Emotion Detection from videos using Artificial Neural Network

Burhanuddin Bhopalwala[1], Amit Sachdeva[2], Abhinav Mishra[3], Satish Chandra[4]
Department of Computer Science and IT
Jaypee Institute of Information Technology, Noida, India
burhanuddinbhopalwala.cse@gmail.com[1], amitsachdeva45@gmail.com[2], abhim4757@gmail.com[3], satish.chandra@jiit.ac.in[4]

***Abstract*— In this paper, Emotion Detection from videos using webcam has been addressed. A dynamic approach using a webcam is used to detect the face of the person in a real time environment. In the algorithm, Artificial Neural Network is used for emotion detection in human faces. Denoising and deblurring algorithms are used to improve the quality of the image received from a webcam.Testing of the image using values generated in the training phase and make results on behalf of that. Due to all above methods, efficiency is improved as a final result.his electronic document is a "live" template and already defines the components of your paper [title, text, heads, etc.] in its style sheet.**

***Keywords*— Artificial Neural Network; Eigenvector; Eigenfaces; Harrcascade; Denoising; Deblurring**

I. INTRODUCTION

An Emotion is a mental and physiological state which involves behavior, thoughts and feeling. Firstly Charles Darwin had traced the emotions of the species and believed that emotions to be species-specific rather than culture-specific. Social psychologists have discovered that there is strong evidence for the universal facial expressions of seven emotions – anger, contempt, disgust, fear, joy, sadness, and surprise To detect emotions from face, different image strategies have been used earlier using single image.

There are many face detection algorithms to locate a human face in a scene – easier and harder ones. Most of the existing algorithms use static images for face and emotion detection. In this work a dynamic approach has been used to detect emotions from webcam. Artificial neural network has been used for face recognition and emotion detection. Input is face detected of a person and and the output layer is emotion of the person. Methods of machine learning method has been used to implement the algorithm that is Harrcascade method which helps to detect the face, mouth and eyes of the person by converting normal image to grey image. Eigen Vector implementation is used to take average of all inputs (face) and decide our results from training data.

A dynamic approach is used here to detect the image of the person using a webcam and improve its quality using deblurring and denoising algorithms and get results by training and testing the machine. Next section covers literature studies done related to emotion detection from images. Then the advantage of videos over images is discussed. Section 3 describes the proposed algorithm which is further divide into 4 steps. Then implementation details and result are discussed. Finally conclusion and reference are mentioned.

II. LITERATURE SURVEY

Shahrin Azuan Nazeer, 2007 [1] used 3 layer feed forward Artificial Neural Network, using Sigmoid Function as an Activation function. Input was taken by accessing the webcam of the Device using Python Libraries and Google API's in Desktop and in Android Application. And then passing the Video instances as a Input to the ANN and then in the hidden layers we are performing below Algorithms and then passing the data can be pass to the activation function so that Machine could learn Effectively and then after training the weights generating the Final Output as an Emotion Detection in Real Time.

Eigen Vector Implementation was done by Daniel Georgescu, 2011 [2] , taking the size of the faces through well-defined functions, for all the faces inside the face detected while training (X1, X2, ---, Xn); Then Root Mean Square of the Eigen Values was taken. Philip Ian Wilson [3], after detecting faces we are used the Harr Cascade for feature extraction in the Faces like Eyes, Mouth, Nose and Lips. N. Hemalatha [4] used Non-Local Means (NLM) filtering algorithm for Denoising and Deblurring of input set of images.

III. PROPOSED APPROACH

A. Eigen Vector : Daniel Georgescu [2] implemented Eigen Vector by taking the size of the faces through well-defined functions, for all the faces inside the face detected while training (X1, X2, ----------, Xn). Let X= (summation of all values)/number of faces, Then while testing X' and if X' > X, then face is not detected. If X'==X, then evaluate again. If X' < X, then face is detected.

But we took the mean of these eigenvalues instead of using the Root Mean Square Formula. Because Root Mean Square method is deviated towards those images which are having a pure face, minimum noise.

