International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 10, Issue 8, August-2022, Impact Factor: 7.429, Available online at: www.ijaresm.com

Facial feature Extraction and Emotional Analysis Using Machine Learning

Chitra Joshi[1], Jalal Mujawar[2], Prattipati Sri Vaishno Sai Suren[3], Gnanadeep Manduva[4], Siddavarapu Sai Chandra Kaushik[5], Rhuthvik Dendukuri[6]

----------**********----------

1. INTRODUCTION

The Importance of Emotion Recognition is that Human face detection plays very important role in many applications such as video surveillance, human computer interface, face recognition, face image database management , etc. Such Facial expressions are important cues for a non-verbal communication among several human beings. This is only possible because humans can recognize emotions quite accurately and efficiently.

An automatic facial emotion recognition has many commercial uses and can be used as a biological modelfor cognitive processing and analysing of human brain.

Collectively they can enhance their applications like monitoring and surveillance analysis, biomedical image, smart room's intelligent robots, and human computer interfaces and driver's alertness system and can play a vital role in the field of security and crime investigations.

This project basically aims to classify all the emotions on a person's face into one of seven categories, using deep convolutional neural networks. The proposed method will use HAAR Cascade classifier to detect the face in an image.

Objective of the project:

The objectives of the project are as follows: -

- To find a suitable dataset using which we can train the AI.
- To create a module suitable for our dataset.
- Train the module until a high enough accuracy is reached and savethe resultant model.
- This project aims to predict the emotions of the person byhis/her facial expression with high Accuracy.
- here are seven types of human emotions shown, that are universallyrecognised across different cultures namely anger, happiness, disgust, fear, sadness, surprise and neutral.

Innovation component in the project:

1. The project one of the most challenging datasets available for the topic.work.
2. The use of keras made the project much more efficient to manage and
3. The use of Python along with tensorFlow and OpenCV is the best way of analyzing image data for machine learning that is available rightnow.

Work done and implementation

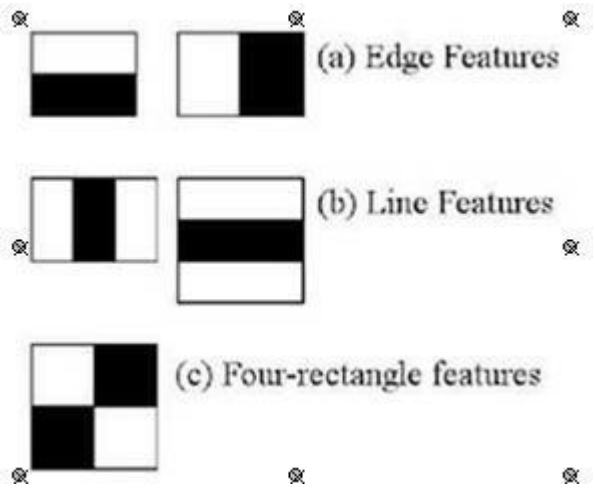
Methodology adapted:

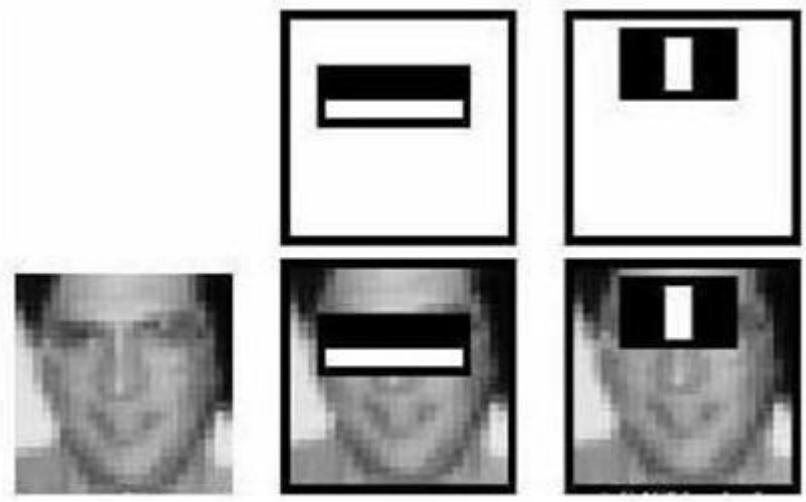
Algorithm

- First, the **haar cascade** method is used to detect faces ineach frame of the webcam feed.
- The region of image containing the face is resized to **48x48** and is passed as input to the CNN.
- The network outputs a list of **softmax scores** for the sevenclasses of emotions.
- The emotion with maximum score is displayed on the screen.

1) Face detection-

Face Detection is the fundamental step in any of the operations carried out in the face recognition process. The Haar Feature-based Cascade Classifier is a widely used mechanism for detecting faces. In order to train a classifier to detect faces, two large sets of images are formed, with one set containing images with faces, and the other set without. These images are then used to generate classifier models. The classifier is generated by extracting Haar features from the positive and negative images.





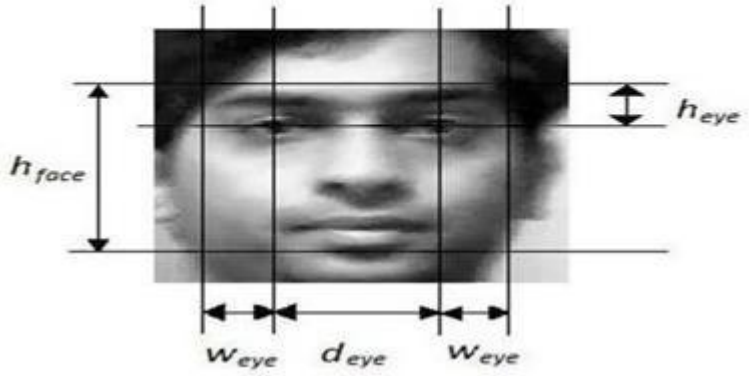
2) Face capture

The very first step in face recognition is to collect face samples. This is carried out inthree basic steps as follows:

1. Detect the face.
2. Crop the cardinal section of the face.
3. Save the face image.

The detection of the face is achieved using **Haar Feature-based Cascade Classifiers**. Typically, the accuracy of face recognition is highly dependent on the quality and variety of the sample images. The variety of sample images can be obtained by capturing multiple images with multiple facial expressions for the same face.

$$h_{face} = K_f \, d_{eye} \quad \text{(eq. 1)}$$
$$h_{eye} = K_e \, h_{face} \quad \text{(eq. 2)}$$
$$w_{eye} = K_{we} \, h_{face} \quad \text{(eq. 3)}$$



3) Emotion recognition

Here we tried with **fer2013** dataset then we made some classification in the as our needmake it to train our model We have converted the csv file into a dataset of images in the PNG format for training/testing and provided this as the dataset.

For training data model, we have make a python code which grab all the classified imagesfromfolders and map it with its emotion.

Architecture diagram:

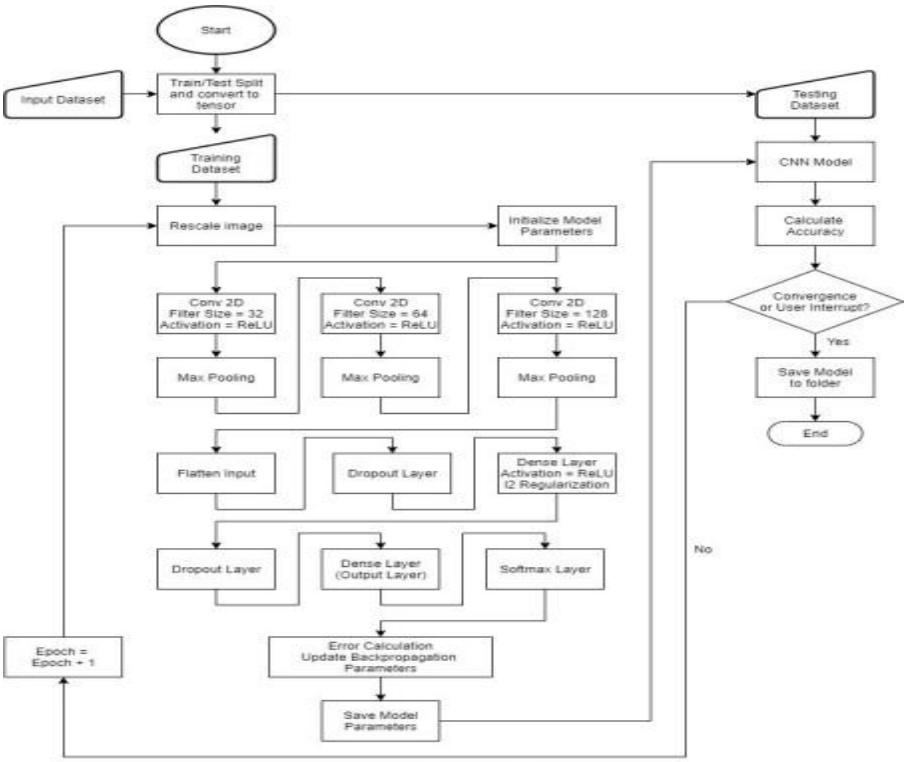


Figure 1-CNN Block Diagram

Dataset used:

The model is trained on the FER-2013 dataset which was published on International Conference on Machine Learning (ICML). This dataset consists of 35887 grayscale, 48x48 sized face images with seven emotions - angry, disgusted, fearful, happy, neutral,sad and surprised

Fer2013 is a challenging dataset. The images are not aligned and some of them are incorrectly labelled as we can see from the following images. Moreover, some samplesdo not contain faces.

This makes the classification harder because the model have to generalize well and berobust to incorrect data.

The best accuracy results obtained on this dataset, as far as I know, is 75.2% described in this paper: [Facial Expression Recognition using Convolutional Neural Networks: State of the Art, Pramerdorfer & al. 2016]

But as our AI trained so intensively that the test accuracy reached 86.2% in 50 epochs.

Tools used:

We used anaconda environment as it was easy to load other packages into it.We also used Jupiter notebooks for the coding section (Python 3).
OpenCV, Tensorflow to handle images.

HAAR Cascade classifier to detect the face in an image.

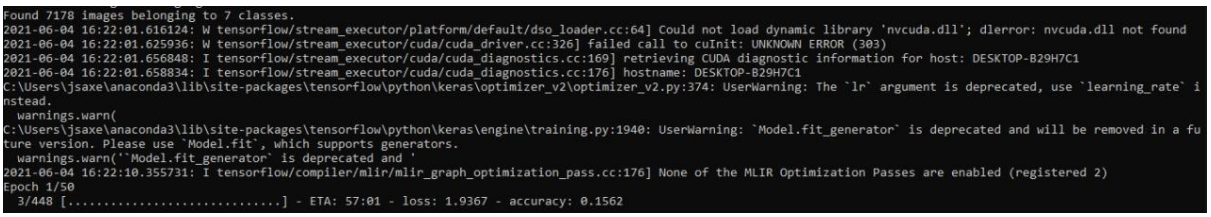
Matplotlib to plot graph of accuracy.

Screenshot and Demo:Training-

```
train_generator = train_datagen.flow_from_directory(
        train_dir,
        target_size=(48,48),
        batch_size=batch_size,
        color_mode="grayscale",
        class_mode='categorical')

validation_generator = val_datagen.flow_from_directory(
        val_dir,
        target_size=(48,48),
        batch_size=batch_size,
        color_mode="grayscale",
        class_mode='categorical')
```

```
if mode == "train":
    model.compile(loss='categorical_crossentropy',optimizer=Adam(lr=0.0001, decay=1e-6),metrics=['accura
    model_info = model.fit_generator(
            train_generator,
            steps_per_epoch=num_train // batch_size,
            epochs=num_epoch,
            validation_data=validation_generator,
            validation_steps=num_val // batch_size)
    plot_model_history(model_info)
    model.save_weights('model.h5')
```

