Emotion Detection using Neural Network using Dynamic Approach

Amit Sachdeva(amitsachdeva45@gmail.com)1

Burhannudin Bhopalwala (burhannudinbhopalwala.cse@gmail.com)2

Abhinav Mishra(abhim4757@gmail.com)3

1 Computer Science and Engineering, Jaypee Institute of Information Technology, Sector 62, Noida, India

2 Computer Science and Engineering, Jaypee Institute of Information Technology, Sector 62, Noida, India

3 Information Technology, Jaypee Institute of Information Technology, Sector 62, Noida, India

**Abstract: *In this paper, Emotion Detection from videos using webcam has been addressed. Firstly, the current methodologies used in image processing is identified and get decent efficiency by using a static method of image processing. In this paper, dynamic approach (frames from videos) 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 which helps to train the machine so that the machine can be used for the testing phase. Along with that, denoising and deblurring algorithm is used to improve the quality of the image recieved from a webcam as an input and it leads to improving the efficiency of the machine. After denoising and deblurring, 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.***

***Keywords:*** Artificial Neural Network, Haarcascade, Denoising, Eigenvector.

1. **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. To detect emotions from face, different image strategies have been used earlier using single image.

Here artificial neural network is implemented unlike normal static image processing methodologies, neural network consists of a large number of simple processing elements simultaneously. Human brain consist of billons of neurons which are connected with each other which help to perform various activities at a time through massive interaction among them.

Artificial Neural Network are mostly in computer science research work which is based on a large collection of neural network having three layer system: Input Layer, Hidden Layer and Output Layer. In this method, we train our inputs with desired outputs and then test various inputs for final result. In this paper, Input is face detection of a patient, Hidden layer is the process in which we used our algorithm and the output layer is emotion of the person.

In a paper, various 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.

So using a dynamic approach 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.

Section 1 have the literary studies we have done before our projects.

Section 2 describes the algorithm of the project which is further divide into 4 steps.

1. **Advantages of Videos using Webcam over normal images:**
2. 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
3. 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.
4. 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.
5. **Proposed Solution for Real Time Face Detection:**
6. **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.

1. **Harrcascade feature extraction:**

We use this method to detect mouth, eyes, nose and various features of human face. Using Haarlike 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.

1. **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* isequally distributed over the signal. An image containing noise can be described as follows:

X (I, J) = η I, J probability (p)

Y (I, J) = probability 1-p

* To ***remove Gaussian noise***working: Later ADD; Gaussian filter working;
* To ***remove Impulsive Noise***using **Non Local Mean (NLM filter) Algorithm**:

NLM filter 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.

Step 1: 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:

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 greatly reduces, while preserving the blur information in the statically in dependent direction to the filter.

1. **Full algorithm used in paper:**

TrainSet ()

{

1. Start webcam
2. Read face using cap.read() function
3. Convert image to greyscale image
4. Use greyscale image to detect the mouth, face and eyes using detectMultiScale(roi\_gray)
5. For each time face detection
6. detect left eyes and right eyes and mouth
7. Then use filter that allow opencv to only detect eyes and lips (filter is mouth come half down the face)
8. Then take ratio of size of eyes and mouth with face

X’= size of eyes (length of eyes \* breadth of eyes) / size of face (length of face \* breadth of face)

Y’= size of mouth (length of mouth\*breadth of eyes) / size of face (length of face \* breadth of face)

1. Repeat 5 to 8 steps for 3 times and take average of that and store that in

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
3. Detect left eyes and right eyes and mouth
4. Then 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)

1. Store M’ and E’ in an array
2. 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

1. Then Detect Emotions

For Shocking

Sh= ((number of times value[i] =1)/ number of times loop run)\*100

For Smiling

Sm= ((number of times value1 [i] =1)/number of times loop run)\*100

For neutral

If sh<10 and sm< 10

Status = neutral

Else if sh>sm

Status = shocking

Else

Status = smiling

}

*Explanation of Algorithm*:

Train Set ()

In our 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”)

Testing Set ()

In this portion 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.

