Walchand College of Engineering, Sangli.

(An Autonomous Institute)

Department
Of
Computer Science and Engineering

TY CSE Mini Project-I
Report
On

AGE AND GENDER PREDICTION

Submitted by

Omkar Sharad Patil **(2019BTECS00104)**
Pratik Babaso Chougule **(2019BTECS00102)**
Pravin Santosh Lokhande **(2019BTECS00084)**

Under the Guidance
of

Mr. S. S. Sontakke
Guide
Computer Science & Eng. Dept,
WCE, Sangli.

2021-2022

Walchand College of Engineering, Sangli
(An Autonomous Institute)

Department
Of
Computer Science and Engineering

CERTIFICATE

This is to certify that the Project Report entitled, **"AGE AND GENDER PREDICTION"** submitted by Mr. Omkar Patil, Mr. Pratik Chougule, Mr. Pravin Lokhande to Walchand College of Engineering, Sangli, India, is a record of bonafide Project work of course *"Mini Project-1"* carried out by him/her under my/our supervision and guidance and is worthy of consideration for the award of the degree of Bachelor of Technology in Computer Science & Engineering of the Institute.

Mr. S. S. Sontakke	**Dr. M. A. Shah**
Guide	Head Of Department
Computer Sci. & Eng. Dept,	Computer Sci.& Eng. Dept,
WCE, Sangli.	WCE, Sangli

Acknowledgement

We would like to express our special thanks of gratitude to our guide Mr. S. S. Sontakke as well as our HOD Dr. M. A. Shah who gave us the golden opportunity to do this wonderful project on the topic “Age and Gender Prediction”, which also helped us in doing a lot of research and we came to know about so many new things. We are thankful to them.

Secondly, we would also like to thank the teammates who worked together in finishing this project within limited time. Finally, thanks to all who supported the project.

Declaration

I hereby declare that work presented in this project report titled "**AGE AND GENDER PREDICTION"** submitted by me in the partial fulfillment of the requirement of the award of the degree of **Bachelor of Technology (B. Tech)** Submitted in the **Department of Computer Science & Engineering, Walchand College of Engineering, Sangli**, is an authentic record of my project work carried out under the guidance of Mr. S. S. Sontakke

Date:
25-11-2021

Place: Sangli

(Omkar Patil)
2019BTECS00104

(Pratik Chougule)
2019BTECS00102

(Pravin Lokhande)
2019BTECS00084

Table of Contents

1 Project title 6

2 Abstract 7

3 Introduction and Related work 8

4 Problem statement 8

5 Objectives 9

6 Methodology 10

7 Project Diagrams 11

8 Testing 12

9 Results and Conclusion 16

10 References 17

1 Project title

Age and Gender Prediction Using CNN

2 Abstract

Age and gender identification have become a major part of the network, security, and care. It has a common use in age specific content access for children. It is also used by social media for delivering layered ads and marketing to extend its reach. Age and Gender Prediction has become relevant to an increasing number of applications.

A simple convolutional network architecture is proposed to make a noticeable improvement in this field using existing methods. Using deep CNN, model is trained to an extent that accuracy of Age and Gender become 79% using HAAR Feature-based Cascade Classifiers is an effective method proposed by Paul Viola and Michael Jones. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

We have applied a simple convolutional architecture that can be used even when the amount of learning data is limited.

3 Introduction and Related work

Age and gender, two of the key facemask attributes, play a very initial role in social communications, making age and gender approximation from a single image an important task in intelligent applications, such as access control, human- computer interaction, law application, marketing intelligence and visual observation, etc. It can be used to suppose the age and gender of the user and use this information to make modified product and understanding for each user.

Early classification of age was done by calculation ration between different features of face like nose, eyes, mouth, chin etc. After localizing calculating their sizes and distances, ratio between them is calculated to predict age by using conventional methods. In the past when computers were not efficient then it was a great job to make computer perform tasks as the interaction of people vary to each computer. Every person cannot respond to computer in similar way.

Thus the methods presented above, we have noticed that we used dataset which is more difficult and challenging

4 Problem statement

To create a platform which helps to predict the age and gender of a person using Convolutional Neural Network.

