Face Detection and Expression Analysis

Utkarsh Yadav[1] ,Pratyush Raj Das[2] and Chinmai S Naregal[3]

Vellore Institute Of Technology, Chennai, India .

ABSTRACT The goal is to design and implement a novel face detection and expression recognition system using available technology and define its importance in modern image processing which enhances human - computer interactions for various smart applications and to study how contextually aware systems or robots can enhance our day to day life.

I. INTRODUCTION

Facial expression plays a major role in the interaction between people and machines. From human facilities to clinical procedures, facial expressions can be valuable identification. Applications focused upon the emotional recognition like HCI, Social Robot, Patient Monitoring Cartoon, Warning device play an important role in analyzing facial expression.

The face detection and expression recognition system should be able to take live video input from the robot's camera (in this case the webcam) and read the video frame by frame and detect the facial features such as left eye, right eye and nose/mouth , using the classifiers and correctly identify the facial expression of the person. A CNN model will be built to study the training set in order to accurately predict the facial expression.

III. FACE DETECTION PROCESS

Each live video frame is converted to grayscale after loading it. The majority of the images we encounter are RGB channel images (Red, Green, Blue). As a result, when OpenCV reads an RGB image, it stores it in the BGR (Blue, Green, Red) channel. This BGR channel must be converted to grey channel for image recognition purposes. The reason for this is because grey channel is simple to process and computationally less intensive because it just has one black-and-white channel.

IV. METHODOLOGIES USED

The system will be built using the Python language for its simplicity and its wide availability of packages that supports the pre-processing necessary for the working of the system and the reusability and scalability it provides for constant updating.

OpenCV : OpenCV is an open-source library.

TensorFlow : TensorFlow APIs use Keras to allow users to make their own machine learning models.

Neural Networks

The system will have an experimental method of workflow and the following conditions are considered favorable for the correct detection of a particular individual's face and recognition of the person's expression so it is assumed as such:

The user is in a well lit surrounding

The user is in a neutral background

The CNN model has been sufficiently trained and tested.

II. PROPOSED WORK

The project will take live video input from the user's webcam and parses frame by frame image to the program. Using the Haar Cascade classifier we will be able to identify distinct facial features like eyes, nose and mouth. The trained CNN model is loaded onto the system. This test image of detected face is sent the loaded CNN model for expression analysis and classification and the predicted emotion is shown in the output. Classification Models

V. IMAGE PREPROCESSING

We have used multiple pre-processing methods such as increasing dimensions, image rotation, image rescaling,etc.

V.A IMAGE ROTATION

The images that are being provided as input have the face in a vertical position. Due to this the emotions of a tilted face might not get recognized. We have used Image data generator function of keras to rotate the image at various angles so that the emotions can be recognized even at an angle.

V.B IMAGE RESCALING

We have rescaled the image using Image Data Generator of keras so that the quality of the image imporves and our model can predict the emotion more accurately.

V.A INCREASING DIMENSIONS

We have increased the dimensions of the image so that the image get enhanced.

VI BUILDING THE CNN MODEL

We have used sequential() to build the model in CNN. It is imported from keras and builds the model layer by layer.

VI.A CONV_2D

This layer creates a convolution kernel that is wind with layers input which helps produce a tensor of outputs. They deal with our input images, which are seen as 2-dimensional matrices.

We have added four Conv2D layers to our model. This result was giving the maximum accuracy to our model and upon increasing the amount of layers, we were not able to extract more features as there is a limit.

VI.B FLATTEN

We have used flatten() function to reduce the dimension of the image. It will improve the quality of the image and also reduce the processing time of each image.

VI.C FULLY CONNECTED LAYER

We have used dense() of keras as our neural network layer. W have used two activations, relu and softmax.

Relu applies the rectified linear unit activation function as returns value only when the we get positive input.

VI.D ADAM OPTIMIZER

Adam optimizer is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments.

Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems.

