

MODIFIED GOOGLNET ARCHITECTURE FOR FACE RECOGNITION FROM VISIBLE IMAGES

Submitted in partial fulfilment of the requirements for the award

Bachelor of Engineering degree in

ELECTRONICS AND COMMUNICATION ENGINEERING

By

MANDARAPU RAVI TEJA (37130236)

KONDREDDY SAIKIRAN REDDY(37130204)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade “A” by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119
www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work Of **MANDARAPU RAVI TEJA, (37130236)** and **KONDREDDY SAIKIRAN REDDY, (37130204)** who carried out the project entitled **“MODIFIED GOOGLNET ARCHITECTURE FOR FACE RECOGNITION FROM VISIBLE IMAGES”** under my supervision from

INTERNAL GUIDE
Dr.N.M.NANDITHA

Head of the Department
(DR. T.RAVI)

Submitted for Viva voce Examination held on _____

INTERNAL EXAMINER

(Name In Capital Letters with
Signature)

EXTERNAL EXAMINER

(Name In Capital Letters with
Signature)

DECLARATION

I **MANDARAPU RAVI TEJA** hereby declare that the Project Report entitled “MODIFIED GOOGLNET ARCHITECTURE FOR FACE RECOGNITION FROM VISIBLE IMAGES“ done by us under the guidance of “**Dr.N.M.NANDITHA**”, Professor & Dean, (School of Electrical and Electronics Engineering) at **Sathyabama Institute of Science and Technology** is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in **ELECTRONICS AND COMMUNICATION ENGINEERING**.

DATE:

SIGNATURE OF THE CANDIDATE

PLACE:

MANDARAPU RAVI TEJA

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to the Board of Management of **SATHYABAMA** for their kind encouragement in doing this Internship and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. N. M. NANDHITHA M.E.,PhD., Prof. & Dean, School of Electrical and Electronics Engineering** and **Dr. T. RAVI M.E.,PhD., Head of the Department, DEPARTMENT of ELECTRONICS AND COMMUNICATION ENGINEERING** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.N.M.NANDITHA, Prof. and Dean (School of Electrical and Electronics)** for her valuable guidance, suggestions and constant encouragement paved the way for the successful completion of my internship.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of **Electronics and Communication Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

To study the performance of googlenet for face recognition and to improve the sensitivity and specificity by modifying the architecture of googlenet. Capturing the faces from a webcam and recognising faces stored on a database of faces by matching them to images on a database. Detecting the faces of people and labelling their names and shown as images with labelled faces as output. In this model we used conventional neural networks. The system should be able to detect faces from a webcam. The system should be able to automatically reveal the number of faces present on video. The system should be able to match detected faces cropped from an image to those on a database on the system. And the system should be able to draw a rectangular box on the face.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	COVER PAGE	1
	BONAFIDE CERTIFICATE	2
	DECLARATION	3
	ACKNOWLEDGEMENT	4
	ABSTRACT	5
	TABLE OF CONTENTS	6
	LIST OF FIGURES	8
1	INTRODUCTION	9
	1.1 ABOUT FACE RECOGNITION	9
	1.2 ROLE OF OPENCV IN FACE RECOGNITION	10
2	LITERATURE SURVEY	13
	2.1 EXISTING SYSTEM	13
	2.2 LITERATURE SURVEY	14

3	WORKFLOW	18
4	PROPOSED WORK	21
	4.1 GOOGLNET ARCHITECTURE	22
	4.2 RESULTS AND DISCUSSION	28
5	CONCLUSION & FUTURE WORK	35

LIST OF FIGURES

FIGURE	FIGURE NAME	PAGE NO:
1.1	FIND FACES IN PICTURES	10
1.2	FIND AND MANIPULATE FACIAL FEATURES IN PICTURES	11
1.3	IDENTIFY FACES IN PICTURES	11
1.4	ROTATION OF IMAGE BY 360 DEGREES AND MATCHING	12

3.1	BLOCK DIAGRAM OF PROPOSED WORK	18
3.2	UML DIAGRAM	19
3.3	DATA FLOW DIAGRAM	20
3.4	ARCHITECTURE DIAGRAM	20
4.1	INCEPTION MODULE	23
4.2	THE STRUCTURE OF THE INCEPTIONS USED IN THE MODEL	26
4.3	GOOGLNET NETWORK WITH BELLS AND WHISTLES	27
4.4	STRAIGHT FORWARD FACES OUTPUT WITH NAME LABELLED	28
4.5	FACE DETECTION IN GROUP PHOTO	29
4.6	FACE NOT PRESENT IN A DATASET	29
4.7	FACES PRESENT IN DATASET	30
4.8	FACE DETECTION WHEN FACE POSITION (UP & DOWN)	32
4.9	FACE DETECTION WHEN FACE POSITION (LEFT & RIGHT)	34

LIST OF TABLES

TABLE	TABLE NAME	PAGENO
4.1	Comparison of Googlenet Architecture with other architectures	24

CHAPTER 1

INTRODUCTION

PROBLEM STATEMENT:

The Proposed work aims at studying the performance of googlenet for face recognition and to improve the sensitivity and specificity by modifying the architecture of googlenet.

OBJECTIVES:

- To acquire of face images from websites.
- To preprocess the images.
- To evaluate performance of conventional googlenet for face recognition.
- To modify googlenet architecture.
- To pretrain the network.
- To modify Googlenet for face recognition.
- To evaluate performance and comparison of the above two networks.

1.1 ABOUT FACE RECOGNITION

We address the issue of 3-D face following, which is a principal issue with regards to human PC communication. The information comprises in a video transfer procured by a webcam highlighting the client's face. Our endeavors center around continuous and strong full 3-D face-following within

the sight of outward appearance changes, wide posture varieties, and incomplete impediments. The distance to the camera ought to conform to the prerequisites of a facial intelligent application, yet can change as indicated by the central length utilized. Additionally, we require a close to front facing present at instatement. The proposed approach incorporates the utilization of a conventional 3-D face model to follow progressively the 3-D head movement of clients and produce exact and finished face models. Our application is made out of three fundamental modules: starting 3-D model fitting, 3-D head following, and re-obtaining. The uses of the 3-D tracker and 3-D model are not talked about here. We need a 3-D model, which can either be a nonexclusive model, or a specific one recovered from an information base. In the underlying 3-D displaying step, the model is distorted symmetrically to the central hub to fit to the client's face in an information picture, by coordinating with 2-D facial tourist spots extricated at run time. The underlying 3-D reproduction of the face model is a basic advance.

Utilizing a specific calculation to recognize human appearances inside an Image (Detect Human Faces on Farm full with that is the point at which you use Face Detection. On the other side, Face acknowledgment portrays a biometric innovation that goes far past perceiving when a human face is available. It really endeavors to build up whose face it is. In this article, we are managing Face Detection program utilizing your Webcam with Python.

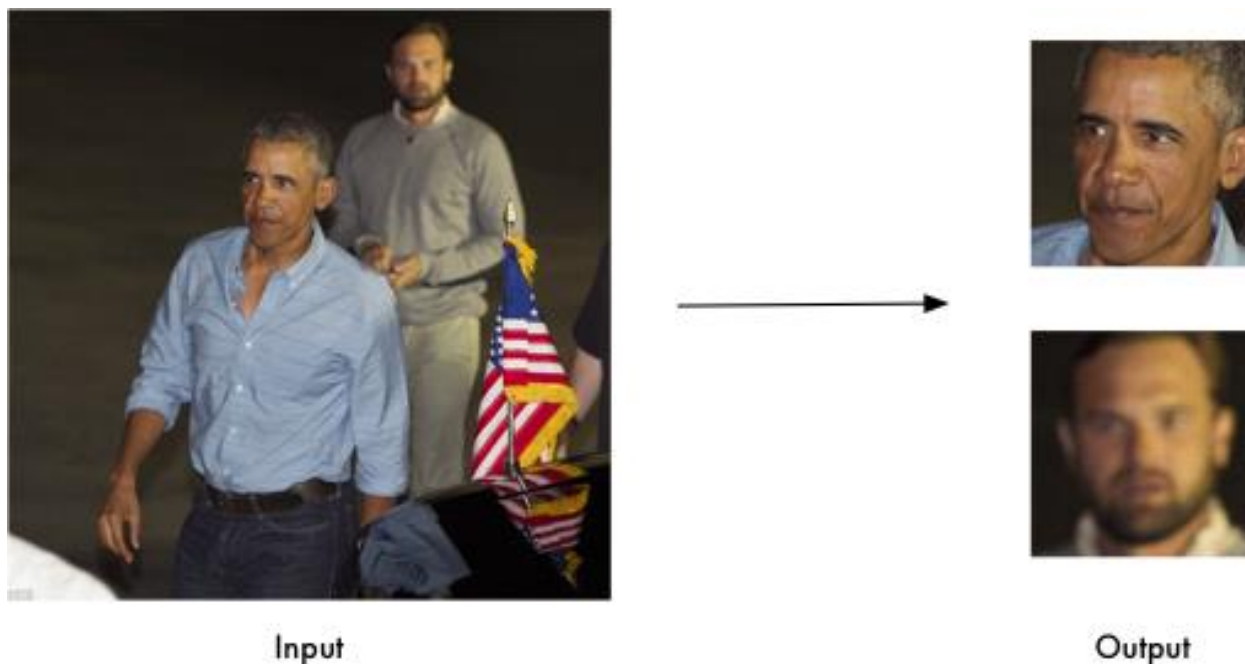


Figure 1.1-Finding faces in pictures



Input



Output

Figure 1.2-Finding and manipulating facial features in pictures



Input



Picture contains
"Joe Biden"

Output

Figure 1.3-Identification of faces in picture



Figure 1.4-Rotation of image by 360degrees and matching

1.2 ROLE OF OPENCV IN FACE RECOGNITION

The developing interest in PC vision of the previous decade. Fuelled by the consistent multiplying pace of registering power at regular intervals, face location and acknowledgment has risen above from a recondite to a mainstream space of exploration in PC vision and one of the better and effective utilizations of picture examination and calculation based arrangement. Due to the inherent idea of the issue, PC vision isn't just a software engineering space of examination, yet additionally the object of neuro-logical and mental investigations, basically in view of the overall assessment that propels in PC picture preparing and understanding exploration will give bits of knowledge into how our mind functions and the other way around. In view of general interest and interest in the matter, the creator has proposed to make an application that would permit client admittance to a specific machine dependent on a top to bottom examination of an individual's facial highlights. This application will be created utilizing Intel's open source PC vision.

