusions and future works are presented in Section VII.

II. Overview of the Proposed System

The flowchart of the proposed facial expression based robot control system is illustrated in Fig. 1.

An algorithm is developed which captures images from a live video under varying lighting conditions. Initially, face detection is done which involves a series of steps like skin color classification, noise removal, edge detection and mathematical morphology. The face region with maximum area is then detected since the user's face will be nearest to the camera and will be of biggest size. Certain facial features are also detected to confirm if the biggest region detected is actually a face.

Manuscript received June 05, 2012; revised July 5, 2012.

Rohin Mittal is with the Apache Design Solutions Pvt. Ltd., Bengaluru, India (e-mail: rohinmittal2007@gmail.com).

Prateek Srivastava is with the NTPC Ltd., Singrauli, India (email: prateek.annie@gmail.com)

Alpha George is with the Samsung Software India R and D Center, Noida, India (email: alphageee@gmail.com)

Asim Mukherjee is with the Electronics and Communication Engineering Department at Motilal Nehru National Institute of Technology, Allahabad, India (email: asimmkj@mnnit.ac.in)

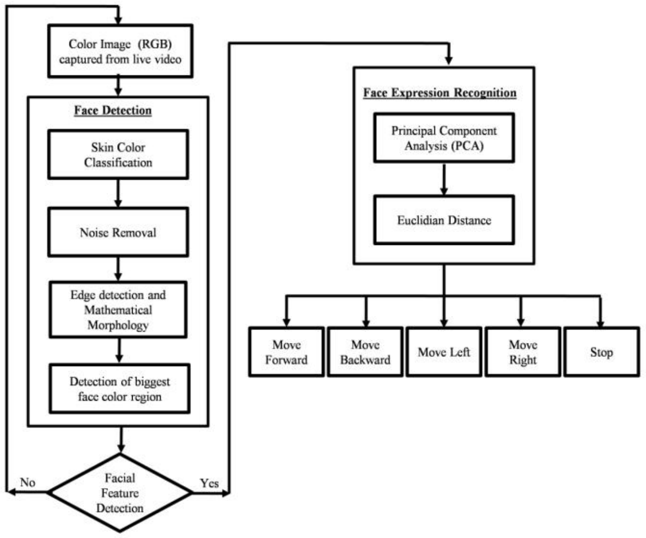
Color Image (RGB) captured from live video
Face Detection
Skin Color Classification
Noise Removal
Edge detection and Mathematical Morphology
Detection of biggest face color region
Facial Feature Detection
No
Yes
Face Expression Recognition
Principal Component Analysis (PCA)
Euclidian Distance
Move Forward
Move Backward
Move Left
Move Right
Stop

Fig. 1. Flowchart of the proposed algorithm of facial expression based robot control system.

To recognize the facial expressions from the detected face, eigenspace analysis [11] is used. Once the facial expressions are detected, controlling signals are sent to the robot using parallel port which makes the robot move forward, backward, left, right and stop.

III. Face Detection

Face detection deals with an image processing problem of locating facial region in a given image. It is the first task that needs to be successfully achieved before further steps such as facial expression recognition can be carried out. The most popular face detection algorithm is the use of color information in which regions with skin color are first detected. Some recent publications that have reported this study include [12]-[17]. These studies reveal that color information can be used for extraction of a face from an image.

A. Skin Color Segmentation

Most of the work on face detection is based on color models like RGB (red, green, and blue), HSI (hue, saturation, and intensity), *YCbCr* (luminance, blue-difference and red-difference chroma components), etc.

RGB color model is an additive color model and hence, the RGB components may vary if the lighting condition changes. This raises difficulties in detection of faces in varying lighting conditions. In *YCbCr* color model, *Y*, *Cb* and *Cr* components represent luminance, blue difference and red difference respectively. [18], [19] conclude that, across different human faces, pixels belonging to skin region have similar *Cb* and *Cr* values. Fairness or darkness of the skin is determined by the *Y* component. Thus, *Cb* and *Cr* values can be used to filter out skin region from non-skin region. Therefore, we introduce a color transformation model, where the RGB color space is converted to *YCbCr* color space. In this paper, we use a 2MP, Microsoft *VX*-3000 web camera to capture images of 640 × 320 pixel resolution in RGB color space. The image captured in Fig. 2(a) is in uint8 format since it requires less memory. But many mathematical functions can only be applied to the image in double format; hence the captured image is first converted to double precision in which the data values are floating-point numbers in the range [0, 1] where the value 0 corresponds to black and the value 1 corresponds to white. The image is then transformed from RGB to YCbCr using (1).