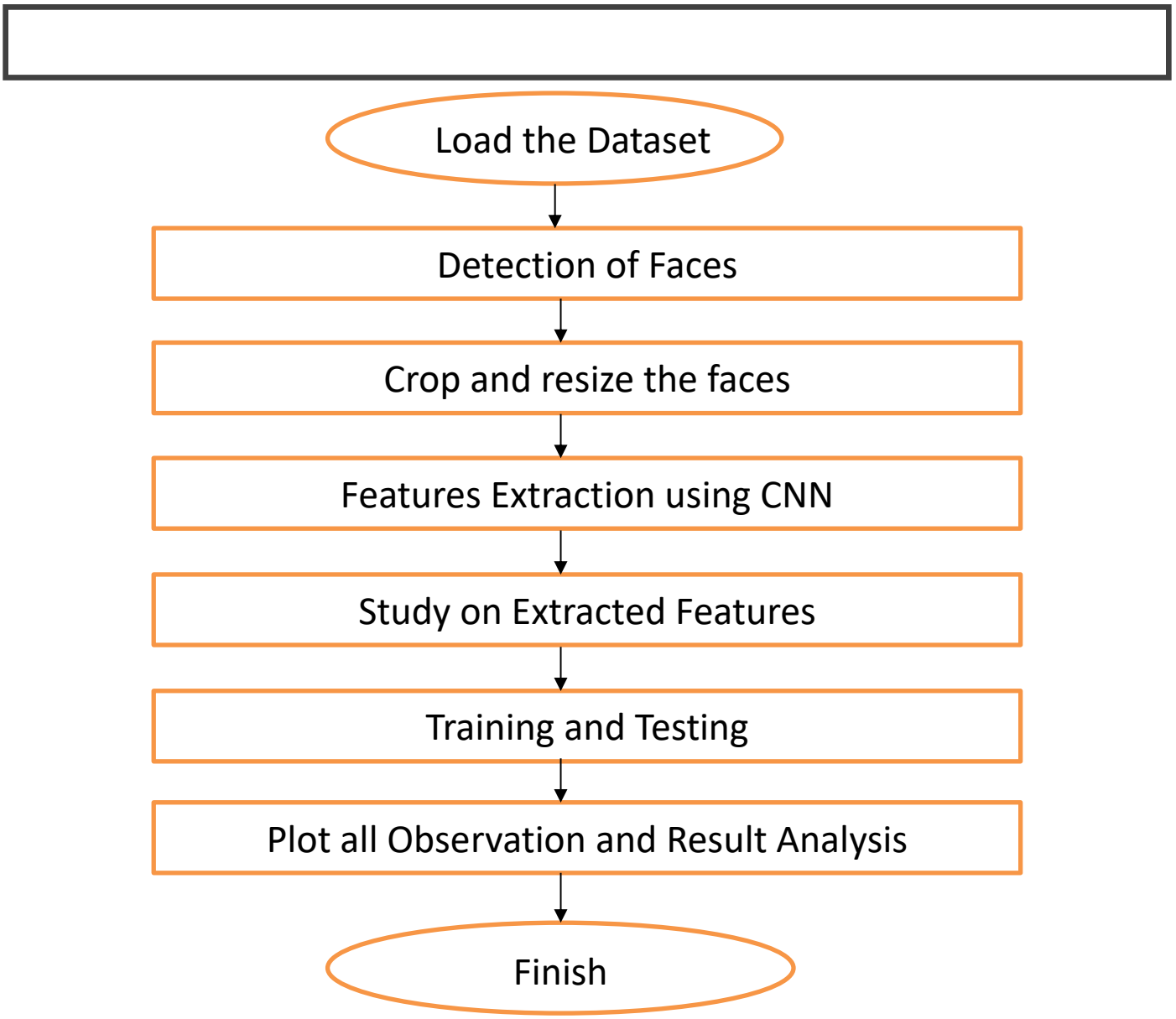
5 Objectives

- **To study the concept of Machine Learning (Convolutional Neural Network Technology), OpenCV and Python.**
- **To collect images from Kaggle Dataset.**
- **To implement a CNN model for age and gender prediction.**
- **To train and test CNN model.**
- **To install/host the website on the cloud.**

