Classroom Attendance Using Face Detection and Raspberry-Pi

P VENKATALAVANYA, B SRIVARSH,PSVS SURYAPRAKASH[1], M HRITHIK,[2]

[123]Student, Dept. of Electronics and Communication Engineering,
[2]Associate Professor, Dept. Electronics and Communication Engineering
TKR COLLEGE OF ENGINNERING AND TECHNOLOGY

Abstract - *In class, it's crucial for pupils to be present. When done manually, it typically consumes a lot of class time that could be used for learning. Face recognition-based automation of the attendance system is the suggested fix for the current issue. Any human can be recognised mostly by their face. This tutorial explains how to use a Raspberry Pi to detect and recognise faces in real time. This project uses the OpenCV open source image processing library to define an effective algorithm. Face Detection, Face Preprocessing, Face Training, Face Recognition, and Attendance Database are the five modules that make up our methodology. To identify the faces of the kids, a face database is compiled. The student database, which consists of the faces of all the students, is used to train the algorithm initially. While gathering student photos and taking attendance throughout training and testing, the system uses an intuitive user interface to maximise the user experience. Numerous more applications where face recognition can be used for authentication can be used with this project. Utilizing a Raspberry Pi reduces product costs and improves usefulness because it can be connected to any device to take attendance. This project employs a modified version of Viola-Cascades Jones's Haar's technique for face detection, LBP histograms for face identification, and updates the database using both MYSQL and SQLite (the lite version of SQL for the Raspberry Pi). The system will automatically send messages to the department head and the guardians of absent students informing them of the student's attendance in the class and updating the student's record accordingly.*

Key Words: OpenCV, Raspberry Pi, Haar cascade, LBPH recognizer, Viola-Jones framework

1. INTRODUCTION

The present day attendance system is manual. It wastes a considerable amount of time both for teachers and students. The waiting time of the students is increased if attendance is taken manually. There are still chances for proxies in the class when attendance is taken manually. Manual attendance always a have a cost of human error. Face is the essential recognizable proof for any human. So automating the attendance process will increase the

productivity of the class. To make it available for every platform we have chosen the Raspberry pi 3 for face recognition. A Webcam is associated with the Raspberry Pi module. Face identification separates faces from non-faces and those countenances that can be perceived. This module can be utilized for different applications where face acknowledgment can be utilized for validation. In this proposed system we take the attendance using face recognition which recognizes the face of each student during the class hours.

2. BLOCK DIAGRAM

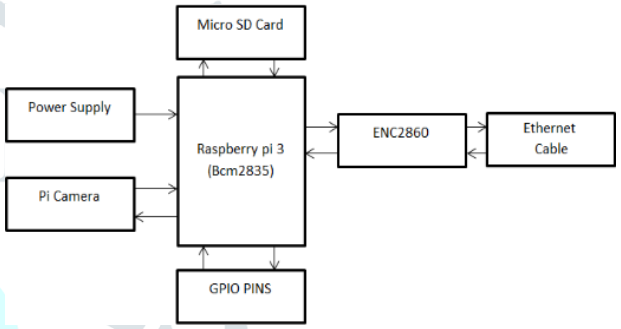

Fig -1: Block diagram of Proposed Approach

2.1 Raspberry Pi 3

The Raspberry Pi is a small, inexpensive computer the size of a credit card that connects to a computer monitor or TV and operates with a regular keyboard and mouse. With the help of this competent small gadget, individuals of all ages may learn about computing and how to programme in languages like Scratch and Python. To connect to the internet, this Raspberry Pi has an Ethernet chip called the ENC28J60.

Fig -2: Raspberry Pi 3

2.2 Camera

A camera is an optical instrument for recording or capturing images, which may be stored locally, transmitted to another location, or both. The images may be individual still photographs or sequences of images constituting videos or movies. The camera is a remote sensing device as it senses subjects without any contact.

2.3 GPIO Pins

General-purpose input/output (GPIO) is a generic pin on an integrated circuit or computer board whose behavior—including whether it is an input or output pin—is controllable by the user at run time.

