Explanation of data:

Below is the lfw image set.

http://vis-www.cs.umass.edu/lfw/lfw.tgz

http://vis-www.cs.umass.edu/lfw/lfw-names.txt

CelebA Dataset

The below has no label on the people's names. We have not yet used this database. But looks to be a potential good database. We can not use this for commercial use. As discussed earlier we will not be using any of this data for our training. We will use it only to understand the effort involved and to understand the training algorithms. We will have to create our own data set for training.

CelebA Dataset

VGGFace2 Dataset with normal pic.  
Vggface2: A dataset for recognising faces across pose and age

It has be downloaded via torrent, It contains the train and test data, The train data has 3miillion images of 861 variations of size 151gb. For training we use train data after converting the pic initial size 256 by 256 .. convert to 182 by 182 for training purpose. After 160 epochs we are getting 99.2% accuracy.

VGGFace2 High Definition Picture.  
VGGFace2-HQ - Google Drive

It has be downloaded directly, It contains the train data, the pic initial size 512 by 512. after converting pic to 182 by 182 or 60 by 160 for training purpose, we could achieve accuracy of 83% after 50 epochs.

Note# Observation was because of HD quality of images the algorithm not able to find much difference, so need to have low or medium quality images

Approach

**System Details:**

**Hardware -**

CPU - AMD EPYC 7542 32 - core @ 2.9Ghz

GPU - NVidia Tesla 4 with 15 GB

Memory - 48.8 gb

Hardisk type & Size

**OS - Windows Server 2019**

Windows with build version & patch details.

**Setup:**

Software Installed -

Python38 (set path C:\Python\Python38; C:\Python\Python38\Scripts;)

Python Release Python 3.8.13

NVIDIA (set path C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common; C:\Program Files\NVIDIA Corporation\NVSMI;C:\Program Files\NVIDIA Corporation\Nsight Compute 2020.3.0\)[Here Cuda ]

CUDA Toolkit Archive

Select 11.2 version

nvcc: NVIDIA (R) Cuda compiler driver  
Copyright (c) 2005-2020 NVIDIA Corporation  
Built on Mon\_Nov\_30\_19:15:10\_Pacific\_Standard\_Time\_2020  
Cuda compilation tools, release 11.2, V11.2.67  
Build cuda\_11.2.r11.2/compiler.29373293\_0

Facenet (set path e:\Project\Face\facenet\src;)[All modules required are present in requirements.txt]

Tensorflow - [All modules required are present in requirements.txt]

absl-py==0.10.0  
astunparse==1.6.3  
cachetools==4.2.4  
certifi==2021.10.8  
chardet==4.0.0  
cycler==0.11.0  
cycler==0.11.0  
flatbuffers==1.12.0  
gast==0.4.0  
google-auth==1.35.0  
google-auth-oauthlib==0.4.6  
google-pasta==0.2.0  
grpcio==1.34.0  
h5py==3.1.0  
idna==3.3  
imageio==2.14.0  
joblib==1.1.0  
Keras-Preprocessing==1.1.2  
kiwisolver==1.4.2  
Markdown==3.3.6  
matplotlib==3.5.1  
numpy==1.19.2  
oauthlib==3.2.0  
opencv-python==3.4.17.63  
opt-einsum==3.3.0  
Pillow==9.1.0  
protobuf==3.20.0  
pyasn1==0.4.8  
pyasn1-modules==0.2.8  
pyparsing==3.0.8  
python-dateutil==2.8.2  
pytest==7.1.1  
psutil==5.9.0  
requests==2.27.1  
requests-oauthlib==1.3.1  
rsa==4.8  
scikit-learn==1.0.2  
scipy==1.8.0  
six==1.15.0  
tensorboard==2.5.0  
tensorboard-plugin-wit==1.8.1  
tensorflow==2.5.0  
tensorflow-gpu==2.5.0  
tensorflow-estimator==2.5.0  
termcolor==1.1.0  
tf-slim==1.1.0  
threadpoolctl==3.1.0  
typing-extensions==3.7.4  
urllib3==1.26.9  
Werkzeug==2.1.1  
wrapt==1.12.1

Path and Configuration -

Location of program, data, datasets & models:

E:\Project\Face\facenet

E:\Project\Face\facenet\data

E:\Project\Face\facenet\datasets

E:\Project\Face\facenet\models\facenet[Date wise models]