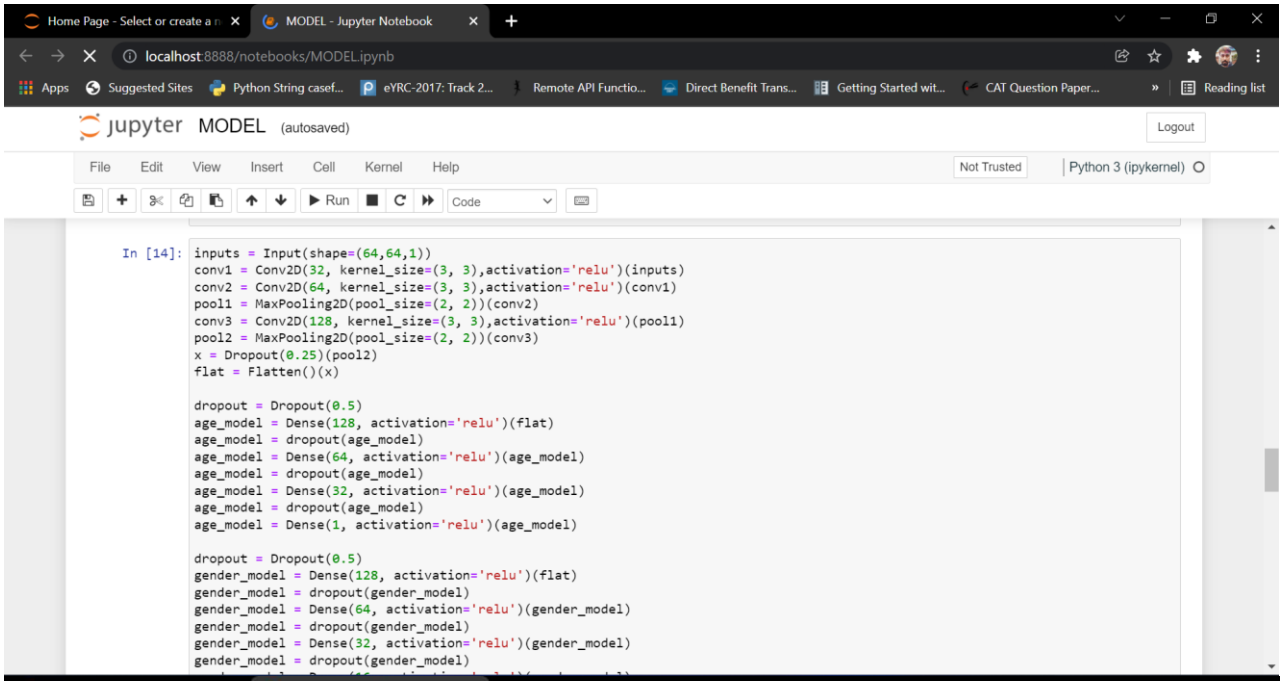
6 Methodology

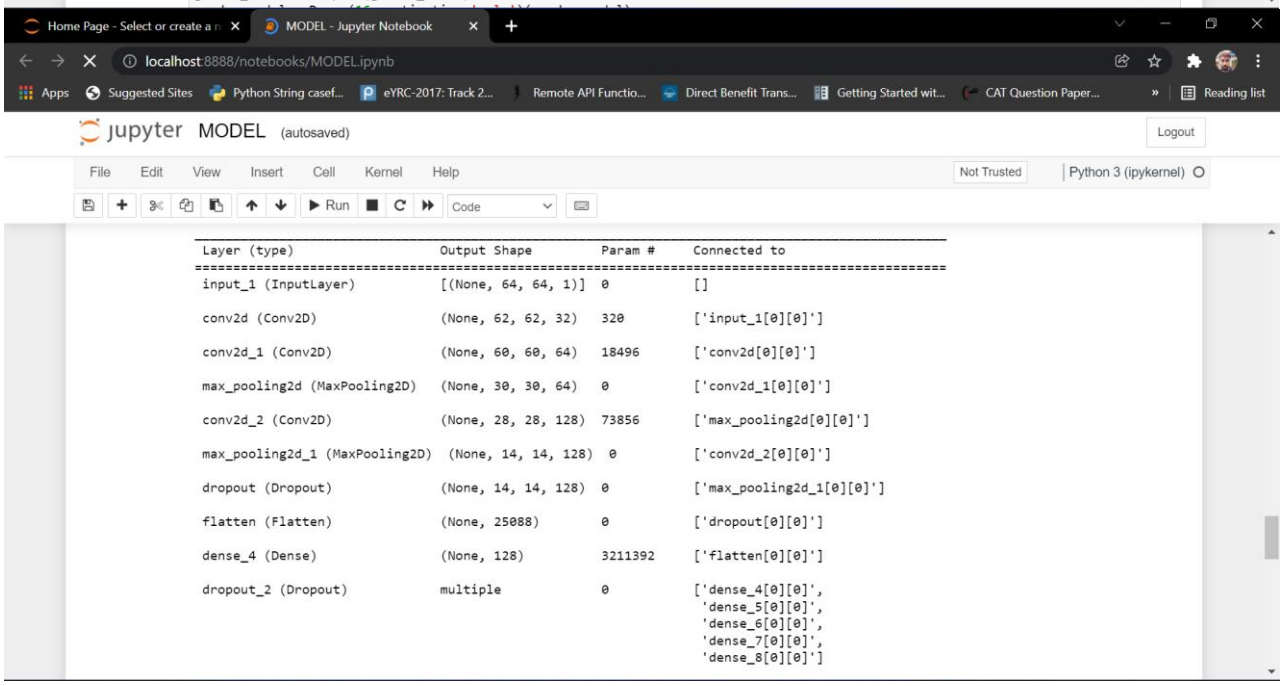
- **Obtain the image of a person.**
- **Process the image and then apply a CNN model.**
- **Use of images from the Kaggle Dataset for the data processing and prediction.**
- **Prediction of the age and gender of the user.**
- **Training the model to maintain the accuracy.**
- **Testing the trained model.**
- **Installing the model on the website.**