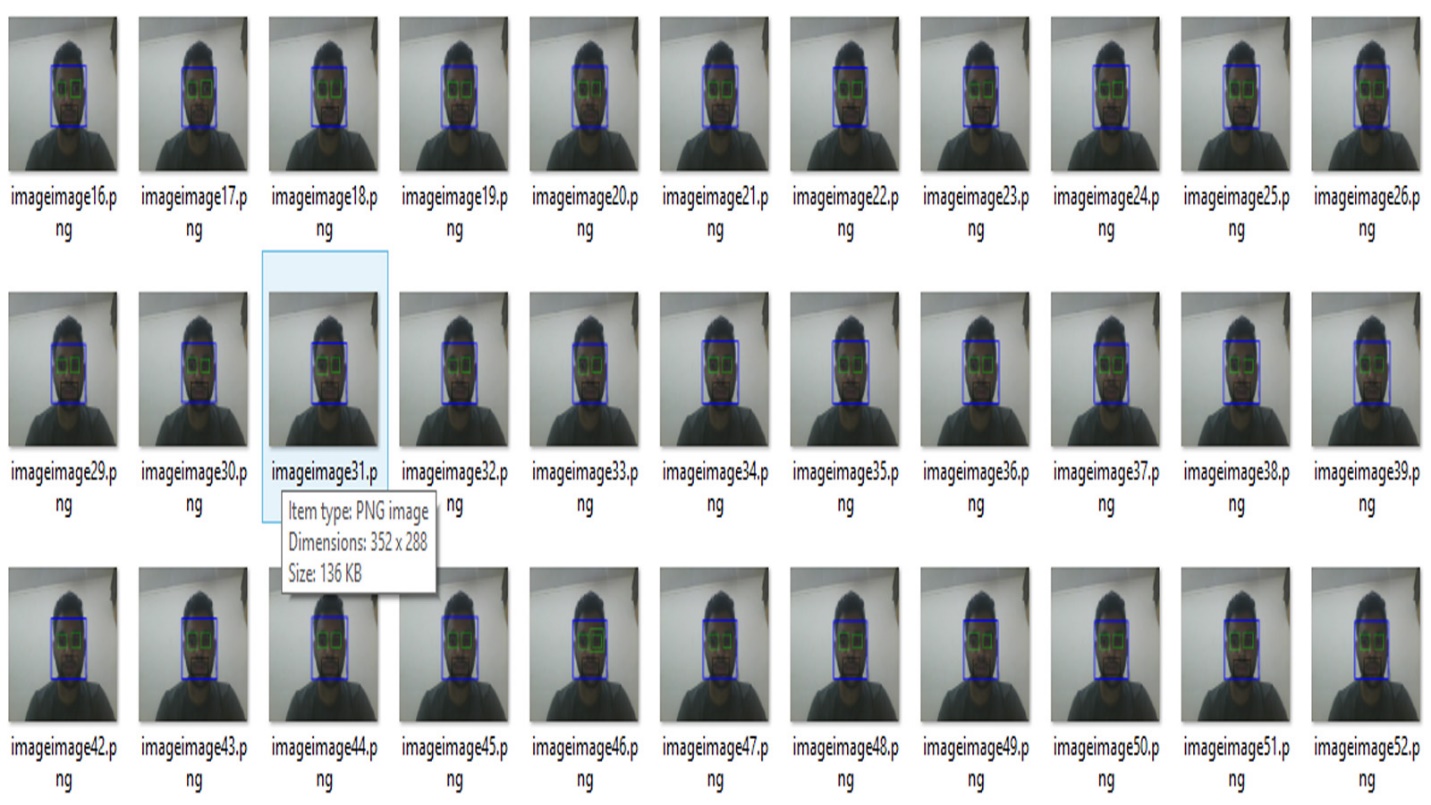
Take percentage to detect shocking and smiling

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

1. **Implementation:**

****