Human Tracking System using Face Detection and Recognition

Nitesh Patel[1], Siddhi Mahamunkar[2], Prathamesh Malage[3]

Guided By: Arvind Sangale[4]

[1,2,3,4]Electronics and Telecommunication Engineering, Rajiv Gandhi Institute of Technology, Mumbai-400053, Maharashtra, India

---***---

Abstract – *Human Tracking system using face detection and recognition is an attempt to help track people swiftly with the aid of python language and integrating the OpenCV and smtplib and NumPy modules. In India, with a population of 1.3 billion it is a humongous task to locate people. This proposed system aims at making the task of finding people simple and informing the concerned authorities with emails along with the timestamp and image of the recognized person. This system creates a database of people in the detection phase and with the help of cameras a missing person or an on-the-run criminal can be tracked. This system incorporates the LBPH algorithm for face detection and recognition process.*

***Key Words*: Python language, OpenCV, SMTPLIB, NumPy, LBPH, Recognition process.**

1. INTRODUCTION

Consider metropolitan cities like Mumbai, Banglore, Delhi where the average population per square kilometre is 27k ,17k and 19k respectively. Now, imagine a situation where a 5-year old kid gets lost in these cities. This becomes an extremely tedious and over-whelming task for the authorities to track the child. Now, imagine a scenario where a person could be tracked within hours, the amount of time, efforts and resources that could be saved are unimaginable. All you would have to do is upload the photo of the missing person in the database, the system would track the person if he/she comes near a camera and send an email along with timestamp and image of the person to the officials such as Police and family members. This system is designed with the help of Python-3 language and its various modules such as OpenCV which deals with computer vision and comes with a 'Face Recognizer' class. Smtplib module which is used for sending emails to the officials.

1.1 Local Binary Pattern Histogram (LBPH)

There are various algorithms for the face recognition process. Here, the system has incorporated the Local Binary Pattern Histogram or LBPH algorithm.

LBP is a powerful tool for feature and when used with Histogram of gradients its performance is significantly improved in some datasets.

The LBP extracts the feature in the following manner

- Divide the examined window into cells.
- For each pixel in a cell, compare the pixel to each of its 8 neighbors . Follow the pixels along a circle, i.e. clockwise or counterclockwise.
- Where the center pixel's value is greater than the neighbor's value, write "0". Otherwise, write "1". This gives an 8-digit binary number.
- Compute the histogram, over the cell, of the frequency of each "number" occurring. This histogram can be seen as a 256-dimensional feature vector.
- Optionally normalize the histogram.
- Concatenate (normalized) histograms of all cells. This gives a feature vector for the entire window.

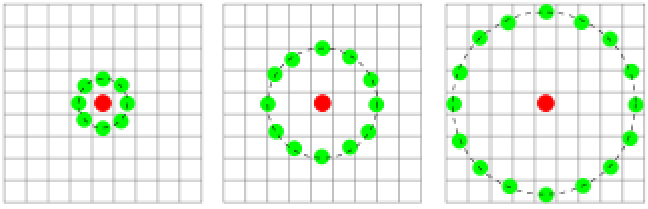


Fig-1: Calculation of LBP

2. WORKING OF PROPOSED SYSTEM

The project on Human Tracking System using Face Detection and Recognition has been done with the help of various Python modules such as OpenCV, SMTP, NumPy etc. The detection phase or the collecting datasets of people is done with the help of Haar Classifier. In Haar Cascades a lot of positive and negative images are trained in order to extract prominent features of the face such as eyes, nose, lips etc. By using this technique, we can store images of people in a database. By clicking the images in various scenarios such as illumination, pose variation, light exposure, we can have a greater accuracy in the recognition phase. Apart from the Haar cascade frontal face file, we have also incorporated the Haar profile face file in order to increase the accuracy. This enables us to recognize the people even if they are not facing towards the camera but are facing towards right or left side. In order to capture the images, we have used laptop's webcam. A detailed scenario of the face detection process can be understood from the flowchart below. In the recognition phase, we are first asked to enter the name of the person who is to be searched or recognized. The laptop's webcam the searched for the person. When the person or the entity is recognized we are given an alert via E-mail. For sending E-mails, we have used Python's SMTP package. This package enables us to send E-mails to concerned authorities and include attachments. The E-mail attachment includes the photo of the recognized person along with the ip-address on which the person was recognized.

