jupyter vgg face fayaz Last Checkpoint: 16 hours ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Run Code

In [1]:
```
from keras.applications import VGG16

# VGG16 was designed to work on 224 x 224 pixel input images sizes
img_rows = 224
img_cols = 224

# Re-loads the VGG16 model without the top or FC layers
model = VGG16(weights = 'imagenet',
                 include_top = False,
                 input_shape = (img_rows, img_cols, 3))

# Here we freeze the last 4 layers
# Layers are set to trainable as True by default
for layer in model.layers:
    layer.trainable = False

# Let's print our layers
for (i,layer) in enumerate(model.layers):
    print(str(i) + " "+ layer.__class__.__name__, layer.trainable)
```

```
Using TensorFlow backend.
```

```
0 InputLayer False
1 Conv2D False
2 Conv2D False
3 MaxPooling2D False
4 Conv2D False
```

```
18 MaxPooling2D False
```

n [2]:
```python
def addTopModel(bottom_model, num_classes, D=256):
    """creates the top or head of the model that will be
    placed ontop of the bottom layers"""
    top_model = bottom_model.output
    top_model = Flatten(name = "flatten")(top_model)
    top_model = Dense(D, activation = "relu")(top_model)
    top_model = Dense(D, activation = "relu")(top_model)
    top_model = Dense(D, activation = "relu")(top_model)
    top_model = Dropout(0.3)(top_model)
    top_model = Dense(num_classes, activation = "sigmoid")(top_model)
    return top_model
```

n [3]:
```python
from keras.models import Sequential
from keras.layers import Dense, Dropout, Activation, Flatten
from keras.layers import Conv2D, MaxPooling2D, ZeroPadding2D
from keras.layers.normalization import BatchNormalization
from keras.models import Model

num_classes = 2

FC_Head = addTopModel(model, num_classes)
```

block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
dense_1 (Dense)	(None, 256)	6422784
dense_2 (Dense)	(None, 256)	65792
dense_3 (Dense)	(None, 256)	65792
dropout_1 (Dropout)	(None, 256)	0
dense_4 (Dense)	(None, 2)	514

Total params: 21,269,570
Trainable params: 6,554,882
Non-trainable params: 14,714,688

In [4]:

```
from keras.preprocessing.image import ImageDataGenerator

train_data_dir = 'fayaz/train/'
validation_data_dir = 'fayaz/validation/'


train_datagen = ImageDataGenerator(
      rescale=1./255,
      rotation_range=20,
      width_shift_range=0.2,
      height_shift_range=0.2,
      horizontal_flip=True,
      fill_mode='nearest')

validation_datagen = ImageDataGenerator(rescale=1./255)

# Change the batchsize according to your system RAM
train_batchsize = 32
val_batchsize = 16
img_rows=224
img_cols=224

train_generator = train_datagen.flow_from_directory(
        train_data_dir,
        target_size=(img_rows, img_cols),
        batch_size=train_batchsize,
        class_mode='categorical')

validation_generator = validation_datagen.flow_from_directory(
```

```
Epoch 1/5
25/25 [==============================] - 386s 15s/step - loss: 1.1430 - accuracy: 0.7126 - val_loss: 5.3990e-04 - val_accuracy:
1.0000

Epoch 00001: val_loss improved from inf to 0.00054, saving model to family_vgg.h5
Epoch 2/5
25/25 [==============================] - 371s 15s/step - loss: 0.0064 - accuracy: 1.0000 - val_loss: 1.8340e-04 - val_accuracy:
1.0000

Epoch 00002: val_loss improved from 0.00054 to 0.00018, saving model to family_vgg.h5
Epoch 3/5
25/25 [==============================] - 365s 15s/step - loss: 4.4188e-04 - accuracy: 1.0000 - val_loss: 5.9798e-07 - val_accu
acy: 1.0000

Epoch 00003: val_loss improved from 0.00018 to 0.00000, saving model to family_vgg.h5
Epoch 4/5
25/25 [==============================] - 373s 15s/step - loss: 0.3579 - accuracy: 0.9756 - val_loss: 8.6017e-05 - val_accuracy:
1.0000

Epoch 00004: val_loss did not improve from 0.00000
Epoch 5/5
25/25 [==============================] - 2503s 100s/step - loss: 3.1120e-04 - accuracy: 1.0000 - val_loss: 3.1444e-08 - val_ac
uracy: 1.0000
```