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING SYSTEM:

- This strategy can distinguish the face in various postures and various degrees of enlightenment. The pivot method for 360° is utilized.
- 3-D face remaking strategies can be ordered into two classes: information driven techniques and format fitting strategies. While techniques that fall into the primary class are very much adjusted to continuous applications, strategies falling into the subsequent one produce more pleasant outcomes however are additional tedious and one-sided towards the info format.
- By utilizing the LFW data set which contains 13,233 pictures of 5,749 individuals recognized and acquired an Accuracy of 99.27%.

LIMITATIONS:

- The datasets LFW and YTF are not freely accessible for preparing.
- Gotten a low exactness in YTF dataset contrasted with facenet.
- Need isn't given to Data.
- Preparing of calculations is just done on explicit datasets like LFW and YTF.
- Time utilization high

2.2 LITERATURE SURVEY

Serial No	Author	Title	Objective
1	HamdiA.R.Ali, Mohammed Alghaili	Face Identification with Deep Learning and Filter Algorithm	Here they are using 360 degrees rotation technique. So each face is rotated from 0 to 360 degrees and compares with database images. if the image matches with database images then its shown as matched. Else if shown as not matched. And here they obtained accuracy of 92.7%
2	P.Luo, X.Wang Z.Zhu, X.Tang	Deep learning Identity preserving face space	Face acknowledgment with huge posture and brightening varieties is a difficult issue in PC vision. This paper tends to this test by proposing another learning based face portrayal: the face character saving (FIP) highlights. In contrast to traditional face descriptors, the FIP highlights can altogether diminish intra-character fluctuations, while keeping up discriminative ness between personalities. Additionally, the FIP highlights separated from a picture under any posture and brightening can be utilized to reproduce its face picture in the sanctioned view.
3	Xiaouu tang, Xiaogang wang, Yi sun	Hybrid deep learning for face verification	This paper proposes a cross breed convolutional network (ConvNet)-A vital commitment of this work is to learn undeniable level social visual highlights with rich personality closeness data. The profound ConvNets in our model beginning by separating nearby social visual

			highlights from two face pictures in correlation, which are additionally prepared through various layers to extricate undeniable level and worldwide social highlights. To keep sufficient discriminative data, we utilize the last secret layer neuron enactments of the ConvNet as highlights for face confirmation rather than those of the yield layer.
4	Y. Wen, W. Liu, Le song, Z. Yu et al, Ming li, Bhiksha Raj.	Sphereface: deep hypersphere embedding for face recognition	This paper tends to profound face acknowledgment (FR) issue under open-set convention, where ideal face highlights are required to have more modest maximal intra-class distance than insignificant between class distance under a reasonably picked metric space. In any case, barely any current calculations can adequately accomplish this rule. To this end, we propose the rakish softmax (A-Softmax) misfortune that empowers convolutional neural organizations (CNNs) to adapt precise discriminative highlights.
5	Y.Sun, X.Tang, X.Wang,	Deep learning face representation from predicting 10,000 classes	This paper proposes to become familiar with a bunch of significant level component portrayals through profound learning, alluded to as Deep secret Identity highlights (DeepID), for face confirmation. We contend that DeepID can be successfully scholarly through testing multi-class face ID undertakings, while they can be summed up to different assignments (like confirmation) and new characters inconspicuous in the preparation set.

			Additionally, the speculation ability of DeepID increments as more face classes are to be anticipated at preparing.
6	Y.Sun, X.Tang, X.Wang,	Deep Learning Face Representation by Joint Identification-Verification	The vital test of face acknowledgment is to create powerful component portrayals for diminishing relational varieties while developing between close to home contrasts. Paper explains it very well may be all around tackled with profound learning andutilizing both face distinguishing proof and check signals as management. The Deep ID confirmation highlights (DeepID2) are learned with deliberately planned profound convolutional networks.
7	M. A. Ranzato, M. Yang, L. Wolf, Y. Taigman	Web-scale training for face identification	Scaling AI techniques to exceptionally enormous datasets has pulled in impressive consideration as of late, because of simple admittance to universal detecting and information from the web. We study face acknowledgment and show that three particular properties effectively affect the adaptability of profound convolutional networks (CNN): The bottleneck of the organization fills in as a significant exchange learning regularizer, and rather than the basic intelligence, execution immersion may exist in CNN's (as the quantity of preparing tests develops); we propose an answer for easing this by supplanting the innocent irregular subsampling of the preparation set with a bootstrapping interaction.

PROPOSED SYSTEM:

We had the option to create exact appearances for human items from RGB channel pictures containing confined articles. We showed our outcomes on the Multi Human Parsing Dataset with mean pixel level exactness. Proposed organization can recognize non front facing faces and various countenances from a solitary picture. The strategy can discover applications in cutting edge assignments like facial part recognition.

In the mid 90's various calculations were produced for face acknowledgment and expansion in the requirement for face recognition. Frameworks were intended to manage video web based. Henceforth the motivation behind why there is still on-going exploration on this framework.

With the latest innovation of facial acknowledgment and discovery methods, business items have arisen on the business sectors. Notwithstanding the business achievement a couple of issues are still to be investigated. Face Recognition in two essential undertakings.

Benefits:

- High accuracy compared to existing system

CHAPTER 3

WORKFLOW

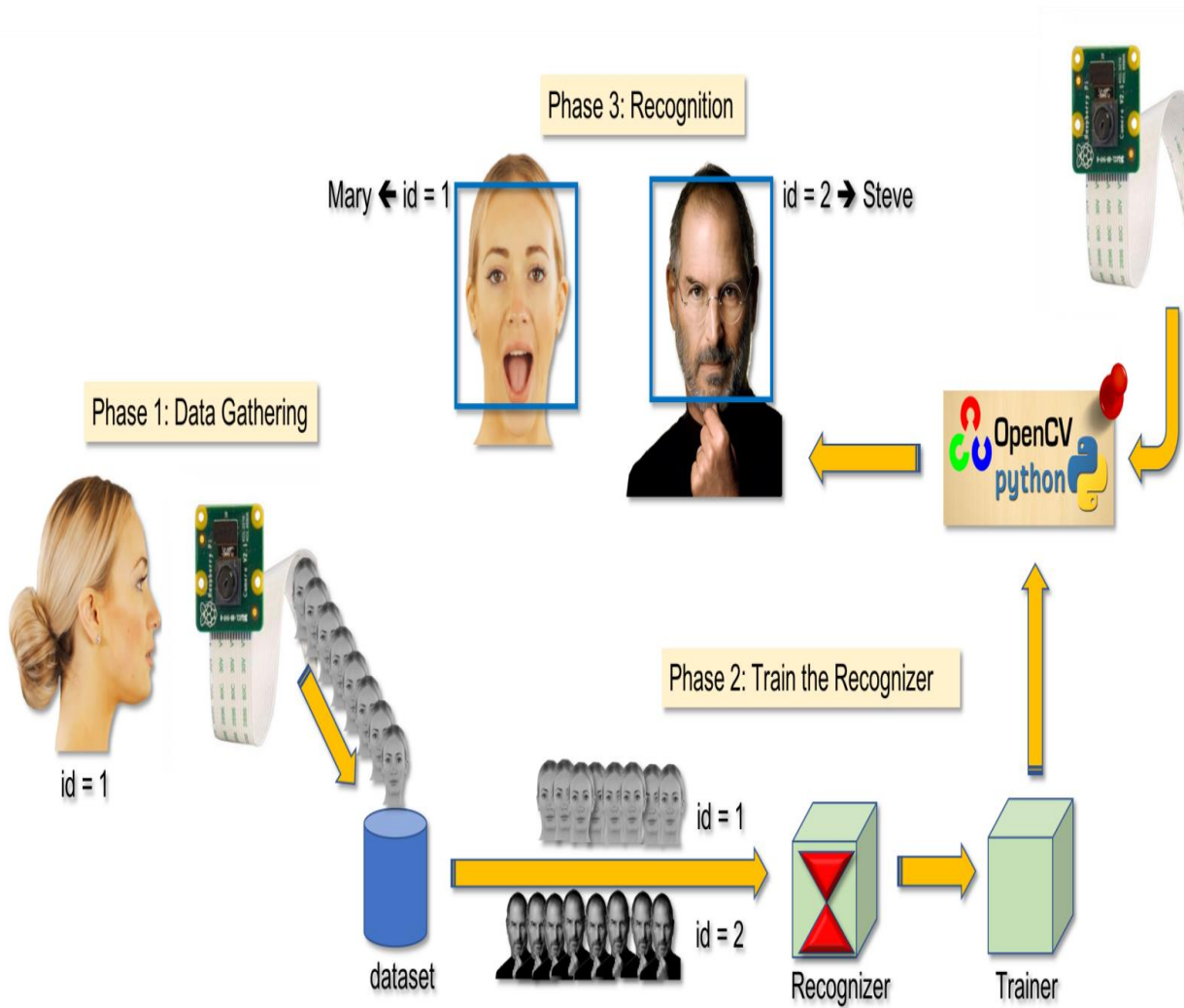


Figure 3.1-Proposed work (WORKFLOW MODEL)

PHASES:

1. Data Gathering

2. Train the Recognizer
3. Recognition

UML DIAGRAM:

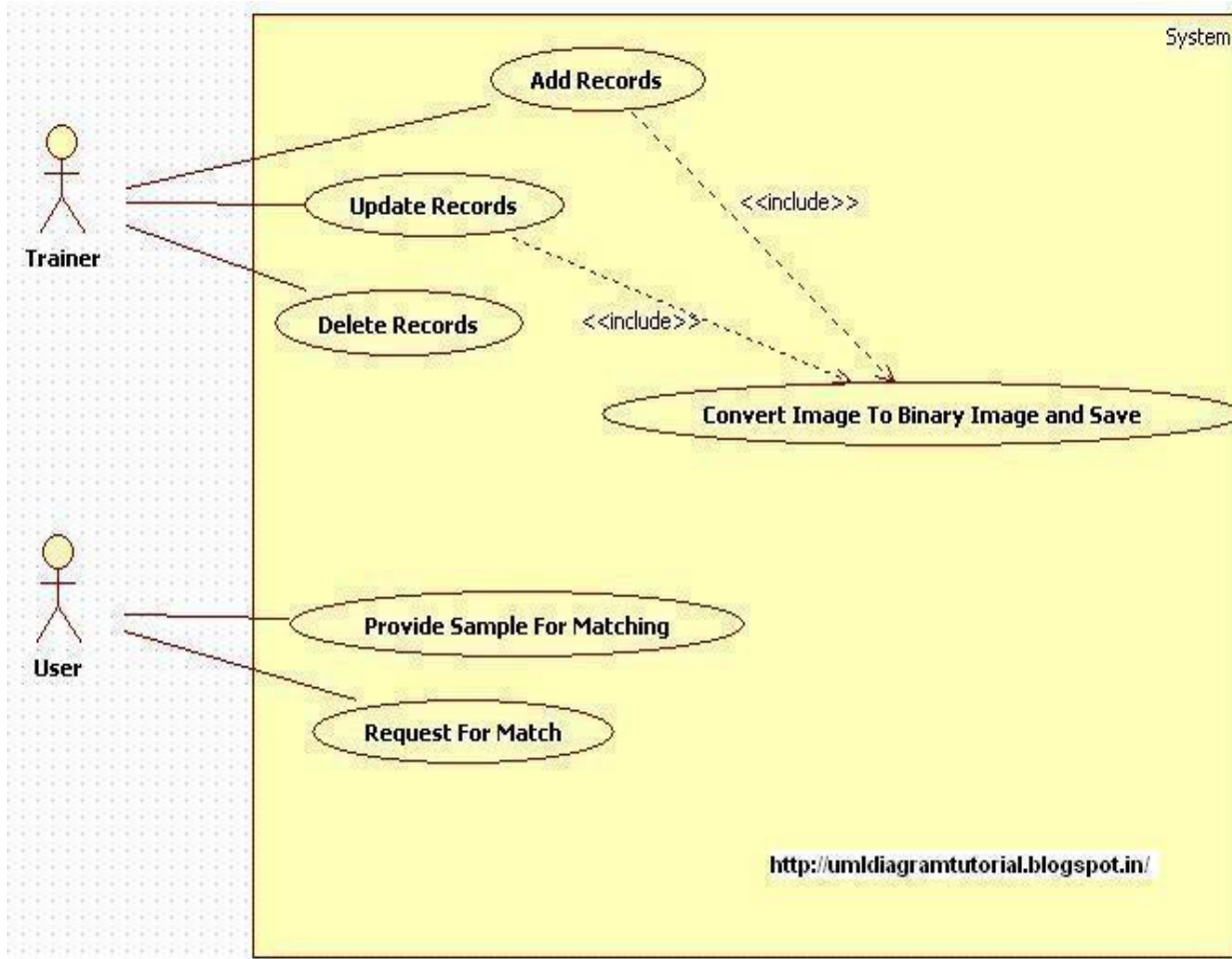
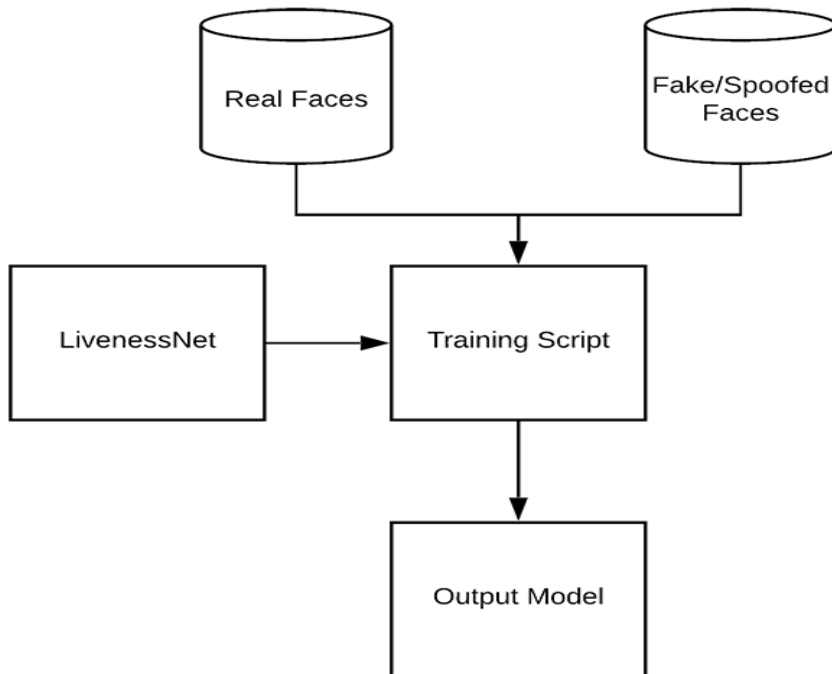


Figure 3.2-Uml diagram



ARCHITECTURE DIAGRAM:

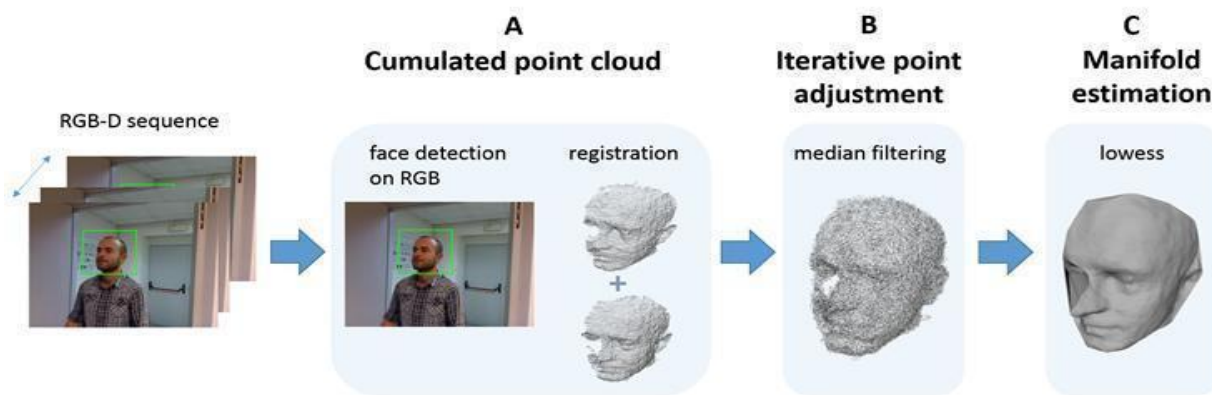


Figure 3.4-Architecture diagram

CHAPTER 4

PROPOSED WORK

Ubuntu is a Operating System(OS) it is build on Linux distribution and generally it is of free. The most recent standard delivery is 20.10 ("Groovy Gorilla"), which is upheld for a very long time. For example, any presently upheld Ubuntu variant and in Fedora). The default record director is GNOME Files, once in the past called Nautilus.

The entirety of the application programming introduced as a matter of course is free programming. Likewise, Ubuntu rearranges some equipment drivers that are accessible just in paired arrangement, yet such bundles are unmistakably set apart in the limited part.

Python is one of the popular programming language. It has many features and most of the programmers use python. And many programmers shifting to python programming language. It is spreaded to every stream. Many digitalization projects are carried out using this language. Top most trending technologies like AI, Machine-learning, DataScience uses python. The debugging will be very easy using python. And face-recognition applications can also be achieved using this.

OpenCV is a package used in python its used for vision related things. Intel created OpenCV package initially. OpenCV upholds the profound learning structures TensorFlow, Torch/PyTorch and Caffe. The entirety of the new turns of events and calculations in OpenCV are presently evolved in the C++ interface

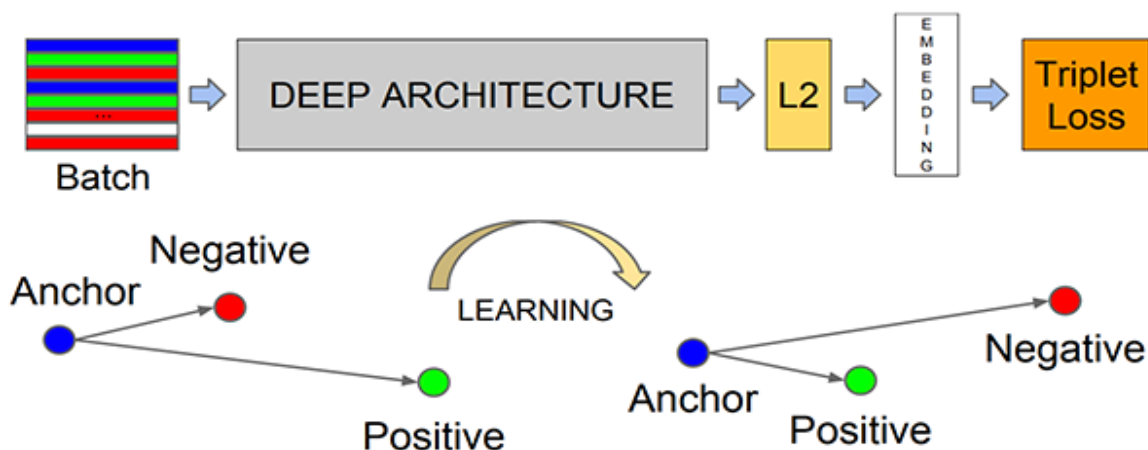
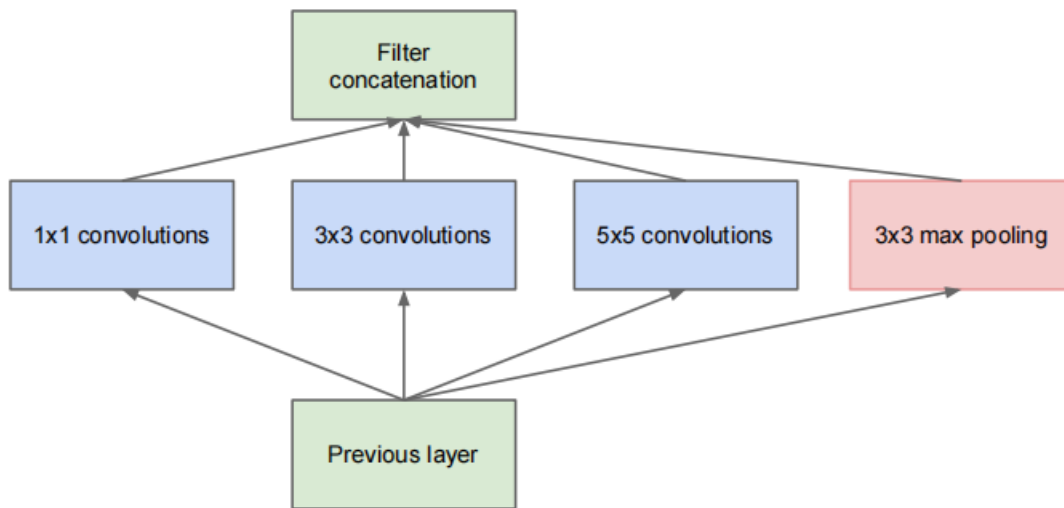


Figure 4.1-Deep learning architecture processing and learning, training of data-model.