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0.257 & 0.504 & 0.098 \\ -0.148 & -0.291 & 0.439 \\ 0.439 & -0.368 & -0.071 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{1}$$

The RGB image is converted to *YCbCr* image as shown in Fig. 2(b). If the number of pixels falling into a particular region with a certain (*Cb*, *Cr*) value exceeds a threshold value, that pixel is considered as a skin color and the image is transformed to a binary image *F* using (2).

$$\begin{aligned} F\,(Cb, Cr) &= 1 \text{ if } (Cb, Cr) \in S \\ &= 0 \text{ if } (Cb, Cr) \notin S \end{aligned} \tag{2}$$

where *S* represents a threshold range.

B. Noise Removal

After converting the image to a binary image, there may still be some pixels which are detected as skin pixels but actually are not skin pixels. These pixels are considered as noise. In order to remove high frequency noise, a low pass filter by a 5× 5 mask is implemented. The numbers of white pixels are calculated in blocks of 5 × 5. Every pixel of the block is set to white if the number of white pixels is more than half of total pixels in that block; else this 5 × 5 block is transformed to a complete black block. The algorithm for noise removal can be described as follows:

1) Consider a 5 × 5 block in the image of size *H* × *W*.
2) Count total number of 1's in the block. Let the count value be *S*.
3) If *S* > 20,
   - Goto step 4 else,
   - Goto step 5.
4) Convert every pixel of the block to 1.
5) Convert every pixel of the block to 0.
6) Goto step 1 with next block.

Fig. 2(c) shows that high frequency noise has been removed.

C. Edge Detection and Mathematical Morphology

The controlling commands given to the robot will be from a single user. Therefore, detection of multiple faces will produce wrong results. Since the user closest to the camera will be the main user controlling the robot and will have the face with maximum area, this region is considered for further processing.

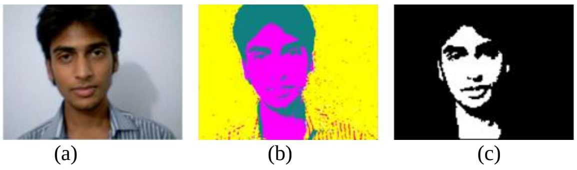
(a) (b) (c)

Fig. 2. Output at different stages of face detection. (a) Input image, (b) Image after color space transformation, (c) Skin representing region after noise removal.

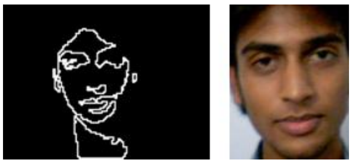

Fig. 3. (a) Edge detection (b) Region with maximum area.

This is done using Sobel edge detector. The detector performs a simple, quick to compute, 2-D spatial gradient measurements on an image. Two 3× 3 kernels (h_x and h_y) are used which are convolved with the original binary image F to calculate approximations of the derivatives – one for edges in horizontal direction and other for edges in vertical direction. G_x in (3) and G_y in (4) are the two images which at each point contain the vertical and horizontal derivative approximations.

$$G_x = h_x * F \tag{3}$$

$$G_y = h_y * F \tag{4}$$

where, $h_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$, $h_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

At each point in the image, the resulting gradient approximations G_x and G_y can be combined to give the gradient magnitude G using (5).

$$G = \sqrt{G_x^2 + G_y^2} \tag{5}$$

Edge dilation (Mathematical Morphology) using a 3 x 3 square as structuring element is performed after edges are detected to get proper enclosed boundaries. Fig. 3(a) shows the image after edges are detected and dilated. The detected closed boundaries are filled with white pixels and the region with largest area is considered as the face and the image is cropped as shown in Fig. 3(b).

D. Detecting Facial Features

The region in Fig. 3(b) may be an object which is actually not a face but where the chrominance values coincide with those of the skin color e.g., leather, wood, clothes and hair. If such objects are detected to be of biggest size, then this region is discarded at this stage and a new image is captured. This is done by directly locating eyes and mouth based on measurements derived from the color space components of an image. Rein-Lien Hsu, Mohamed Abdel-Mottaleb, Anil *K.* Jain in [20] analyzed the chrominance components and indicated that high *Cb* and low *Cr* values are found around the eyes and the chrominance component *Cr* is greater than *Cb* near the mouth areas.

After eyes and mouth are detected, two conditions are verified. First, the line joining the eyes is perpendicular to the line joining the mouth to the mid-point of the line joining the eyes, Fig. 4(a). Second, the triangle joining all the three coordinates is an isosceles triangle, Fig. 4(b). If these two conditions are satisfied, the detected face is resized to $M \times M$ else the camera captures a new image and follows the same procedure.

