# Real Time Emotion Detection System for Retail Stores

**By**

**ADIL AHMED 19BCE1012**
**MAYANK SHARMA 19BCE1145**
**SHAIL PATEL 19BCE1414**

A Project Report Submitted To

**Dr. Geetha S**
**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

In partial fulfillment of the requirements for the course of
**CSE4019 - Image Processing**
In
**B.Tech COMPUTER SCIENCE AND ENGINEERING**

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

**Vandalur - Kelambakkam Road**
**Chennai - 600127**

**APRIL 2022**

# DECLARATION

We hereby declare that the project entitled **“Real Time Emotion Detection System for Retail Stores”** submitted by us to the School of Computer Science and Engineering, Vellore Institute of Technology, Chennai Campus, Chennai 600127 in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology – Computer Science and Engineering** is a record of bonafide work carried out by us**.** I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma of this institute or of any other institute or university.

Signature

Mayank Sharma 19BCE1145
Shail Patel 19BCE1414
Adil Ahmed 19BCE1012

# CERTIFICATE

The project report entitled “**Real Time Emotion Detection System for Retail Stores**” was prepared and submitted by **Mayank Sharma (Register No: 19BCE1145), Adil Ahmed(Register No:19BCE1012) and Shail Patel(Register No: 19BCE1414)**. It has been found satisfactory in terms of scope, quality and presentation as partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology – Computer Science and Engineering** in Vellore Institute of Technology, Chennai, India.

# ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. Geetha S,** for her consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.We are extremely grateful to **Dr. Ganeshan R**, Dean of the School of Computer Science and Engineering, VIT Chennai, for extending the facilities of the School towards our project and for his unstinting support. We express our thanks to our Head of the Department **Dr. P Nithyanandam** for his support throughout the course of this project. We also take this opportunity to thank all the faculty of the School for their support and their wisdom imparted to us throughout the course. We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

# CONTENTS

# ABSTRACT

Emotion recognition systems play an essential function in many fields, especially picture processing, scientific technological know-how, and system studying. As per human needs, the effect and capability use of programmed emotion recognition have been developing in a huge scope of utilizations, consisting of humanPC conversation, robot control and driver nation commentary. Anyhow, so far, lively acknowledgment of outward appearances from photos and recordings is yet a trying out errand because of the problem in exactly extricating the helpful passionate highlights. Those highlights are often spoken to in diverse systems, for example, static, dynamic, factor based geometric or place based appearance. Facial improvement highlights, which comprise position and shape adjustments, are with the aid of and are brought about by way of the developments of facial additives and muscle tissues at the face in an enthusiastic manner. Emotion recognition structures have many programs. and it plays an essential part in fault detection and in gaming applications. In this task the emotion recognition is dynamic and no longer like uploading the photo and finding the emotion. And that is executed with the help of the idea of machine getting to know referred to as Convolutional Neural network. That is one of the maximum acquainted deep mastering standards. The principle motto of using this concept is to hold accuracy. The CNN includes many intermediate states which play a crucial role in producing correct output. The layers of CNN are the enter layer, hidden layer and output layer. The hidden layer is used to replace weight, bias and activation function. If we use the CNN method the undesirable components which aren't necessary for the emotion recognition can be eliminated as it should be. CNNs help reduce the elimination assignment in a simpler way and with a minimal range of steps.

**Dataset Used:**

https://www.kaggle.com/msambare/fer2013

# INTRODUCTION

The facial additives, especially the important additives will continually display signs of changing their function when the emotions of the character are modified. As a result, a comparable element in various pictures is maximum a part of unique positions. In case, the region of the element may be twisted or bent out of the ordinary shape because of facial muscle traits. For instance there are 3 photos, the mouth position of the first pics provides distinct shapes from that inside the 1/3 photo. in line with the above state of affairs the precise feeling, the geometric - primarily based position and look-primarily based shape usually modifications starting with one photograph then the next image inside the database just as in recordings. In current years, the recognition machine is used in the discipline of dynamic research with applications in some precise fields. as an instance human and system affiliation, neural science, laptop illustrations. transport safety via identifying the motive force's weakness. Emotion recognition systems have performed a vital function in device interface which allows to make communique among system and human in an efficient and less complicated way. A few programs use the face and thumb for the individual recognizable proof and get right of entry to manage. but, the execution of the face location undoubtedly influences the execution of the giant number of makes use of .

Diverse strategies have been proposed for figuring out human faces in pics. They can be ordered into 4 categories. they're records based strategies, encompass based strategies, template based totally strategies and appearance primarily based methods. While used personally, these techniques were now not capable of taking care of an extensive range of problems like role, look and impediments. simultaneously it's miles smarter to paintings with a few progression or parallel techniques. The enormous one of the outward appearance techniques are mainly focused on 5 important lessons, for example angry, glad, neutral, bowled over, sad.

A CNN is one of the system studying or deep mastering algorithms. In this algorithm the input document is the photo and assigning some crucial weights and biases to the unique thing within the particular photo is in a position to distinguish one fee from another cost. The wishes of Pre—processing techniques are virtually very less when in comparison to other picture class strategies and strategies. The motivation of the domain is to empower machines to peer into the sector as humans do. Likewise we use a number of facts for some tasks. as an example, photo and video acknowledgment, photo Processing and Matching pattern, Fingerprint matching and so on.

## Our Objective:

Shopping experience performs a key function in determining enterprise success in a retail context. In truth, it could have an effect on patron buy chance, patron pride and consumer loyalty. The greater the buying experience is able to engage customers in a manner that creates thrilling and noteworthy occasions, the extra it influences purchaser delight and purchaser choice-making to purchase. However, presenting entertainment and organizing humorous and innovative activities aren't enough to make certain best consumers enjoy (CX). corporations have to control all of the clues they may be sending to clients consistent with a properly-conceived and comprehensive CX approach. This first off requires the know-how of how humans behave and experience in each touchpoint that represents the client adventure. This will be carried out via defining a new device able to tune the client's emotional nation and ship an update to the retailers about the facial response of the patron on seeing diverse merchandise. To this give up, this assignment proposes a tool able to reveal the customer’s buying revel in through the analysis of behavioral information extracted from facial expressions.

