Artificial Intelligence

INT-404

Emotion Recognition

Under the guidance of

Mr. DIPEN SAINI

Assistant Professor of CSE Dept.

Work Division

Name	Roll no.	Work
Amit Kumar	56	Building the Neural network for Processing our data.
Himanshu Kumar	06	Training the model using keras
Etesh Agarwal	74	Defining the propertied for training data set and validation data set.
Ankit Kumar Giri	75	Testing the trained model using webcam. i.e facial recognition.

Lovely Professional University, Phagwara, Punjab

CERTIFICATE

This is to certify that the project work entitled as “EMOTION RECOGNITION OF FACE ” is being Submitted by
HIMANSHU KUMAR(06),
AMIT KUMAR(56) ,
ETESH AGARWAL(74),
ANKIT GIRI (75),
in the partial fulfillment for the award of the Degree of Bachelor of Technology in “COMPUTER SCIENCE AND ENGINNERING” in the academic during 2019-2020.

Under the esteemed Guidance of

Mr. DIPEN SAINI

Assistant Professor CSE Dept.

Emotion Detection

ABSTRACT

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisherface method. Facial image recognition Eigenface method is based on the reduction of face-dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigen faces was formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image. The area of this project face detection system with face recognition is Image processing. The software requirements for this project is matlab software.

Keywords: face detection, Eigen face, PCA, matlab

Extension: There are vast number of applications from this face detection project, this project can be extended that the various parts in the face can be detect which are in various directions and shapes.

Popular recognition algorithms include:

1. Principal Component Analysis using Eigenfaces, (PCA)
2. Linear Discriminate Analysis,
3. Elastic Bunch Graph Matching using the Fisherface algorithm

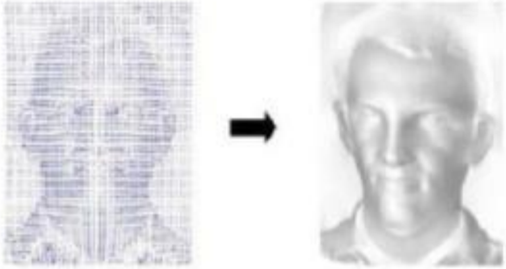


Figure 2 -Photometric stereo image.

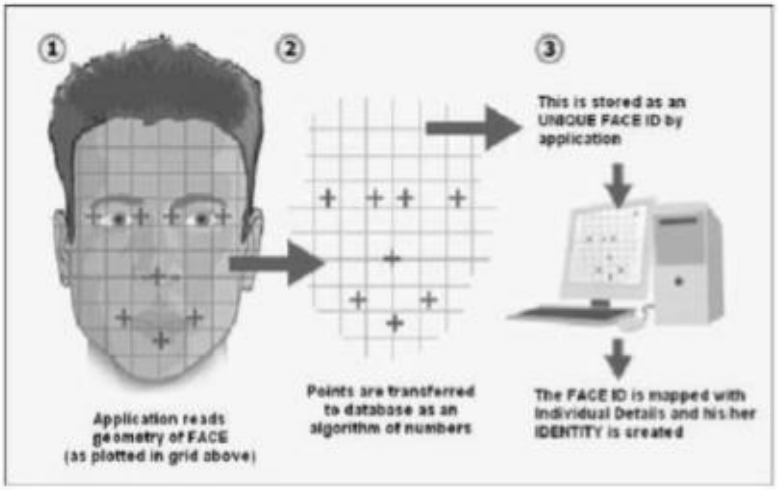

①
Application reads geometry of FACE (as plotted in grid above)
②
Points are transferred to database as an algorithm of numbers
③
This is stored as an UNIQUE FACE ID by application
The FACE ID is mapped with Individual Details and his/her IDENTITY is created

Figure 3 - Geometric facial recognition.