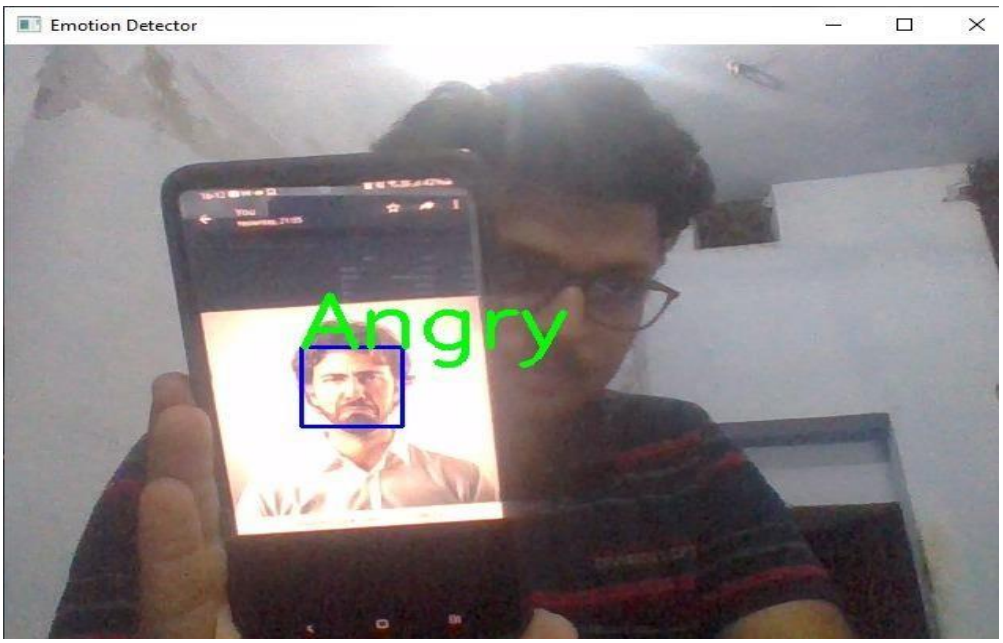
Ss of model training:



Testing-

```
while True:
    # Grab a single frame of video
    ret, frame = cap.read()
    labels = []
    gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(gray,1.3,5)

    for (x,y,w,h) in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(255,0,0),2)
        roi_gray = gray[y:y+h,x:x+w]
        roi_gray = cv2.resize(roi_gray,(48,48),interpolation=cv2.INTER_AREA)


        if np.sum([roi_gray])!=0:
            roi = roi_gray.astype('float')/255.0
            roi = img_to_array(roi)
            roi = np.expand_dims(roi,axis=0)

        # make a prediction on the ROI, then lookup the class

            preds = classifier.predict(roi)[0]
            print("\nprediction = ",preds)
            label=class_labels[preds.argmax()]
            print("\nprediction max = ",preds.argmax())
            print("\nlabel = ",label)
            label_position = (x,y)
            cv2.putText(frame,label,label_position,cv2.FONT_HERSHEY_SIMPLEX,2,(0,255,0),3)
        else:
            cv2.putText(frame,'No Face Found',(20,60),cv2.FONT_HERSHEY_SIMPLEX,2,(0,255,0),3)
        print("\n\n")
    cv2.imshow('Emotion Detector',frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

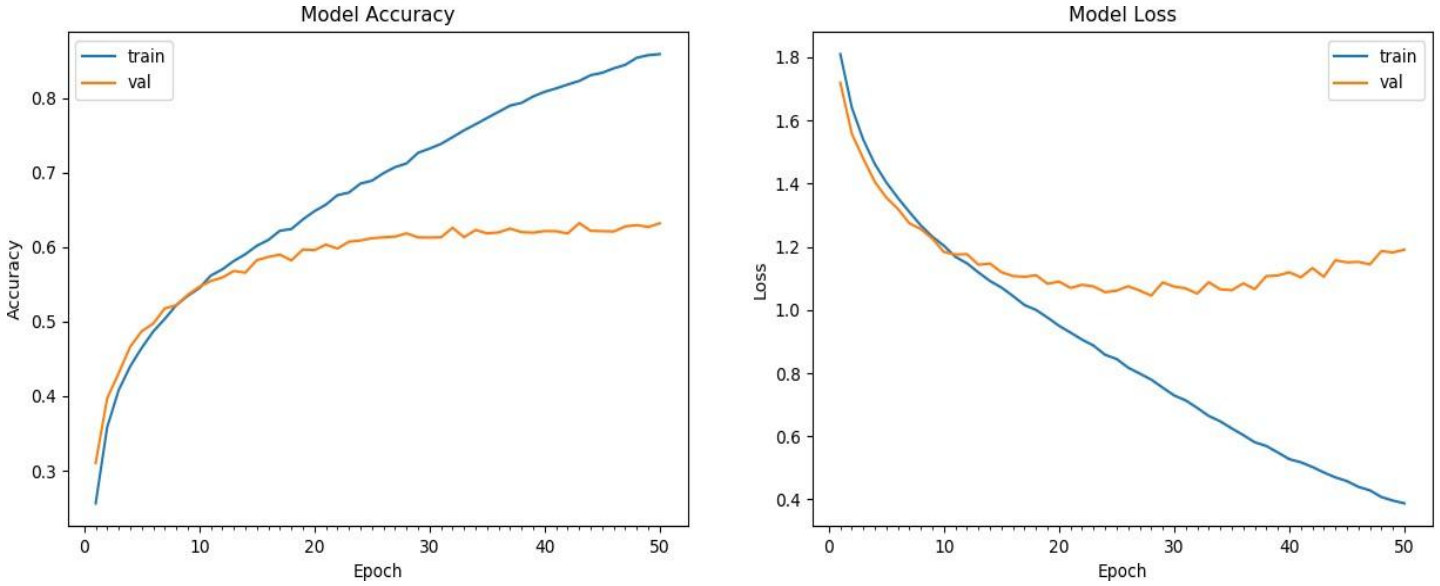
Outcome Screenshots:



RESULTS AND DISCUSSION

We conclude the projects by hoping that you got a fair idea and understoodthe whole pipeline on how you can make an emotion detection model.

We initially defined network and trained it so that it can classify the correct emotion and then we made use of that trained model to classify the emotions in real-time. The model does give good accuracy and takes much time to train. We can make use of other pre-trained architecture like ResNet and Mobile Net to train your network for better performance of the model.



With our method the highest accuracy we were able to reach was 86.2percent.

The average test accuracy reached 63.5 percent.

With rapid advancements in Artificial intelligence, the use cases and applications you can build using AI are increasing rapidly. You can always think of what use cases you can build after reading an article like this usingthe power of CNN.

REFERENCES

[1] Abadi, Mart, Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... others. (2016). Tensorflow: A system for large-scale machine learning. In 12th $USENIX$ Symposium on Operating Systems Design and Implementation ($OSDI$ 16) (pp. 265–283)

[2] Joseph, A. and Geetha, P., 2020. Facial emotion detection using modified eyemap–mouthmap algorithm on an enhanced image and classification with tensorflow. *The Visual Computer*, *36*(3), pp.529-539..

[3] Abdulsalam, W.H., Alhamdani, R.S. and Abdullah, M.N., 2019. Facial emotion recognition from videos using deep convolutional neural networks. *International Journal of Machine Learningand Computing*, *9*(1), pp.14-19.

[4] Caramihale, T., Popescu, D. and Ichim, L.,2018. Emotion classification using a tensorflowgenerative adversarial network implementation. *Symmetry*, *10*(9), p.414.