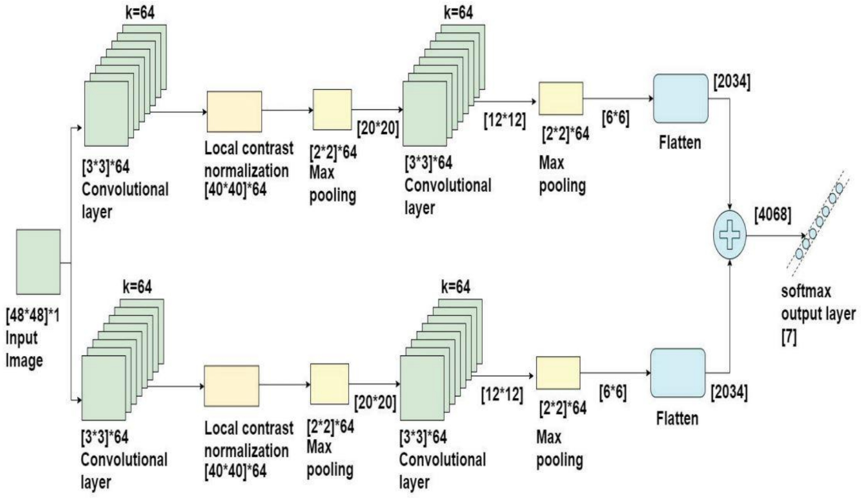
# RELATED WORK

Specific techniques are used for facial features reputation, each of which include different methodologies. Dividing the face into separate movement units or maintaining it as a whole for further processing appears to be the first and the primary difference among the main procedures. In each of those processes, two distinct methodologies, specifically the ‘Geometric based’ and the ‘look-based totally’ parameterizations, may be used. making use of the whole frontal face photograph and processing it as a way to grow to be with the classifications of 6 commonplace facial features prototypes: disgust, worry, pleasure, marvel, sadness and anger; outlines the first approach. here, it's assumed that each of the above cited feelings have function expressions on face and that’s why reputation of them is important and sufficient. instead of the use of the face pix as an entire, dividing them into some sub-sections for similarly processing paperwork up the main idea of the second approach for facial features evaluation. As expression is greater related with diffused changes of a few discrete functions such as eyes, eyebrows and lip corners; those fine-grained modifications are used for reading automated recognition. There are foremost techniques which might be utilized in each of the above defined processes. Geometric based total Parameterization is an antique manner which consists of tracking and processing the motions of some spots on photograph sequences, first off supplied via Suwa et al to recognize facial expressions. Cohn and Kanade afterward attempted geometrical modeling and tracking of facial capabilities with the aid of claiming that each AU is supplied with a particular set of facial muscle tissue. The dangers of this technique are the contours of those features and additives have to be adjusted manually on this body, the problems of robustness and difficulties come out in cases of pose and illumination adjustments whilst the tracking is carried out on images, as actions & expressions have a tendency to alternate both in morphological and in dynamical senses, it will become tough to estimate widespread parameters for movement and displacement. Therefore, ending up with robust choices for facial actions below these various conditions will become tough. As opposed to monitoring spatial factors and using positioning and movement parameters that vary inside time, color (pixel) statistics of related areas of the face are processed in look based Parameterizations; so as to attain the parameters that are going to form the characteristic vectors. extraordinary capabilities along with Gabor, Haar wavelet coefficients, together with function extraction and choice techniques along with PCA, LDA, and Adaboost are used within this framework. For type problems, algorithms like device gaining knowledge of, Neural community, aid Vector system, Deep mastering, Naive Bayes are used. Raghuvanshi A. et al have built a facial features popularity machine upon recent studies to classify pics of human faces into discrete emotion classes using convolutional neural networks. Alizadeh, Shima, and Azar Fazel have developed a facial expression popularity gadget using Convolutional Neural Networks primarily based on the Torch version.

## PROPOSED SYSTEM

We advocate a Convolutional Neural Networks primarily based real time device so that you can be dedicated in the direction of imparting retailers, live facial response updates or facial comments on the outlets' email. This will help the stores in understanding and determining diverse matters consisting of what type products are favored with the aid of what category of clients, age-smart interest of customers, which merchandise are appreciated and which aren't and in the end what emblem is favored the most by means of maximum of the customers. Furthermore, the system can also help shopkeepers to recognise earlier about people with wrong intentions which include shoplifting. The system may be hooked up and used in local well known shops and also in small stores promoting day by day use products or even such things as clothes, bags, poached meals items.

## SYSTEM DESIGN



# MODULES

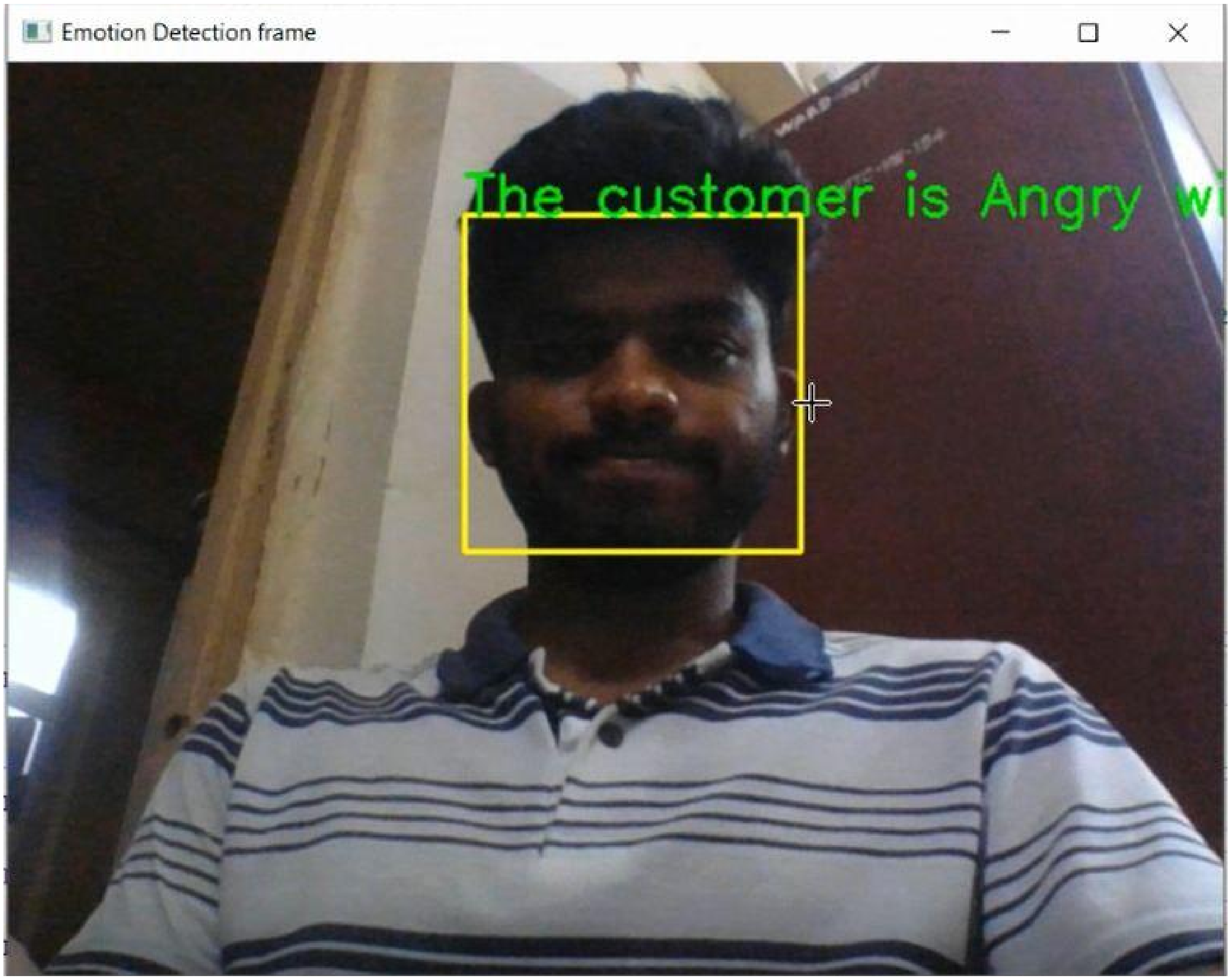
The proposed System has two modules, namely, CNN for Training and Testing data and Real Time Emotion Detection using Webcam as discussed below:

1. **Convolutional Neural Network for Training and Testing data:**
   Model is made by a 30 layered CNN Algorithm on the dataset taken from Kaggle in Python Language. A total of 80 epochs are taken to build the model. Accuracy achieved is 63%, which is decent considering the previous works done in the same field, as discussed in the Related Works section of the report.

2. **Real Time Emotion Detection:**
   The CNN model is saved and integrated with the webcam to take Real Time input and detect the emotions of the customers in the retail stores to get an idea on what type of products and of which brand is most liked by the customers and hence, increase sales to earn more profits.

# OUTPUT

**Emotion Detected:**



The customer is Angry wi

# CONCLUSION

On this undertaking, the expressions of the faces are successfully diagnosed through processing the dataset that includes various facial expressions that are then coded in python or type. Our proposed architecture is recognizing the emotion of the human face dynamically. Here, the primary parameter taken into consideration is the location of the eyes and the mouth. The emotion is diagnosed consistent with the placement change of eyes and mouth. right here similarly, it additionally sends an email telling the emotional fame of the customers to the retailer. This would assist the stores in knowing and finding out numerous things consisting of what type merchandise are favored with the aid of what class of clients, age-wise interest of customers, which merchandise are preferred and which aren't and sooner or later what emblem is preferred the maximum by using most of the customers. Furthermore, the device can also help shopkeepers to recognise in advance about people with incorrect intentions together with shoplifting. The gadget may be set up and used in local standard shops and additionally in small stores promoting each day use merchandise or maybe things like clothes, baggage, poached meals objects.

