

Guided Assignment On Deep Learning

Name: Shreya Shrivastava

Course: Artificial Intelligence And Machine Learning

Batch Four

Duration: 12 Months

Problem statement: Face recognition such as that used in our phones, relies on two important steps. **First step is face detection**, which can be implemented using simple classifiers (such as Haar Cascade) or CNNs. **Second part involves the recognition** of the identity of the person, which further uses a CNN for this task. Implement a face recognition network using **Haar Cascade for detection**, followed by using **VGG-19 for the task of recognition**. Make sure you use a pre-trained VGG model and freeze the weights of starting layers before fine tuning the model.

Prerequisites:

The libraries as well as things required in order for the program to work:

- I. **Python 3.6** : The following url <https://www.python.org/downloads/> can be referred to download python. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: <https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/>. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic. Second option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url : <https://www.anaconda.com/download/>
- II. **ADDITIONAL PACKAGES** : You will also need to download and install below 3 packages- Numpy, Pillow and Keras after you install ei-

ther python or anaconda from the steps above. If you have chosen to install python 3.6, then run the following commands in command prompt/terminal to install these packages :

NUMPY: pip install -U numpy

PIL: pip install -U pillow

KERAS: pip install -U keras

If using Anaconda then run the following commands in anaconda prompt to install these packages:

NUMPY: conda install -c anaconda numpy

PIL: conda install -c anaconda pillow

KERAS: conda install -c keras

III. METHODS USED:

A. HAAR CASCADE CLASSIFIER

B. VGG-19

THE PROJECT :

1. Importing the libraries.

```
import cv2
from PIL import Image
from numpy import asarray, expand_dims
from keras_vggface.vggface import VGGFace
from keras_vggface.utils import preprocess_input
from keras_vggface.utils import decode_predictions
```

2. Extracting the Faces using Haar Cascade Classifier.

```
def extract_faces(filename, req_size = (224, 224)):
    face_cascade = cv2.CascadeClassifier
    (["/Users/shreyashrivastava/opt/anaconda3/pkgshare/libopencv-4.5.2-py39h852ad08_1/share/opencv4/haarcascades/haarcascade_frontalface_default.xml"])
    img = cv2.imread(filename)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.1, 4)
    for (x, y, w, h) in faces:
        face = img[y:y+h, x:x+w]
        image = Image.fromarray(face)
        image = image.resize(req_size)
        face_array = asarray(image)
    return face_array
```

3. Performing face recognition.

```
img='/Users/shreyashrivastava/Desktop/XYZ.png'
#img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
pixels = extract_faces(img)
# convert one face into samples
pixels = pixels.astype('float32')
samples = expand_dims(pixels, axis=0)
# prepare the face for the model, e.g. center pixels
samples = preprocess_input(samples, version=2)
# create a vggface model
model = VGGFace(model='resnet50')
# perform prediction
yhat = model.predict(samples)
# convert prediction into names
results = decode_predictions(yhat)
# display most likely results
for result in results[0]:
    print('%s: %.3f%%' % (result[0], result[1]*100))
```

4.Results.



Original Image of Sharon Stone

```
Sharon_Stone': 98.834%
Anita_Lipnicka': 0.422%
Noelle_Reno': 0.075%
Paulina_Gretzky': 0.048%
Therese_Johaug': 0.031%
```

Predictions