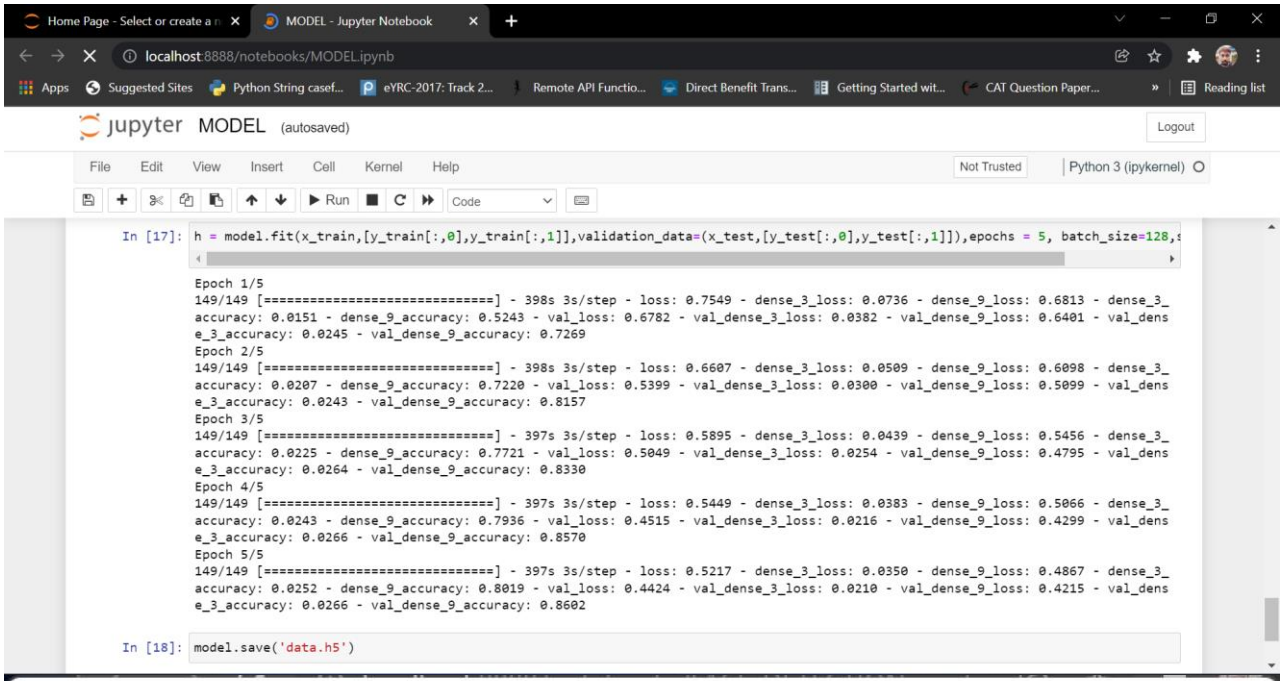
7. Project diagrams

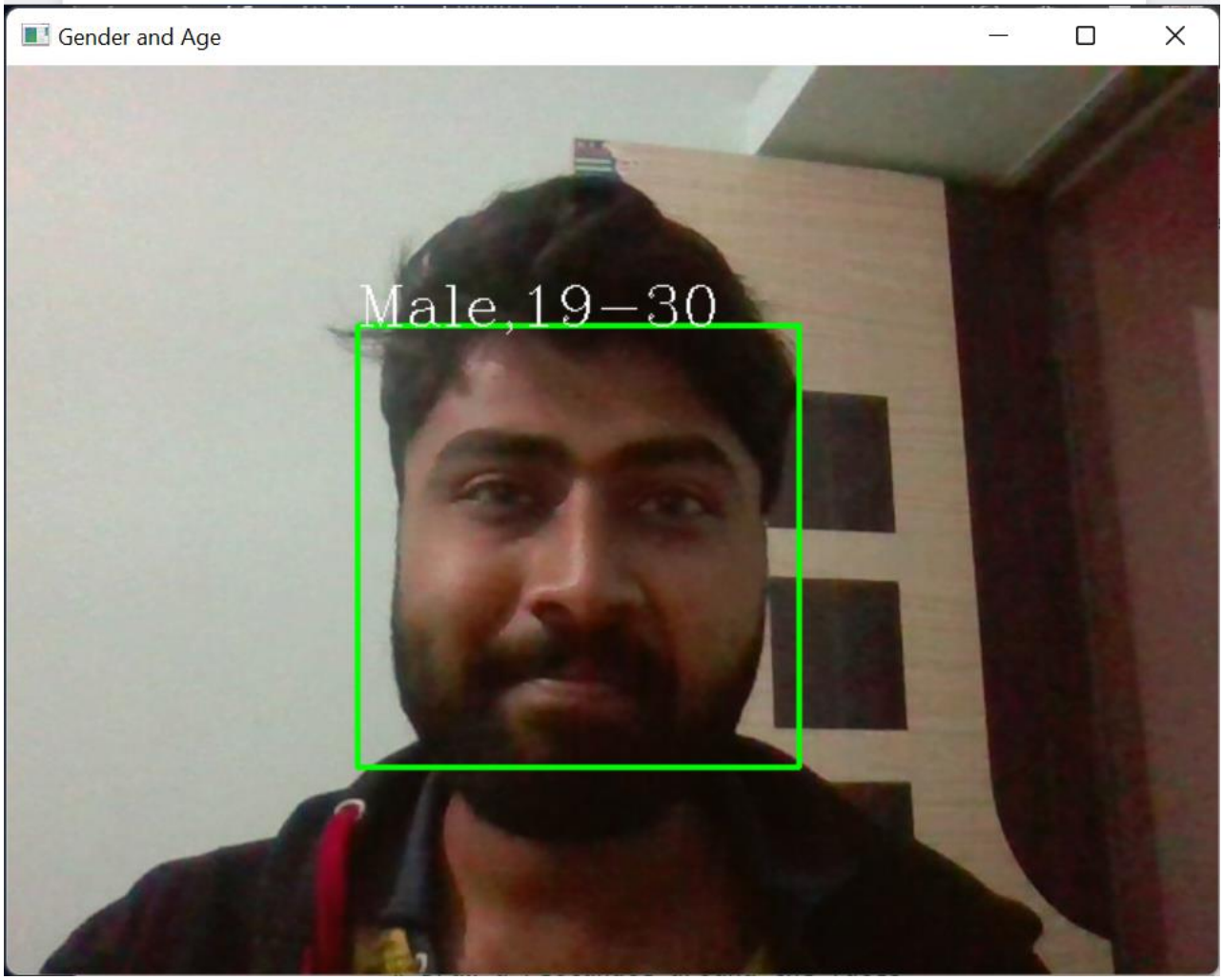


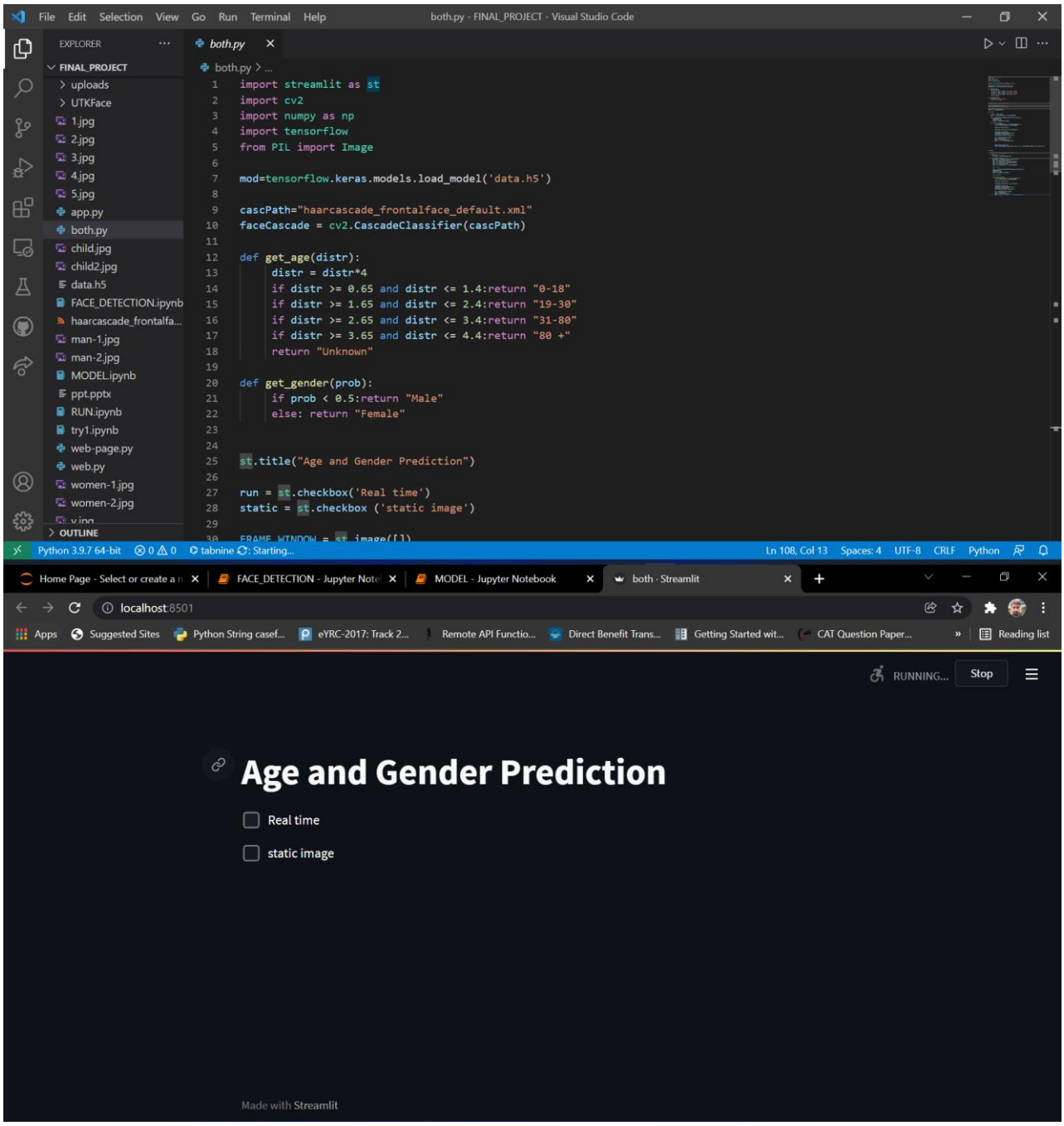
8. Testing

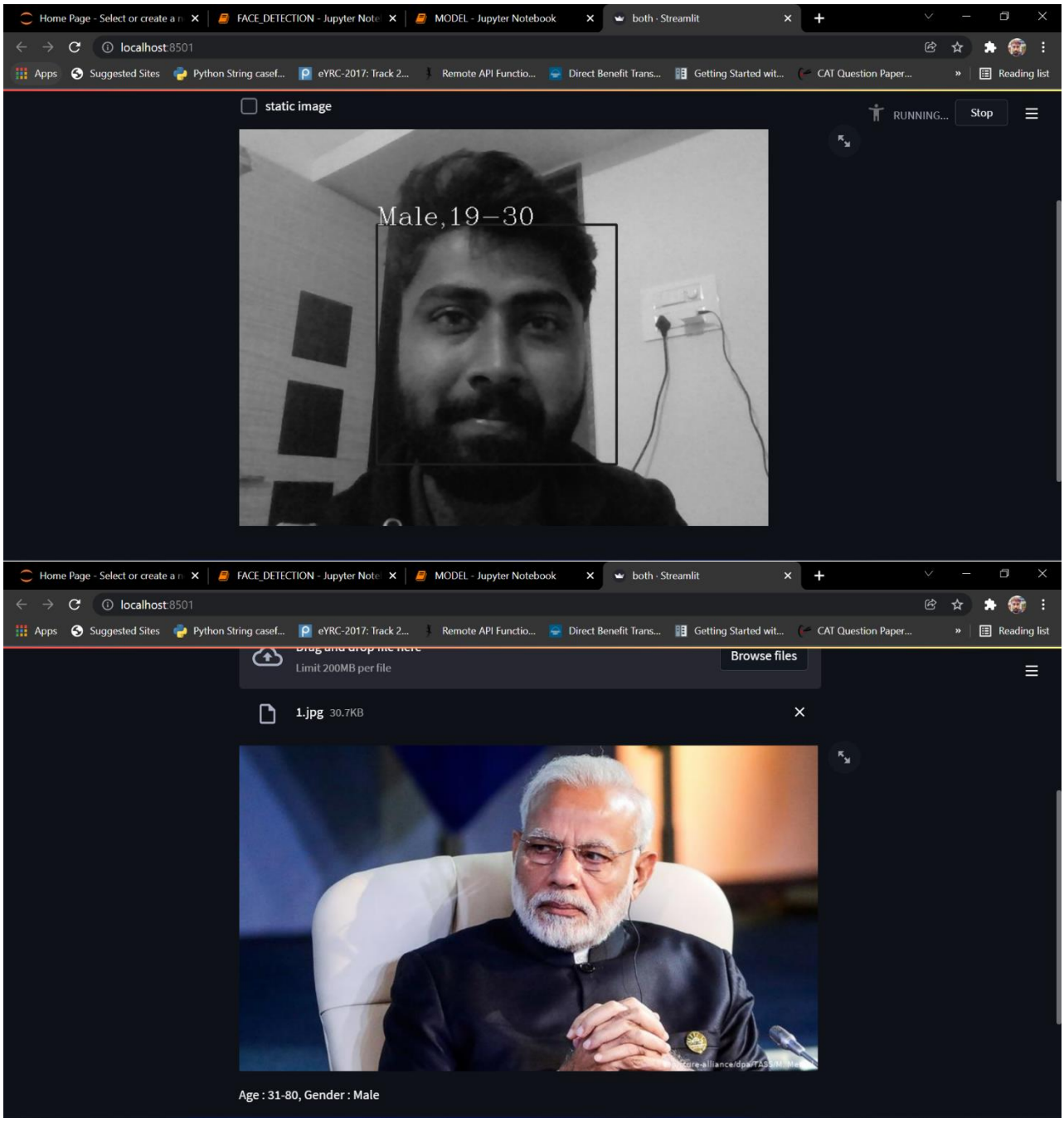

Home Page - Select or create a n
MODEL - Jupyter Notebook
localhost:8888/notebooks/MODEL.ipynb
Apps
Suggested Sites
Python String casef...
eYRC-2017: Track 2...
Remote API Functio...
Direct Benefit Trans...
Getting Started wit...
CAT Question Paper...
Reading list
jupyter MODEL (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [14]:
inputs = Input(shape=(64,64,1))
conv1 = Conv2D(32, kernel_size=(3, 3),activation='relu')(inputs)
conv2 = Conv2D(64, kernel_size=(3, 3),activation='relu')(conv1)
pool1 = MaxPooling2D(pool_size=(2, 2))(conv2)
conv3 = Conv2D(128, kernel_size=(3, 3),activation='relu')(pool1)
pool2 = MaxPooling2D(pool_size=(2, 2))(conv3)
x = Dropout(0.25)(pool2)
flat = Flatten()(x)
dropout = Dropout(0.5)
age_model = Dense(128, activation='relu')(flat)
age_model = dropout(age_model)
age_model = Dense(64, activation='relu')(age_model)
age_model = dropout(age_model)
age_model = Dense(32, activation='relu')(age_model)
age_model = dropout(age_model)