2.1 Face Detection

The first step of this proposed system is to create a database of people. This system captures images of every person. The images in the database vary in terms of parameters such as poses, facial expressions, and illumination. Each person's images are stored in a file along with his/ her name which is later useful in recognition phase. Furthermore, Python's Tkinter module has been incorporated. With this we can wrap this entire system in a Graphical User Interface (GUI) to make it more user friendly. This is enabled using the Haar Cascade Classifiers. This requires a lot of positive and negative images to be trained and then the features are extracted from it. To extract it there are certain Haar features.

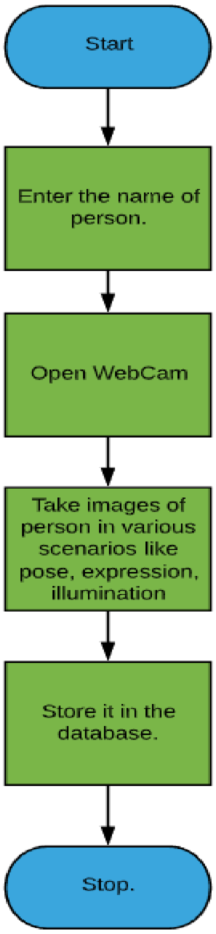

Start
Enter the name of person.
Open WebCam
Take images of person in various scenarios like pose, expression, illumination
Store it in the database.
Stop.

Fig-2- Flowchart of Image Detection process.

2.2 Face Recognition

Once the datasets have been created, then it's time for the recognition process. In this phase, first the email address of officials such as police or email address of kins is to be entered. Then the next step is to enter the name of the person, who has been missing or a criminal who has escaped. As soon as this is done, the cameras that are connected to the PC starts, like the webcam or USB camera and then searches for the person. It compares the histogram of every face in front of the camera with the histogram of missing person stored in the database. Then it returns the closest histogram value. If the name of entered person is not in the database then the system will flash a message saying that the person does not exist in database. In this scenario, a person's latest passport size photo could be uploaded and then he/she can be searched. If the person is found then the image of the person is sent along with timestamp. For the recognition process we have implemented LBPH algorithm, there are other techniques such as eigenfaces and fisherfaces which are affected by light and LBPH is an improvement on them. As a result, we have incorporated LBPH algorithm for face detection. The LBP does not look at the image but compares each pixel to its neighboring pixel. Take a 3x3 window, and move it across each part in the picture, all the pixel values are compared with the middle pixel and if that pixel is greater than the reference pixel i.e. the middle pixel then it is equated to 1 or else it is equated to 0, as shown in the above figure. When you finish doing this on the whole image, you will have a list of local binary patterns. Once you get a list of local binary pattern and convert it into a decimal number as shown above, you create a histogram of it. Now, if you have 100 images in the training sets there will be 100 such histogram samples. LBPH will extract them and store them for later recognition. The algorithm also keeps track of which histogram belongs to which person. When a new face appears in front of the camera, it creates a histogram of that image and compares it with the histogram that are already stored in it. It then returns the name of the person with which the newly generated histogram best matches.