# FUTURE WORKS

1. First and foremost, we intend to increase the Accuracy of the model to get more precise and correct results.
2. We will also work to integrate voice recognition in the system to get more understanding of the person’s emotions and get more outcomes.
3. We will also try to implement the system in E Learning platforms to change the difficulty level of practice questions for school students based on their facial expressions while solving them.

# REFERENCES

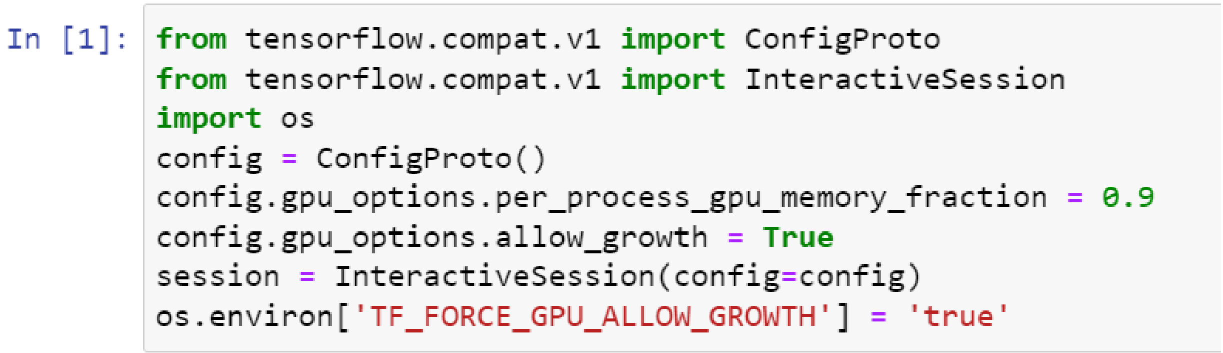
1. Siyue Xie and Haifeng Hu, "Facial Expression Recognition Using Hierarchical Features with Deep Comprehensive Multi-Patches Aggregation Convolutional Neural Networks" IEEE Transactions on Multimedia, 2019.
2. Stefanos Eleftheriadis, Ognjen Rudovic and Maja Pantic, "Discriminative Shared Gaussian Processes for Multiview and ViewInvariant Facial Expression Recognition", IEEE Transactions on Image Processing, Vol – 24, Jan 2019.
3. Chao Qi, Min Li, Qiushi Wang, Huiquan Zhang, Jinling Xing, Zhifan Gao and Huailing Zhang, "Facial Expressions Recognition Based on Cognition and Mapped Binary Patterns", IEEE Access, 2018.
4. Mohammad Reza Mohammadi, Emad Fatemizadeh, and Mohammad H. Mahoor, "Intensity Estimation of Spontaneous Facial Action Units Based on Their Sparsity Properties", IEEE TRANSACTIONS ON CYBERNETICS, 2018.
5. Xi Yin and Xiaoming Liu, "Multi-Task Convolutional Neural Network for Pose-Invariant Face Recognition" IEEE Transactions on Image Processing, 2018.
6. MH Siddiqi, R Ali, AM Khan, "Human Facial Expression Recognition using Stepwise Linear Discriminant Analysis and Hidden Conditional Random Fieldsy" IEEE Transactions, 2017.
7. Q Mao, Q Rao, Y Yu, M Dong "Hierarchical Bayesian Theme Models for Multi-pose Facial Expression Recognition" IEEE Transactions, 2017.
8. Mao, Q., Rao, Q., Yu, Y., & Dong, M. "Hierarchical Bayesian Theme Models For Multipose Facial Expression Recognition." IEEE Transactions On Multimedia Vol. 19, 2017.
9. "Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation". DeepLearning 0.1. LISA Lab.
10. Suwa, M.; Sugie N. and Fujimora K. A Preliminary Note on Pattern Recognition of Human Emotional Expression, Proc. International Joint Conf, Pattern Recognition
11. Raghuvanshi, Arushi, and Vivek Choksi. "Facial Expression Recognition with Convolutional Neural Networks." Stanford University
12. Alizadeh, Shima, and Azar Fazel. "Convolutional Neural Networks for Facial Expression Recognition." Stanford University

13. Seung Ho Lee, Konstantinos N. (Kostas) Plataniotis and Yong Man Ro, “Intra-Class Variation Reduction Using Training Expression Images for Sparse Representation Based Facial Expression Recognition”, IEEE Transactions on Affective Computing, Vol – 5,

14. Simonyan, Karen, and Andrew Zisserman. ”Very deep convolutional networks for large-scale image recognition.” arXiv preprint arXiv:1409.1556

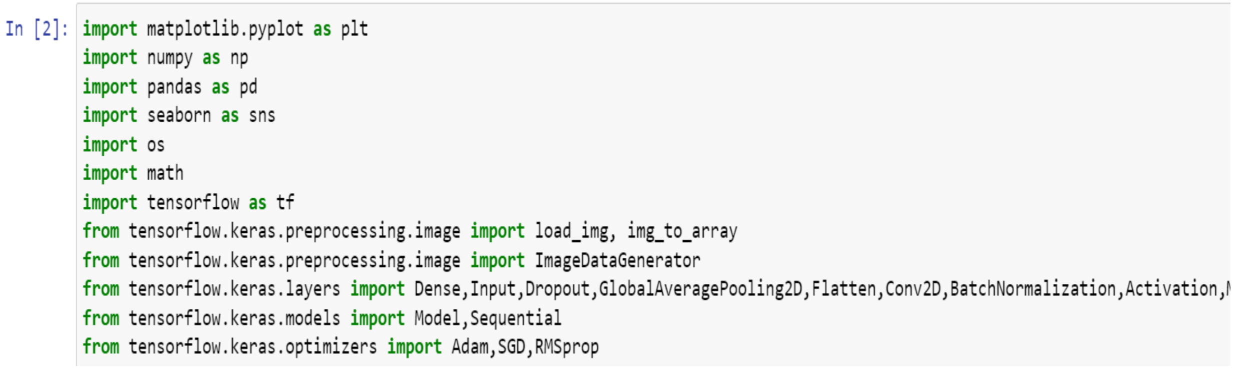
15. LeCun, Yann. "LeNet-5, convolutional neural networks".

# COMPLETE PROGRAM



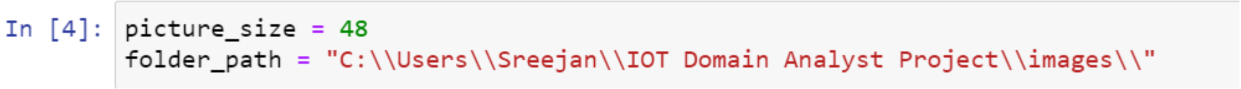
In [1]:

```
from tensorflow.compat.v1 import ConfigProto
from tensorflow.compat.v1 import InteractiveSession
import os
config = ConfigProto()
config.gpu_options.per_process_gpu_memory_fraction = 0.9
config.gpu_options.allow_growth = True
session = InteractiveSession(config=config)
os.environ['TF_FORCE_GPU_ALLOW_GROWTH'] = 'true'
```