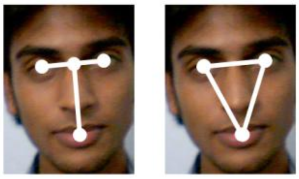

Fig. 4. (a) Line joining eyes is perpendicular to the line joining mouth to the midpoint of the line joining eyes. (b) Isosceles triangle joining eyes and mouth.

IV. Facial Expression Recognition

Once the face is detected by the model described in section III, facial expression is recognized using eigen-space analysis [11]. The facial expression recognition algorithm is user dependent which means that robot control system is to be trained using different expressions of the user. For this, a set of 135 images called Training Set is created and classified in the following expressional classes:

1) Image001 to Image027 = Happiness
2) Image028 to Image055 = Surprise
3) Image056 to Image083 = Anger
4) Image084 to Image111 = Sadness
5) Image112 to Image135 = Neutral

The training for a particular user is done only once. Since the images are captured and stored in the directory, there is no need to train the system again. Thereafter, a face space or the basis set is deduced using Principal Component Analysis (PCA).

Let the images after face detection be I_1, I_2, $I_3 \ldots I_N$. Problems arise when recognition is done on high dimensional space, therefore these face images of size $M \times M$ are first mapped into a low dimensional space of size $M^2 \times 1$. Hence, every face image I_i ($M \times M$ matrix) is represented as a vector Γ_i ($M^2 \times 1$ matrix) where i=1,2,3…N. We compute the mean image Ψ_{train} of the training data using (6).

$$\Psi_{train} = \frac{1}{N} \sum_{n=1}^{N} \Gamma_n \tag{6}$$

where, Ψ_{train} is the mean image vector, Γ_n is the training image vector, N is the number of training images.

The mean image vector Ψ_{train} is subtracted from the training image vectors Γ_n using (7) and the covariance matrix C is calculated using (8).

$$\Phi_n = \Gamma_n - \Psi_{train}\text{, n=1,2.......N} \tag{7}$$

$$C = \frac{1}{N} \sum_{n=1}^{N} \Phi_n \Phi_n^T = AA^T \tag{8}$$

where, A = $\left[\Phi_1 \Phi_2 \Phi_3 \ldots . \Phi_N \right]$ of dimension $M2 \times N$.

The next step consists of finding the eigenvectors, u_i of the covariance matrix, C (or AA^T) which represents the "face space". AA^T can have up to M^2 eigenvalues and eigenvectors, calculation of which is impractical but A^TA can have up to N eigenvalues and eigenvectors. The N eigenvalues of A^TA

(along with their corresponding eigenvectors) correspond to the N largest eigenvalues of AA^T (along with their corresponding eigenvectors). Therefore, we compute the eigenvectors of matrix A^TA (N x N matrix). The eigenvectors of AA^T and A^TA are related as shown in (9).

$$u_k = Av_k \qquad (9)$$

where u_k and v_k are the eigenvectors of AA^T and A^TA respectively. From these eigenvectors, the weights for each image in the training set are computed as in (10).

$$\omega_{ik} = u_k^T(\Gamma_i - \Psi_{train}), \qquad k=1,2,3\ldots\ldots N \qquad (10)$$

These weights form a vector $\Omega^T = [\omega_1\omega_2\omega_3\ldots.\omega_N]$ which describes the contribution of each eigenface in representing the input image.

During the testing phase, an image Γ_{test} is captured from the live video and is projected on the face space as shown in (11).

$$\omega_{test.k} = u_k^T(\Gamma_{test} - \Psi_{train}),\ k=1,2,3\ldots\ldots N \qquad (11)$$

The Euclidian distance of this projected test image $\omega_{test.k}$ is calculated from all the projected train images Ω and the minimum value is chosen in order to find out the train image which is most similar to the test image. The test image is assumed to fall in the same class of the facial expression that the closest train image belongs to.

V. Hardware Implementation

This section describes the design of hardware circuit to drive the robot after recognizing the facial expressions. The hardware consists of a 74HC244 buffer IC, L293D dual H-Bridge motor driver, a parallel port and two DC motors. The block diagram of the hardware circuit is shown in Fig. 5.

