PROJECT UPDATE

FACIAL EMOTION DETECTION

Group 12

Gayatri Annapurna Vadlani

Mohammed Johny Shaik

GitLink: https://github.com/Gayatri345/CSCE5222.git

Problem statement:

We are planning to implement our project to detect facial emotions from video which uses Open CV haarcascades algorithm on video for face detection, facial key features and input it to the Deep Learning model to detect the emotion from the video. We are planning to work on emotions like angry, disgust, happy, sad, surprise and neutral.

TIMELINES:

Module Working on	**Study**	**Coding**	**Result**	**Due date**
Dataset	Done	Done	Done	Oct-10
Extracting facial features	Done	Done	Done	Oct-15
Developing a Neural Network Model	Done	In Progress	-	Oct-30
Experimenting with already available DeepFace for emotion recognition and calculating percentage of emotion.	Done	Done	Done	Oct-20
Video input to the model and calculate percentage of emotions. Removal of noise from the input and inputting video for facial recognition	In Progress	In Progress	In Progress	Nov 1-Nov 7
Training and Plotting the accuracies, predictions and confusion matrix.	Done	In Progress	In Progress	Nov5-Nov10

Error check and final code evaluation	In Progress	In Progress	In Progress	By Nov-15

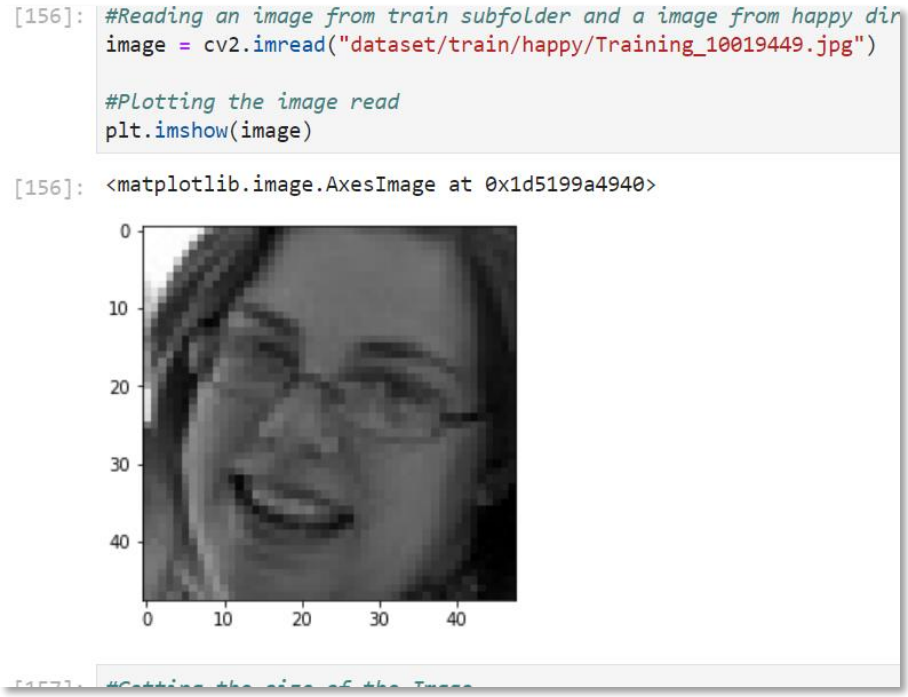
Results/Progress till date:

1. **DataSet:**

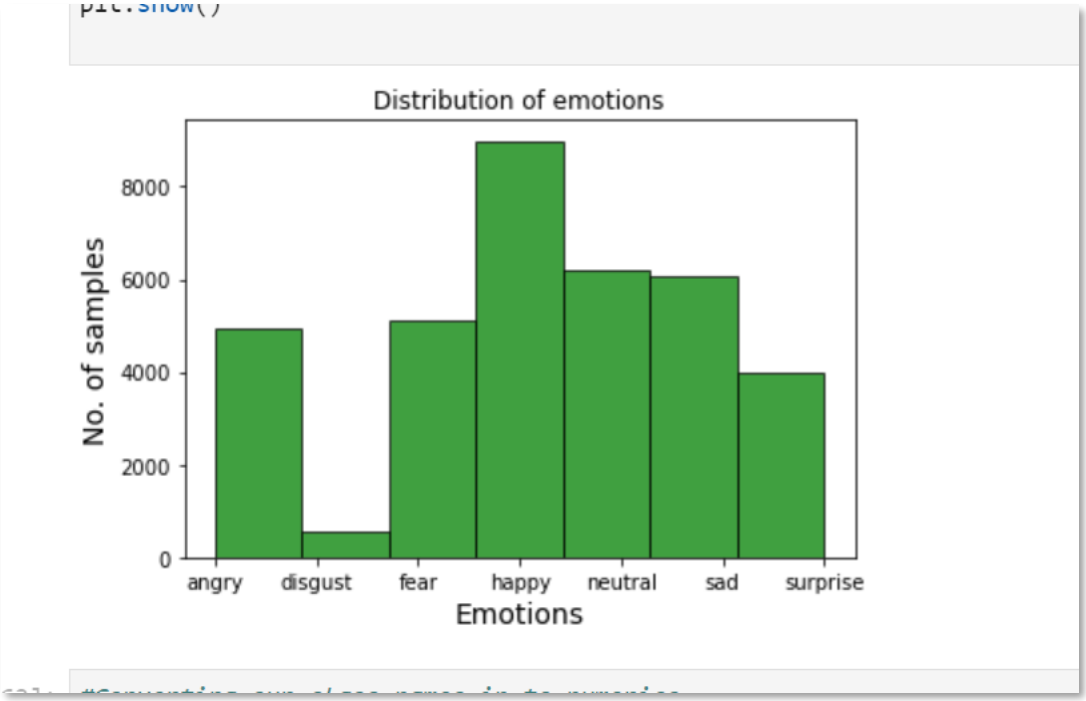
   We worked on the dataset, we studied different classes of data available and converted the required data in to test.csv file and train.csv files which we are planning to use as input for our developed model for testing and training.