age_model = Dense(1, activation='relu')(age_model)
dropout = Dropout(0.5)
gender_model = Dense(128, activation='relu')(flat)
gender_model = dropout(gender_model)
gender_model = Dense(64, activation='relu')(gender_model)
gender_model = dropout(gender_model)
gender_model = Dense(32, activation='relu')(gender_model)
gender_model = dropout(gender_model)


Home Page - Select or create a n
MODEL - Jupyter Notebook
localhost:8888/notebooks/MODEL.ipynb
Apps
Suggested Sites
Python String casef...
eYRC-2017: Track 2...
Remote API Functio...
Direct Benefit Trans...
Getting Started wit...
CAT Question Paper...
Reading list
jupyter MODEL (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Not Trusted
Python 3 (ipykernel)
Run
Code
Layer (type) Output Shape Param # Connected to
input_1 (InputLayer) [(None, 64, 64, 1)] 0 []
conv2d (Conv2D) (None, 62, 62, 32) 320 ['input_1[0][0]']
conv2d_1 (Conv2D) (None, 60, 60, 64) 18496 ['conv2d[0][0]']
max_pooling2d (MaxPooling2D) (None, 30, 30, 64) 0 ['conv2d_1[0][0]']
conv2d_2 (Conv2D) (None, 28, 28, 128) 73856 ['max_pooling2d[0][0]']
max_pooling2d_1 (MaxPooling2D) (None, 14, 14, 128) 0 ['conv2d_2[0][0]']