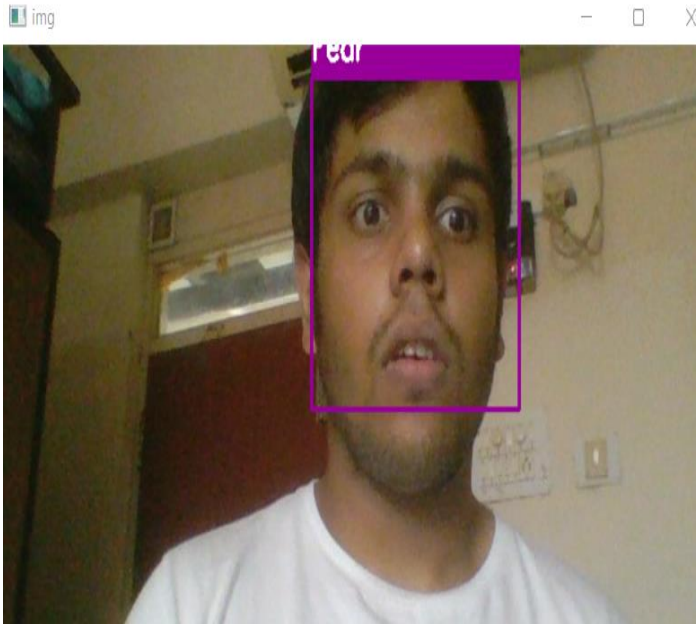
VII CAPTURING IMAGE AND DISPLAYING EMOTION

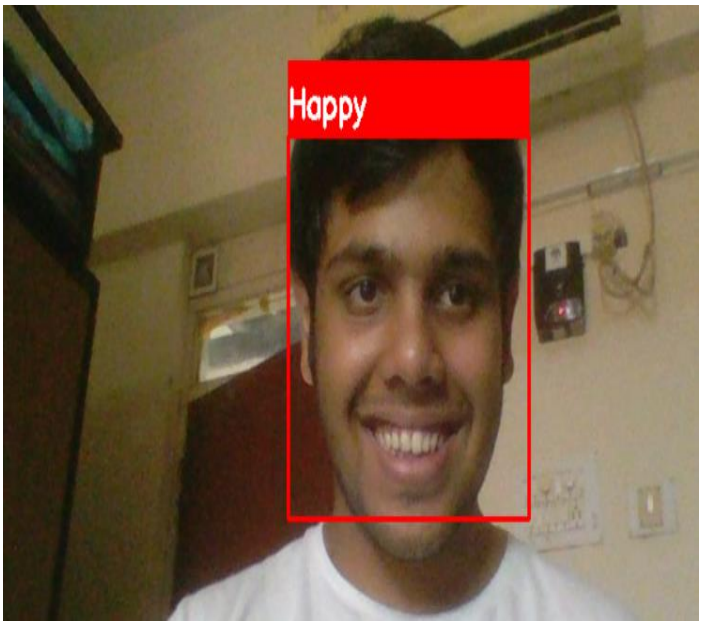
We take image input through a live feed. The algorithm detects the face of a person in the surroundings and marks a boundary around it. Each emotion boundary has its name on the top and a different colour.

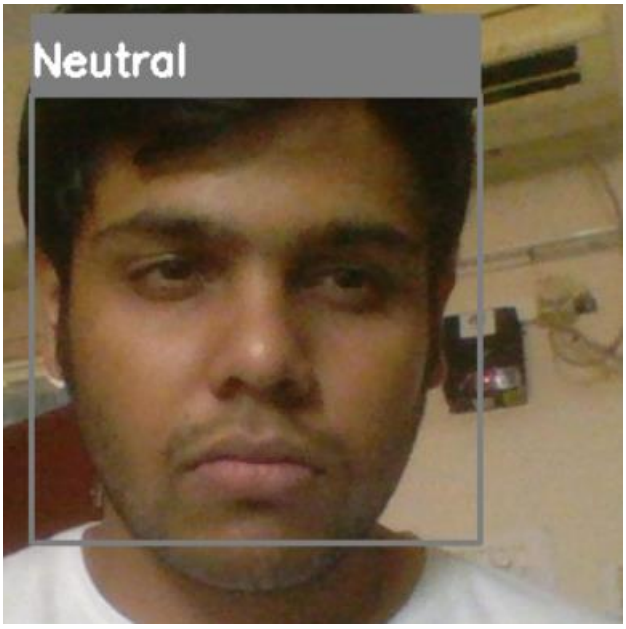
We have stored each colour in an array. When the input arrives we pass it through a face_cascade.detectMultiScale function that detects object of different sizes so that even if person is a little far away, emotion can be detected. Then the images are passed to the algorithm and it matches emotion. The result is passed onto an if-else statement and the corresponding emotion that gets matched will be displayed.