   We also plot a histogram to understand the distribution of different classes in the dataset FER-2013.

Sample training image imported from our dataset FER-2013:



Class Distribution using Histogram Plot:



Exporting the data required to csv files:

```
[174]: df_train.to_csv('dataset/train.csv')

[175]: df_test.to_csv('dataset/test.csv')

[176]: df_encoded.to_csv('dataset/datasetFER_2013.csv')
```

2. **Facial feature extraction:**

   We used haarcascade algorithm to extract facial frontal-feature. We verified the same by using image of happy face as shown in the below picture.

   Reading an image:

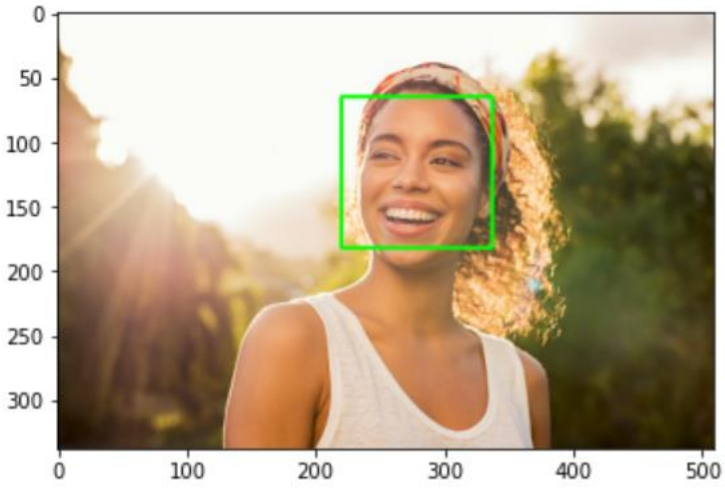
```
[6]: plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
```

```
[6]: <matplotlib.image.AxesImage at 0x188c3078c70>
```


```
[7]: plt.imshow(cv2.cvtColor(image, cv2.COLOR_RGB2GRAY),cmap='gray'
```

Detecting Face in the picture using haarcascade, and drawing a bounding box around the face.

```
|: <matplotlib.image.AxesImage at 0x188d03865b0>
```

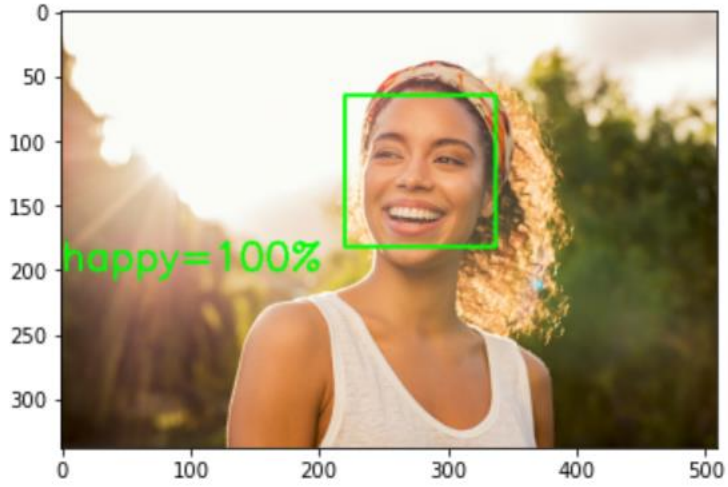


Getting amount of emotions:

We want to generate the percentage of emotion on the picture. For this we applied some python coding logic and tried to display the percentage of dominant emotion.

Here we are testing by using DeepFace in python library. Later this model will be replaced by our custom developed model deep neural network.

```
<matplotlib.image.AxesImage at 0x1890636ebe0>
```



FUTURE DEVELOPMENTS:

1. Developing model for facial emotion recognition.
2. Inputting video and using haarcascade to detect faces in the video.
3. Trying input a noise video and study the accuracy before and after removing noise in the video.
4. Compare the model with DeepFace model.
5. Generating confusion matrix, and predictions with valid and test set.