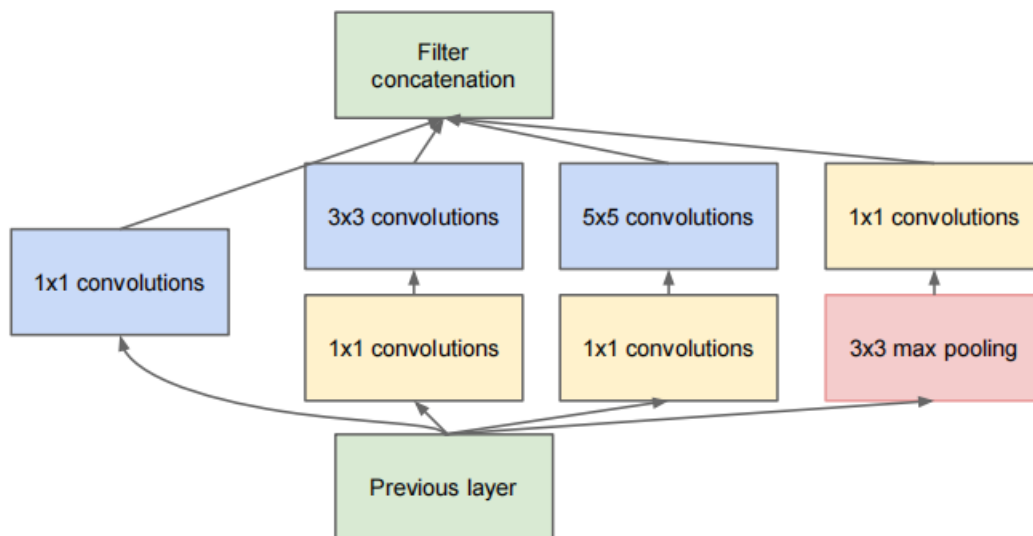
Numpy is a package used in python for performing mathematical operations. And using numpy we can store data in the form of arrays. Arrays can be a single dimension or multi dimension. For storing data in the form of matrices we use two dimensional numpy array. Numpy package is used for data analytics or any operation for doing mathematical calculations using data.

4.1 GOOGLNET ARCHITECTURE

Googlenet Architecture is one of the efficient architecture available. We can achieve maximum accuracy using this. For deep learning applications the best way to choose is GoogleNet. In this architecture for our project because for face recognition applications efficiency will be high. And it has 22 layers. In this case, the images are given directly to the input layers. However, the image must be a three Dimensional input (Red, Green and Blue). The initial layers extract the features namely edges, points and lines. Pooling layers are used for reducing the feature maps and the next convolutional layer aggregates the overall features from the above primary features. These features are then converted into positive numbers and zeros with Rectified Linear Activation function and are then fed to fully connected network.



(a) Inception module, naïve version



(b) Inception module with dimensionality reduction

Figure 4.2-Inception module (a&b)

Team	Year	Place	Error (top-5)	Uses external data
SuperVision	2012	1st	16.4%	no
SuperVision	2012	1st	15.3%	Imagenet 22k
Clarifai	2013	1st	11.7%	no
Clarifai	2013	1st	11.2%	Imagenet 22k
MSRA	2014	3rd	7.35%	no
VGG	2014	2nd	7.32%	no
GoogLeNet	2014	1st	6.67%	no

Team	Year	Place	mAP	external data	ensemble	approach
UvA-Eurovision	2013	1st	22.6%	none	?	Fisher vectors
Deep Insight	2014	3rd	40.5%	ImageNet 1k	3	CNN
CUHK DeepID-Net	2014	2nd	40.7%	ImageNet 1k	?	CNN
GoogLeNet	2014	1st	43.9%	ImageNet 1k	6	CNN

Table 4.1-Comparison of Googlenet Architecture with other architectures

type	patch size/ stride	output size	depth	#1×1	#3×3 reduce	#3×3	#5×5 reduce	#5×5	pool proj	params	ops
convolution	7×7/2	112×112×64	1							2.7K	34M
max pool	3×3/2	56×56×64	0								
convolution	3×3/1	56×56×192	2		64	192				112K	360M
max pool	3×3/2	28×28×192	0								
inception (3a)		28×28×256	2	64	96	128	16	32	32	159K	128M
inception (3b)		28×28×480	2	128	128	192	32	96	64	380K	304M
max pool	3×3/2	14×14×480	0								
inception (4a)		14×14×512	2	192	96	208	16	48	64	364K	73M
inception (4b)		14×14×512	2	160	112	224	24	64	64	437K	88M
inception (4c)		14×14×512	2	128	128	256	24	64	64	463K	100M
inception (4d)		14×14×528	2	112	144	288	32	64	64	580K	119M
inception (4e)		14×14×832	2	256	160	320	32	128	128	840K	170M
max pool	3×3/2	7×7×832	0								
inception (5a)		7×7×832	2	256	160	320	32	128	128	1072K	54M
inception (5b)		7×7×1024	2	384	192	384	48	128	128	1388K	71M
avg pool	7×7/1	1×1×1024	0								
dropout (40%)		1×1×1024	0								
linear		1×1×1000	1							1000K	1M
softmax		1×1×1000	0								

Table 4.2-GoogleNet incarnation of the Inception architecture

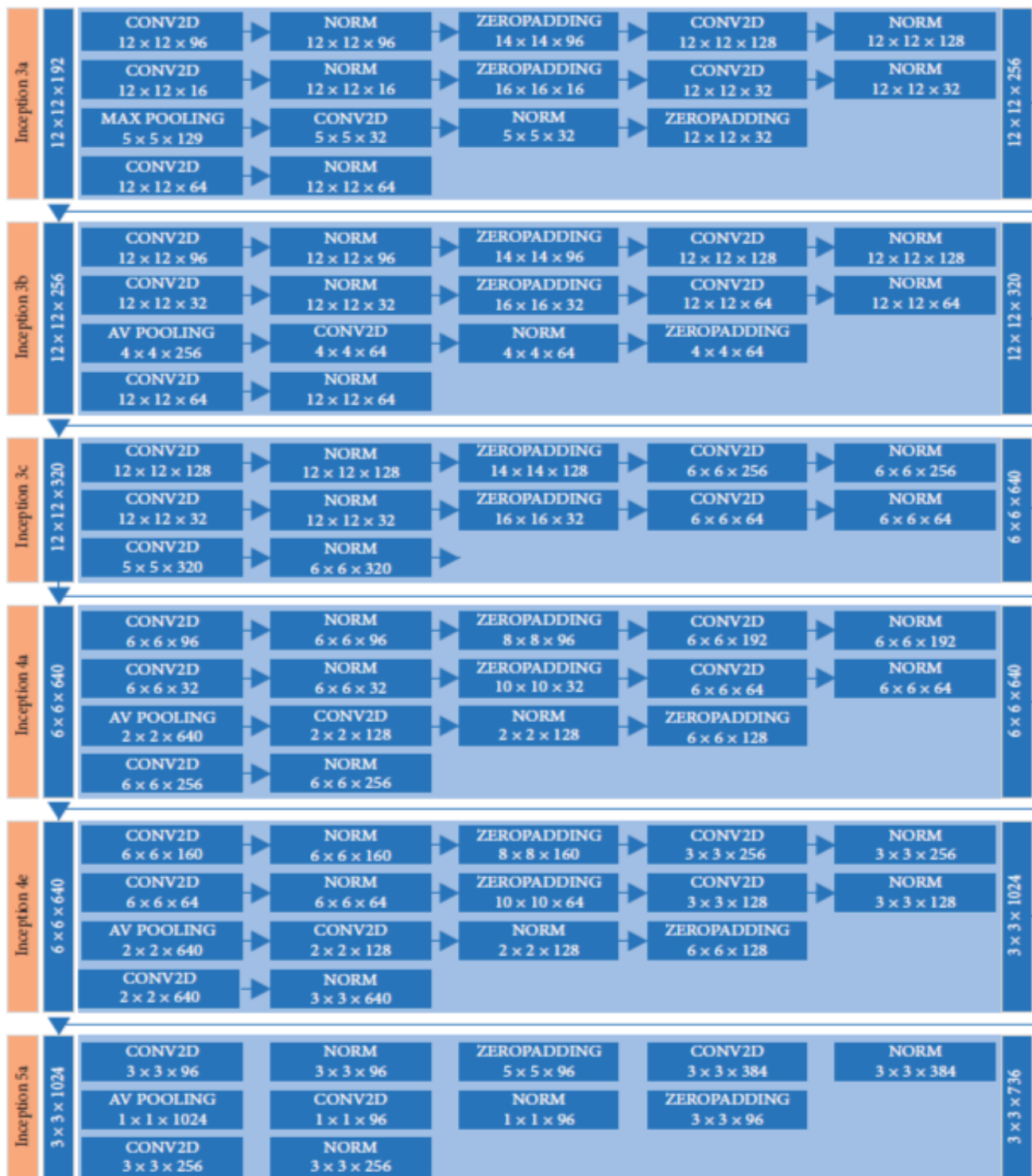


FIGURE 3: The structure of the inceptions used in the model.

Figure 4.3-The structure of the inceptions used in the model



Figure 3: GoogLeNet network with all the bells and whistles.

4.2 RESULTS AND DISCUSSION

The screenshots of the output are given below: It is found that the images match perfectly with the given input images. Real time images are used for testing and the proposed software works well for the real time images also.