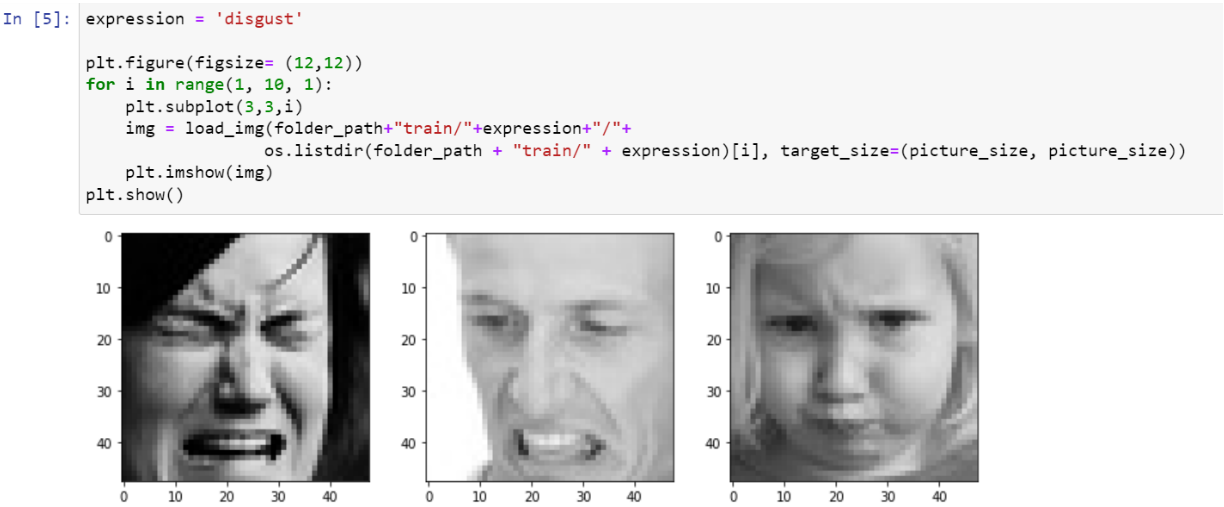
In [2]:

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import os
import math
import tensorflow as tf
from tensorflow.keras.preprocessing.image import load_img, img_to_array
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import Dense,Input,Dropout,GlobalAveragePooling2D,Flatten,Conv2D,BatchNormalization,Activation,
from tensorflow.keras.models import Model,Sequential
from tensorflow.keras.optimizers import Adam,SGD,RMSprop
```

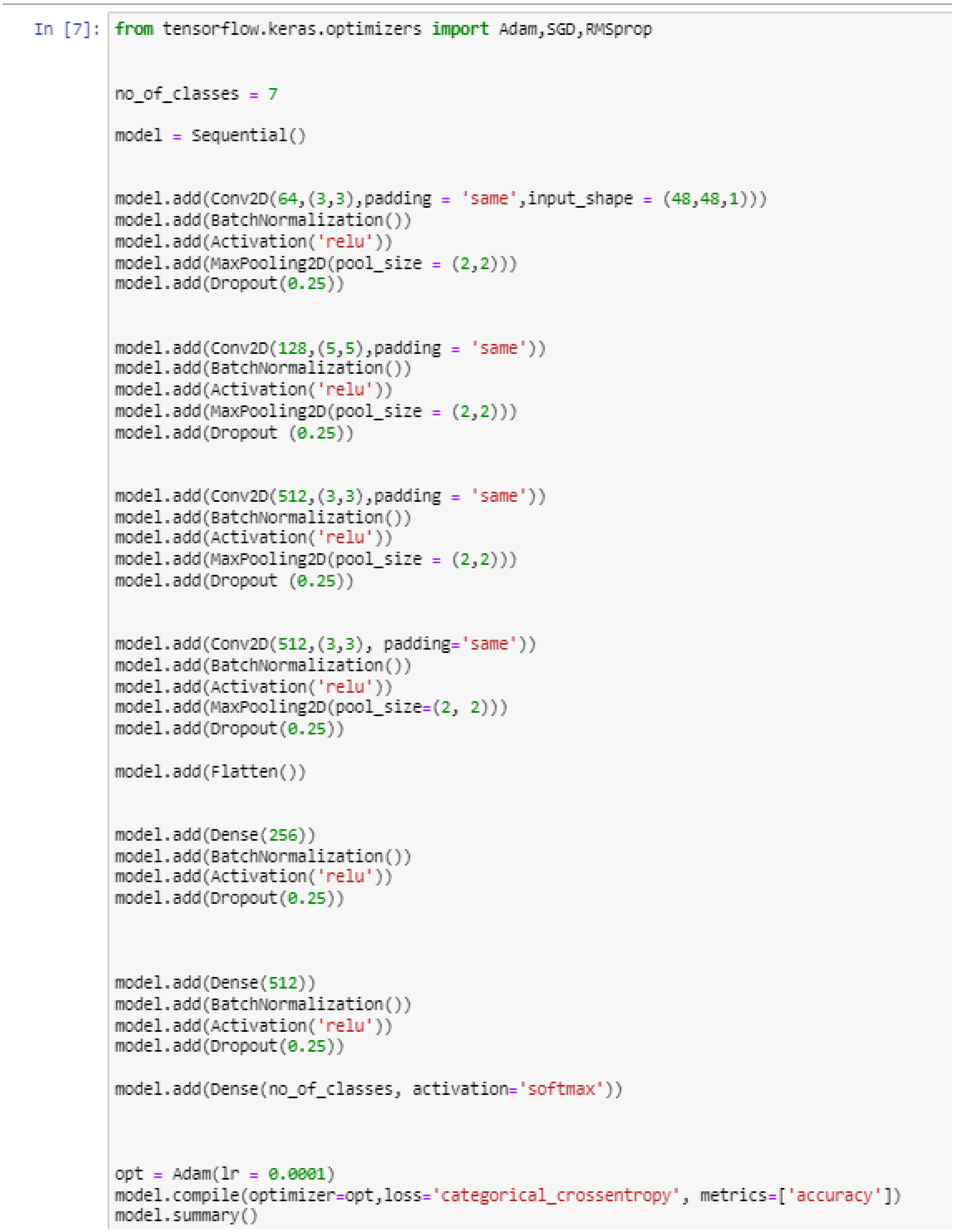


In [4]:

```
picture_size = 48
folder_path = "C:\\Users\\Sreejan\\IOT Domain Analyst Project\\images\\"
```

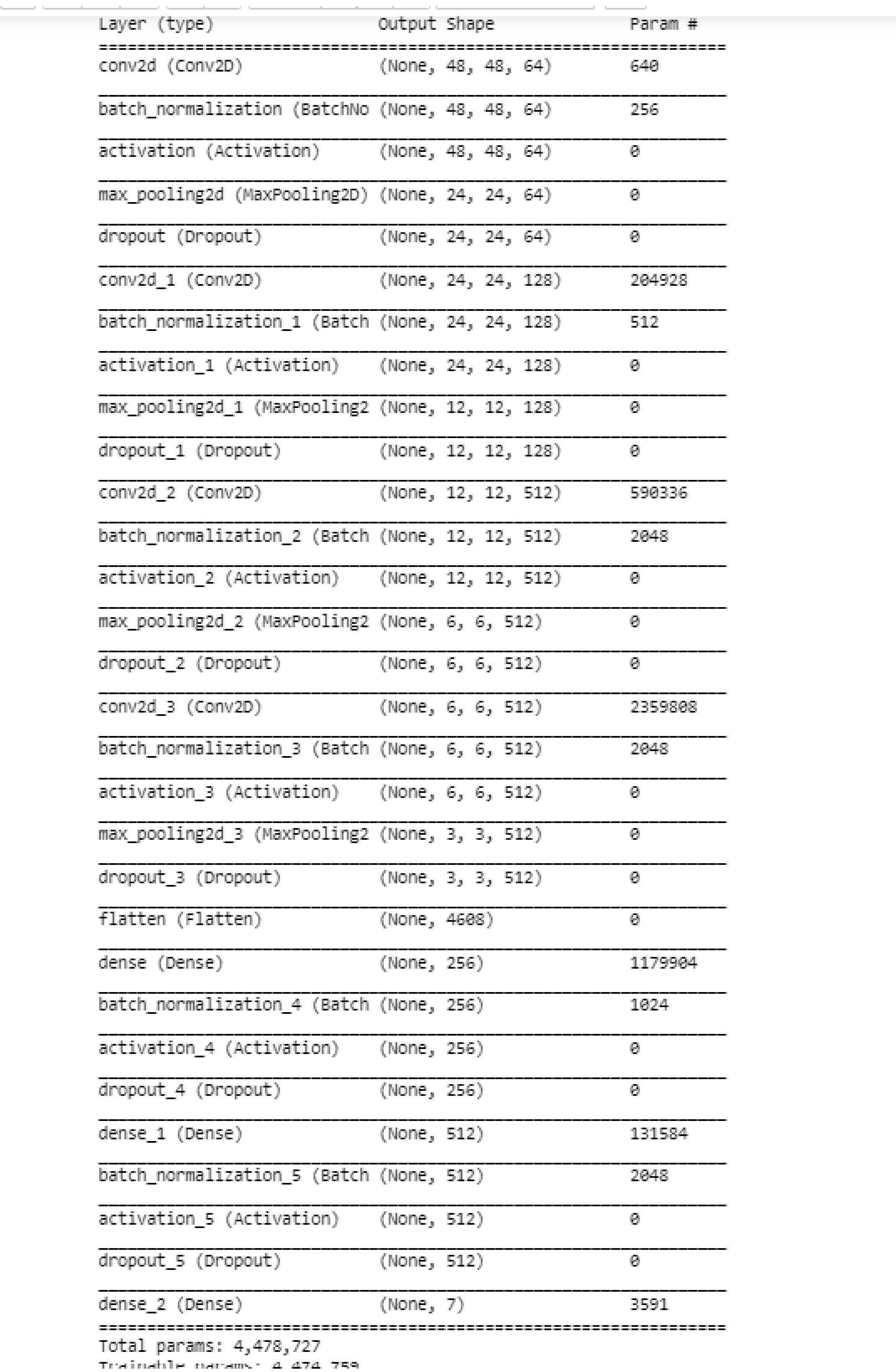