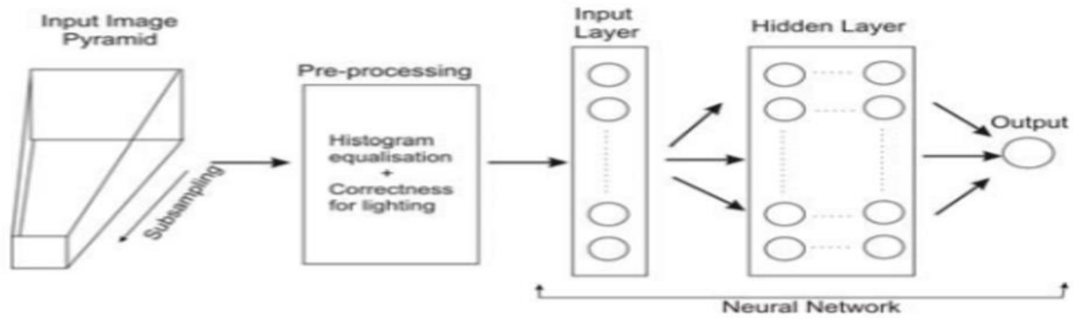
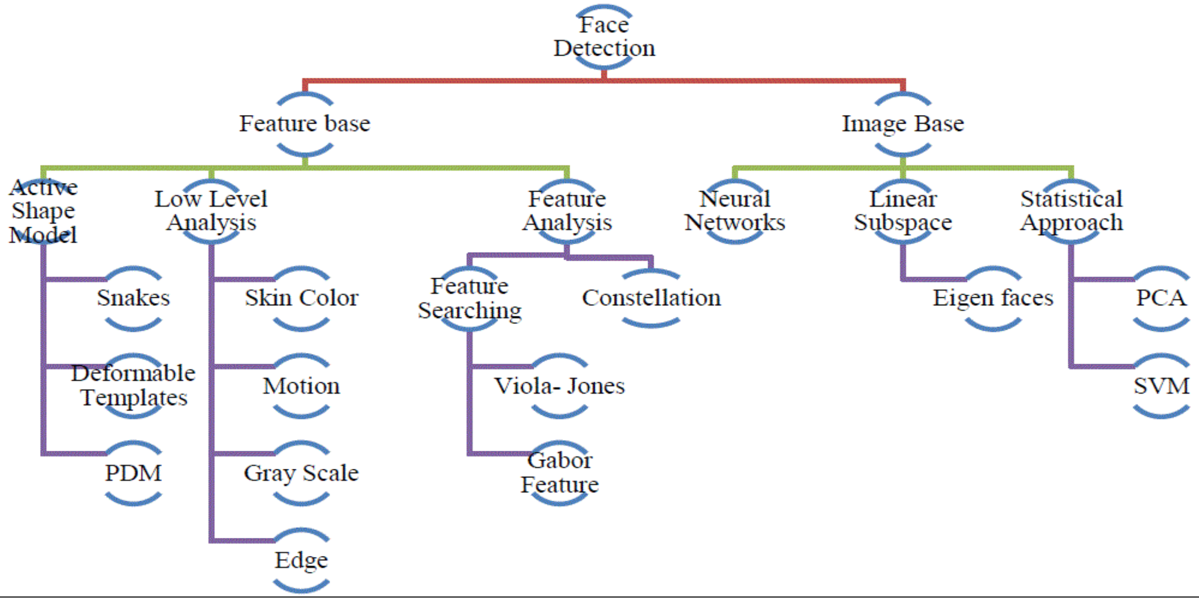


