Real Time Facial Expression Recognition for Nonverbal Communication

Md. Sazzad Hossain[1] and Mohammad Abu Yousuf[2]
[1]Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Bangladesh
[2]Institute of Information Technology, Jahangirnagar University, Bangladesh

Abstract: *This paper represents a system which can understand and react appropriately to human facial expression for nonverbal communications. The considerable events of this system are detection of human emotions, eye blinking, head nodding and shaking. The key step in the system is to appropriately recognize a human face with acceptable labels. This system uses currently developed OpenCV Haar Feature-based Cascade Classifier for face detection because it can detect faces to any angle. Our system can recognize emotion which is divided into several phases: segmentation of facial regions, extraction of facial features and classification of features into emotions. The first phase of processing is to identify facial regions from real time video. The second phase of processing identifies features which can be used as classifiers to recognize facial expressions. Finally, an artificial neural network is used in order to classify the identified features into five basic emotions. It can also detect eye blinking accurately. It works for the active scene where the eye moves freely and the head and the camera moves independently in all directions of the face. Finally, this system can identify the natural head nodding and shaking that can be recognized in real-time using optical flow motion tracking and find the direction of head during the head movement for nonverbal communication.*

Keywords: *Haar-cascade classifier, facial expression, artificial neural network, template matching, lucas-kanade optical flow.*

Received April 4, 2015; accepted November 29, 2015

1. Introduction

Human communication has two main aspects such as verbal (auditory) and non-verbal (visual). Facial expression, body movement and physiological reactions are the basic units of non-verbal communication. By using mathematical algorithm a machine can describe the interpretation of human facial characteristics. Human Nonverbal Communication Computing aims to investigate how people utilize nonverbal aspects of their communication to coordinate their activities and social relationships. Nonverbal communication is a fundamental yet challenging research topic. Motion analysis methods have been widely investigated and employed for Nonverbal Communication Computing. It includes face tracking, expression recognition, body reconstruction and eye tracking, and group activity analysis.

Face plays significant role in social communication. This is a 'window' to human personality, emotions and thoughts. According to the psychological research conducted by Mehrabian [18], nonverbal part is the most informative channel in social communication. Verbal part contributes about 7% of the message, vocal 34% and facial expression about 55%. Due to that, face is a subject of study in many areas of science such as psychology, behavioural science, medicine and lastly computer science.

In the field of computer science much effort is put to explore the ways of automation the process of real time face detection for attention detection as nonverbal communication. The main issue is to provide appropriate face representation, which remains robust with respect to diversity of facial appearances for communication.

The aim of this paper is to focus on a key element for improving human-computer interactions and to answer the question of whether computers can recognize and understand humans emotions, eye blinking, head nodding or shaking and based on that information take appropriate actions for nonverbal communication in real time environment.

Organization of paper is such that Section two provides lliterature review of current research in relation to nonverbal communication such as face feature extraction, feature representation, facial expression recognition, eye blinking and head nodding. Section three describes the details proposed methodology of facial expression recognition, eye blinking, head nodding which are related to nonverbal communication system. Section four introduces experimental results of nonverbal communication. Section five concludes the paper with future aspect related with nonverbal communication. Finally comprises of references.

2. Literature Review

There are several items requires for nonverbal communication. Facial expression is one of the most important items for such communication. Automatic face detection and recognition video are two challenging problems in real time environment. Fast image processing would be needed [3] for dealing with real time capturing from a camera. The capability to automatically recognize the state of emotion in a human being could be extremely useful in a wide range of area. An example could be an interacting video-game or an active safety device for a vehicle driver. The last application could be really useful to alert a driver about the onset of an emotional state that might endanger his safety or the safety of others, e.g., when his degrees of attention decay below a given threshold [22]. The six basic emotions anger, happiness, surprise, sadness, fear and disgust are introduced by Ekman and Friesen [8].

Facial Emotion detection can be done using various algorithms such as Principle Component Analysis (PCA), Artificial Neural Network (ANN), Support Vector Machine, Gabor Features and Genetic Algorithm [28], Mean Shift Algorithm [37] etc., In this research, we have implemented Fast Artificial Neural Network for classification of emotions.

Facial expression recognized by Shan *et al.* [25] based on statistical local features is Local Binary Patterns (LBP). This used Boosted-LBP to extract the most discriminate LBP features and support vector machine classifiers for achieving best recognition performance. Zhang [36] works within an architecture which is based on two layer perceptron and applied Gabor wavelet coefficients as a feature. For detecting smile, Ito *et al*. [14], firstly used the skin colour detection for the face area detection. Then detect the feature points like two eyes nose and mouth and on the basis of lip length, lip angle and mean intensity of cheeks area measure the smile and achieve 80% accuracy. Shan [26] used the intensity differences between pixels in the gray-scale face images as features and provides 85% accuracy by examining 20 pairs of pixels and 88% accuracy with 100 pairs of pixels on GENKI database. Wang *et al.* [34] proposed real time facial expression recognition using AdaBoost technique in which they detected face using Haar features and then identify six basic expressions from the features extracted from the detected face area. In addition, Wang *et al.* [33] proposed object identification and representation in complex traffic scene based on the colour features and objects of interest are represented by using a Minimum Bound Region (MBR) with a reference coordinate. Finally, they determine the efficient parameters and demonstrate that single and multiple known objects in complex scenes can be identified by their proposed method. The facial expressions using a Hierarchical Radial Basis Function Network by Applying PCA to local regions such as eyes and lips has successfully classified by Lin [16]. However, the accuracy of this method relies on the accurate detection of features like lips and eyes, which in itself is a difficult task. PCA has also been used in [13] to classify facial expressions from static images by extracting local features called local binary pattern [1].