```
In [5]: expression = 'disgust'

        plt.figure(figsize= (12,12))
        for i in range(1, 10, 1):
            plt.subplot(3,3,i)
            img = load_img(folder_path+"train/"+expression+"/"+
                          os.listdir(folder_path + "train/" + expression)[i], target_size=(picture_size, picture_size))
            plt.imshow(img)
        plt.show()
```

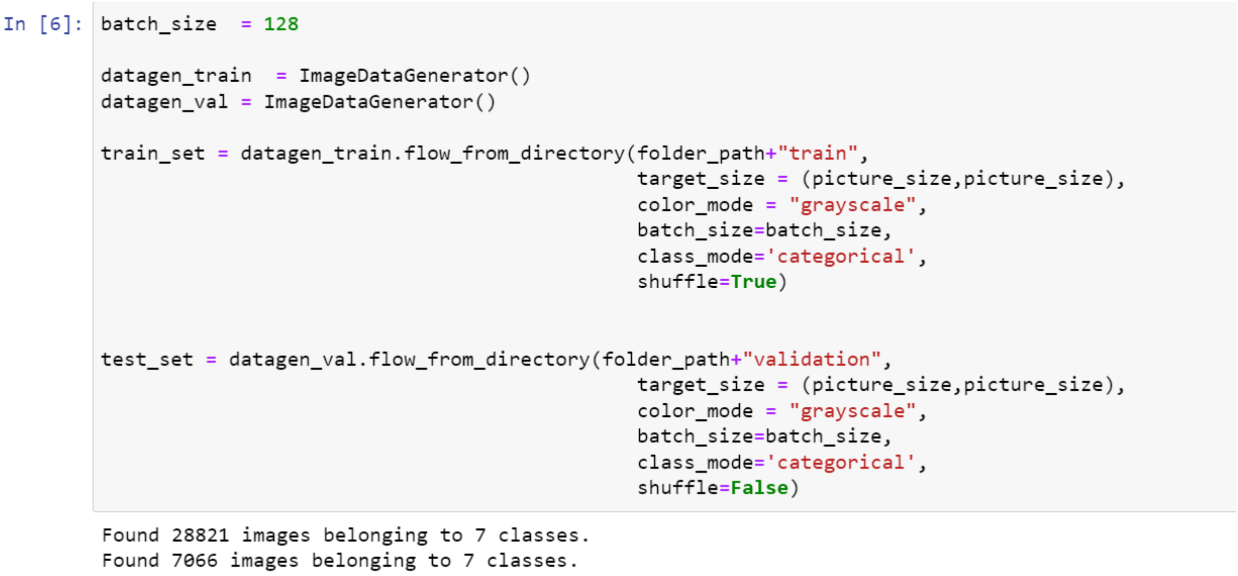


In [7]:

```
from tensorflow.keras.optimizers import Adam,SGD,RMSprop


no_of_classes = 7

model = Sequential()


model.add(Conv2D(64,(3,3),padding = 'same',input_shape = (48,48,1)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
model.add(Dropout(0.25))


model.add(Conv2D(128,(5,5),padding = 'same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
model.add(Dropout (0.25))


model.add(Conv2D(512,(3,3),padding = 'same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size = (2,2)))
model.add(Dropout (0.25))


model.add(Conv2D(512,(3,3), padding='same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

model.add(Flatten())


model.add(Dense(256))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(Dropout(0.25))



model.add(Dense(512))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(Dropout(0.25))

model.add(Dense(no_of_classes, activation='softmax'))



opt = Adam(lr = 0.0001)
model.compile(optimizer=opt,loss='categorical_crossentropy', metrics=['accuracy'])
model.summary()
```

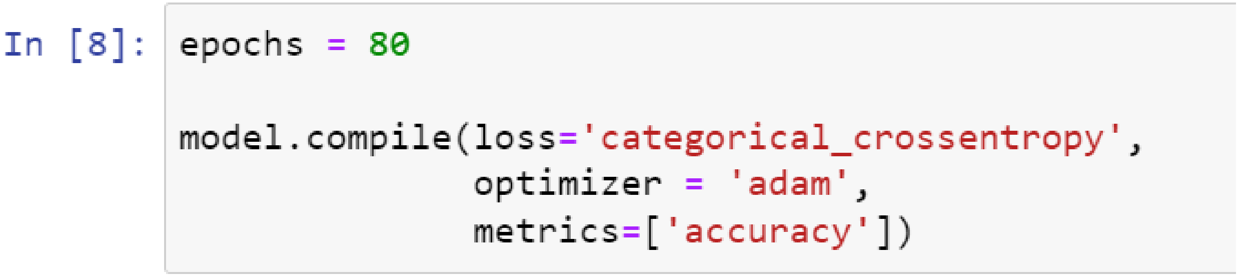