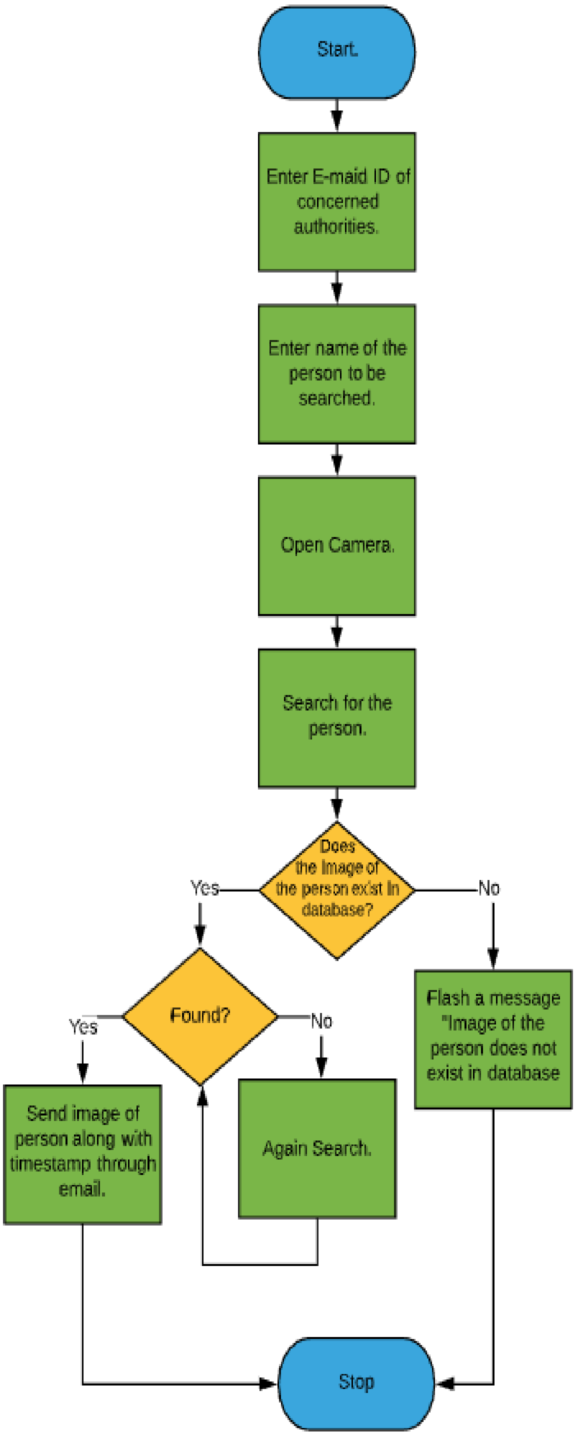
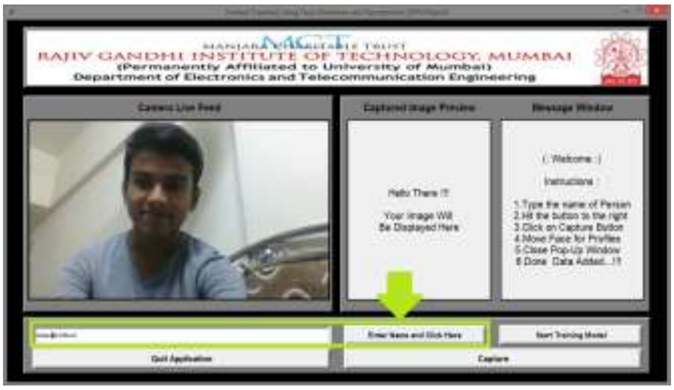

Start.
Enter E-maid ID of concerned authorities.
Enter name of the person to be searched.
Open Camera.
Search for the person.
Does the Image of the person exist in database?
Yes
No
Found?
Yes
No
Flash a message "Image of the person does not exist in database
Send image of person along with timestamp through email.
Again Search.
Stop

Fig 3- Flowchart of Image Recognition process.

3. RESULTS

3.1 Dataset Creation:

In the first part we enter the name of the person you want to register or add into the database in the text field and click on the button next to the text field and the name will be replaced by a file path.


RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
(Permanently Affiliated to University of Mumbai)
Department of Electronics and Telecommunication Engineering
Camera Live Feed
Captured Image Preview
Message Window
Enter Name and Click Here
Start Training Model
Quit Application
Capture

Hello There !!!

Your Image Will Be Displayed Here

(: Welcome :)

Instructions :

1.Type the name of Person
2.Hit the button to the right
3.Click on Capture Button
4.Move Face for Profiles
5.Close Pop-Up Window
6.Done Data Added...!!!

Fig 3: Entering the name for creating database.

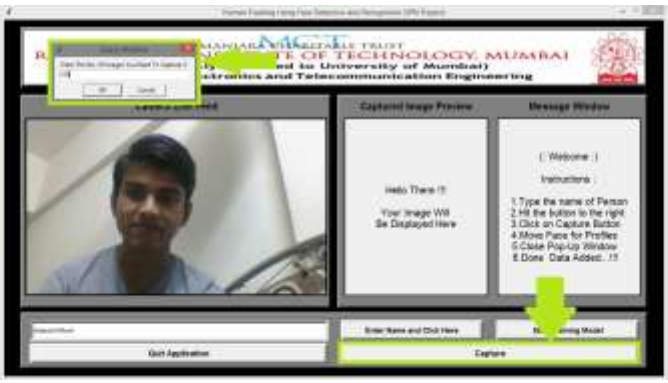

RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
Captured Image Preview
Message Window
Enter Name and Click Here
Quit Application
Capture

Fig 4: Enter number of images to capture.

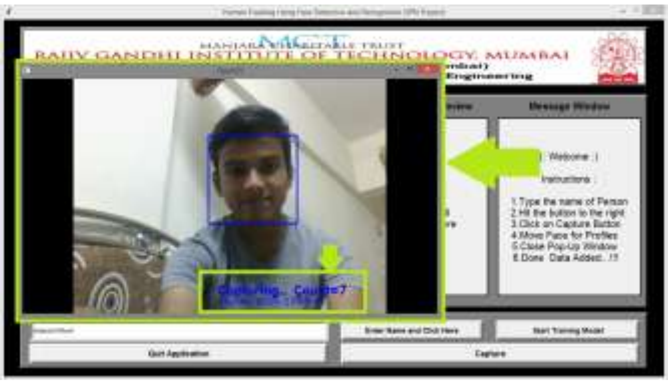
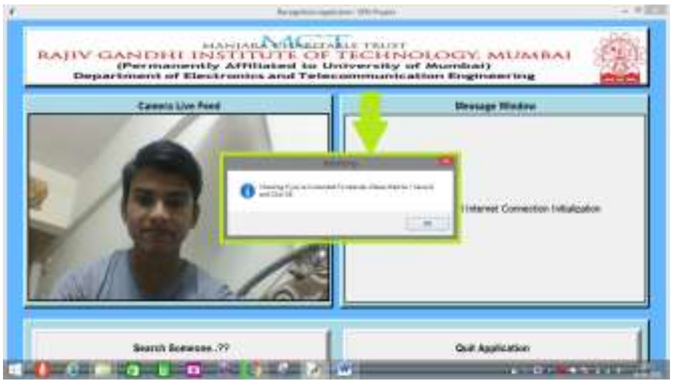


Fig 5: Total count of images along with timestamp.

3.2 Recognition:

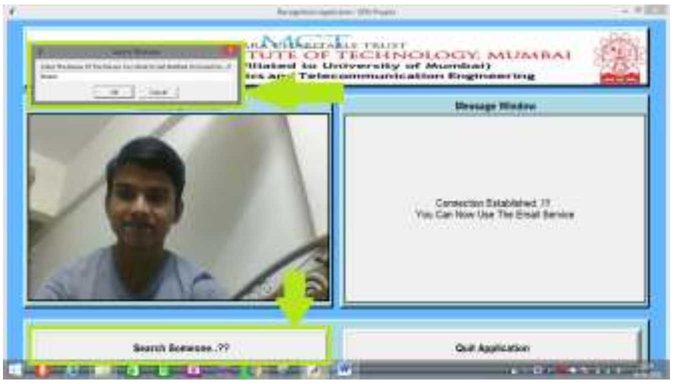
Initialization is done by checking the internet connection. Internet connection is required as we'll be using email service to notify the concerned authorities about the search details.


RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
(Permanently Affiliated to University of Mumbai)
Department of Electronics and Telecommunication Engineering
Camera Live Feed
Message Window
OK
Search Someone..??
Quit Application

Internet Connection Initialization

Fig 6: Check internet connection.

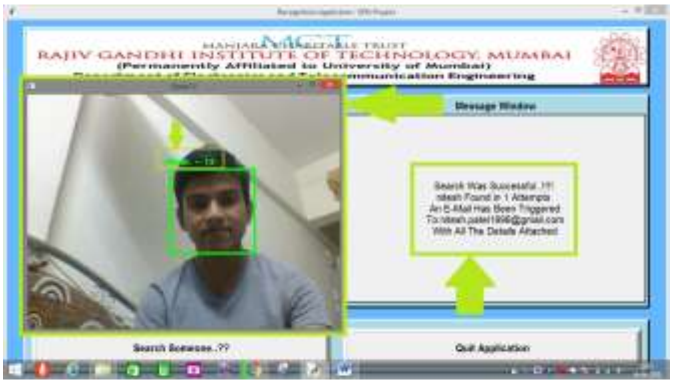
Once initial setup is done, we can now enter the name of the person we want to search by clicking on the search button as shown in the image.



Connection Established ..!!
You Can Now Use The Email Service

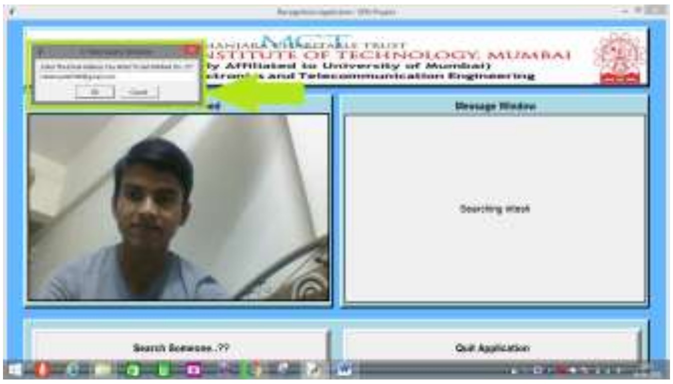
Fig 7: Enter name of person to search.

Camera window will appear showing the live feed from the camera and start detecting all the faces present in the frame. Program will start searching the faces stored in the database and keep on continuously matching it to the image taken directly from the camera frame. If faces match a confidence level of the match will be displayed on the image. If it matches the name of query successful search message will be printed.


RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
(Permanently Affiliated to University of Mumbai)
Message Window
Search Someone..??
Quit Application

Search Was Successful ..!!!
nitesh Found in 1 Attempts
An E-Mail Has Been Triggered
To:nitesh.patel1998@gmail.com
With All The Details Attached

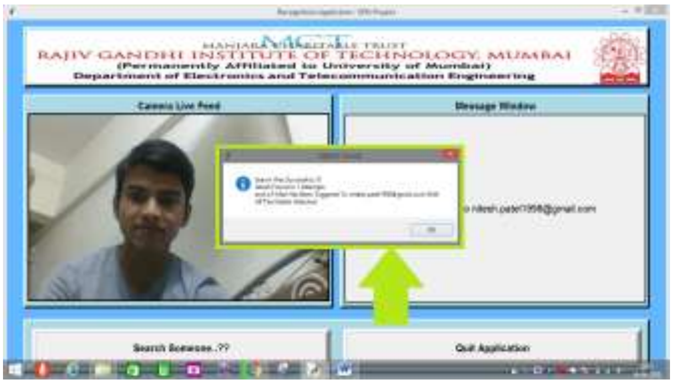
Fig 8: Person found along with his/her name.