OUTPUT SCREENSHOTS:

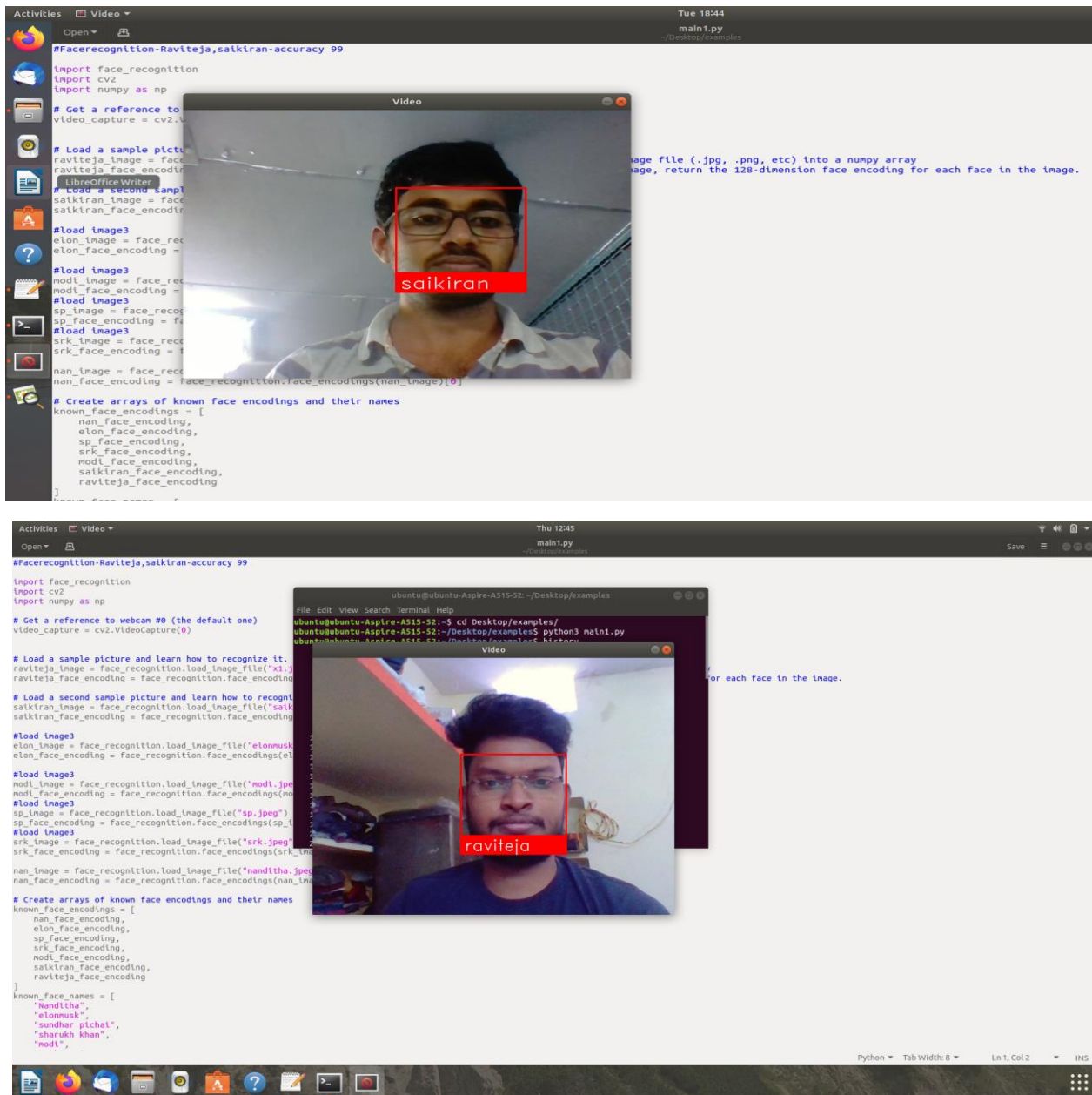


Figure 4.4- Straight forward faces output with name labelled

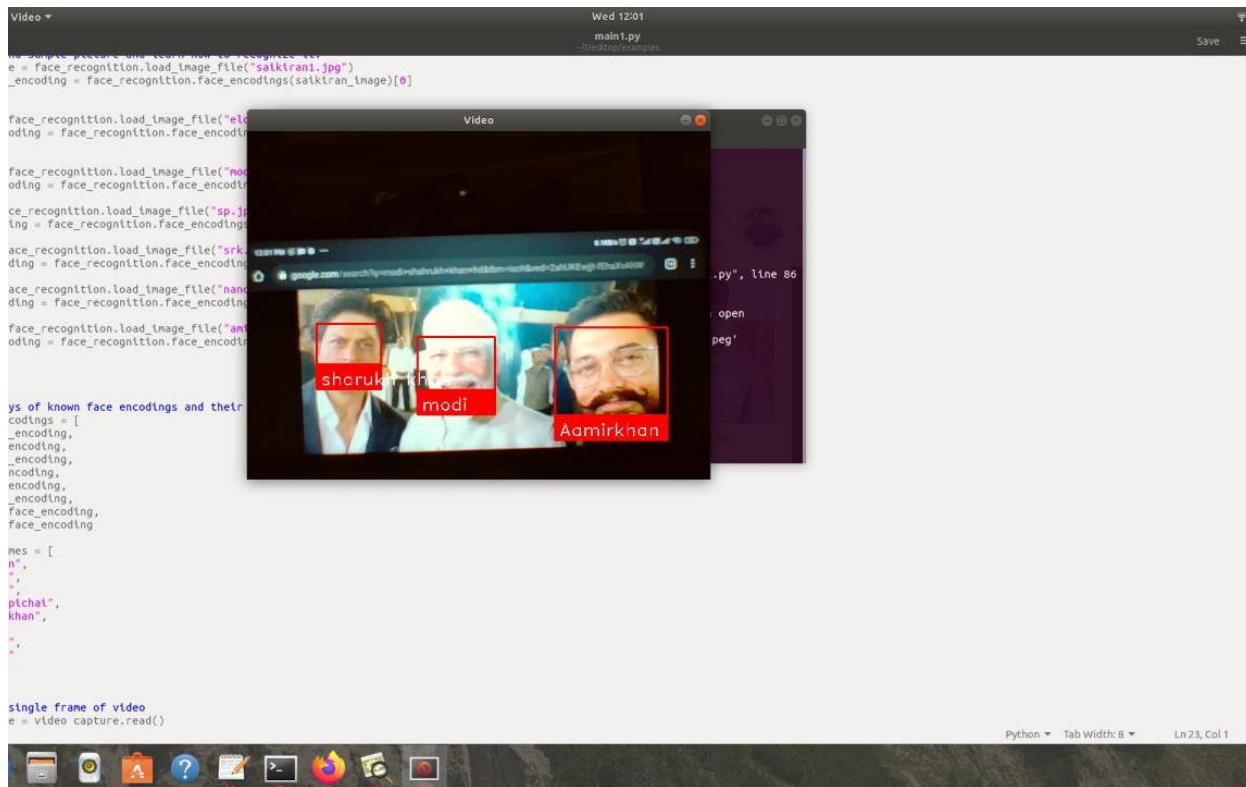


Figure 4.5 - Face detection in a group photo