Eye blinking is one of the means of non-verbal communication to attract human attention. There have been so many research dedicated to the implementation of eye tracking systems which were designed to detect eye movements. However, most of them were expensive to use and not practical. The human eye tracking, blinking and eye movement can be recorded with relatively high reliability by unremarkable techniques.

A thorough survey on work related to eye and blink detection methods is presented by Grauman *et al*. [12], as well as Magee *et al*. [17]. Since the implementation of the eye blink detection system by Grauman *et al*. [12], a number of considerable contributions and advancements have been made in the Human Computer Interaction (HCI) field. A blink detection algorithm for human computer interaction also has been proposed by Morris [21], in which the initialization step requires motion detection to locate the eyelids. There still have not been many blink detection related systems designed to work with in-expensive USB webcams [10, 11]. There are some other feature detection systems that use more expensive and less portable alternatives, such as digital and inferred cameras for video input [2, 19, 24, 35]. A side from the portability concerns, these systems is also typically unable to achieve the desirable higher frame rates of approximately 30 frames per seconds that are common with USB cameras.

A very large percentage of our communication is nonverbal, which includes all expressive signals, signs and cues that are used to send and receive messages apart from manual sign language and speech. Nonverbal gestures perform different functions [5]. Head nods and shakes can be used as a gesture to fulfil a semantic function (e.g., nod head instead of saying yes), to communicate emotions (e.g., nodding enthusiastically with approval) and as conversational feedback (e.g., to keep the conversation moving). Table 1 shows some of the semantic functions and emotions associated with head nods and shakes. A system that could detect head nods and head shakes would be an important component in an interface that is expected to interact naturally with people.

Table 1. Affective meanings of head nods/shakes.

Head Nods	**Head Shakes**
Approval	Disapproval
Understanding	Disbelief
Agreement	Negation

Head nod, which is a vertical up-and-down movement of the head rhythmically raised and lowered is an positive cue, widely used throughout the world to show agreement, understanding and approval [7, 9, 20]. Head shake is rotation of the head horizontally from side-to-side and is nearly a universal sign of disapproval, disbelief, and negation [7, 9, 20]. Although, head nod goes mostly with positive intent/emotions and head shake with negative intent/emotions there might be certain exceptions. For example, head nod might occur with a feeling of temper too. Head nods are also used as a conversational response, so a person, even if he does not agree, may nod his head while he is listening. Detection of head nods and shakes is difficult in Real time, as the head movements during a nod or shake are fast, small and jerky, causing many video-based face trackers to fail. A Hidden Markov Model (HMM) [23] based classifier is used to detect the occurrence of head nods and head shakes.

Kawato and Ohya [15] have described a system to detect head nods and head shakes in real time by using a "circle frequency filter" which can directly detecting and tracking the "between-eyes". It is a discrete Fourier transform of points lying on a circle, together with skin colour information and templates. Head shakes and head nods are detected based on pre-defined rules applied to the positions of "between-eyes" in consecutive frames. In this research paper, the system uses motion analysis technique to detect nodding and shaking.

In this paper, at first, we develop a system that can detect facial emotions in a real time video stream where an ANN is used in order to classify the identified features into five basic emotions accurately out of six emotions. Then implement an eye blinking system accurately in real time environment using template matching technique. Finally, develop a system which can recognize head nodding and shaking in real-time using optical flow motion tracking and find the direction of head during the head movement for nonverbal communication. In this research, we use Haar-like classifier algorithm supplied from OpenCV library.

3. Proposed Methodology

3.1. Emotion Detection

Firstly, we present the details of our learning framework for automatic facial emotion detection in real time. We divide the task into several steps, i.e., face detection and location, facial expression, feature extraction and facial emotion classification. The framework of this system is demonstrates in Figure 1.

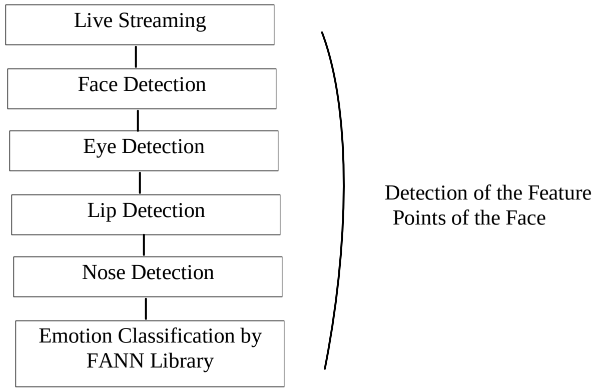


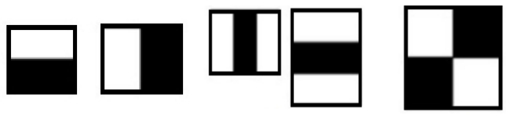
Figure 1. Framework of emotion detection system.

3.1.1. Live Streaming

Live streaming is the fundamental step of image acquisition in real time environment, where image frames are received using streaming media like web camera.

3.1.2. Face Detection

We use Haar-classifier to detect face in an image owing to its high detection accuracy and real time performance [1]. Haar-like features are rectangular features in a digital image, which are used in object detection. Generating a strong classifier out of the weak classifiers we need to combines of such features in a cascading method. Figure 2 shows some Haar-like features. The value of a rectangular Haar-like feature is the difference between the sum of the pixels within the white rectangular regions and that within the black rectangular regions.



a) Two-rectangle. b) Three-rectangle. c) Fourrectangle.

Figure 2. Different types of Haar-like features.

For fast computation of rectangular Haar-like features, integral images are computed (Figure 3). It is the intermediate representation of the original image was firstly used by Viola and John [31, 32] in image processing and by using this integral image rectangle feature is calculating very fast. Each element of the integral image contains the sum of all pixels located on the up-left region of the image. Using this, we compute the sum of rectangular areas in the image, at any position or scale. The integral image I(x, y) is computed efficiently in a single pass over the image using Equation (1)

$$I(x, y) = i(x, y) + I(x-1, y) + I(x, y-1) - I(x-1, y-1) \quad \ldots \quad (1)$$

In Equation (1), $I(x, y) = \sum_{\substack{x' \le x \\ y' \le y}} i(x', y')$ and i(x, y) is the intensity of the image point (x, y).