dropout (Dropout) (None, 14, 14, 128) 0 ['max_pooling2d_1[0][0]']
flatten (Flatten) (None, 25088) 0 ['dropout[0][0]']
dense_4 (Dense) (None, 128) 3211392 ['flatten[0][0]']
dropout_2 (Dropout) multiple 0 ['dense_4[0][0]',
'dense_5[0][0]',
'dense_6[0][0]',
'dense_7[0][0]',
'dense_8[0][0]']


Home Page - Select or create a n
MODEL - Jupyter Notebook
localhost:8888/notebooks/MODEL.ipynb
Apps
Suggested Sites
Python String casef...
eYRC-2017: Track 2...
Remote API Functio...
Direct Benefit Trans...
Getting Started wit...
CAT Question Paper...
Reading list
jupyter MODEL (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Not Trusted
Python 3 (ipykernel)
Run
Code
In [17]:
h = model.fit(x_train,[y_train[:,0],y_train[:,1]],validation_data=(x_test,[y_test[:,0],y_test[:,1]]),epochs = 5, batch_size=128,s
Epoch 1/5
149/149 [==========] - 398s 3s/step - loss: 0.7549 - dense_3_loss: 0.0736 - dense_9_loss: 0.6813 - dense_3_
accuracy: 0.0151 - dense_9_accuracy: 0.5243 - val_loss: 0.6782 - val_dense_3_loss: 0.0382 - val_dense_9_loss: 0.6401 - val_dens
e_3_accuracy: 0.0245 - val_dense_9_accuracy: 0.7269
Epoch 2/5