In [7]:
```
train_generator.class_indices
```

Out[7]:
```
{'fayaz': 0, 'sajid': 1}
```

In [11]:
```
import cv2
import numpy as np


face_classifier = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

def face_detector(img, size=0.5):

    # Convert image to grayscale
    gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(gray, 1.3, 5)
    if faces is ():
        return img, []

    for (x,y,w,h) in faces:
```

```
cv2.destroyAllWindows()
```

```
[[0. 1.]]
[[0. 1.]]
[[1.235573e-33 1.000000e+00]]
[[0. 1.]]
[[1.19842185e-26 1.00000000e+00]]
[[9.9998081e-01 2.9885233e-04]]
[[6.6675086e-32 1.0000000e+00]]
[[0. 1.]]
[[2.7838362e-16 1.0000000e+00]]
[[9.561052e-23 1.000000e+00]]
[[0. 1.]]
[[0. 1.]]
[[0. 1.]]
[[0. 1.]]
[[0. 1.]]
[[1.1854787e-38 1.0000000e+00]]
[[6.948093e-26 1.000000e+00]]
[[1.0016725e-37 1.0000000e+00]]
[[1.2228546e-21 1.0000000e+00]]
```

In [ ]:

Desktop/fayaz/ | vgg face fayaz - Jupyter | Untitled2 - Jupyter Notebo

localhost:8888/notebooks/Desktop/tt/vgg%20face%20fayaz.ipynb

jupyter vgg face fayaz Last Checkpoint: 15 hours ago (unsaved changes)

Logout

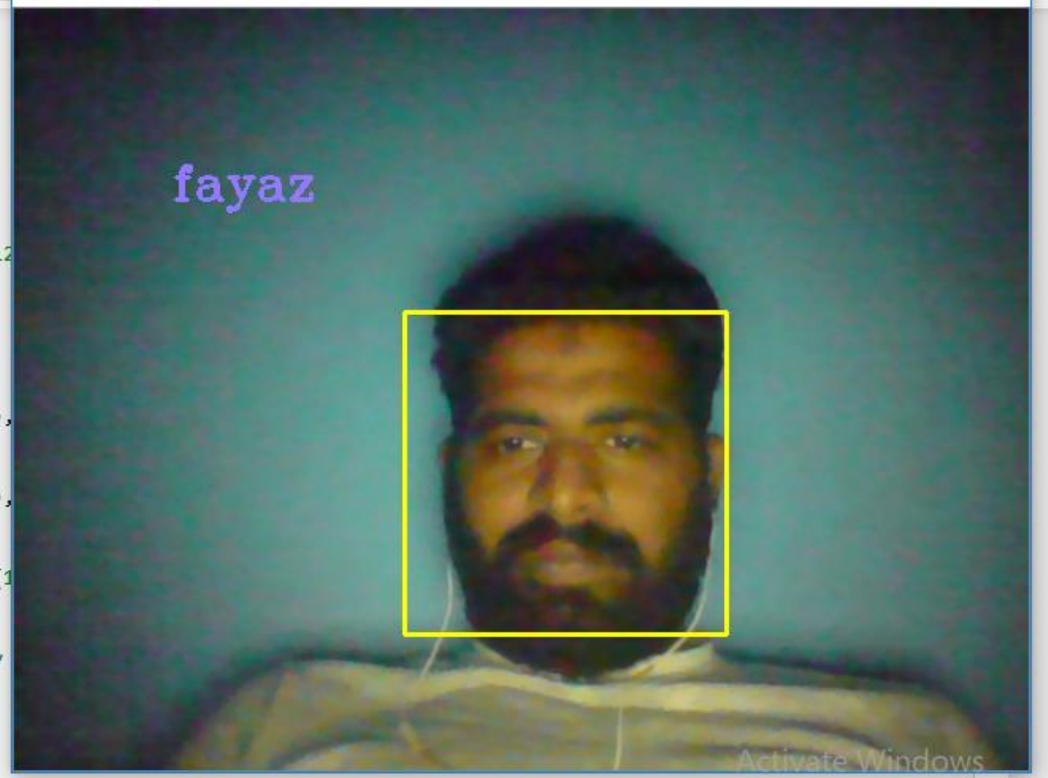
File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

```
while True:

    ret, frame = cap.read()

    image, face = face_detector(frame)

    face=np.array(face)
    face=np.expand_dims(face,axis=0)
    if face.shape==(1,0):
        cv2.putText(image,"I don't know", (100, 12
        cv2.imshow('Face Recognition',image)
    else:
        result=model.predict(face)
        print(result)
        if result[0][0] == 1.0:
            cv2.putText(image,"sajid", (100, 120),
            cv2.imshow('Face Recognition',image)
        elif result[0][0] == 0.0:
            cv2.putText(image,"fayaz", (100, 120),
            cv2.imshow('Face Recognition',image)
        else:
            cv2.putText(image,"Not recognized", (1
            cv2.imshow('Face Recognition',image)

    if cv2.waitKey(1) == 13: #13 is the Enter Key
        break

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

Face Recognition



Activate Windows
Go to Settings to activate Windows.