2.4 Power Supply

The primary necessity for the project's work is the power supply. The mains line provides the necessary DC power supply for both the base unit and the charging unit. The secondary of a transformer with a 12V–12V centre tap is utilised for this. We receive a 5V power supply from this transformer.

2.4 SD Card

The OS required for raspberry pi is raspbian and the minimum recommended card size is 8 GB.

3. PROPOSED APPROACH

The total system is divided into 3 modules- Database creation, Training the dataset, Testing, sending alert messages as an extension.

1. Database creation

a) Initialize the camera and set an alert message to grab the attention of the students.
b) Get user id as input
c) convert the image into gray scale, detect the face and
d) Store it in database by using given input as label up to 20 frames.

2. Training

a) Initialize LBPH face recognizer.
b) Get faces and Id's from database folder to train the LBPH face recognizer.
c) Save the trained data as xml or yml file.

3. Testing

Load Haar classifier, LBPH face recognizer and trained data from xml or yml file.

a) Capture the image from camera,
b) Convert it into gray scale,
c) Detect the face in it and
d) Predict the face using the above recognizer.

This proposed system uses Viola Jones algorithm [1] for face detection which uses modified Haar Cascades for detection.

Raspberry Pi is the main component in the project. We will be using USB webcam to capture photos. We can access Raspberry Pi's console either by using SSH in laptop or by using Keyboard and mouse with the display device like TV connected to Pi. Firstly, the algorithm needs a lot of positive images and negative images to train the Haar cascades classifier. Positive images are images with clear faces where negative images are those without any faces.

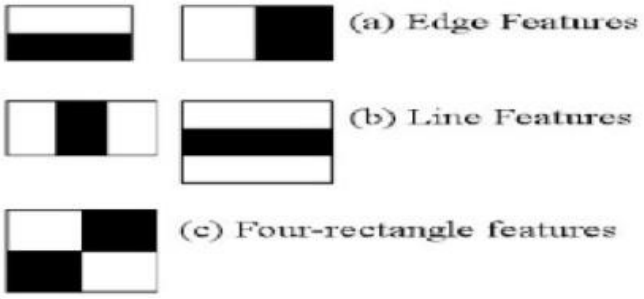

Fig -3: Haar Cascades

Each feature is represented by a single value that is derived from the difference between the sums of the pixels in the white and black rectangles. Numerous features are calculated using all available classifier sizes and positions. The arithmetic computations appear to take a lengthy time as the number of classifiers rises. We employ the idea of an integral image to prevent this. Integral images are a type of data structure used in image processing that contains a method for quickly and effectively calculating the sum of values in a subset of a rectangular grid. Using the formula, an integral image is produced.

$$I_{\sum}(x,y) = \sum_{\substack{x' \le x \\ y' \le y}} i(x',y')$$

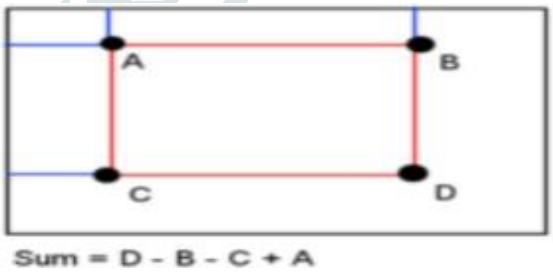

Fig -4: Integral image

To solve the complexity of the number of classifiers applied for calculation we use Adaboost machine learning algorithm, which is inbuilt in OpenCV library that is cascade classifier, to eliminate the redundancy of the classifiers. Any classifier which has a probability of 50% of more in detection is treated as weak classifier. The Sum of all weak classifier gives a strong classifier which makes the decision about detection. Although it is very vague to classify with one strong classifier we use the cascade of classifiers. Classification takes place in stages, if the

selected region fails in the first stage, we discard it. We don't use the classifiers on that region which is discarded. The region which passes all the stages i.e. all strong classifiers is treated as the detected face. Detected Faces are passed to the Face recognition phase. In this phase we use Local Binary Patterns algorithm for face recognition. Local binary patterns are simple at the same time very efficient texture operator which assigns the pixels of the image by comparing with the adjacent pixels as threshold and which results in a binary result. The detected integral image is subjected to this Local binary pattern which results in decimals are represented as histogram for every integral image. Face recognition is extremely vulnerable to the environment changes like brightness, facial expressions and position. Face preprocessing is the module which reduces the problems that makes the picture unclear to recognize the face such as less brightness and contrast problems and noise in the image and make sure the facial features always be in a constant position. In this project we use histogram equalization for face preprocessing. For efficiency we use separate preprocessing which is histogram equalization for left and right face. So histogram equalization is done three times, firstly for the whole face and the other two for side faces [7].