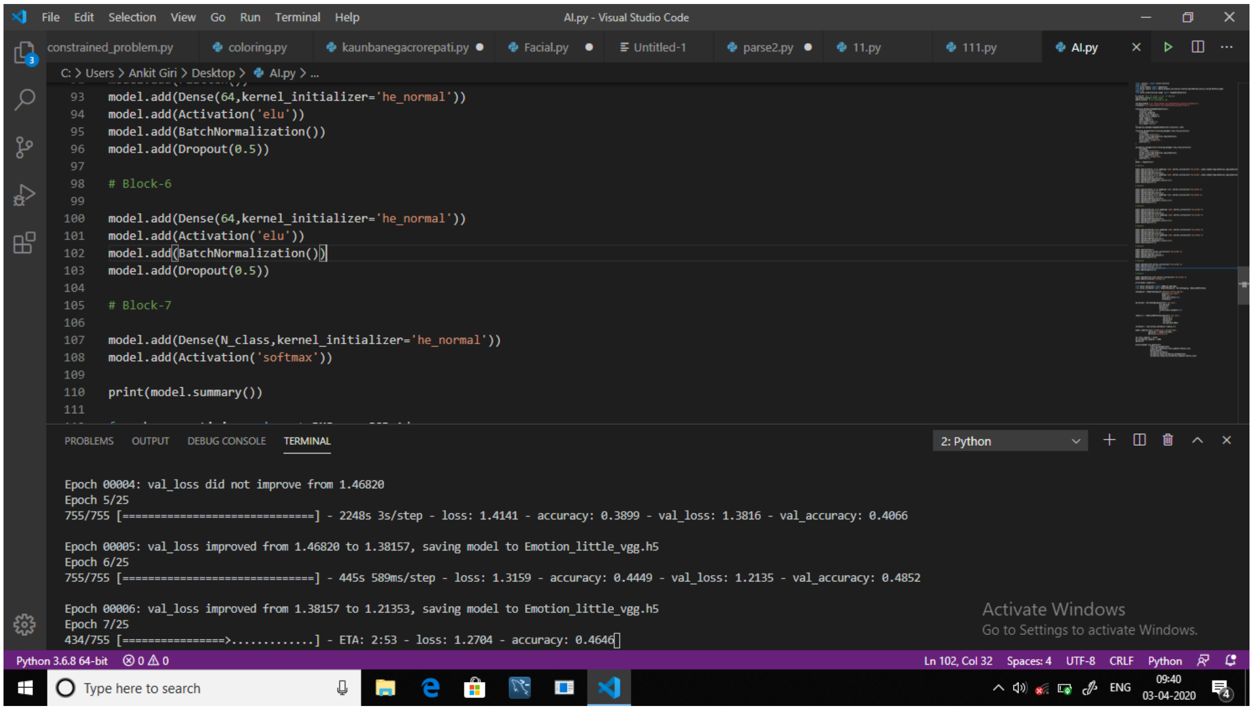
Fig: Face detection algorithm



$$\psi_{u,v}(z) = \frac{\| k_{u,v} \|^2}{\sigma^2} e^{\left(\frac{\|k_{u,v}\|^2\|z\|^2}{2\sigma^2}\right)} \left[e^{i\vec{k}_{u,v}z} - e^{-\frac{\sigma^2}{2}} \right]$$