**Commands Used & Understanding:**

• First we align the image

• Crop only the face

• Keep the face straight

• Command here : python src/align/align\_dataset\_mtcnn.py ./datasets/lfw/raw ./datasets/lfw/lfw\_mtcnnpy\_160 --image\_size 160 --margin 32 --gpu\_memory\_fraction 0.4 Note# we use the above command for LFW images

• Command here : python src/align/align\_dataset\_mtcnn.py ./datasets/lfw/raw ./datasets/lfw/lfw\_mtcnnpy\_160 --image\_size 160 --margin 32 --gpu\_memory\_fraction 0.4 Note# we use the above command for LFW images for training we need to train at 182 by 182 image size

• Command here : python src/align/align\_dataset\_mtcnn.py C:\Project\Face\facenet\datasets\vggface2\VGG-Face2\data\train C:\Project\Face\facenet\datasets\vggface2\VGG-Face2\data\vggface2\_train\_182 --image\_size 182 --margin 44 --gpu\_memory\_fraction 0.4 Note# we use the above command for training images

• We start the training algorithm.

• Softmax

• Command here [LFW Training]: python src/train\_softmax.py --logs\_base\_dir ./logs/facenet/ --models\_base\_dir ./models/facenet/ --data\_dir ./datasets/lfw/lfw\_mtcnnpy\_160/ --image\_size 160 --model\_def models.inception\_resnet\_v1 --lfw\_dir ./datasets/lfw/lfw\_mtcnnpy\_160/ --optimizer ADAM --learning\_rate -1 --max\_nrof\_epochs 150 --keep\_probability 0.8 --random\_crop --random\_flip --use\_fixed\_image\_standardization --learning\_rate\_schedule\_file data/learning\_rate\_schedule\_classifier\_casia.txt --weight\_decay 5e-4 --embedding\_size 512 --lfw\_distance\_metric 1 --lfw\_use\_flipped\_images --lfw\_subtract\_mean --validation\_set\_split\_ratio 0.05 --validate\_every\_n\_epochs 5 --prelogits\_norm\_loss\_factor 5e-4 --gpu\_memory\_fraction 0.3

• Command here [vggface2 Training]: python src/train\_softmax.py --logs\_base\_dir ./logs/facenet/ --models\_base\_dir ./models/facenet/ --data\_dir C:\Project\Face\facenet\datasets\vggface2\VGG-Face2\data\vggface2\_train\_182/ --image\_size 160 --model\_def models.inception\_resnet\_v1 --lfw\_dir ./datasets/lfw/lfw\_mtcnnpy\_160/ --optimizer ADAM --learning\_rate -1 --max\_nrof\_epochs 500 --batch\_size 90 --keep\_probability 0.4 --random\_flip --use\_fixed\_image\_standardization --learning\_rate\_schedule\_file data/learning\_rate\_schedule\_classifier\_vggface2.txt --weight\_decay 5e-4 --embedding\_size 512 --lfw\_distance\_metric 1 --lfw\_use\_flipped\_images --lfw\_subtract\_mean --validation\_set\_split\_ratio 0.01 --validate\_every\_n\_epochs 5 --gpu\_memory\_fraction 0.5

4. Once we have the model we will validate using following command

python src/validate\_on\_lfw.py ./datasets/lfw/lfw\_mtcnnpy\_160/ C:\Project\Face\facenet\models\facenet\20220719-222142/ --distance\_metric 1 --use\_flipped\_images --subtract\_mean --use\_fixed\_image\_standardization

5. Once we have trained the model and validate we will use the following approach for identification:

a)Take two similar images of yourself.

b)Pass it to the newly written python code to validate the given two images.

c)Get the result.

python src/train\_softmax.py --logs\_base\_dir ./logs/facenet --models\_base\_dir ./models/facenet --data\_dir ./data/train --image\_size 160 --model\_def models.inception\_resnet\_v1 --optimizer ADAM --learning\_rate -1 --max\_nrof\_epochs 500 --batch\_size 90 --keep\_probability 0.4 --random\_flip --use\_fixed\_image\_standardization --learning\_rate\_schedule\_file data/learning\_rate\_schedule\_classifier\_vggface2.txt --weight\_decay 5e-4 --embedding\_size 512 --validation\_set\_split\_ratio 0.2 --validate\_every\_n\_epochs 5 --filter\_percentile 95.0 --gpu\_memory\_fraction 0.5