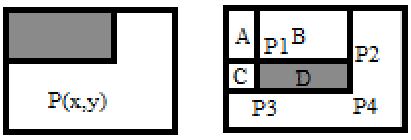


Figure 3. Integral image representations.

Once integral image is calculated, every feature can be obtained by using just 3 additions, irrespective of the scale and location. The optimal set of features and their corresponding thresholds for classification is obtained from AdaBoost algorithm. Adaboost in Haar cascade method is for training by supervised learning to classifying positive and negative sample, to classifying two decisions and the Figure 4 shows how the Adaboost learns weak classifier by cascading weak classifier whose output is just random gauss and make strong classifier therefore reduced the computation time [30]. Apply more strict rules for adding less images and more different Haar feature like Figure 4 at each stage of the cascade, which cannot be indicated as a face image is rejected. When an image is given to a cascade of classifiers and if it passes all the classifiers then this will represent the presence of face with high probability.

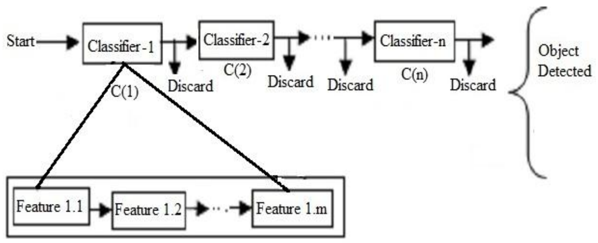


Figure 4. Cascades of feature classifiers.

Opencv haar cascade classifier is applied in our development for not only face detection but also other feature like eye, nose, and mouth detection.

3.1.3. Eye and Brow feature detection

When we detect the face rectangle, we need to identify the left and right eye separately. For calculating the position of left and right eye, the Equations (2) and (3) are used.

$$\text{LeftEye}(\text{FaceRect.x} + \text{FactRect.width}/2.0,\ \text{FactRect.y} + \text{FactRect.width}/2.0) \quad (2)$$

$$\text{RightEye}(\text{FaceRect.x}, \text{FactRect.y}, \text{FactRect.width}/2.0,\ \text{FactRect.width}/2.0) \quad (3)$$

The brow region of interest is calculated by using Equations (4) and (5).

$$\text{RightBrowROI}(\text{FaceRect.x} + \text{FactRect.width}/2.0,\ \text{FactRect.y} + \text{FactRect.height}/6.0,\ \text{FaceRect.width}/2.0 - \text{w/scale}, \text{FaceRect.height}/5.0) \quad (4)$$

$$\text{LeftBrowROI}(\text{FaceRect.x} + \text{w/scale}, \text{FactRect.y} + \text{FactRect.height}/6.0,\ \text{FaceRect.width}/2.0 - \text{w/scale}, \text{FaceRect.height}/5.0) \quad (5)$$

3.1.4. Mouth feature detection

We need to find Region of Interest (ROI) of mouth using Equation (6).

$$\text{MouthROI}(\text{FaceRect.x}, \text{FactRect.y} + \text{FactRect.height}/2.0, \text{FaceRect.width}, \text{FaceRect.height}/2.0) \quad (6)$$

After calculating the mouth contour, we find the lip's height, width, top and bottom.

3.1.5. Nose Feature Detection

The region of interest of nose is calculated by using Equation (7).

$$\text{NoseROI}(\text{FaceRect.x} + \text{FaceRect.width}/4.0, \text{FaceRect.y} + \text{FactRect.height}/2.0, 2.0 * \text{FaceRect.width}/4.0, \text{FaceRect.height}/2.0) \quad (7)$$

Finally, center point of the nose is calculated by using Equation (8).

$$\text{NoseCenter}(\text{Nose.x} + \text{Nose.width}/2.0, \text{Nose.y} + \text{Nose.height}/2.0) \quad (8)$$

3.1.6. Emotion Detection and Classification

Once all the feature detection is done, we calculate the following parameters.

- Width and height of mouth.
- Distance between top and bottom of the left eye.
- Distance between top and bottom of the right eye.
- Distance between the center of left brow's left and center of the nose.
- Distance between the center of left brow's center and center of the nose.
- Distance between the center of left brow and center of the nose.
- Distance between the center of right brow's left and center of the nose.
- Distance between the center of right brow's center and center of the nose.
- Distance between the center of right brow and center of the nose.

Algorithm 1: Emotion detection

Step 1. Start Live streaming
Step 2. Retrieve grabbed video frame into F
Step 3. Copy F into FCopy
Step4. Convert FCopy to gray image and assign the result to GImage
Step 5. Detect rectangular face from GImage using Haar cascades and assign the result to FRects
Step 6. Gather all features from the rectangular face and assign the result to FRArry
Step 7. If found all of the features successfully then calculate the difference between natural and emotional features of the face and assign the result to FRIn
Step 8. Send FRIn to the neural network and collect the output of the network.
Step 9. Analyze the output and write appropriate message
Step 10. Repeat step 3 to 9 until streaming end

For artificial neural network implementation, an open source library called Fast Artificial Neural Network (FANN) is used. Both fully connected and sparsely connected multilayer artificial networks can be implemented using this library. The FANN library is fast on systems with no floating point processor. It is super fast on standard workstation machines and it is very versatile to allow for new functionality to be implemented and easy to use. During the experiment, we create an ANN by applying FANN library and finally we are ready to feed our input to the neural network. For best performance we create our neural network which is shown in Table 2.

Table 2. Importan setting of neural network for emotion classification.