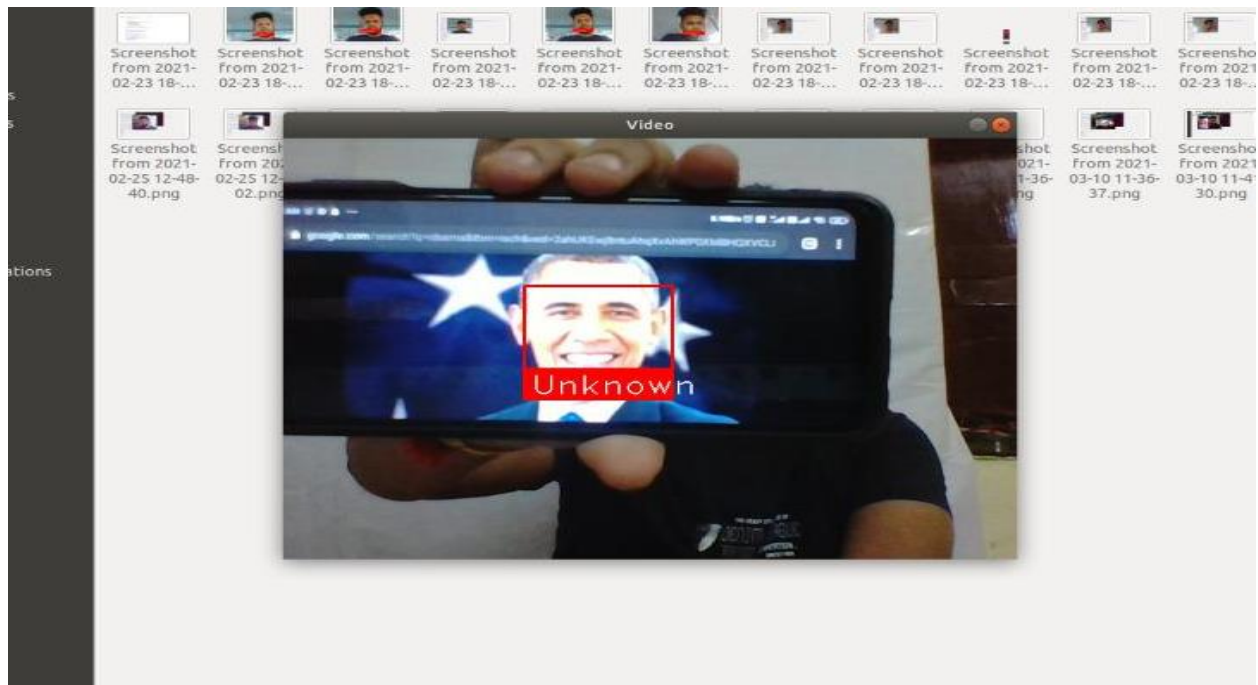


Figure 4.6- Face not present in a dataset

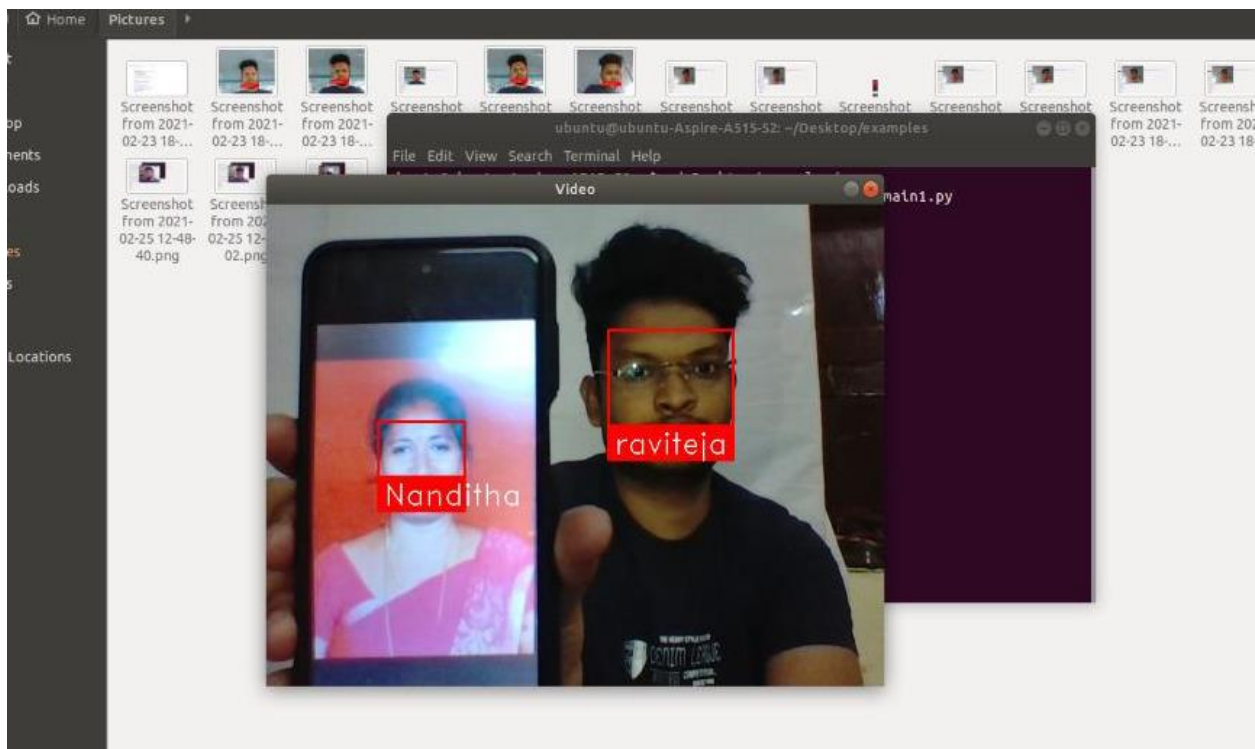
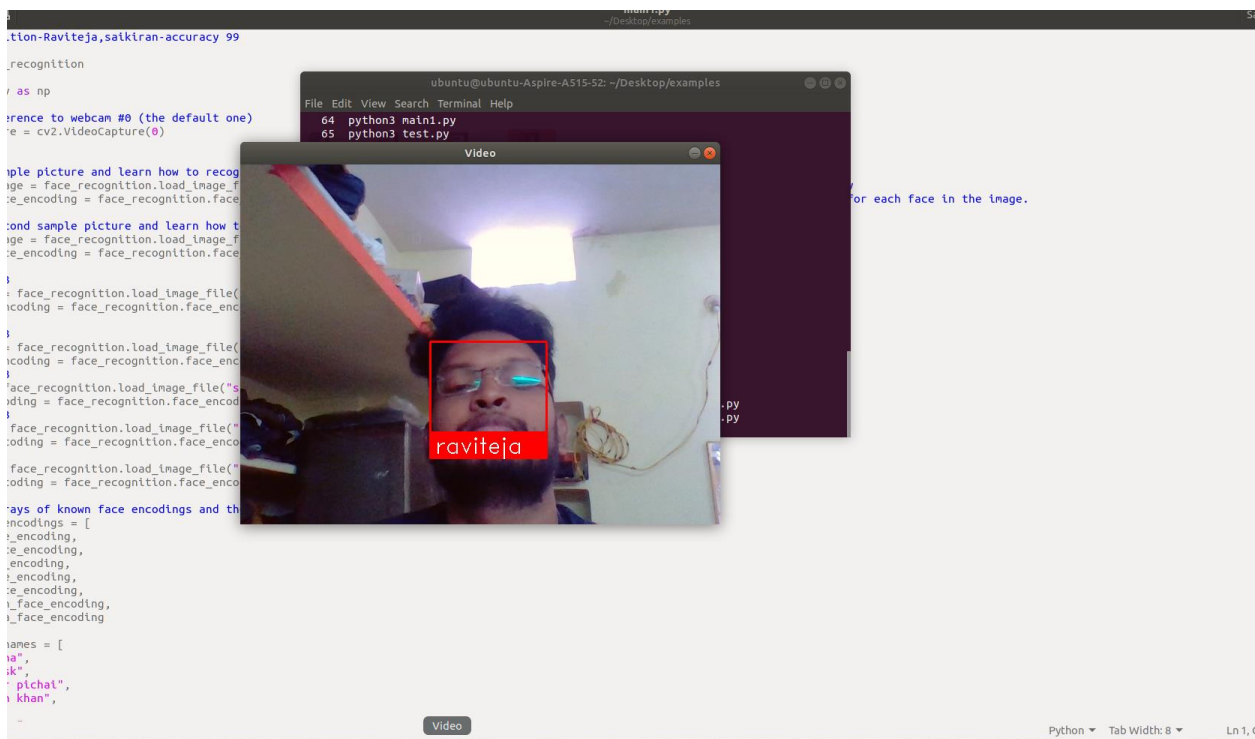
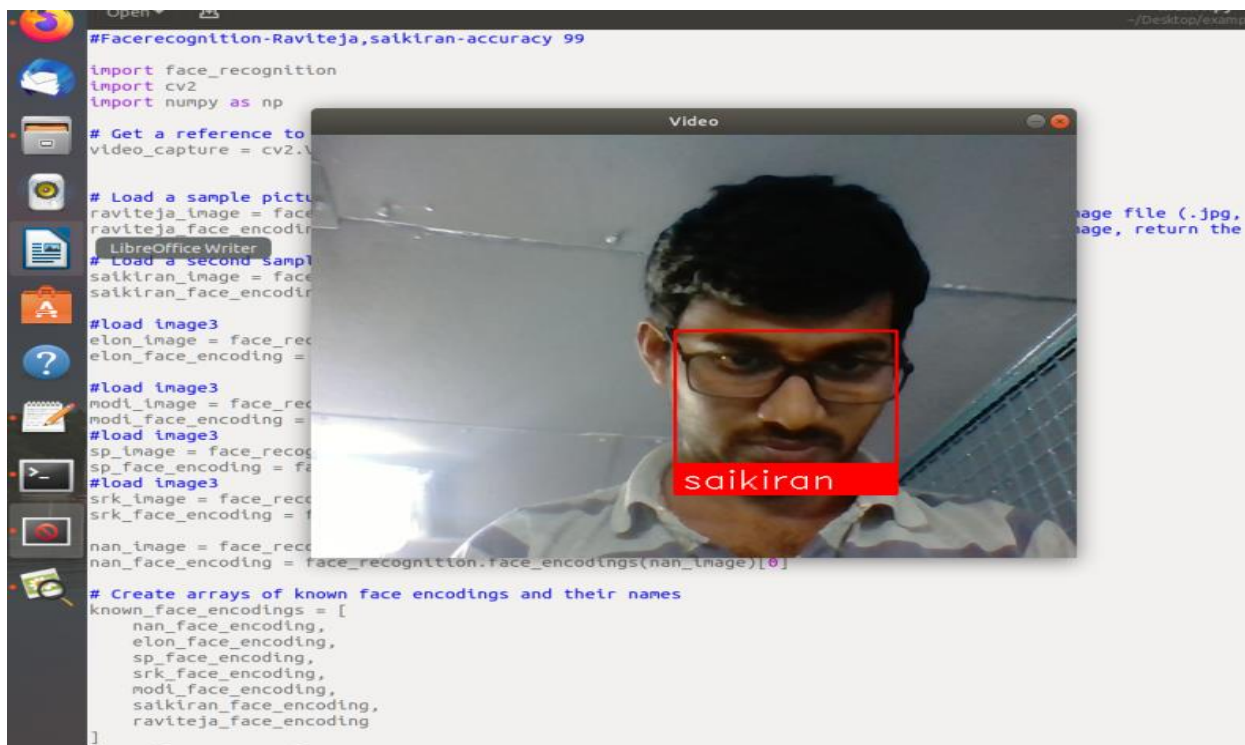
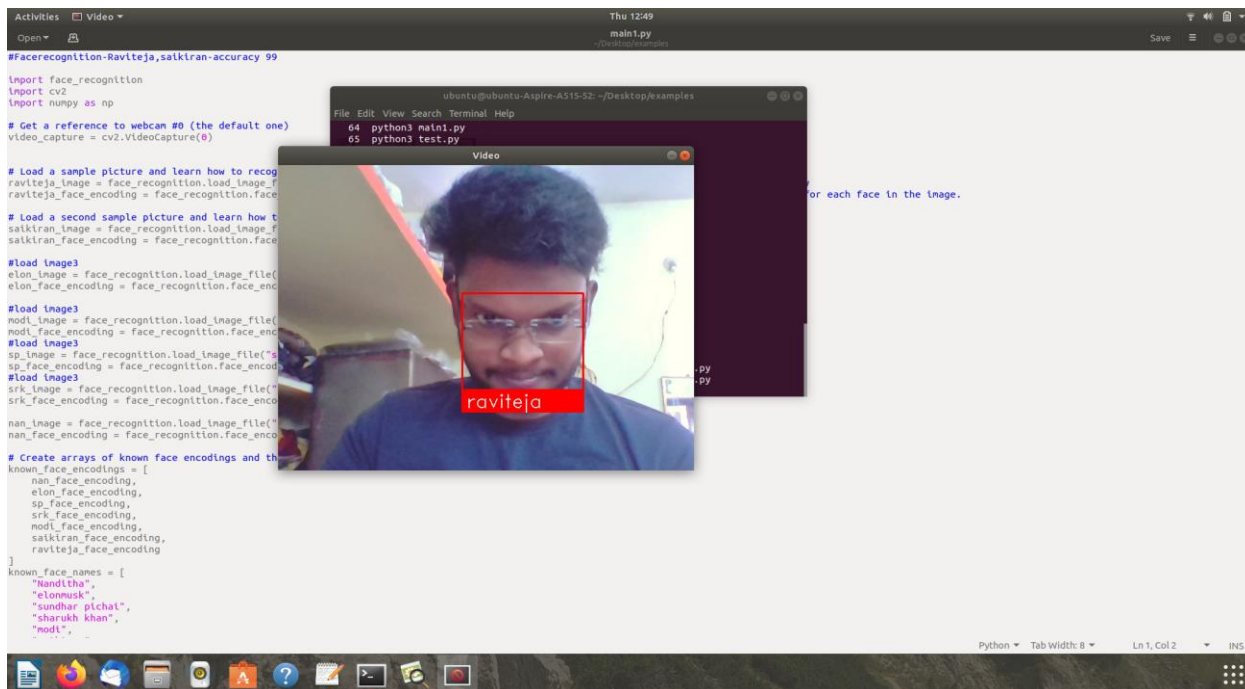