Searching nitesh

Fig 9: Enter the email to send a notification to.

Once you enter the email address on which you want to send the details a successful search notification will pop up indicating the search process to be complete.


RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI
(Permanently Affiliated to University of Mumbai)
Department of Electronics and Telecommunication Engineering
Camera Live Feed
Message Window
OK
Search Someone..??
Quit Application

nitesh.patel1998@gmail.com

Fig 10: Confirmation of E-mail being sent

The detected person's image, name along with the IP address of the computer holding the camera is sent to the person whose email address we had entered through our project's official email address htfdrproject@gmail.com. We can Check the inbox of the receiver's email address for the details in this case nitesh.patel1998@gmail.com as shown in the image below.

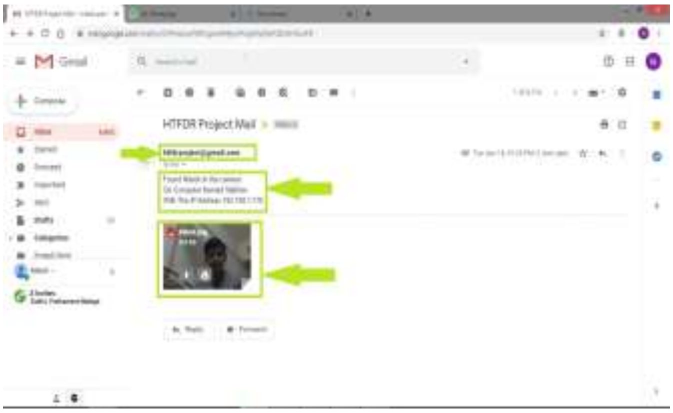


Fig 11: E-mail being sent to the desired e-mail address.

4. CONCLUSION

Thus, the system of human tracking system using face detection and recognition has been successfully implemented. Through this project, we have contemplated about various scenarios that influence the system. Also, concepts such as Haar Classifier and LBPH algorithm for face detection were understood.

5. FUTURE SCOPE

Thus, the system of human tracking system using face detection and recognition has been successfully implemented. Through this project, we have contemplated about various scenarios that influence the system. Also, concepts such as Haar Classifier and LBPH algorithm for face detection were understood.

ACKNOWLEDGEMENT

We would like to acknowledge our Professors, Mr. Arvind Sangale and Mr. Satish Bhoyar, without their staunch support, motivation and vision, it would not have been possible for us to move ahead with this project and successfully implement it.

REFERENCES

[1] Sujata G. Bhele, V. H. Mankar "A review paper on face recognition techniques" October 2012.

[2] Kavita, Ms. Manjeet Kaur "A survey paper for face recognition technologies." RIEM, Rohtak.

[3] Divya Meena, Ravi Sharan, "An approach to face detection and recognition".

[4] Shalini Singh, Md Golam Mohiuddin, Swati Pandit, Amigo Banerjee, Indrasom Gangopadhyay, Indrajit Das "Face Detection and Recognition by Haar Cascade Classifier, Eigen Face and LBP Histogram".

BIOGRAPHIES

Prathamesh Malage
B.E EXTC Department
Rajiv Gandhi Institute of Technology, Versova
Team Member

Siddhi Mahamunkar
B.E EXTC Department
Rajiv Gandhi Institute of Technology, Versova
Team Member

Prathamesh Malage
B.E EXTC Department
Rajiv Gandhi Institute of Technology, Versova
Team Member

Arvind Sangale
Assistant Professor
EXTC Department
Rajiv Gandhi Institute of Technology, Versova
Team Member