FANN parameter for emotion
num_layers=3
learning_rate=0.700000
connection_rate=1.000000
network_type=0
learning_momentum=0.000000
training_algorithm=2
train_error_function=1
train_stop_function=0
cascade_output_change_fraction=0.010000
quickprop_decay=-0.000100
quickprop_mu=1.750000
rprop_increase_factor=1.200000
rprop_decrease_factor=0.500000
rprop_delta_min=0.000000
rprop_delta_max=50.000000
rprop_delta_zero=0.100000
cascade_output_stagnation_epochs=12
cascade_candidate_change_fraction=0.010000
cascade_candidate_stagnation_epochs=12
cascade_max_out_epochs=150
cascade_min_out_epochs=50
cascade_max_cand_epochs=150
cascade_min_cand_epochs=50
cascade_num_candidate_groups=2
bit_fail_limit=3.49999994039535522461e-01
cascade_candidate_limit=1.00000000000000000000e+03
cascade_weight_multiplier=4.00000005960464477539e-01
cascade_activation_functions_count=10
cascade_activation_functions=3 5 7 8 10 11 14 15 16 17
cascade_activation_steepnesses_count=4
cascade_activation_steepnesses=2.50000000000000000000e-01 5.00000000000000000000e-01 7.50000000000000000000e-01 1.00000000000000000000e+00
layer_sizes=11 5 5

3.2. Eye Blinking Detection

In this research paper, we present the details of our system for automatic eye blinking detection in real time environment. We divide the task into three steps, i.e., face detection & location of face region, eye detection & eye tracking and finally template matching for eye blinking detection. The flowchart of this system is demonstrated in Figure 5. Face detection procedure has already been described in sub section 3.1.2.

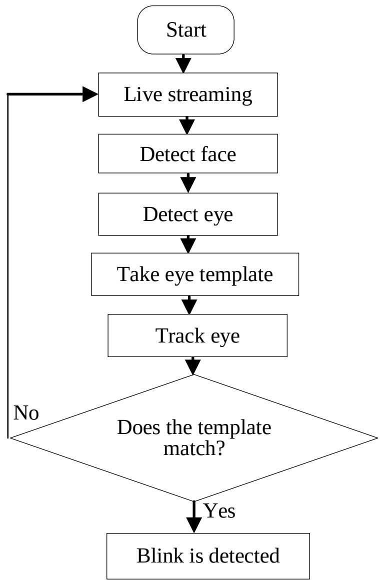


Figure 5. Flow diagram of eye blinking system.

3.2.1. Eye Detection and Tracking

This paper uses fast eye detection and tracking system that takes the input from webcam. In order to achieve high speed performance, the system needs to perform the face and eye detection only once at the program start-up. After the eye is successfully detected, an eye template is created at runtime and will be used for tracking the eye using template matching method.

During the eye tracking process each frame is converted to grayscale to reduce processing time. If the eye is successfully located, the program function will return an eye template and its region of interest. The OpenCV eye tracking method takes the current frame, eye template, and eye ROI as the input. The eye template is used for locating the eye in the given frame with template matching method. If the eye successfully located, the ROI will be updated for the new location of the eye. The template matching is performed in a search window to increase the speed

3.2.2. Template Matching and Blink Detection

Template matching is a technique for finding areas of an image that match (which are similar) to a template image (patch). During the first template matching, we need to load offline template for close and open eyes and then we create online template and match with the previous template. We wait a little time for matching with template and after that we take a new one and continue this process. Actually, we check the transition time between open and close eye. If this transition time passes a threshold time, we identify an eye blink.

The summary of template matching technique is as follows:

- Loads an input image from video and an image patch (template).

- Perform a template matching procedure by using the OpenCV function matchTemplate with any of the method through (a) to (f). The user can choose the method by entering its selection in the Trackbar.
- Normalize the output of the matching procedure
- Localize the location with higher matching probability.
- Continuously check the frame and wait for threshold time to detect eye blink.

a. method=CV_TM_SQDIFF

$$R(x,y) = \sum_{x',y'} (T(x',y') - I(x+x',y+y'))^2 \tag{9}$$

method=CV_TM_SQDIFF_NORMED

$$R(x,y) = \frac{\sum_{x',y'} (T(x',y') - I(x+x',y+y'))^2}{\sqrt{\sum_{x',y'} T(x',y')^2 \cdot \sum_{x',y'} I(x+x',y+y')^2}} \tag{10}$$

b. method=CV_TM_CCORR

$$R(x,y) = \sum_{x',y'} (T(x',y') \cdot I(x+x',y+y')) \tag{11}$$

c. method=CV_TM_CCORR_NORMED

$$R(x,y) = \frac{\sum_{x',y'} (T(x',y') \cdot I(x+x',y+y'))}{\sqrt{\sum_{x',y'} T(x',y')^2 \cdot \sum_{x',y'} I(x+x',y+y')^2}} \tag{12}$$

d. method=CV_TM_CCOEFF

$$R(x,y) = \sum_{x',y'} (T'(x',y') \cdot I'(x+x',y+y')) \tag{13}$$

where

$$T'(x',y') = T(x',y') - 1/(w \cdot h) \cdot \sum_{x'',y''} T(x'',y'')$$

$$I'(x+x',y+y') = I(x+x',y+y') - 1/(w \cdot h) \cdot \sum_{x'',y''} I(x+x'',y+y'')$$

e. method=CV_TM_CCOEFF_NORMED

$$R(x,y) = \frac{\sum_{x',y'} (T'(x',y') \cdot I'(x+x',y+y'))}{\sqrt{\sum_{x',y'} T'(x',y')^2 \cdot \sum_{x',y'} I'(x+x',y+y')^2}} \tag{14}$$

Algorithm 2: Eye blinking detection

Step 1. Start Live streaming
Step 2. Retrieve grabbed video frame at 320 X 240 resolutions into F
Step 3. Detect rectangular face from F using OpenCV Haar cascades classifier and assign the result to FRects
Step 4. Find the eye region from rectangular face F with same classifier and assign the result to EImg.
Step 5. Find eye blink by identifying the transition between open and close eye by template matching algorithm using online template EImg and two offline open and close eye template.
Step 6. Repeat step 3 to 5 until streaming end

3.3. Head Nodding and Shaking

Head nods and head shakes are non-verbal gestures used often to communicate intent, emotion and to perform conversational functions. We describe a computer vision-based system that detects head nods and head shakes in real time and can act as a useful and basic interface to a machine. The directions of head movements, determined using the optical flow Lucas–Kanade method to detect when a head nod/shake occurs.