```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 48, 48, 64)        640
_________________________________________________________________
batch_normalization (BatchNo (None, 48, 48, 64)        256
_________________________________________________________________
activation (Activation)      (None, 48, 48, 64)        0
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 24, 24, 64)        0
_________________________________________________________________
dropout (Dropout)            (None, 24, 24, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 24, 24, 128)       204928
_________________________________________________________________
batch_normalization_1 (Batch (None, 24, 24, 128)       512
_________________________________________________________________
activation_1 (Activation)    (None, 24, 24, 128)       0
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 12, 12, 128)       0
_________________________________________________________________
dropout_1 (Dropout)          (None, 12, 12, 128)       0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 12, 12, 512)       590336
_________________________________________________________________
batch_normalization_2 (Batch (None, 12, 12, 512)       2048
_________________________________________________________________
activation_2 (Activation)    (None, 12, 12, 512)       0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 6, 6, 512)         0
_________________________________________________________________
dropout_2 (Dropout)          (None, 6, 6, 512)         0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 6, 6, 512)         2359808
_________________________________________________________________
batch_normalization_3 (Batch (None, 6, 6, 512)         2048
_________________________________________________________________
activation_3 (Activation)    (None, 6, 6, 512)         0
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 3, 3, 512)         0
_________________________________________________________________
dropout_3 (Dropout)          (None, 3, 3, 512)         0
_________________________________________________________________
flatten (Flatten)            (None, 4608)              0
_________________________________________________________________
dense (Dense)                (None, 256)               1179904
_________________________________________________________________
batch_normalization_4 (Batch (None, 256)               1024
_________________________________________________________________
activation_4 (Activation)    (None, 256)               0
_________________________________________________________________
dropout_4 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_1 (Dense)              (None, 512)               131584
_________________________________________________________________
batch_normalization_5 (Batch (None, 512)               2048
_________________________________________________________________
activation_5 (Activation)    (None, 512)               0
_________________________________________________________________
dropout_5 (Dropout)          (None, 512)               0
_________________________________________________________________
dense_2 (Dense)              (None, 7)                 3591
=================================================================
Total params: 4,478,727
Trainable params: 4,474,759
```

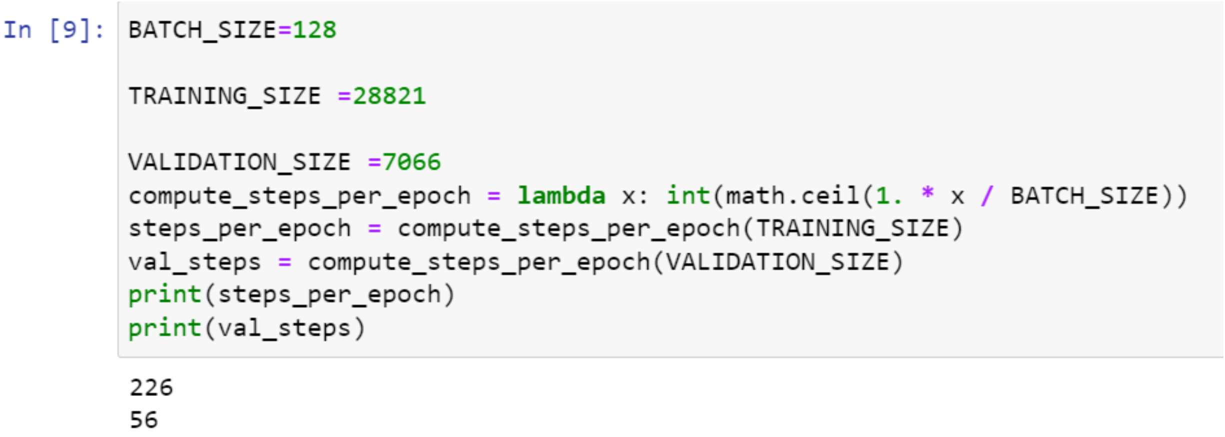


```
In [6]: batch_size  = 128

datagen_train  = ImageDataGenerator()
datagen_val = ImageDataGenerator()

train_set = datagen_train.flow_from_directory(folder_path+"train",
                                              target_size = (picture_size,picture_size),
                                              color_mode = "grayscale",
                                              batch_size=batch_size,
                                              class_mode='categorical',
                                              shuffle=True)


test_set = datagen_val.flow_from_directory(folder_path+"validation",
                                              target_size = (picture_size,picture_size),
                                              color_mode = "grayscale",
                                              batch_size=batch_size,
                                              class_mode='categorical',
                                              shuffle=False)
```

```
Found 28821 images belonging to 7 classes.
Found 7066 images belonging to 7 classes.
```

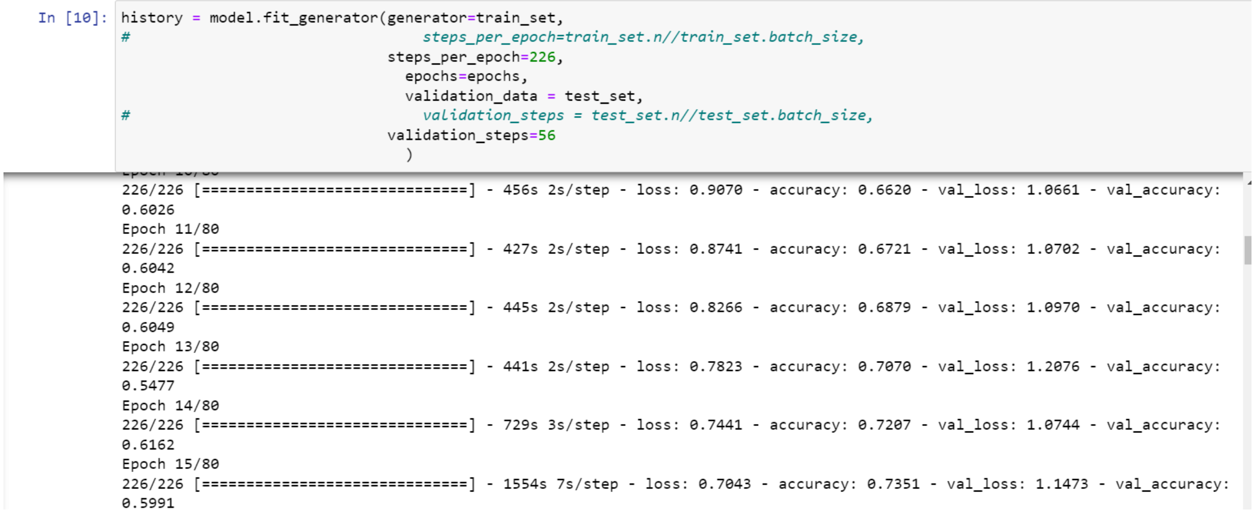


```
In [8]: epochs = 80

model.compile(loss='categorical_crossentropy',
              optimizer = 'adam',
              metrics=['accuracy'])
```



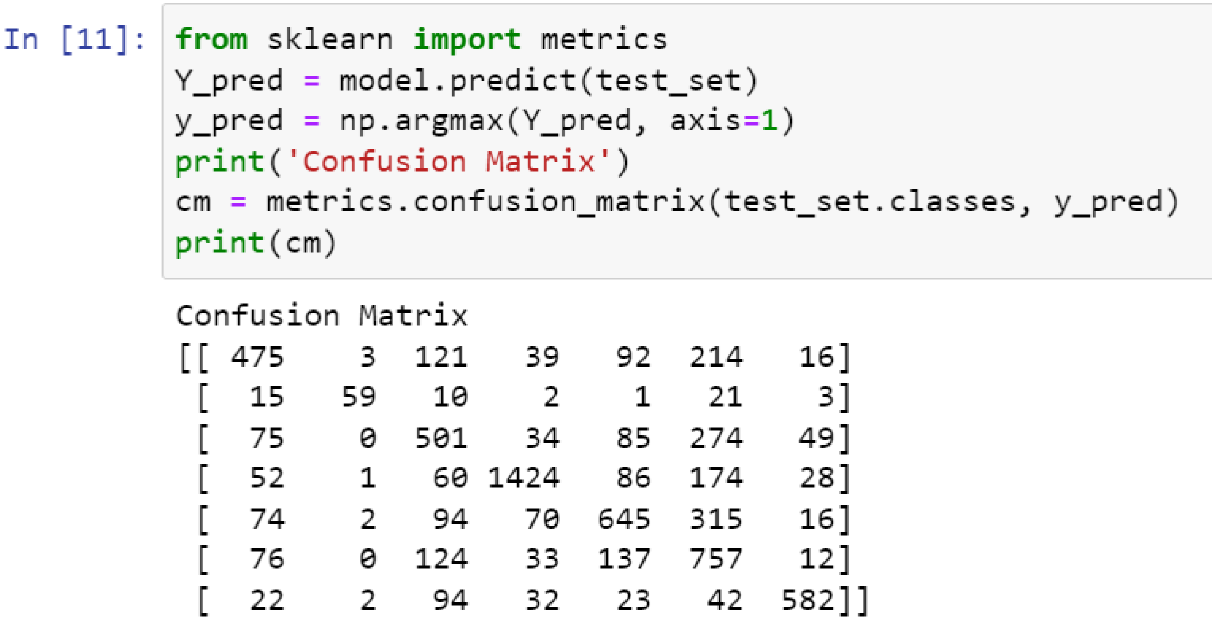
```
In [9]: BATCH_SIZE=128