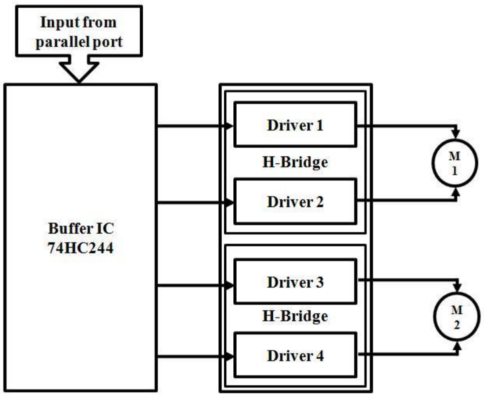

Fig. 5. Block diagram of hardware circuit.

TABLE I: Working of One H-Bridge of L293D Motor Driver.

INPUTS		DESCRIPTION
Input 1	Input 2	
0	0	Motor Stops
0	1	Clockwise
1	0	Anti-clockwise
1	1	Motor Stops

TABLE II: Working of Robot Control System.

INPUTS				CONTROL SYSTEM'S OUTPUT
*M*1		*M*2		
IN1	IN2	IN3	IN4	
0	0	0	0	Robot Stops
0	1	0	1	Turns Right
1	0	1	0	Turns Left
1	0	0	1	Move Forward
0	1	1	0	Move Backward

L293D dual H-bridge motor driver consist of two H-bridges and two DC motors can be interfaced which can be controlled in both clockwise and counter clockwise direction. Working of one of the H-bridges of the driver is shown in Table I.

After the expression recognition algorithm recognizes the expression of the user, it sends the corresponding controlling signal to the robot using parallel port. This circuit needs high current which should not be withdrawn from the parallel port. Hence, a buffer IC is used between the parallel port and H-Bridge. As shown in Table II, the robot turns right when both the motors *M*1 and *M*2 rotate in clockwise direction and the robot turns left when both the motors *M*1 and *M*2 rotate in anti-clockwise direction. For forward and backward motion, the motors rotate in opposite direction.

VI. Experimental Results

To assess the validity of the proposed system, the user was asked to change the facial expression 150 times. The accuracy of our proposed system at various stages is shown in Table III. The overall accuracy of the system was found to be 97.3%.

The robot was found to successfully recognize the facial expressions and move forward, backward, right, left and stopped when the expression of the user was Happiness, Sadness, Surprise, Anger and Neutral respectively. Moreover, as shown in Fig. 6, our system requires the smaller user motion than any other robot control system. This tells us that the proposed system is more comfortable than any other conventional methods.

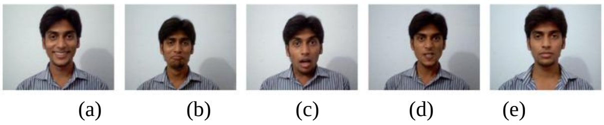

Fig. 6. Control commands of the designed system (a) Forward (b) Backward (c) Right (d) Left (e) Stop..

TABLE III: Accuracy at Various Stages.

Stage	Accuracy
Face Detection	99.3%
Facial Expression	97.3%
Robot Movement	100%
Overall Accuracy	*97.3%*

VII. Conclusion and Future Work

In this paper, a robot control system using human facial expressions is presented. The major advantage of this system is that it does not require any physical body movement and is really comfortable for people with extreme physical disabilities. Moreover, using YCbCr color space model, we achieved very good results for face detection in different lighting environments, such as indoor white light, indoor yellow light, and outdoor sunlight. After face detection, computational complexity was reduced during facial expression recognition because the eigenface analysis was done on a part of the acquired image i.e. face region only.

The main limitation of the robot control system is that since the expressions like happiness, sadness, surprise, anger and neutral are used as controlling commands, even when expressed naturally, they may be mistaken for a command.

In order to overcome the limitation mentioned, future developments will be focused on studying the nervous system activities of the user during the expressions like happiness, sadness, surprise, anger and neutral and mapping them to control the robot. This work can further be extended to real world applications like intelligent wheel chairs, human computer interaction and security systems.

References

[1] D. J. S. Ju and E. Y. Kim, “Intelligent wheelchair (IW) interface using face and mouth recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 551-564, Jun. 1999.

[2] P. M. Faria, R. A. M. Braga, E. Valgode, and L. P. Reis, “Interface framework to drive an intelligent wheelchair using facial expressions,” in *proc. Third IEEE International Conference on Automatic Face and Gesture Recognition (FG ’98)*, Nara, Japan, pp. 124-129, April 1998.

[3] B.-K. Shim, K.-W. Kang, W.-S. Lee, J.-B. Won, and S.-H. Han, “An intelligent control of mobile robot based on voice command,” in *proc. International Conference on Control, Automation and Systems 2010*, South Korea, pp. 2107, October 2010.

[4] E. Hjelmas and B. K. Low, “Face detection: A survey,” *Computer*

[5] *Vision and Image Understanding (CVIU)*, vol. 83, no. 3, pp. 236–274, Sept. 2001.