The most important part of this system is to find good features which must have at least texture and corner, compute convex hull, find centroid, apply optical flow methods etc. Face detection and retrieving frame from video is the same process which already explained earlier. Detection of head nodding and shaking is based on face detection. The block diagram of the detection of head nodding and shaking is depicted in Figure 6.

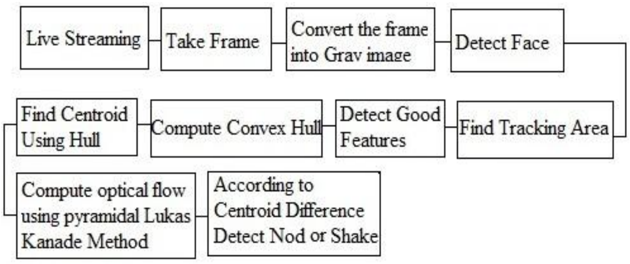


Figure 6. Block diagram of head nodding and shaking.

3.3.1. Track Face and Find Feature

After successfully detection of face, we need to identify the tracking area. Allocate the area which is called ROI. Now, we need to find good features. Good features are identified by using the following OpenCV function goodFeaturesToTrack.

Void good Features To Track (InputArray image, OutputArray corners, int maxCorners, double qualityLevel double minDistance, InputArray mask=noArray(), int blockSize=3, bool useHarrisDetector=false, double k=0.04)

The OpenCV function finds the most prominent corners in the specified image region, as described in [27].

- Function calculates the corner quality measure at Map(x,y) are rejected.every source image pixel using the cornerMinEigenVal() or cornerHarris().
- Function performs a non-maximum suppression (the local maximums in *3 x 3* neighborhood are retained).
- The corners with the minimal eigenvalue less than qualityLevel.$\max_{x,y}$ qualityMeasure.
- The remaining corners are sorted by the quality measure in the descending order.

- Function throws away each corner for which there is a stronger corner at a distance less than maxDistance.

The function can be used to initialize a point-based tracker of an object. Good features computed on a different coordinate system and later they are shifted to their original location.

3.3.2. Compute Convex Hull

X is a bounded subset of the plane, the convex hull may be visualized as the shape formed by a rubber band stretched around X like the Figure 7.

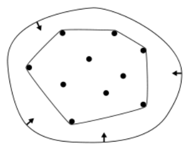


Figure 7. Idea of Convex Hull.

The following OpenCV function is used to find the Convex Hull.
void convexHull(InputArray points, OutputArray hull, bool clockwise=false, bool returnPoints=true)

3.3.2.1. Calculate Centroid

We detect the polygon area by using the Convex Hull. Multiply the polygon area six times and divide it with the all good feature's X and Y coordinate value to find the centroid.

3.3.2.2. Optical flow motion

Optical flow is the pattern of apparent motion of image objects between two consecutive frames caused by the movement of object or camera. It is 2D vector field where each vector is a displacement vector showing the movement of points from first frame to second. Consider the image shown in Figure 8.

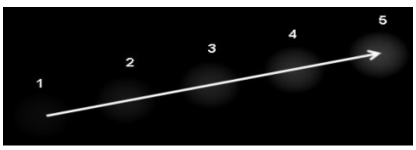


Figure 8. Optical flow method.

It shows a ball moving in 5 consecutive frames. The arrow shows its displacement vector.
Optical flow works on several assumptions:

- The pixel intensities of an object do not change between consecutive frames.
- Neighbouring pixels have similar motion.

Consider a pixel $I(x,y,t)$ in first frame. It moves by distance (dx, dy) in next frame taken after dt time. So, since those pixels are the same and intensity does not change, we can derive Equation (15).

$$I(x,y,t) = I(x+dx, y+dy, t+dt) \tag{15}$$

Then, take Taylor series approximation of right-hand side, remove common terms and divide by dt to get the Equation (16).

$$f_x u + f_y v + f_t = 0 \tag{16}$$

where,

$$f_x = \frac{\delta f}{\delta x}; f_y = \frac{\delta f}{\delta y}$$
$$u = \frac{dx}{dt}; v = \frac{dy}{dt}$$

The Equation (16) is called Optical Flow equation. Here, f_x and f_y, they are image gradients along the x and y direction. Similarly f_t is the gradient along time. But (u,v) is unknown. We cannot solve this equation with two unknown variables. So several methods are provided to solve this problem and one of them is Lucas-Kanade.

- Lucas-Kanade Method

We have seen an assumption before, that all the neighbouring pixels will have similar motion. Lucas-Kanade method takes a 3x3 patch around the point. So all the 9 points have the same motion. We can find (f_x, f_y, f_t) for these 9 points. So now our problem becomes solving 9 equations for 9 points with two unknown variables which are over-determined. A better solution is obtained with least square fit method. Equation (17) is the final solution which is two equations with two unknown problems and solve it to get the solution.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \Sigma_i f_{x_i}^{\ 2} & \Sigma_i f_{x_i} f_{y_i} \\ \Sigma_i f_{x_i} f_{y_i} & \Sigma_i f_{y_i}^{\ 2} \end{bmatrix}^{-1} \begin{bmatrix} -\Sigma_i f_{x_i} f_{t_i} \\ -\Sigma_i f_{y_i} f_{t_i} \end{bmatrix} \tag{17}$$

We give some points to track and receive the optical flow vectors of those points.