```
[[0. 1.]]
[[0. 1.]]
```

Type here to search

5:38 AM
5/20/2020

Desktop/fayaz/ | vgg face fayaz - Jupyter | Untitled2 - Jupyter Notebo

localhost:8888/notebooks/Desktop/tt/vgg%20face%20fayaz.ipynb

jupyter vgg face fayaz Last Checkpoint: 15 hours ago (unsaved changes)

Logout

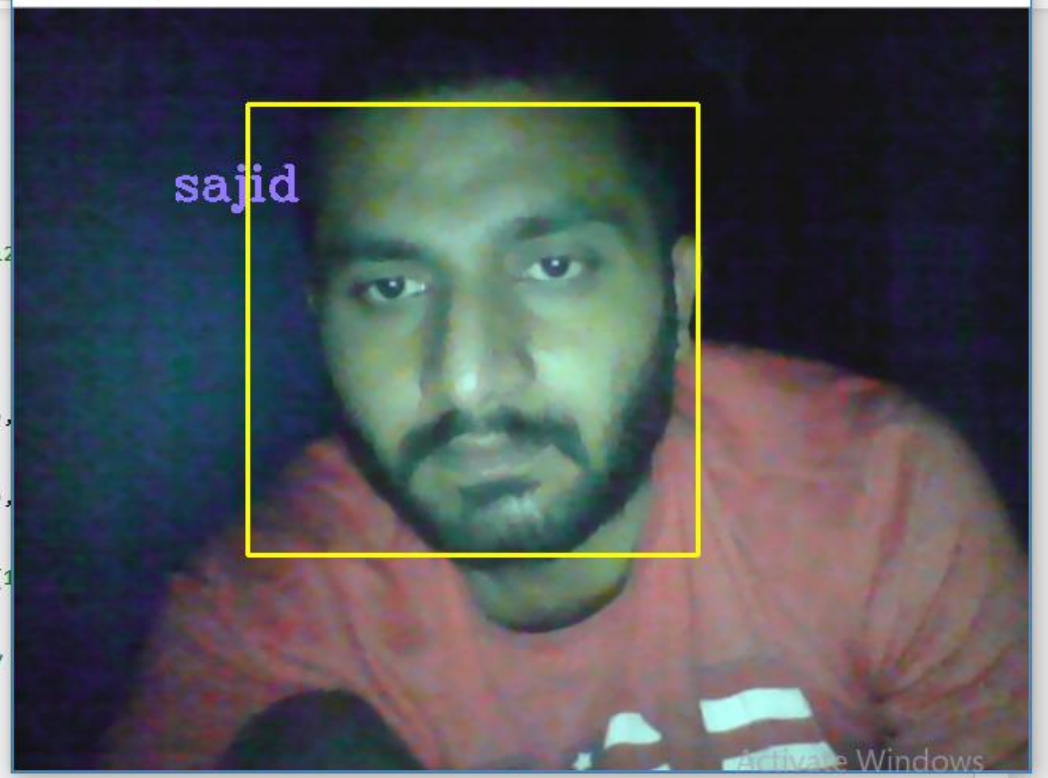
File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3

Run | Code

```
while True:

    ret, frame = cap.read()

    image, face = face_detector(frame)

    face=np.array(face)
    face=np.expand_dims(face,axis=0)
    if face.shape==(1,0):
        cv2.putText(image,"I don't know", (100, 12
        cv2.imshow('Face Recognition',image)
    else:
        result=model.predict(face)
        print(result)
        if result[0][0] == 1.0:
            cv2.putText(image,"sajid", (100, 120),
            cv2.imshow('Face Recognition',image)
        elif result[0][0] == 0.0:
            cv2.putText(image,"fayaz", (100, 120),
            cv2.imshow('Face Recognition',image)
        else:
            cv2.putText(image,"Not recognized", (1
            cv2.imshow('Face Recognition',image)

    if cv2.waitKey(1) == 13: #13 is the Enter Key
        break

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

Face Recognition



Activate Windows
Go to Settings to activate Windows.

```
[[0. 1.]]
[[0. 1.]]
```

Type here to search

5:40 AM
5/20/2020