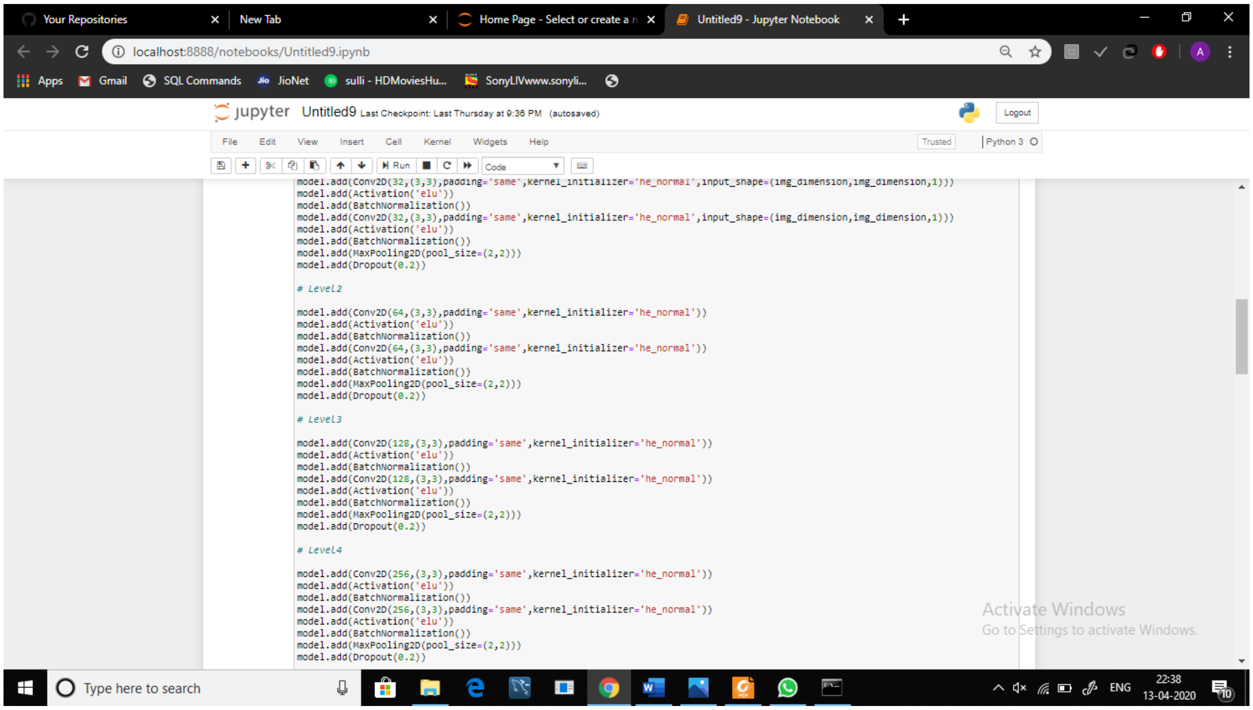
Where

$\phi_u = \frac{u\pi}{8}, \quad \phi_u \in [0,\pi)$ gives the frequency,

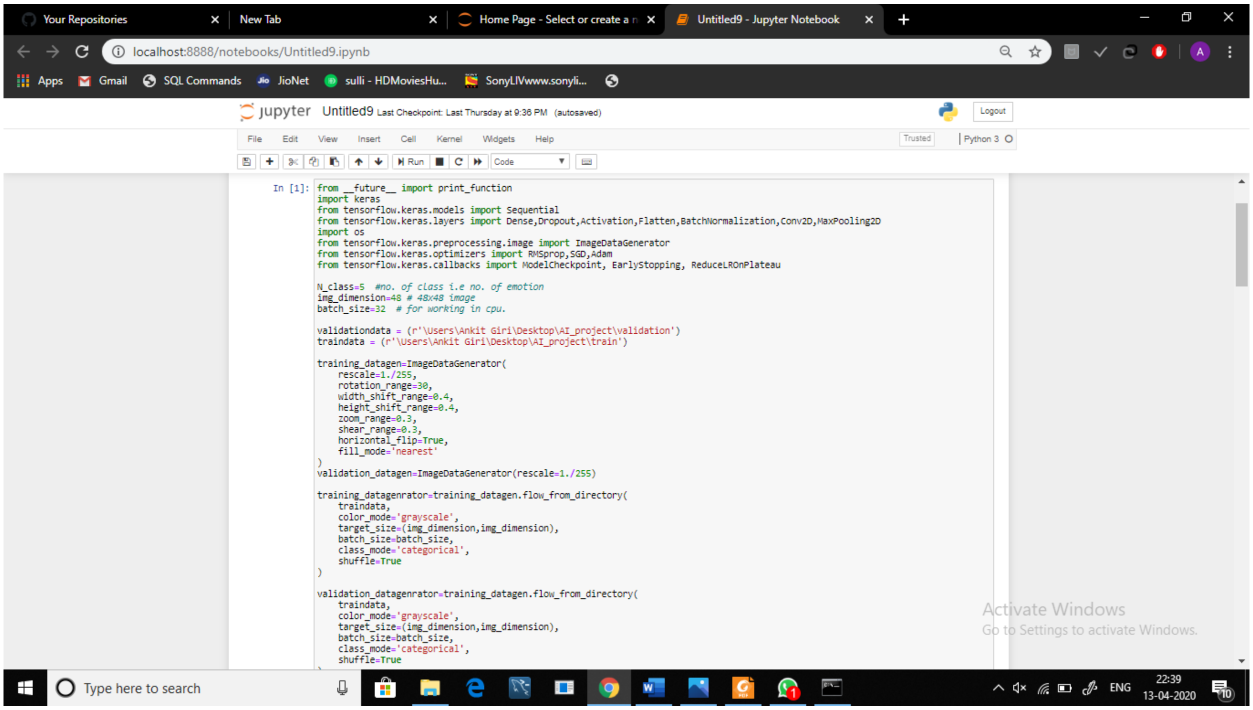


```
model.add(Dense(64,kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))

# Block-6

model.add(Dense(64,kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))

# Block-7

model.add(Dense(N_class,kernel_initializer='he_normal'))
model.add(Activation('softmax'))

print(model.summary())
```