But, Mean of Eigen Values method has deviated towards all the set of images which contain minimal Noise and a fair human face. Using Mean of Eigen Values formula we are taking and accepting all the images which are having decent noise and contains face as an object and not only biased for pure and noise free human face image. Hence our modified algorithm is:

For all the faces inside the face detected while training (X1, X2, ----, Xn), taking the Mean of the Eigen Values,

Let X= (summation of all values)/number of faces. Then while testing X' and if X' > X, then face is not detected! If X'==X, then evaluate again. If X' < X, then face is detected.
For Denoising and Deblurring we did the following changes for better efficiency.
We used NLM method because it is very efficient in preserving the details of the image while denoising. According to the proposed NLM method the input image after passing the Eigen Value Face Detection phase, the Mean of the pixels of the image is taken and for the all the pixels in the image if (Pixel intensity << Mean Pixel intensity OR Pixel intensity >> Mean Pixel Intensity). Then Dirty Bit Detected else pixel is noise free. Move to the next pixel.

But in our implementation, we took the input image and divided into various quadrants and the mean of these quadrants has to been taken. And then for the pixels in the input image:

If (Pixel intensity of a particular Quadrant << Mean Pixel intensity of that Quadrant OR Pixel intensity of a particular Quadrant >> Mean Pixel Intensity of that Quadrant). Then Dirty Bit Detected Else. Pixel is noise free. Move to the next pixel.

B. ADVANTAGES OF VIDEOS USING WEBCAM OVER NORMAL IMAGES:

1.Normal images not always give result proper result while using opencv. Sometimes it give error as opencv are not able to detect colour combination which is not problem faced while using videos from webcam

2. Images taken as a feature set are not always reliable and we can rely on videos. Moreover, normal images have a lot of noises which is not commonly found in videos.

3. Each person has its mouth and eyes to face ratio. In other words, we can say that every person has different face shape so we cannot take generic feature set and detect emotion of new person but in videos, we can find emotions according to person's face as a person face plays important role in emotions.

C. PROPOSED SOLUTION FOR REAL TIME FACE DETECTION:

1. Artificial Neural Network Algorithm:
Using Video Instances as an Input and then passing into the 3 Layer Feed Forward Neural Network in which the Flow of information is from left to right only. And after Passing the Input performing the BELOW ALGORITHMS and the n passing the DATA to the Activation Function and after updating the weights we are generating the Final Output as a Real Time Emotion Detection.

2. Eigen Vector Implementation:
Calculating the size of the faces through well-defined functions, for all the faces inside the face detected while training (X1, X2, ----------, Xn); Then take average of the Eigen Values. Let say X= (summation of all values)/number of faces Then while testing X and if X' > X, then face is not detected!! If X'==X, then evaluate again. If X' < X, then face is detected.

3. Harrcascade feature extraction:
We use this method to detect mouth, eyes, nose and various features of human face. Using Haar like Features are digital image features used in object recognition. They owe their name to their intuitive similarity with Haar wavelets and were used in the first real-time face detector. Its bit expensive as we used RGB pixel at each pixel so its time taking. A simple Rectangular Haar like features can be sum of pixels of areas insider the rectangle.
Harrcascade is machine learning algorithm based where we train it with different images positive or negative using Artificial Neural Network. It is then used to detect objects in other images. It checks our lips, face, and eyes using inbuilt xml files and inbuilt function by converting normal image to grey image.

3. Denoising and Deblurring:

While using webcam image is not always clear due to weak quality of camera and there is a lot of noise in image, so we use denoising and deblurring of image. Denoising means removal of dirty pixels from the image whereas Deblurring means replacing of dirty pixels with new pixels. Noise is of 2 types: Impulsive noise means changing part of any image with dirty pixels which only changes in particular part not on whole image. Gaussian noise is equally distributed over the signal.

To remove Impulsive Noise using Non Local Mean (NLM filter) Algorithm [4]: NLM filter used for impulsive Noise. Detects uncorrupted intensity which is equal to average weighted for all pixel and weighted value are directly proportional to neighbour pixel and its surrounding pixel similarity. NLM algorithm are very helpful in preserving image details.

Divide the entire images into quadrants and compute the RGB pixel of all the quadrant. If (Difference in the RGB pixel – mean quadrant value) > Experimental value). Then Noise = true, dirty bit identified -> noise identified. Otherwise if (Difference < Experimental value). Then Dirty bit! = true
Otherwise Perfect pixel, clear Image pixel.