TRAINING_SIZE =28821

VALIDATION_SIZE =7066
compute_steps_per_epoch = lambda x: int(math.ceil(1. * x / BATCH_SIZE))
steps_per_epoch = compute_steps_per_epoch(TRAINING_SIZE)
val_steps = compute_steps_per_epoch(VALIDATION_SIZE)
print(steps_per_epoch)
print(val_steps)
```

```
226
56
```



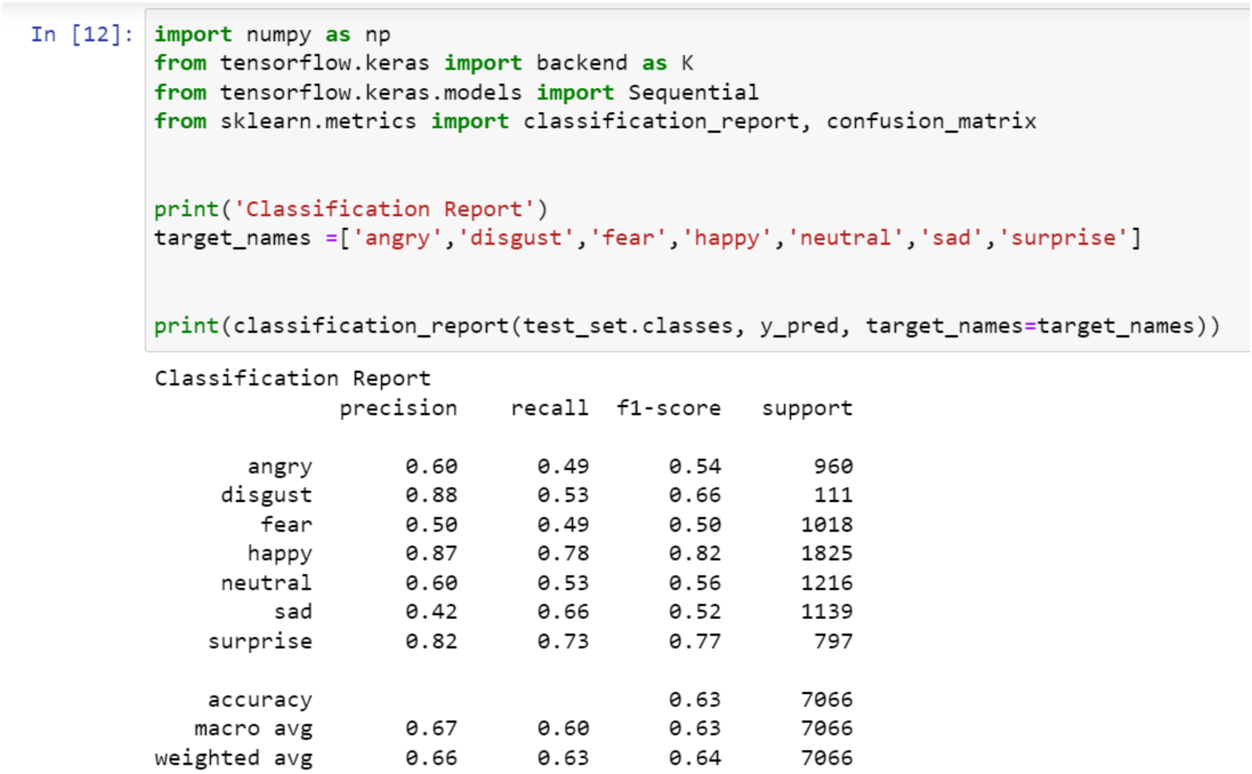
In [10]:
```
history = model.fit_generator(generator=train_set,
#                                 steps_per_epoch=train_set.n//train_set.batch_size,
                              steps_per_epoch=226,
                                epochs=epochs,
                                validation_data = test_set,
#                                 validation_steps = test_set.n//test_set.batch_size,
                              validation_steps=56
                                )
```

```
Epoch 10/80
226/226 [==============================] - 456s 2s/step - loss: 0.9070 - accuracy: 0.6620 - val_loss: 1.0661 - val_accuracy:
0.6026
Epoch 11/80
226/226 [==============================] - 427s 2s/step - loss: 0.8741 - accuracy: 0.6721 - val_loss: 1.0702 - val_accuracy:
0.6042
Epoch 12/80
226/226 [==============================] - 445s 2s/step - loss: 0.8266 - accuracy: 0.6879 - val_loss: 1.0970 - val_accuracy:
0.6049
Epoch 13/80
226/226 [==============================] - 441s 2s/step - loss: 0.7823 - accuracy: 0.7070 - val_loss: 1.2076 - val_accuracy:
0.5477
Epoch 14/80
226/226 [==============================] - 729s 3s/step - loss: 0.7441 - accuracy: 0.7207 - val_loss: 1.0744 - val_accuracy:
0.6162
Epoch 15/80
226/226 [==============================] - 1554s 7s/step - loss: 0.7043 - accuracy: 0.7351 - val_loss: 1.1473 - val_accuracy:
0.5991
```



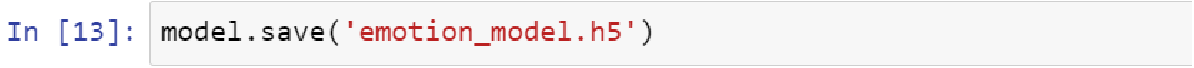
In [11]:
```
from sklearn import metrics
Y_pred = model.predict(test_set)
y_pred = np.argmax(Y_pred, axis=1)
print('Confusion Matrix')
cm = metrics.confusion_matrix(test_set.classes, y_pred)
print(cm)
```

```
Confusion Matrix
[[ 475    3  121   39   92  214   16]
 [  15   59   10    2    1   21    3]
 [  75    0  501   34   85  274   49]
 [  52    1   60 1424   86  174   28]
 [  74    2   94   70  645  315   16]
 [  76    0  124   33  137  757   12]
 [  22    2   94   32   23   42  582]]
```

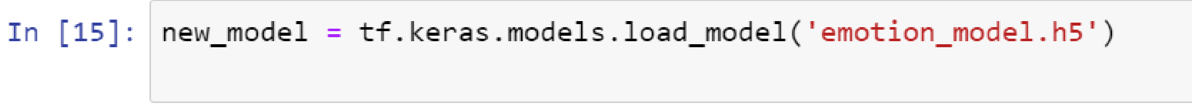


```
In [12]: import numpy as np
         from tensorflow.keras import backend as K
         from tensorflow.keras.models import Sequential
         from sklearn.metrics import classification_report, confusion_matrix


         print('Classification Report')
         target_names =['angry','disgust','fear','happy','neutral','sad','surprise']


         print(classification_report(test_set.classes, y_pred, target_names=target_names))
```

```
Classification Report
              precision    recall  f1-score   support

       angry       0.60      0.49      0.54       960
     disgust       0.88      0.53      0.66       111
        fear       0.50      0.49      0.50      1018
       happy       0.87      0.78      0.82      1825
     neutral       0.60      0.53      0.56      1216
         sad       0.42      0.66      0.52      1139
    surprise       0.82      0.73      0.77       797

    accuracy                           0.63      7066
   macro avg       0.67      0.60      0.63      7066
weighted avg       0.66      0.63      0.64      7066
```

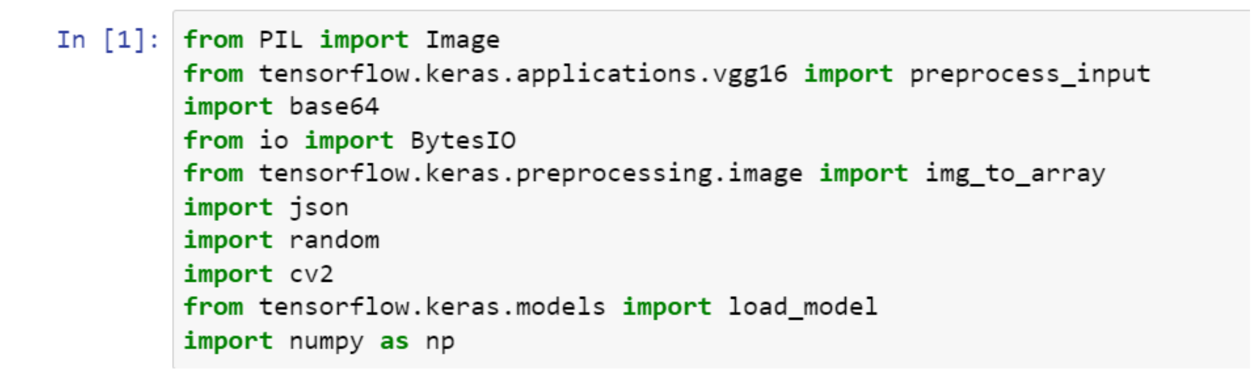


