Name : Tejas Redkar
PRN : 1032210937
Panel- C, Batch : C2
Roll NO : PC-44

AIES Lab Assignment - 8

* Title - Implement a Neural network for a real-life application

* Aim - Implement & Understand working of Neural network for a real-life application : Face Recognition with Python.

* Objective - To study & implement face recognition using Python & the open-source library OpenCV

* Theory

1) Difference between Machine learning & Deep learning

Machine learning is a broader concept that involves the development of algorithms & models that enable computers to learn & make predictions or decisions based on data. It encompasses various techniques, including traditional statistical methods & newer approaches like neural networks.

Deep learning is a subset of machine learning that specifically focuses on neural networks with multiple layers (deep neural networks). It is inspired by the structure & function of the human brain, allowing it to automatically learn hierarchical representations of data. Deep learning has been particularly successful in tasks such as image & speech recognition.

2) What is open CV? How does it work?

Open CV (Open Source Computer Vision) is a library of programming functions for computer vision tasks. It provides tools for images & video processing, enabling computer vision applications.

3) What are Neural Networks? How working of open CV? ~~Write in brief about them~~ is different from Neural Network?

Neural Networks are computational models inspired by the human brain, processing information through interconnected nodes. Open CV is a computer vision library, while neural networks focus on learning patterns from data.

* Input - Input an image with a human face in it.

* Output - Algorithm will ~~decide~~ detect faces of all humans present in the image.

* Algorithm - Neural Network.

* Platform - Linux

* FAQ's

Q1) Explain cascade & classifier in detail.

Ans A cascade classifier is a series of interconnected stages, each containing a simple classifier. It efficiently eliminates non-object regions in early stages, allowing more complex, computationally expensive classifiers to focus on potential object regions, enhancing the overall speed of object detection in computer vision applications.

Q2) What are other cascades provided by OpenCV ? Write in brief about them.

Ans OpenCV provides pre-trained cascades for various object detection tasks. Examples include face detection, eye detection, full body detection, license plate detection & more. These cascades are trained to recognize specific patterns in images for different applications.

1/12/23

CODE:

```
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense

# Sample input data (features)
X = np.array([[0, 0],
              [0, 1],
              [1, 0],
              [1, 1]])

# Sample output data (labels)
y = np.array([0, 1, 1, 0])

# Build a simple feedforward neural network
model = Sequential()
model.add(Dense(4, input_dim=2, activation='relu'))  # Hidden layer with 4 neurons and
ReLU activation
model.add(Dense(1, activation='sigmoid'))          # Output layer with 1 neuron and
Sigmoid activation

# Compile the model
```

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

# Train the model
model.fit(X, y, epochs=500, verbose=0)

# Test the model
sample_input = np.array([[0, 0]])
predicted_output = model.predict(sample_input)
print(f"Sample Input: {sample_input}")
print(f"Predicted Output: {predicted_output}")
```

Input:

```
[[0 0]]
```

Output:

```
[[0.03575368]]
```