To remove Deblurring from image [4]: Deblurring means to remove dirty pixels with new pixels. A method is proposed for handle noise in image deblurring based on theory and practical. The main observation is that apply an iterative filtering process, so that the noise level in an image get greatlyreduces, while preserving the blur information in the statically independent direction to the filter.

To remove Gaussian noise working [5]: Gaussian Filter we are using for Gaussian Noise. Unlike Impulsive Noise

Gaussian Noise is distributed equally over the Image or a Signal. For 2D Image we are using standard Gaussian Function to remove the Gaussian Noise.

G(x,y)=1/2*pie*((sigma)^2)*e^-(x^2+y^2)/2((sigma)^2)

Difference between Denoising and Deblurring of an Image [6]: Here the main difference between Denoising and Deblurring of an Image is finding out the DIRTY pixels inside Image and Denoising is the process to remove those DIRTY BITS by appropriate neighboring average BIT Pixel.

Detailed Algorithm

TrainSet ()

1. Read face using
2. Convert image to greyscale image
3.Use greyscale image to detect the mouth, face and eyes using detectMultiScale(roi_gray)
    For each time face detection using Haar cascade
    detect left eyes and right eyes and mouth
    use filter to detect eyes and lips
    (filter is mouth come half down the face)
4. Apply Denoising and Deblurring
5.Ratio of size of eyes and mouth with face
    Let X'= size of eyes (length of eyes * breadth of eyes) / size of face (length of face * breadth of face)
and
Y'= size of mouth (length of mouth*breadth of eyes) / size of face (length of face * breadth of face).
Repeat 3 to 5 steps for 3 times and take average of that and store that in using Eigen vector

X"=(X'1+ X'2 +X'3)/3
Y"=(Y'1+Y'2+Y'3)/3

TestingSet ()

1. Repeat 1 to 3 steps of TrainSet ()
2. For each time face detection using Haar cascade
3. Detect left eyes and right eyes and mouth and improve quality of image using Denoising and Deblurring
4. Take ratio of size of eyes and mouth with face

E'= size of eyes (length of eyes * breadth of eyes) / size of face (length of face * breadth of face)

M'= size of mouth (length of mouth*breadth of eyes) / size of face (length of face * breadth of face)

5. Store M' and E' in an array
6. Repeat step 2 to 5 for 50 times

If X"<E' then value[i] =1 else value[i] = 0
Similarly if Y"<M' then value1 [i] =1 else value1 [i] =0

7. Detect Emotions

7.1 Shocking: Sh= ((number of times value[i] =1)/ number of times loop run)*100
7.2 Smiling: Sm= ((number of times value1 [i] =1)/number of times loop run)*100

7.3Neutral:
    If sh<10 and sm< 10
        Status = neutral
    Else if sh>sm
        Status = shocking
    Else
        Status = smiling
// and similarly check for other 8 emotions.

In the TrainSet () algorithm we first start webcam and read face and then using haarcascade detect face, mouth and eyes and then take ratio of size of mouth to face and ratio of size of eyes to face. Repeat it for 3 times and take average of our ratios (X" and Y")

In the TestingSet () algorithm we take ratio of size of mouth to face and ratio of size of eyes to face. Repeat it for 50 times and then take value[i] =1 if X'' <E' else value[i] = 0 and similarly value1 [i] = 1 if Y" < M' else value1 [i] =0.

Sh= ((number of times value[i] =1)/ number of times loop run)*100. Sm= ((number of times value1 [i] =1)/number of times loop run)*100. Neutral if Sh < 10 and Sm < 10
And similarly check for other 8 emotion also.

IV RESULTS

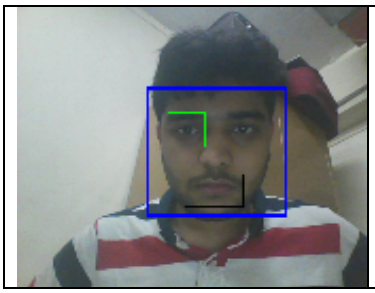
The proposed work is implemented to teach different emotions of the Asperger Syndrome patient. This can help the patient to understand emotion of another person easily. It has great importance in the medical field. We have implemented the work on two different platform – desktop and Android. For that, different algorithms and Artificial Neural Network for inputs learning have been used. Haarcascade is used for feature extraction, Denoising and Deblurring algorithms are used for improving the quality of image and also improve the efficiency of the results. We have tested the application on 100 people and get 90 plus accurate results.