Figure 4.7- Faces present in dataset





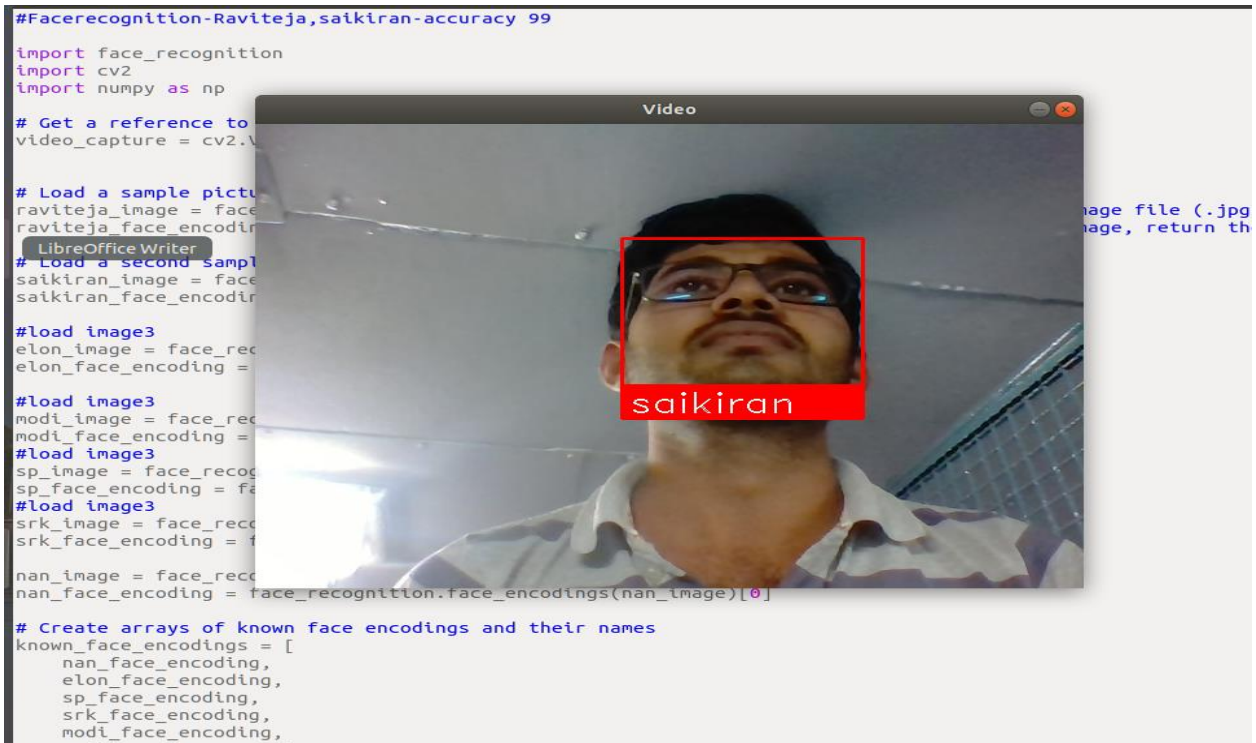
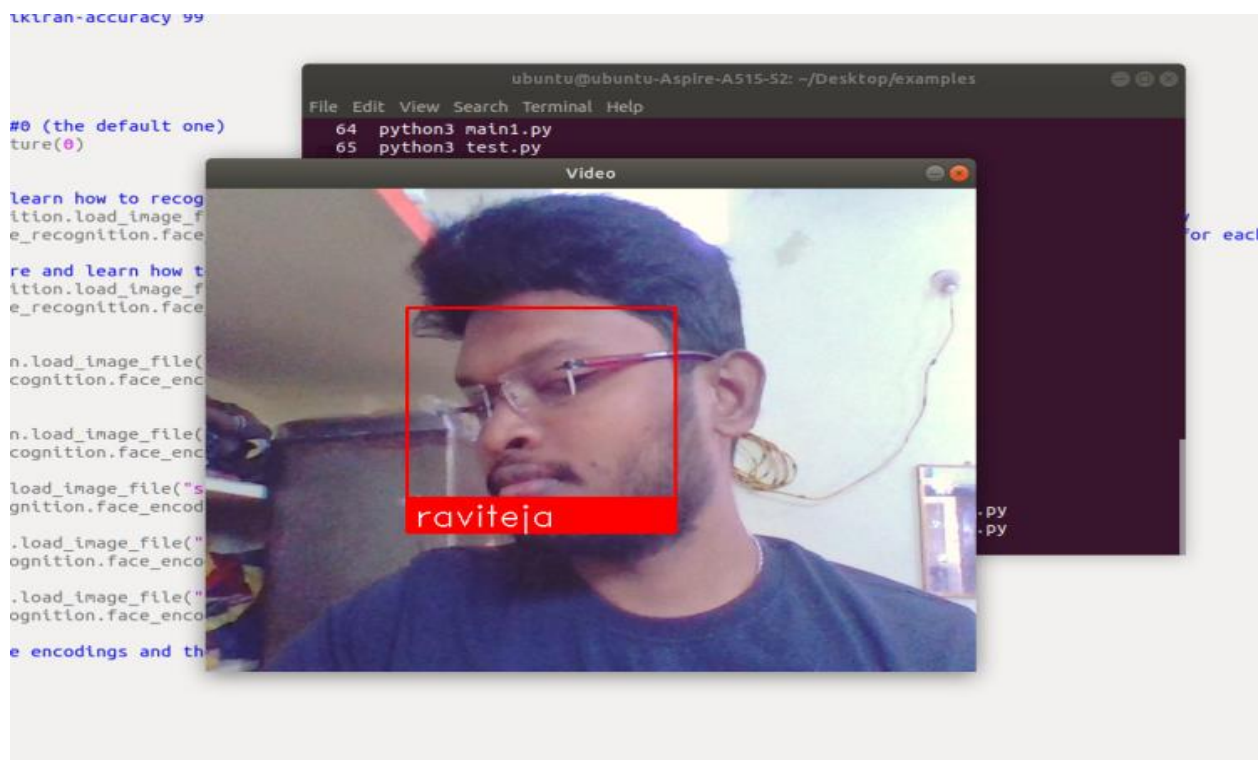


Figure 4.8- Face detection when face position(up&down)



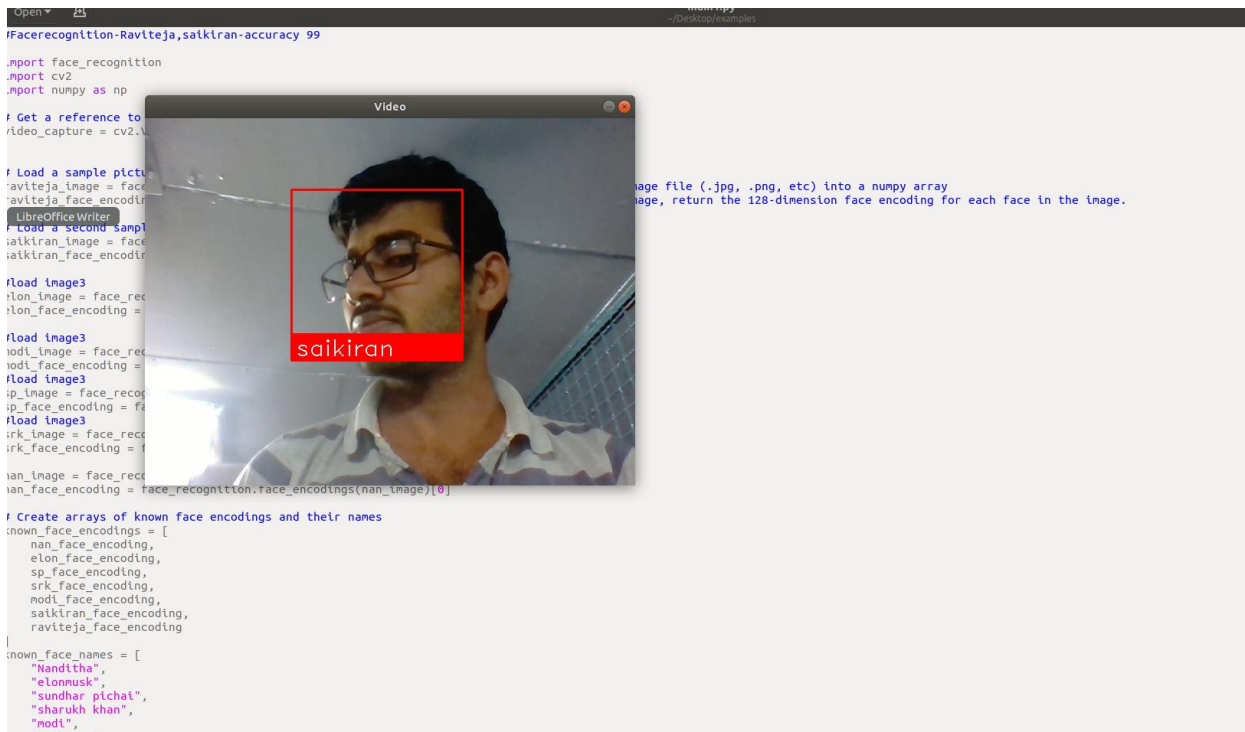


Figure 4.9 Face detection when face position(left & right)

CHAPTER 5

CONCLUSION & FUTURE WORK

Studied the performance of Googlenet for face recognition and to improve the sensitivity and specificity by modifying the architecture of Googlenet. Taking the faces from a webcam and detecting the faces of people and labelling their names. In this work, which is a face recognition theme is implemented by deep learning techniques and obtained an accuracy of 99.38%. Here face-recognition package is used for so that faces are detected using this. And also we are using OpenCV package for converting images to numpy array. This system was much convenient for the user and it gives better security. For future development we want to add an application like Car driver detection to this project and want to increase the efficiency of the project.