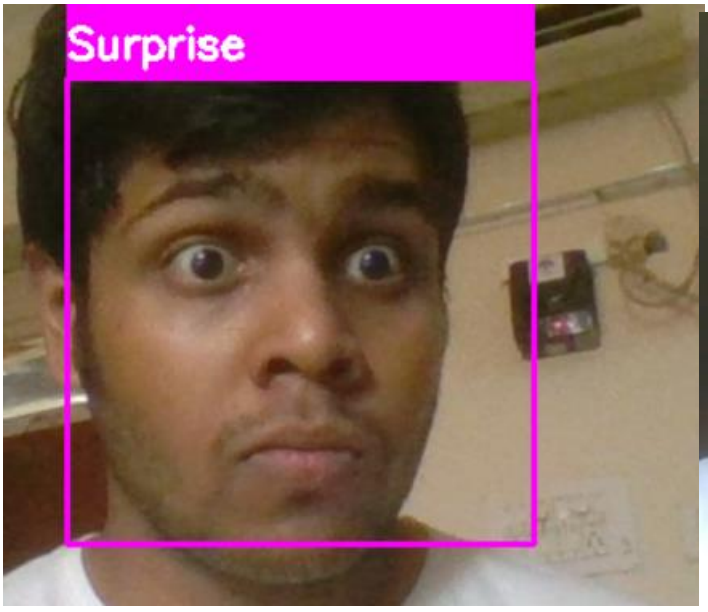
OUTPUT

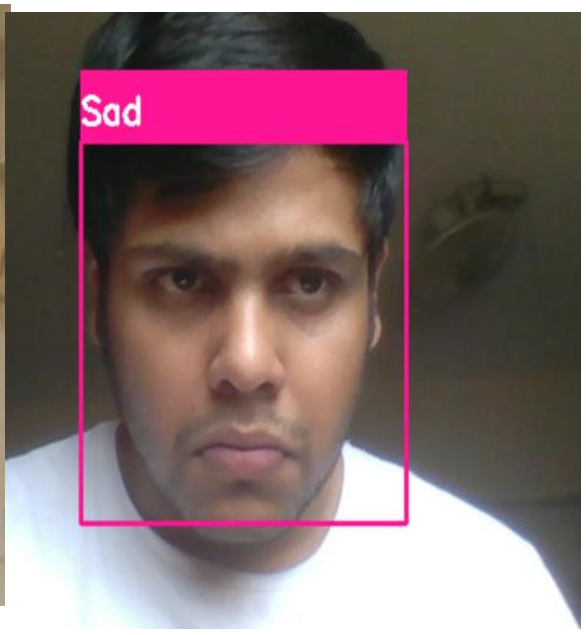
As mentioned before, we have 6 distinct emotions that can be identified by our novel model and they are shown below:











VII. CONCLUSION

Finally, we can conclude that our model is a success and that through live image detection, it can identify 6 separate emotions on a face i.e happy, sad, surprise, fear, neutral and angry. Through the use of CNN and other mathematical models, we can achieve our goal.

ACKNOWLEDGMENT

We would like to acknowledge out teacher, Dr Geetha S for being a constant source of support for us throughout the semester and we would like to thank her especially for her for always being there to clear our doubts and helping us. We

would also like to than our friends and family for helping us along the way.