OpenCV provides all these in a single function calcOpticalFlowPyrLK().

For the function calcOpticalFlowPyrLK() we pass the previous frame, previous points and next frame. It returns next points along with some status numbers which has a value of 1 if next point is found, else zero.

The overall nonverbal communication system is shown in Figure 9.

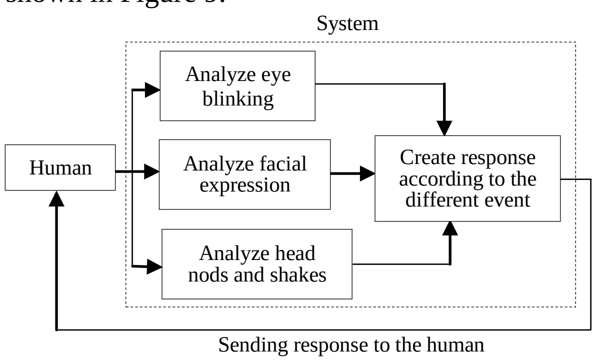


Figure 9. Nonverbal communication system.

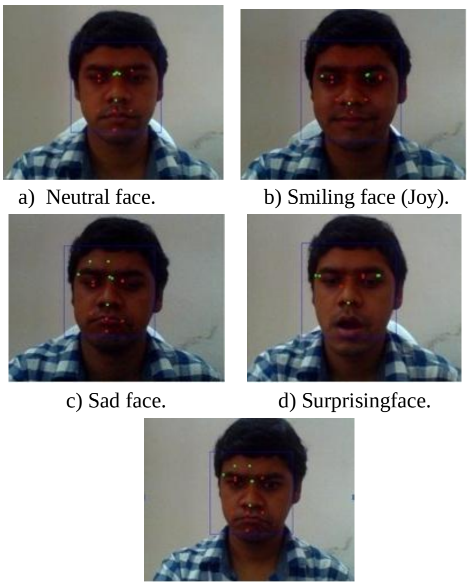
Algorithm 3: Head nodding and shaking detection

Step 1. Start Live streaming
Step 2. Initialize the system by taking the frame input from camera
Step 3. Detect rectangular face from using Voila and Jones method
Step 4. Define tracking area and find the best region of interest.
Step 5. Identify good features with OpenCV function goodFeaturesToTrack.
Step 6. Find the Convex Hull after finding the features by using OpenCV function convexHull.
Step 7. Calculate Centroid of the tracking area polygon.
Step 8. Apply Lucas-Kanade sparse optical flow OpenCV Method calcOpticalFlowPyrLK to find head nod and shake.
Step 9. Repeat step 3 to 8 until streaming end

4. Experimental Result

4.1. Result of Emotion Detection

For the experiment we need 10 inputs and we have 6 outputs to classify the emotions. The learning rate of neural network is 0.7. The output of the FANN for neutral face and facial expression classification accuracy is shown in Tables 3 and 4 and the resultant images are shown in Figure 10.



a) Neutral face. b) Smiling face (Joy).

c) Sad face. d) Surprisingface.

e) Angry face.

Figure 10. Emotion detection using proposed system.

Table 3. Output of different expression.

Feeling	Output value
Anger	-0.0171519, 1, -0.00325582, 0.0172076
Surprise	0.323947, 1, 0.983592, -0.993025
Sadness	-0.0274133, 0.00141051, 0.983483, -0.0072828
Joy	-0.0171183, 1, -0.00329069, 0.0172078
Neutral	2.14583e-005, 0.000135782, 2.75418e-005, 0.981064

Table 4. Facial expression classification.

Method	Classification Accuracy %	Input type
Rapid Facial Expression Classification Using Artificial Neural Networks [4]	73.3%	Static image
Facial Expression Classification Using Multi Artificial Neural Network [29]	83%	Static image
CANNY PCA Artificial Neural Network [30]	85.7%	Static image
Proposed System (Facial Expression Classification with FANN)	88%	Video

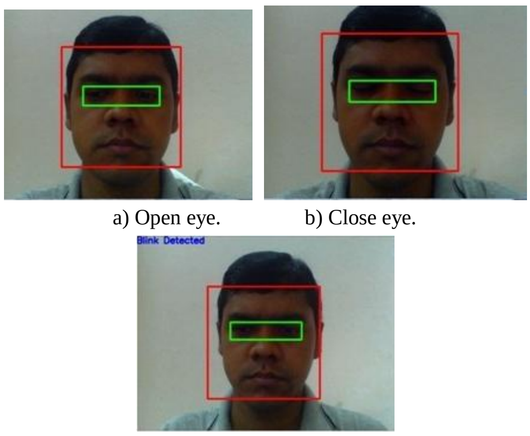
Table 5. Comparison among Classification Rate of Methods

Feeling	Correct Classifications	Classification Accuracy %
Anger	8/10	80
Surprise	9/10	90
Sadness	8/10	80
Joy	9/10	90
Neutral	10/10	100

The average facial expression classification of our proposed method FANN is 88%. The Table 5 shows comparison table with other techniques. In contrast, our system is tested in real time video input and performs well than others.

4.2. Result of Eye Blinking Detection

The system was primarily developed and tested on a standard machine. Video was captured with an integrated webcam. To evaluate our system, we counted the successful blinking detection rate using the Equation (19). The successful blink detection rate of our system among 100 blinking is 88.33%. The output images for eye blinking detection of our proposed system are shown in Figure 11.



a) Open eye. b) Close eye.

c) Eye Blink detected.

Figure 11. Detection of Eye Blinking using proposed system.