149/149 [==========] - 398s 3s/step - loss: 0.6607 - dense_3_loss: 0.0509 - dense_9_loss: 0.6098 - dense_3_
accuracy: 0.0207 - dense_9_accuracy: 0.7220 - val_loss: 0.5399 - val_dense_3_loss: 0.0300 - val_dense_9_loss: 0.5099 - val_dens
e_3_accuracy: 0.0243 - val_dense_9_accuracy: 0.8157
Epoch 3/5
149/149 [==========] - 397s 3s/step - loss: 0.5895 - dense_3_loss: 0.0439 - dense_9_loss: 0.5456 - dense_3_
accuracy: 0.0225 - dense_9_accuracy: 0.7721 - val_loss: 0.5049 - val_dense_3_loss: 0.0254 - val_dense_9_loss: 0.4795 - val_dens
e_3_accuracy: 0.0264 - val_dense_9_accuracy: 0.8330
Epoch 4/5
149/149 [==========] - 397s 3s/step - loss: 0.5449 - dense_3_loss: 0.0383 - dense_9_loss: 0.5066 - dense_3_
accuracy: 0.0243 - dense_9_accuracy: 0.7936 - val_loss: 0.4515 - val_dense_3_loss: 0.0216 - val_dense_9_loss: 0.4299 - val_dens
e_3_accuracy: 0.0266 - val_dense_9_accuracy: 0.8570
Epoch 5/5
149/149 [==========] - 397s 3s/step - loss: 0.5217 - dense_3_loss: 0.0350 - dense_9_loss: 0.4867 - dense_3_
accuracy: 0.0252 - dense_9_accuracy: 0.8019 - val_loss: 0.4424 - val_dense_3_loss: 0.0210 - val_dense_9_loss: 0.4215 - val_dens
e_3_accuracy: 0.0266 - val_dense_9_accuracy: 0.8602
In [18]:
model.save('data.h5')