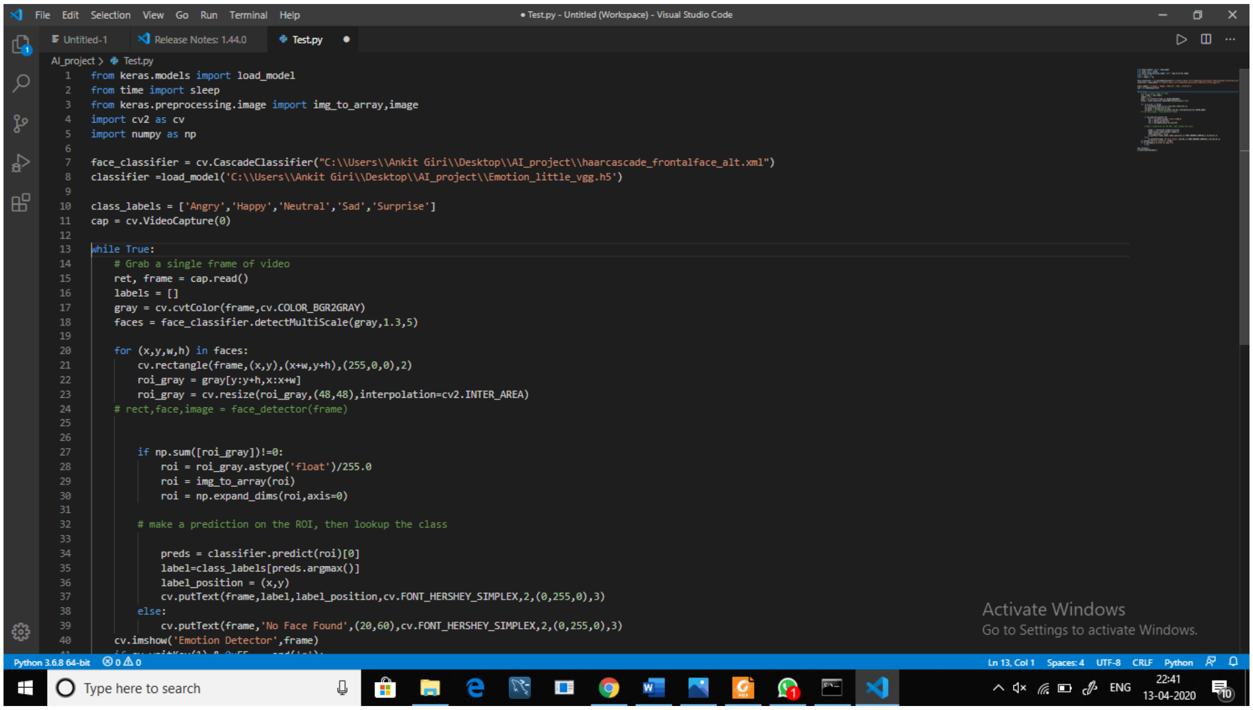
```
Epoch 00004: val_loss did not improve from 1.46820
Epoch 5/25
755/755 [==============================] - 2248s 3s/step - loss: 1.4141 - accuracy: 0.3899 - val_loss: 1.3816 - val_accuracy: 0.4066

Epoch 00005: val_loss improved from 1.46820 to 1.38157, saving model to Emotion_little_vgg.h5
Epoch 6/25
755/755 [==============================] - 445s 589ms/step - loss: 1.3159 - accuracy: 0.4449 - val_loss: 1.2135 - val_accuracy: 0.4852

Epoch 00006: val_loss improved from 1.38157 to 1.21353, saving model to Emotion_little_vgg.h5
Epoch 7/25
434/755 [===============>.............] - ETA: 2:53 - loss: 1.2704 - accuracy: 0.4646
```