We compute the overall detection accuracy and the detection accuracy of the eye blink detection by using Equations (18) and (19), respectively. Where True Positive (TP) is the number of frames that are correctly detected eye blinks; False Negative (FN) is the number of frames that show eye blinks but the program is not detected; False Positive (FP) is the number of frames that are reported as eye blinks but they are not; and True Negative (TN) is the number of frames that are correctly reported as no blinks.

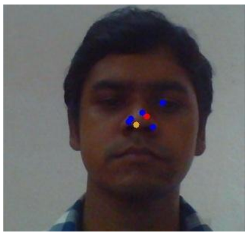
$$Overall = \frac{TP + TN}{TP + FP + FN + TN} \times 100\% \tag{18}$$

$$Detection = \frac{TP}{TP + FN} \times 100\% \tag{19}$$

According to the Equations (18) and (19), an overall accuracy rate of our system is 78% and blink detection accuracy rate is 88.33% which gives better performance than Michael Chauand Margrit Betke's blink detection system [6].

4.3. Result of Head Nodding or Shaking Detection

The system is tested on real time data from ten users gathered in the presence of varied lighting and varied facial expressions. A total of 100 samples were collected with 50 head nods and 50 head shakes. The system achieves a real time recognition accuracy of 79.00% on the sample dataset which is shown in Table 6 and the output image is shown in Figure 12.



a) Center.

b) Turn right.

c) Turn left.

d) Looking down.

e) Looking upward.

Figure 12. Detection of Head Nod and Shake by proposed system.

Table 6. Recognition result for the sample data.

Head Nods/Shakes	Recognized	Misses	Recognition Accuracy %
Nods	40	10	80
Shakes	39	11	78
Overall recognition rate: 79%			

5. Conclusions

Human facial expression recognition is useful to evaluate the mood or emotional state of a subject under observation. For the purpose of non-verbal communication, this research paper briefly overviews automatic emotion recognition in real time using a new approach in which we identify different emotions based on FANN for facial expression classification. However, the uses of ANN for classifying and the number of hidden nodes are identified by experience. It required the high calculating cost for learning process.

In this research paper, a new algorithm is also introduced to detect eye blinking. The proposed method easily detects the blinks and the success rate is high because recent developed OpenCv Haar-classifier method can easily detect face and eye accurately. Our system is designed for blinking detection in the real time environment. The implementation of this technique runs at 25-30 frames per second. The proposed method shows promising results under good lighting conditions. In comparison with other methods, template matching is an efficient algorithm for locating features and it does not require a significant amount of further image processing. However, there are still some weaknesses in these techniques:

- The highest value may not be an eye because it only gives the most similar region compared with the kernel.
- The false detection rate is increased due to variation in skin color or lighting condition changes.
- Traditional calculation of convolution is computationally expensive.

Because of these shortcomings, template matching has not played an important role for object detection for a time so that template matching could be used effectively.

Finally, we present a technique to identify Head nodding and shaking for nonverbal communication. In our research project, we used Lucas-Kanade sparse optical flow computation for Head nods and Head Shakes. Our system can deal with small motions. So it fails when there is large motion. We will implement head nods and head shakes using dense optical flow computation for large motion and finally integrates all these methods and acts as an interface that can use for communication.

References

[1] Ahonen T., Hadid A., and Pietikäinen M., "Face Recognition with Local Binary Patterns," *in proceedings of European Conference on Computer Vision*, Prague, pp. 469-481, 2004.

[2] Bhaskar T., Keat F., Ranganath S., and Venkatesh Y., "Blink Detection and Eye Tracking for Eye Localization," *in proceedings of Conference on Convergent Technologies for Asia-Pacific Region*, Bangalore, pp. 821-824, 2003.

[3] Bouhabba E., Shafie A., and Akmeliawati R., "Support Vector Machine for Face Emotion Detection on Real Time Basis," *in proceedings of 4th International Conference on Mechatronics*, Kuala Lumpur, pp. 1-6, 2011.

[4] Cantelmo N., Rapid Facial Expression Classification Using Artificial Neural Networks, *Northwestern University*, http://steadystone.com/research/ml07/, Last Visited 2007.

[5] Cassell J., *Nudge nudge wink wink Elements of Face-to-Face Conversation for Embodied Conversational Agents*, MIT Press, 2000.

[6] Chau M. and Betke M., "Real Time Eye Tracking and Blink Detection with USB Cameras," *Boston University Computer Science Technical Report*, USA, no. 2005-12, 2005.

[7] Darwin C., *The Expression of the Emotions in Man and Animals*, Oxford University Press, 1998.

[8] Ekman P. and Friesen W., "Constants across Cultures in the Face and Emotion," *Personality and Social Psychology*, vol. 17, no. 2, pp. 124-129, 1971.

[9] Givens D., *Dictionary of Gestures, Signs and body Language Cues*, Center for Nonverbal Studies Press, 2002.

[10] Gorodnichy D., "On Importance of Nose for Face Tracking," *in Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, Washington, pp. 181-186, 2002.

[11] Gorodnichy D., "Second Order Change Detection, and its Application to Blink-controlled Perceptual Interfaces," *IASTED Conference on Visualization, Imaging and Image Processing*, Benalmadena, pp. 140-145, 2003.

[12] Grauman K., Betke M., Lombardi J., Gips J., and Bradski G., "Communication Via Eye Blinks and Eye-brow Rises: Video-based Human-Computer Interfaces," *Universal Access in the Information Society*, vol. 2, no. 4, pp. 359-373, 2003.

[13] Happy S., George A., and Routray A., "A Real Time Facial Expression Classification System using Local Binary Patterns," *in Proceedings of 4th International Conference on Intelligent Human Computer Interaction*, Kharagpur, pp. 1-5, 2012.

[14] Ito A., Wang X., Suzuki M., and Makino S., "Smile and Laughter Recognition Using Speech Processing and Face Recognition from Conversation Video," *in Proceedings of International Conference on Cyberworlds*, Singapore, pp. 437-444, 2005.