Gender and Age
Male,19-30


File Edit Selection View Go Run Terminal Help
both.py - FINAL_PROJECT - Visual Studio Code
EXPLORER
FINAL_PROJECT
uploads
UTKFace
1.jpg
2.jpg
3.jpg
4.jpg
5.jpg
app.py
both.py
child.jpg
child2.jpg
data.h5
FACE_DETECTION.ipynb
haarcascade_frontalfa...
man-1.jpg
man-2.jpg
MODEL.ipynb
ppt.pptx
RUN.ipynb
try1.ipynb
web-page.py
web.py
women-1.jpg
women-2.jpg
OUTLINE
import streamlit as st
import cv2
import numpy as np
import tensorflow
from PIL import Image
mod=tensorflow.keras.models.load_model('data.h5')
cascPath="haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascPath)
def get_age(distr):
distr = distr*4
if distr >= 0.65 and distr <= 1.4:return "0-18"
if distr >= 1.65 and distr <= 2.4:return "19-30"
if distr >= 2.65 and distr <= 3.4:return "31-80"
if distr >= 3.65 and distr <= 4.4:return "80 +"
return "Unknown"
def get_gender(prob):
if prob < 0.5:return "Male"
else: return "Female"
st.title("Age and Gender Prediction")
run = st.checkbox('Real time')
static = st.checkbox ('static image')
Python 3.9.7 64-bit
Ln 108, Col 13 Spaces: 4 UTF-8 CRLF Python
Home Page - Select or create a
FACE_DETECTION - Jupyter Note
MODEL - Jupyter Notebook
both - Streamlit
localhost:8501
Apps
Suggested Sites
Python String casef...
eYRC-2017: Track 2...
Remote API Functio...
Direct Benefit Trans...
Getting Started wit...
CAT Question Paper...
Reading list
RUNNING...
Stop
Age and Gender Prediction
Real time
static image
Made with Streamlit


Home Page - Select or create a n
FACE_DETECTION - Jupyter Note
MODEL - Jupyter Notebook
both · Streamlit
localhost:8501
Apps
Suggested Sites
Python String casef...
eYRC-2017: Track 2...
Remote API Functio...
Direct Benefit Trans...
Getting Started wit...
CAT Question Paper...
Reading list
static image
RUNNING...
Stop
Male,19–30
Limit 200MB per file
Browse files
1.jpg 30.7KB
Age : 31-80, Gender : Male

9. Results and Conclusion

- **CNN can be used to provide improved age and gender classification results, even considering the smaller size of image sets labeled for age and gender.**
- **This model will be beneficiary in many fields and will be user-friendly.**
- **The simplicity of the model implies that more elaborate systems using more training data will be capable of improving results.**
- **It's efficiency can be increased by using more precise CNN architecture.**

10. References

[1] 9 A. A. Zaidan, B. B. Zaidan, A. Al-Haiqi, M. L. M. Kiah, M. Hussain, and M. Abdulnabi, "Evaluation and selection of open-source EMR software packages based on integrated AHP and TOPSIS," J. Biomed. Inform., vol. 53, pp. 390–404, 2015.

[2] M. Vasileiadis, G. Stavropoulos, and D. Tzovaras, "Facial Soft Biometrics Detection on Low Power Devices," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019, p. 0.

[3] F. Dornaika, I. Arganda-Carreras, and C. Belver, "Age estimation in facial images through transfer learning," Mach. Vis. Appl., vol. 30, no. 1, pp. 177–187, 2019.

[4] G. Antipov, M. Baccouche, S.-A. Berrani, and J.-L. Dugelay, "Effective training of convolutional neural networks for face-based gender and age prediction," Pattern Recognit., vol. 72, pp. 15–26, 2017.