Your Repositories
New Tab
Home Page - Select or create a n
Untitled9 - Jupyter Notebook
localhost:8888/notebooks/Untitled9.ipynb
Apps
Gmail
SQL Commands
JioNet
sulli - HDMoviesHu...
SonyLIVwww.sonyli...
jupyter Untitled9 Last Checkpoint: Last Thursday at 9:36 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
22:38
13-04-2020

```
model.add(Conv2D(32,(3,3),padding='same',kernel_initializer='he_normal',input_shape=(img_dimension,img_dimension,1)))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(32,(3,3),padding='same',kernel_initializer='he_normal',input_shape=(img_dimension,img_dimension,1)))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.2))

# Level2

model.add(Conv2D(64,(3,3),padding='same',kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(64,(3,3),padding='same',kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.2))

# Level3

model.add(Conv2D(128,(3,3),padding='same',kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(128,(3,3),padding='same',kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.2))

# Level4

model.add(Conv2D(256,(3,3),padding='same',kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(256,(3,3),padding='same',kernel_initializer='he_normal'))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.2))
```


Your Repositories
New Tab
Home Page - Select or create a n
Untitled9 - Jupyter Notebook
localhost:8888/notebooks/Untitled9.ipynb
jupyter Untitled9 Last Checkpoint: Last Thursday at 9:36 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
22:39
13-04-2020

```python
In [1]: from __future__ import print_function
import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Dropout,Activation,Flatten,BatchNormalization,Conv2D,MaxPooling2D
import os
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.optimizers import RMSprop,SGD,Adam
from tensorflow.keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceLROnPlateau

N_class=5  #no. of class i.e no. of emotion
img_dimension=48 # 48x48 image
batch_size=32  # for working in cpu.

validationdata = (r'\Users\Ankit Giri\Desktop\AI_project\validation')
traindata = (r'\Users\Ankit Giri\Desktop\AI_project\train')

training_datagen=ImageDataGenerator(
    rescale=1./255,
    rotation_range=30,
    width_shift_range=0.4,
    height_shift_range=0.4,
    zoom_range=0.3,
    shear_range=0.3,
    horizontal_flip=True,
    fill_mode='nearest'
)
validation_datagen=ImageDataGenerator(rescale=1./255)

training_datagenrator=training_datagen.flow_from_directory(
    traindata,
    color_mode='grayscale',
    target_size=(img_dimension,img_dimension),
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=True
)

validation_datagenrator=training_datagen.flow_from_directory(
    traindata,
    color_mode='grayscale',
    target_size=(img_dimension,img_dimension),
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=True
```


File Edit Selection View Go Run Terminal Help
• Test.py - Untitled (Workspace) - Visual Studio Code
Untitled-1
Release Notes: 1.44.0
Test.py
AI_project > Test.py
Python 3.6.8 64-bit
Ln 13, Col 1 Spaces: 4 UTF-8 CRLF Python
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
22:41
13-04-2020

```python
from keras.models import load_model
from time import sleep
from keras.preprocessing.image import img_to_array,image
import cv2 as cv
import numpy as np

face_classifier = cv.CascadeClassifier("C:\\Users\\Ankit Giri\\Desktop\\AI_project\\haarcascade_frontalface_alt.xml")
classifier =load_model('C:\\Users\\Ankit Giri\\Desktop\\AI_project\\Emotion_little_vgg.h5')

class_labels = ['Angry','Happy','Neutral','Sad','Surprise']
cap = cv.VideoCapture(0)

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

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


        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]
            label=class_labels[preds.argmax()]
            label_position = (x,y)
            cv.putText(frame,label,label_position,cv.FONT_HERSHEY_SIMPLEX,2,(0,255,0),3)
        else:
            cv.putText(frame,'No Face Found',(20,60),cv.FONT_HERSHEY_SIMPLEX,2,(0,255,0),3)
    cv.imshow('Emotion Detector',frame)
```