IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Smart Attendance System using Image Processing and Rasberry PI

Author

Prof. Dr. Vijay Mane
Dept. of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India

Shubham Indulkar
Dept. of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India

Omkar Khelge
Dept. of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India

Sagar Khandade
Dept. of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India

Amit Mitkari
Dept. of Electronics and Telecommunication
Vishwakarma Institute of Technology
Pune, India

***Abstract*:-**

Face detection is usually the first step towards many face-related technologies, such as face recognition or verification. Real-time face detection not only is one part of the automatic face recognition system but also is developing an independent research subject. In this paper, we have proposed an automated Face Recognition System for Time and Attendance application. The model is developed with the help of real time OpenCV library. It takes a real time image of the class using a dedicated camera module and evaluates the attendance of class using the students data. The attendance is sent to the concerned person via mail to the concerned person.

I. INTRODUCTION

Person Recognition is one of the emerging research fields in image processing. Face detection and face direction estimation are important for face recognition. In personal identification with surveillance cameras, for example, it is necessary to detect the face whose size, position, and pose are unknown. After the face detection, the face direction estimation is useful for the correct face recognition because we can select the face image of the most desirable direction from the face images taken by the multiple cameras.This technology could be used in Classroom Attendance system, where the attendance is automatically taken and mailed to the concerned person. This system is developed in order to avoid the manual drudgery for lecturers in entering the data daily while taking attendance and also to avoid proxy. They are several biometrics used for Person. Recognition like Iris, Fingerprints, Face etc. Since Iris and Fingerprints are very short-distance biometrics. As a worst case, our system should be able to recognise a person who is sitting at the last bench, which might not be possible by using Iris or Fingerprints as a biometric. Hence we go for a medium range Biometric i.e., Face with the help of which we can recognise a person and mark his attendance.

II. RASPBERRY PI CAMERA

The cameras connecting wirelessly with IP address in local network and use for Image Processing is basic idea for this project. Two major steps involve in this whole process. First step is to put the camera in in such a location so that it could take a picture of the entire class. Then the camera has to send a real time image of the entire class to the raspberry pi microcontroller so that the students in the image can be compared to the database and their corresponding attendance could be taken.

III. PROPOSED SYSTEM'S ARCHITECTURE

The system designed consists of following steps. It can be constructed in many modules:

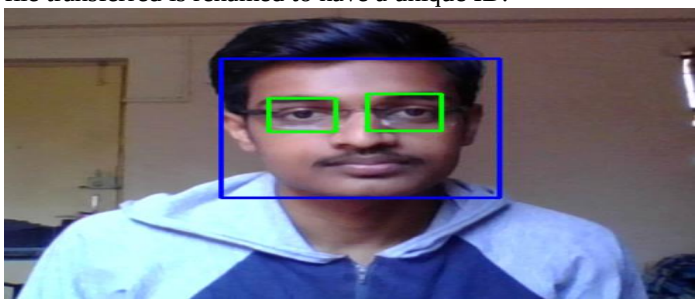
- Image capturing,
- Face Detector and
- Face recognizer.

A) Image Capturing

Images are captured using a module that is a raspberry pi camera module whose link is integrated to the application that is developed using the proposed idea. After an image is captured, the image is sent to microcontroller for processing. Together with the image, the web service accepts the course code. Using this course code, the LMS is aware of which students are enrolled in that class and do face matching only for those students. The camera takes pictures on a given interval until an appropriate image in which the faces of all the students can be detected.