REFERENCES

- [1] Mohammed Alghaili, Hamdi A. R. Ali, “Face Identification with Deep Learning and Filter Algorithm” received 21 November 2019; Accepted 2 April 2020; Published 1 August 2020.
- [2] P. Luo, Z. Zhu, X. Wang, and X. Tang, “Hybrid deep learning for face verification,” in Proceedings of International Conference on the Computer Vision, pp. 113–120, IEEE, Sydney, NSW, Australia, December 2013.
- [3] W. Liu, Y. Wen, Z. Yu et al., “Sphereface: deep hypersphere embedding for face recognition,” in the Proceedings of the Conference of Computer Vision and Pattern Recognition (CVPR), July 2017.
- [4] P. Luo, Z. Zhu, P. Luo, X. Tang, X. Wang, “Recover canonical view faces in the wild with deep neural networks,” pp. 1404–3543, 2014, <https://arxiv.org/abs/1404.3543>.
- [5] X. Wang, Y. Sun, and X. Tang, “Deep learning face representation from predicting 10,000 classes,” in the Proceedings of the Conference on Computer Vision and the Pattern Recognition, pp. 1891–1898, IEEE, Columbus, OH, USA, June 2014.
- [6] X. Wang, Y. Sun, Y. Chen, and X. Tang, “Deep learning face representation by joint identification-verification,” Advances in the Neural Information Processing Systems, pp. 1988–1996, 2014, <https://arxiv.org/abs/1406.4773>.
- [7] M. Yang, Y. Taigman, M. A. Ranzato, and L. Wolf, “Web-scale training for face identification,” in Proceedings of the Conference on Computer Vision and Pattern Recognition, pp. 2746–2754, IEEE, Boston, MA, USA, June 2015.
- [8] Z. Zhu, P. Luo, X. Wang, and X. Tang, “Deep learning and disentangling face representation by multi-view perceptron,” in Proceedings of the NIPS, Quebec, Canada, December 2015.
- [9] K. Q. Weinberger and L. K. Saul, “Distance metric learning for large margin nearest neighbour classification,” Journal of Machine Learning Research, vol. 10, pp. 207–244, 2009.
-

- [10] X. Wang, Y. Sun, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in Proceedings of Conference on Computer Vision and Pattern Recognition, pp. 2892–2900, IEEE, Boston, MA, USA, June 2015.
- [11] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," pp. 1411–7923, 2014, <https://arxiv.org/abs/1411.7923>.
- [12] Y. Wang, J. Shen, S. Petridis, and M. Pantic, "A real-time and unsupervised face re-identification system for human-robot interaction," Pattern Recognition Letters, vol. 128, 2018.
- [13] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in the Proceedings of the International Conference on Computer Vision, pp. 3730–3738, IEEE, Santiago, Chile, December 2015.
- [14] P. Luo, Z. Zhu, X. Wang, and X. Tang, "Deep learning identity-preserving face space," in Proceedings of the IEEE International Conference on the Computer Vision, pp. 1489–1496, IEEE, Sydney, NSW, Australia, December 2013.

APPENDIX

APPLICATIONS OF FACE RECOGNITION

1.FOR UNLOCK PHONES

An assortment of telephones are presently utilizing face acknowledgment to open telephones. The present security for mobile phone is focuses on this technique. By using deep-learning techniques with high complexity we can achieve this.



2.ADVERTISING

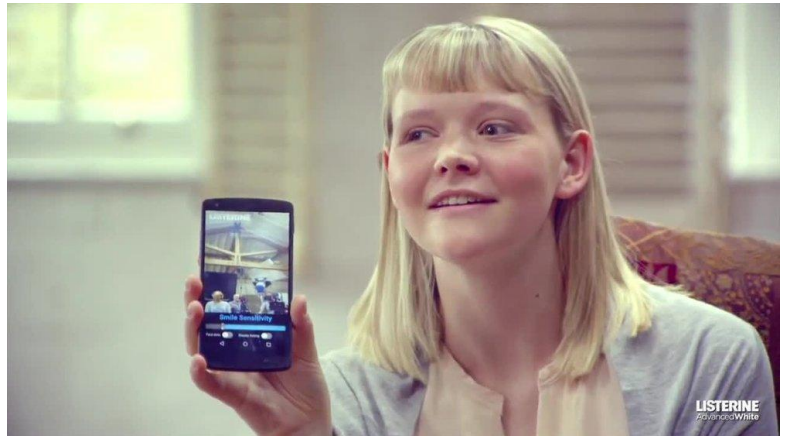
Face acknowledgment can make publicizing more focused by making taught surmises individuals' age and sexual orientation. Organizations like Tesco are as of now anticipating introducing screens at service stations with face acknowledgment worked in. It won't be long until face-acknowledgment turns into an omni-present promoting innovation



3.HELPING THE BLIND

Listerine has built up a weighty facial acknowledgment application that helps the visually impaired utilizing face acknowledgment. The application perceives when individuals are grinning and cautions the visually impaired

individual with a vibration. This can help them better comprehend social circumstances.



4.IDENTIFYING PEOPLE ON SOCIAL MEDIA PLATFORMS

Facebook utilizes face acknowledgment innovation to consequently perceive when Facebook individuals show up in photographs. This makes it simpler for individuals to discover photographs they are in and can propose when specific individuals ought to be labeled in photographs.



5. RECOGNIZING VIPS AT SPORTING EVENTS

Face acknowledgment can be utilized to give fans a superior encounter. Face acknowledgment can in the split second perceive when season ticketholders go to games. Occasion settings can offer them loot, let them skip lines and other VIP advantages that bring about more noteworthy season ticketholder maintenance.



6. SECURE TRANSACTIONS

In China, there is a monetary administrations organization considered Ant Financial that empowers clients to pay for suppers by filtering their countenances.

Clients place orders through a computerized menu, and afterward use face examine as an installment alternative. Subsequent to giving their phone number they would then be able to buy their feast.



7.VALIDATING IDENTITY AT ATMS

It appears to be likely that face outputs will ultimately supplant ATM cards totally. In any case, meanwhile, face acknowledgment can be utilized to ensure that people utilizing ATMs cards are who they say they are. Face acknowledgment is as of now being utilized at ATMs in Macau to ensure people groups' characters.



8.FINDING LOST PETS

Discovering Rover is an application that attempts to assist proprietors with rejoining lost pets. The application utilizes face acknowledgment (yet it's the substance of a creature for this situation)



to coordinate photographs that pet proprietors transfer to a data set of photographs of pets in covers. The application can then quickly alarm proprietors if their pets are found.

CODE

#Face Recognition-Raviteja, saikiran-accuracy 99.38%

```
import face_recognition                #Recognize and manipulate faces from Python
import cv2                             #helps for computer vision problems(helps to convert
image into numpy array)

import numpy as np                     #package used to solve
mathematical problems

video_capture = cv2.VideoCapture(0)    #used to trigger(on) the
camera
```

Load a sample picture and learn how to recognize it.

```
raviteja_image = face_recognition.load_image_file("x1.jpeg")    #Loads an image file (.jpg,
.png, etc) into a numpy array
```

```
raviteja_face_encoding = face_recognition.face_encodings(raviteja_image)[0] #Given an image, return the
128-dimension face encoding for each face in the image.
```

```
saikiran_image = face_recognition.load_image_file("saikiran1.jpg")
```

```
saikiran_face_encoding = face_recognition.face_encodings(saikiran_image)[0]
```

#load image3

```
elon_image = face_recognition.load_image_file("elonmusk.jpg")
```

```
elon_face_encoding = face_recognition.face_encodings(elon_image)[0]
```

#load image4

```

modi_image = face_recognition.load_image_file("modi.jpeg")
modi_face_encoding = face_recognition.face_encodings(modi_image)[0]

#load image5
sp_image = face_recognition.load_image_file("sp.jpeg")
sp_face_encoding = face_recognition.face_encodings(sp_image)[0]

#load image6
srk_image = face_recognition.load_image_file("srk.jpeg")
srk_face_encoding = face_recognition.face_encodings(srk_image)[0]

#load image7
nan_image = face_recognition.load_image_file("nanditha.jpeg")
nan_face_encoding = face_recognition.face_encodings(nan_image)[0]

#load image8
amir_image = face_recognition.load_image_file("amirkhan.jpeg")
amir_face_encoding = face_recognition.face_encodings(amir_image)[0]

# Create arrays of known face encodings and their names
known_face_encodings = [
    amir_face_encoding,
    nan_face_encoding,
    elon_face_encoding,
    sp_face_encoding,
    srk_face_encoding,
    modi_face_encoding,
    saikiran_face_encoding,
    raviteja_face_encoding
]

known_face_names = [

```

```

"Aamirkhan",
"Nanditha",
"elonmusk",
"sundhar pichai",
"sharukh khan",
"modi",
"saikiran",
"raviteja"
]

while True:

    # Grab a single frame of video

    ret, frame = video_capture.read()

    # Convert the image from BGR color (which OpenCV uses) to RGB color (which face_recognition uses)
    rgb_frame = frame[:, :, :-1]

    # Find all the faces and face encodings in the frame of video
    #captures multiple faces
    and separates faces individually

    face_locations = face_recognition.face_locations(rgb_frame)

    face_encodings = face_recognition.face_encodings(rgb_frame, face_locations)

    for (top, right, bottom, left), face_encoding in zip(face_locations, face_encodings):

        # See if the face is a match for the known face(s)

        matches = face_recognition.compare_faces(known_face_encodings, face_encoding)

        name = "Unknown"

```

```

face_distances = face_recognition.face_distance(known_face_encodings, face_encoding)
best_match_index = np.argmin(face_distances)
if matches[best_match_index]:
    name = known_face_names[best_match_index]

# Draw a box around the face
cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)

# Draw a label with a name below the face
cv2.rectangle(frame, (left, bottom - 35), (right, bottom), (0, 0, 255), cv2.FILLED)
font = cv2.FONT_HERSHEY_DUPLEX
cv2.putText(frame, name, (left + 6, bottom - 6), font, 0.8, (255, 255, 255), 1)

# Display the resulting image
cv2.imshow('Video', frame)

# Hit 'q' on the keyboard to quit!
if cv2.waitKey(1) & 0xFF == ord('q'):
    break

# Release handle to the webcam
video_capture.release()          #release the camera
cv2.destroyAllWindows()

```