[6] M.-H. Yang, D. Kriegman, and N. Ahuja, “Detecting faces in images: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 24, no. 1, pp. 34–58, Jan. 2002.

[7] T. Liu, H. Guo, and Y. Wang, “A new approach for color-based object recognition with fusion of color models,” *Congress on Image and Signal Processing*, vol. 3, pp. 456-460, May 2008.

[8] T. Lalanne and C. Lempereur, “Color recognition with a camera: a supervised algorithm for classification,” *IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 198-204, Apr. 1998.

[9] Y. L. Tian, T. Kanade, and J. F. Cohn, “Recognizing action units for facial expression analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 97–115, Feb. 2001.

[10] M. Pantic and L. J. M. Rothkrantz, “Automatic analysis of facial expressions: the state of the art,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 12, pp. 1424–1445, Dec. 2000.

[11] B. Fasel and J. Luettin, *Automatic Facial Expression Analysis: A Survey*, Pattern Recognition, 2003.

[12] M. Turk and A. Pentland, “Eigenfaces for recognition,” *J. Cogn. Neurosci.*, vol. 3, pp. 71-86, 1991.

[13] H. Li and R. Forchheimer, “Location of face using color cues,” in *Proc. Picture Coding Symp.*, Lausanne, Switzerland, March 1993.

[14] K. Sobottka and I. Pitas, “Face localization and facial feature extraction based on shape and color information,” in *Proc. IEEE Int. Conf. Image Processing*, September 1996, vol. III, pp. 483–486.

[15] D. Saxe and R. Foulds, “Toward robust skin identification in video images,” in *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, Killington, VT, October 1996, pp. 379–384.

[16] R. Kjeldsen and J. Kender, “Finding skin in color images,” in *Proc. Int. Conf. Automatic Face and Gesture Recognition*, Vermont, October 1996, pp. 312–317.

[17] D. Chai and K. N. Ngan, “Automatic face location for videophone images,” in *Proc. IEEE TENCON’96*, Perth, Australia, November 1996, vol. 1, pp. 137–140.

[18] Y. J. Zhang, Y. R. Yao, and Y. He, “Automatic face segmentation using color cues for coding typical videophone scenes,” in *Proc. SPIE Visual Commun and Image Processing*, San Jose, CA, February 1997, vol. 3024, pp. 468–479.

[19] D. Chai and K. N. Ngan, “Face segmentation using skin-color map in videophone applications,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 551-564, Jun. 1999.

[20] D. Chai and K. N. Ngan, “Locating facial region of a head-and-shoulders color image,” in *proc. Third IEEE International Conference on Automatic Face and Gesture Recognition* (FG ’98), Nara, Japan, pp. 124-129, Apr. 1998.

[21] R.-L. H. Abdel-Mottaleb and A. K. M. Jain, “Face detection in color images,” *IEEE transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696-706, May 2002.

Rohin Mittal was born in Ghaziabad, India on July 02, 1990. He received his Bachelors of Technology degree (B.Tech.) in Electronics and Communication Engineering from Motilal Nehru National Institute of Technology, Allahabad, India in 2011. Currently, he is working as a R&D Engineer in Apache Design Solutions Pvt. Ltd., Bengaluru, India. He can be contacted at rohinmittal2007@gmail.com.

Prateek Srivastava was born in Pratapgarh, India on April 30, 1989. He received his Bachelors of Technology degree (B.Tech.) in Electronics and Communication Engineering from Motilal Nehru National Institute of Technology, Allahabad, India in 2011. Currently, he is working as an Executive Trainee in NTPC Ltd., Singrauli, India. He can be contacted at prateek.annie@gmail.com

Aplha George was born in Kerala, India on April 11, 1989. He received his Bachelors of Technology degree (B.Tech.) in Electronics and Communication Engineering from Motilal Nehru National Institute of Technology, Allahabad, India in 2011. Currently, he is working as a Software Engineer in Samsung India Software RandD Center, Noida, India. He can be contacted at alphageee@gmail.com.

Asim Mukherjee was born in Kolkata, India on January 06, 1968. He received his Master’s degree in Electrical Engineering with specialization in Communication Systems from IIT Kanpur and Bachelor’s degree in Electrical Engineering from NIT, Hamirpur in 2004 and 1992 respectively. Currently, he is working as an Assistant Professor in Electronics and Communication Engineering Department at Motilal Nehru National Institute of Technology, Allahabad, India. He can be contacted at asimmkj@mnnit.ac.in.