```
In [13]: model.save('emotion_model.h5')
```

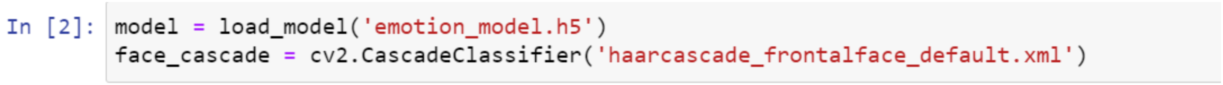


```
In [15]: new_model = tf.keras.models.load_model('emotion_model.h5')
```

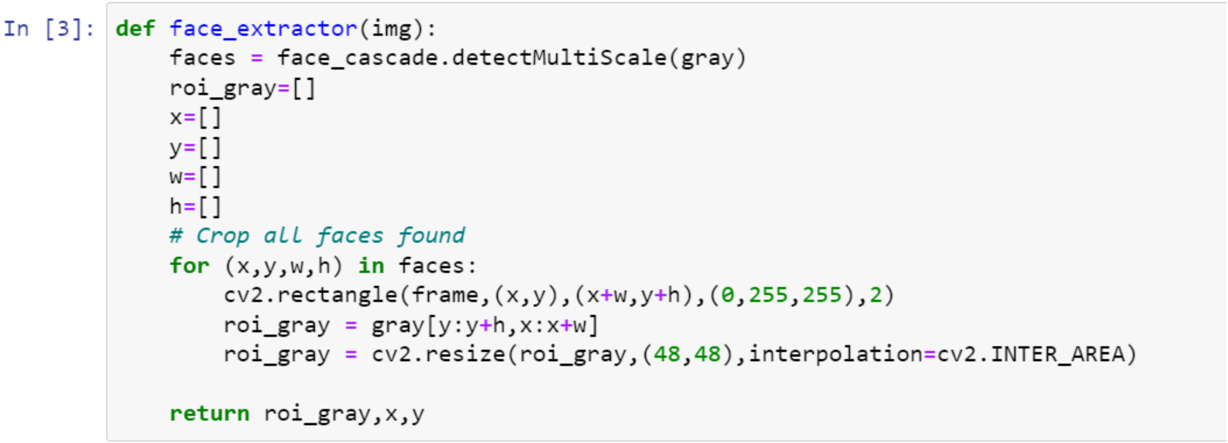
## Emotion detection from trained model



```
In [1]: from PIL import Image
        from tensorflow.keras.applications.vgg16 import preprocess_input
        import base64
        from io import BytesIO
        from tensorflow.keras.preprocessing.image import img_to_array
        import json
        import random
        import cv2
        from tensorflow.keras.models import load_model
        import numpy as np
```

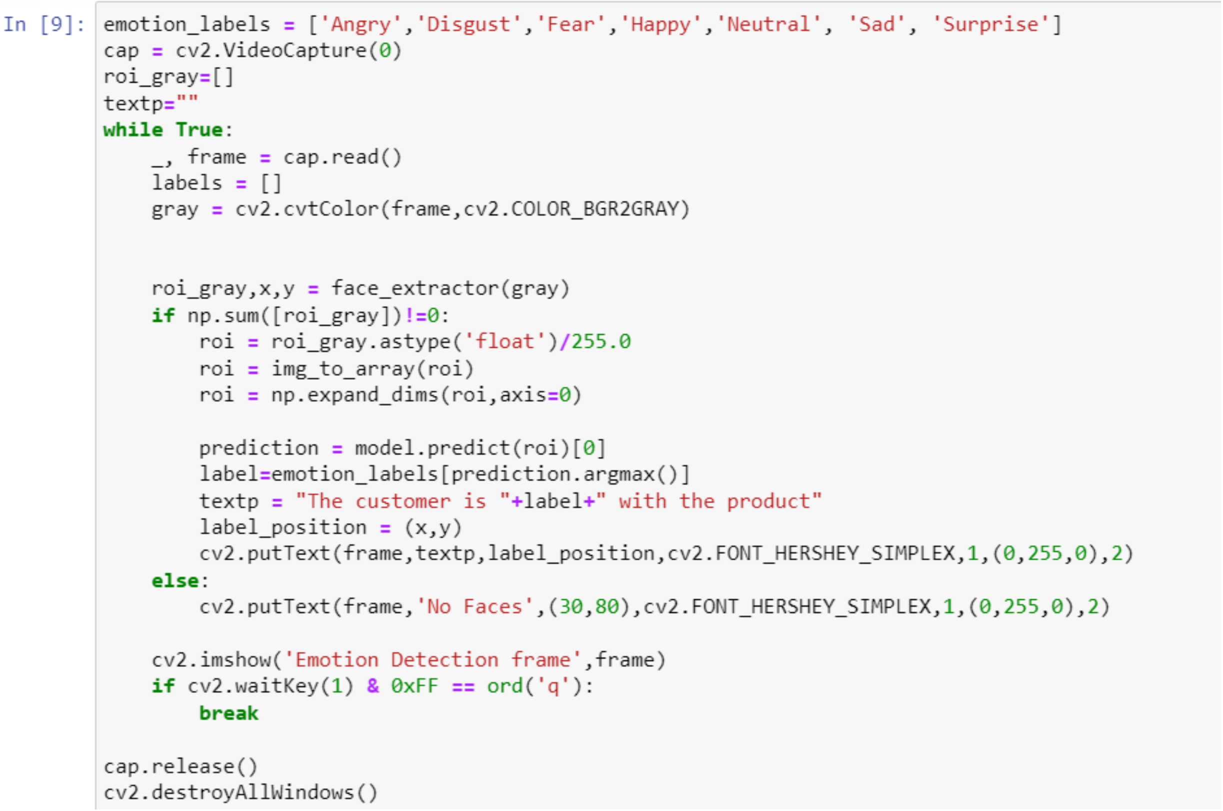


```
In [2]: model = load_model('emotion_model.h5')
        face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
```



In [3]:

```
def face_extractor(img):
    faces = face_cascade.detectMultiScale(gray)
    roi_gray=[]
    x=[]
    y=[]
    w=[]
    h=[]
    # Crop all faces found
    for (x,y,w,h) in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,255),2)
        roi_gray = gray[y:y+h,x:x+w]
        roi_gray = cv2.resize(roi_gray,(48,48),interpolation=cv2.INTER_AREA)

    return roi_gray,x,y
```



In [9]:

```
emotion_labels = ['Angry','Disgust','Fear','Happy','Neutral', 'Sad', 'Surprise']
cap = cv2.VideoCapture(0)
roi_gray=[]
textp=""
while True:
    _, frame = cap.read()
    labels = []
    gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)


    roi_gray,x,y = face_extractor(gray)
    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)

        prediction = model.predict(roi)[0]
        label=emotion_labels[prediction.argmax()]
        textp = "The customer is "+label+" with the product"
        label_position = (x,y)
        cv2.putText(frame,textp,label_position,cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),2)
    else:
        cv2.putText(frame,'No Faces',(30,80),cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),2)

    cv2.imshow('Emotion Detection frame',frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

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