Accuracy:
Table1. Desktop and Android application accuracy on 100 people

Application	Accuracy
Desktop	92%
Android	93%

Following are some screenshots of the results obtained.

A. Images with One Face



Face1:
Detected Emotion parameters:
Anger:0.01522,
Contempt:0.0006,
Disgust:0.00023,
Fear:.0.00331,
Happiness:0.0000,
Neutral:0.53505

Figure 1

B. Video with Two Faces

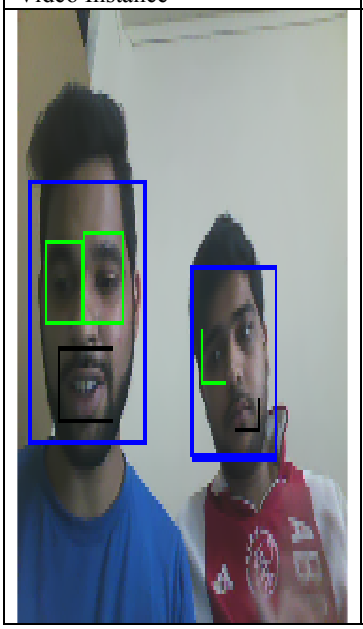
Video Instance	Output
	Detected Emotion parameters: *Face1:* (Left) Anger:0.00004, Contempt:0.00069, Disgust:0.00561, Fear:0.00000, Happiness:0.07457, Neutral:0.91452, Sadness:0.00015, Surprise:0.00005 *Face2:* (Right) Anger:0.000689, Contempt:0.00264, Disgust:0.00666, Fear:0.00097, Happiness:0.00457, Neutral:0.976971, Sadness:0.00815, Surprise:0.000032

Figure 2

C: Video with more than 2 Faces:
When number of faces were more than two, it was found that Android application was working better and faster. One such example is given in figure 3.

Android Application:

Face1(Right most):
Anger:0.00001,
Contempt:0.00761,
Disgust:0.00333,
Fear:0.00003,
Happiness:0.26349,
Neutral:0.79985,
Sadness:0.0,
Surprise:0.0

Face2: (centre)
Anger:0.00000,
Contempt:0.00435,
Disgust:0.00216,
Fear:0.00000,
Happiness:0.18588,
Neutral:0.80761,
Sadness:0.00000,
Surprise:0.0

Face3: (second left)
Anger:0.00019,
Contempt:0.00803,
Disgust:0.00022,
Fear:0.00001,
Happiness:0.12565,
Neutral:0.86477,
Sadness:0.00000,
Surprise:0.00040

Figure 3

IV. FUTURE WORK

The future scope of this research is to develop an application for people with impaired vocal activity so that their emotions could be detected efficiently. An idea is to use it for Asperger Syndrome patient. By proper training and testing better results are achievable.

V. REFERENCES

[1] Shahrin Azuan Nazeer 1, Nazaruddin Omar 2 and Marzuki Khalid 3, "Face Recognition System using Artificial Neural Networks Approach" in, Anna University, Chennai, India. Feb. 22-24, 2007. pp. 420-425.

[2] Daniel Georgescu "A Real-Time Face Recognition System Using Eigenfaces" in Journal of Mobile, Embedded and Distributed Systems, vol. III, no. 4, 2011. ISSN 2067 – 4074.

[3] Philip Ian Wilson 1, Dr. John Fernandez 2 "Facial Feature detection using haar classifiers" in South Central Conference, Texas April, 2006.

[4] N. Hemalatha 1, "Image Denoising and Deblurring Using Non-Local Means Algorithm in Monochrome Images" in International Journal of Engineering Research and General Science Volume 2, Issue 2, Feb-Mar 2014. ISSN 2091-2730.

[5] Gaussian Filtering presentation by the university of Auckland, New Zealand.

[6] Li Zhang 1, Alok Deshpande 2, Xin Chen 3, "Denoising vs. Deblurring: HDR Imaging Techniques using

moving cameras" in National Science Foundation EFRI-0937847, IIS-0845916, and IIS-0916441.