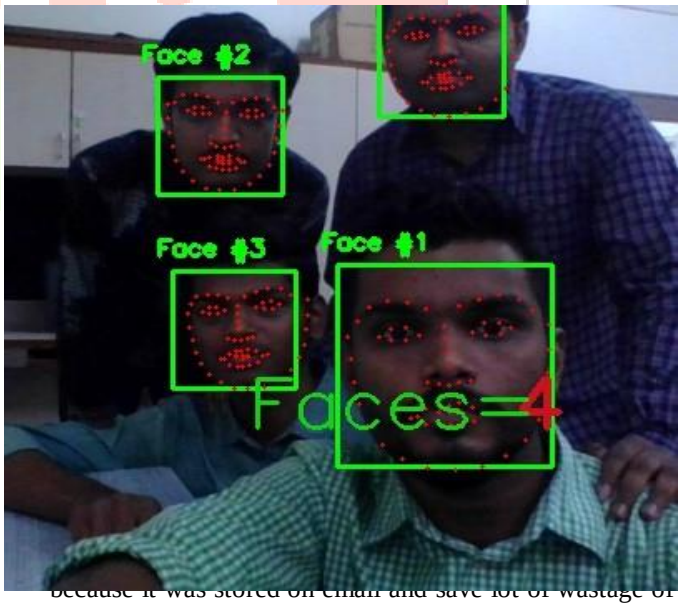
B) Face Detection

Because of processor intensive job of the face detection algorithm, this tool is server based. Detecting a face is in essence an object detection task, where the object of interest in this case is the face. However, many factors can interfere with the face detection algorithms, factors such as face pose, scale, position, rotation, light, image colors etc. The same problems arise when one wants to identify (recognize) a face, with addition to some other obstacles which is discussed shortly. The process of detecting faces from still pictures containing multiple faces can be separated in few steps. There are plenty face detessction algorithms which can effectively detect a face (or any other specific object) in a picture. In the system presented here, most students face the camera frontally hence we chose to use the HAAR classifier for face detection. This classifier is implemented on Intel's Open CV library. The classifier works by training a model using positive face images and negative face images. A positive image is an image that contains the desired object to be detected, in our case this object is a face. A negative image is an image that does not contain the desired object. After the model is trained, it is able to identify face features, which is later, stored on a XML file. A problem faced during this process was the large number of false-positives: objects mistakenly detected as faces. This was not such a big issue for us, since a false-positive does not result in a positive identification during the recognition phase. Because of this, we lowered the detection threshold, so all faces could be detected. After a face has been detected, the rectangle enclosing this face is cropped and processed later by the face recognition module. This rectangle represents a single face, and after being cropped as an image is transferred on server. Each file transferred is renamed to have a unique ID.



C) Face Recognition

Recognizing a face means to identify that particular face from a list of faces on a database. In this project the images of all the students in the class are stored in a database. Same as in face detection, there are many existing algorithms used to identify a face. Our system implements a server based module, programmed in Python which takes benefit of eigenfaces to identify a face. This algorithm has many drawbacks: it depends on scale, pose and the color of the compared images. However the algorithm is very fast, and can compare only to images, thus we do not NEED to have multiple images of a person to train our system. Since our system is setup to capture only frontal images the pose of the face in not an issue. When a face is captured during the face detection phase, it is converted into gray scale. The same conversion is applied to faces on our student image database. We also do background subtraction on our images so other objects do not interfere during the process. Another issue is that faces are subject of change during time (facial hair, eyeglasses etc). Whenever we successfully identify a face, a copy of that face is stored in the database of faces for that student. Together with the image we store the time and date when this image was taken. This way even if a student gradually changes his appearance (e.g., grows a beard) the system is still capable to identify him, since it has multiple images of the same person. On each consequent scan for a student, the recognition module starts comparing images from this database, sorted by date in descending order. This approach was chosen since the latest image of a student on our database is most likely to be more similar to the current captured image. Of course, a drastic change on a student's look causes the system to not identify that particular student. To solve this issue, we have included a module, which lists all unidentified faces and the teacher is able to manually connect a captured face with a student from the list. This image is also stored on our database, as an updated picture of this particular student. This manual recognition process is performed only once. In a subsequent scan, this student is identified



because it was stored on cloud and save lot of wastage of time and hardwork.It is the best application of IOT phenomenon.

CONCLUSION

This face recognition based attendance management system provides accurate attendance information of the students in easy way and email the attendance to the teacher using IOT. This system eradicates the wastage of time while taking manual attendance and also provides correct information. Security system gives high secure for any type of systems instead of using fingerprint or RFID. This system is convenient to user, easy to use and gives better security.. When number of student faces increases the accuracy will decrease slightly. This system gives the student details as output to a storage device and sends an email to the person who manages the corresponding attendances.

ACKNOWLEDGMENT

This paper was supported by our institution Vishwakarma institute of technology. We are thankful to our guide Prof. Vijay Mane who guided us regularly and helped us overcome our difficulties.We are also grateful to Prof. Milind Patwardhan for assistance with timely advise and our department of Electronics and Telecommunication engineering who give us an opportunity to prepare this paper. We also have to express out appreciation to our other prof. of department who resolve the simple conflicts and problem we face us during the course of this project.

REFERENCES

1. "ICTFacts & Figures: The World in 2015"[[more than 7 billion mobile cellular subscriptions worldwide|]]February 2016
2. Dr. A. Rosebrock ,"Practical Python and OpenCV : An introductory ,example driven guide to image processing and computer vision", 3rd edition.
3. http://www.pyimagesearch.com/practical-python-opencv/
4. Security for Education with IP Surveillance Systems, A D----Link
5. white paper, January, 2008
6. http://www.stackoverflow.com/questions/python
7. Q.Chen, "A basic introduction to OpenCV for image processing", DiscoverLab School of Information Technology & Engineering University of Ottawa, January 15 ,2007.
8. http://www.github.com
9. Simple and easy tutor on how to work on OPenCV in Python http://www.youtube.com/channel/sentdex
10. http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html
11. http://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html#ahp04
12. IP webcam using Smartphone supporting WiFi inbuilt or similar device http://www.instructables.com/id/Make-A-Wi-fi-Webcam-From-An-Old-Android-Phone/
13. IEEE paper documentation on Face recognition done with face landmark detection. http://ieeexplore.ieee.org/document/1613000/
14. Basic of how live streaming works while transfering data http://computer.howstuffworks.com/internet/basics/streaming-video-and-audio.htm