[15] Kawato S. and Ohya J., "Real-time Detection of Nodding and Head-shaking by Directly Detecting and Tracking the Between-Eyes," *in Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition*, Grenoble, pp. 40-45, 2000.

[16] Lin D., "Facial Expression Classification using PCA and Hierarchical Radial Basis Function Network," *Journal of Information Science and Engineering*, vol. 22, no. 5, pp. 1033-1046, 2006.

[17] Magee J., Scott M., Waber B., and Betke M., "Eyekeys: A Real-Time Vision Interface Based on Gaze Detection from a Low-Grade Video Camera," *IEEE Workshop on Real-Time Vision for Human-Computer Interaction*, Washington, pp. 159-166, 2004.

[18] Mehrabian A., "Communication without Words," *Psychology Today*, vol. 2, pp. 53-56, 1968.

[19] Moriyama T., Kanade T., Cohn J., Xiao J., Ambadar Z., Gao J., and Imanura M., "Automatic Recognition of Eye Blinking in Spontaneously occurring Behaviour," *in Proceedings of 16th International Conference on Pattern Recognition*, Quebec, pp.78-81, 2002.

[20] Morris D., *Bodytalk: The Meaning of Human Gestures*, Crown Publishers, 1994.

[21] Morris T., Zaidi F., and Blenkhorn P., "Blink Detection for Real-time Eye Tracking," *Networking and Computer Applications*, vol. 25, no. 2, pp. 129-143, 2002.

[22] Paschero M., Vescovo G., Benucci L., Rizzi A., Santello M., Fabbri G., and Frattale F., "A Real Time Classifier for Emotion and Stress Recognition in a Vehicle Driver," *in Proceedings of IEEE International Symposium on Industrial Electronics*, Hangzhou, pp. 1690-1695, 2012.

[23] Rabiner L., "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *IEEE*, vol. 77, no. 2, pp. 257-286, 1989.

[24] Rurainsky J. and Eisert P., "Eye Center Localization using Adaptive Templates," *in Proceedings of the CVPR Workshop on Face Processing in Video*, Washington, pp. 67-67, 2004.

[25] Shan C., Gong S., and McOwan P., "Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study," *Image Vision Computing*, vol. 27, no. 6, pp. 803-816, 2009.

[26] Shan C., "Smile Detection by Boosting Pixel Differences," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 431-436, 2012.

[27] Shi J. and Tomasi C., "Good Features to Track," *in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, pp. 593-600, 1994.

[28] Shojaeilangari S., Yun Y., and Khwang T., "Person Independent Facial Expression Analysis using Gabor Features and Genetic Algorithm," *in Proceedings of 8th International Conference on information Communications and Signal Processing*, Singapore, pp.1-5, 2011.

[29] Thai L., Tat p., and Hai T., "Facial Expression Classification Based on Multi Artificial Neural Network," *in Proceeding of Advance Computing and Applications*, Vietnam, pp. 125-133, 2010.

[30] Thai L., Hai T., and Nguyen N., "A Facial Expression Classification System Integrating Canny, Principal Component Analysis and Artificial Neural Network," *Machine Learning and Computing*, vol. 1, no. 4, pp. 388-393, 2011.

[31] Viola P. and Jones M., "Rapid Object Detection using a Boosted Cascade of Simple Features," *in Proceeding of IEEE Conference Computer Vision Pattern Recognition*, Kauai , pp. 511-518, 2001.

[32] Viola P. and Jones M., "Robust Real-Time Face Detection," *Computer Vision*, vol. 57 no. 2, pp. 137-154, 2004.

[33] Wang H., Mohamad D., and Ismail N., "An Efficient Parameters Selection for Object Recognition Based Colour Features in Traffic Image Retrieval," *The International Arab Journal of Information Technology*, vol. 11, no. 3, pp. 308-314, 2014.

[34] Wang Y., Ai H., Wu B., and Huang C., "Real Time Facial Expression Recognition with Adaboost," *in Proceedings of the Pattern Recognition 17th International Pattern Recognition*, Cambridge, pp. 926-929, 2004.

[35] Wei X., Zhu Z., Yin L., and Ji Q., "A Real-time Face Tracking and Animation System," *CVPR Workshop on Face Processing in Video*, Washington, pp. 71-71, 2004.

[36] Zhang Z, "Feature-Based Facial Expression Recognition: Sensitivity Analysis and Experiments with a Multi-Layer Perceptron," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 13, no. 6, pp. 893-911, 1999.

[37] Zhao-Yi P., Zhi-Qiang W., and Yu Z, "Application of Mean Shift Algorithm in Real-time Facial Expression Recognition," *in Proceeding of International Symposium on Computer Network and Multimedia Technology*, Wuhan, pp. 1-4, 2009.

Md. Sazzad Hossain was born in Dhaka, Bangladesh, in 1984. He received the B.Sc. (Engg.) in Computer Science and Engineering from the Mawlana Bhashani Science and Technology University, Tangail, Bangladesh, in 2008, and the M.Sc. (Engg.) in Computer Science and Engineering from the same university in 2014. Now he is a faculty member of Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University. His current research interests include human computer interaction, embedded system, VLSI design, reconfigurable architecture and cryptography/information security.

Mohammad Abu Yousuf received the B.Sc.(Engineering) degree in Computer Science and Engineering from Shahjalal University of Science and Technology, Sylhet, Bangladesh in 1999, the Master of Engineering degree in Biomedical Engineering from Kyung Hee University, South Korea in 2009, and the Ph.D. degree in Science and Engineering from Saitama University, Japan in 2013. In 2003, he joined as a Lecturer in the Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Tangail, Bangladesh. In 2014, he moved to the Jahangirnagar University as an Assistant Professor in the Institute of Information Technology. His research interests include Medical Image Processing, Human-Robot Interaction and Computer Vision.