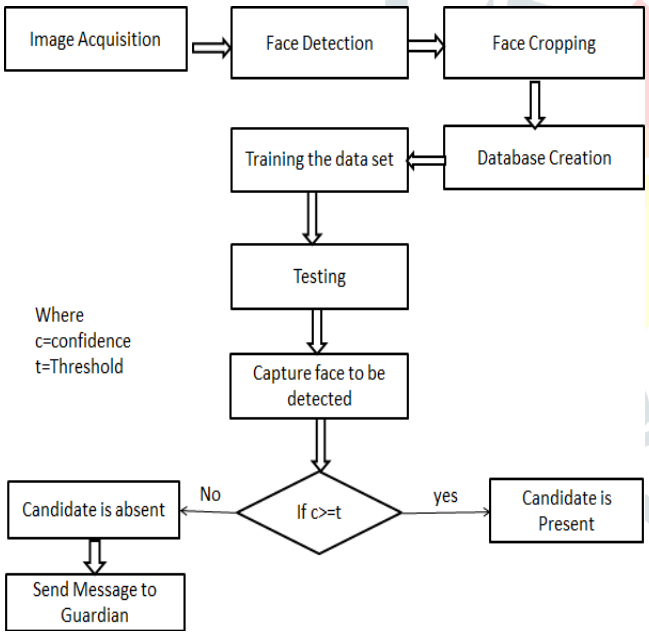

Fig -5: Flow Chart

4. ALGORITHMS

4.1 Python IDE

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. ThePython interpreter is easily extended with new functions and data types implemented in C or C++ (or other languages callable from C). Python is also suitable as an extension language for customizable applications.

4.2 OpenCV

Real-time computer vision is the primary focus of the OpenCV collection of programming functions. Because of its modular design, the package contains a number of shared or static libraries. We are employing an image processing module with features like histograms, colour space conversion, affine and perspective warping, geometric picture transformations, linear and non-linear image filtering, and more. Libraries like the Viola-Jones or Haar classifier, the LBPH (Lower Binary Pattern histogram) face recognizer, and the Histogram of Oriented Gradients are all included in our project (HOG).

A) Image processing module

Picture processing is the process of applying various operations to an image in order to enhance it (by merely emphasising some interesting characteristics) or to extract some valuable information from it. It is a kind of signal processing where the input is an image and the output can either be another image or traits or features related to that image.

Purpose of Image processing

The purpose of image processing is divided into 5 groups. They are:

1. Visualization- Observe the objects that are not visible.
2. Image sharpening and restoration- To create a better image.
3. Image retrieval- Seek for the image of interest.
4. Measurement of pattern– Measures various objects in an image.
5. Image Recognition– Distinguish the objects in an image. [5]

i. Haar Classifier

This object detection framework is to provide competitive object detection rates in real-time like detection of faces in an image. A human can do this easily, but a computer needs precise instructions and constraints. To make the task more manageable, Viola–Jones requires full view frontal upright faces. Thus in order to be detected, theentire face must point towards the camera and should not

be tilted to either side. While it seems these constraints could diminish the algorithm's utility somewhat, because the detection step is most often followed by a recognition step, in practice these limits on pose are quite acceptable [2].

The characteristics of Viola–Jones algorithm which make it a good detection algorithm are:

a) Robust – very high detection rate (true-positive rate) & very low false-positive rate always.
b) Real time – For practical applications at least 2 frames per second must be processed.
c) Face detection only (not recognition) - The goal is to distinguish faces from non-faces (detection is the first step in the recognition process).

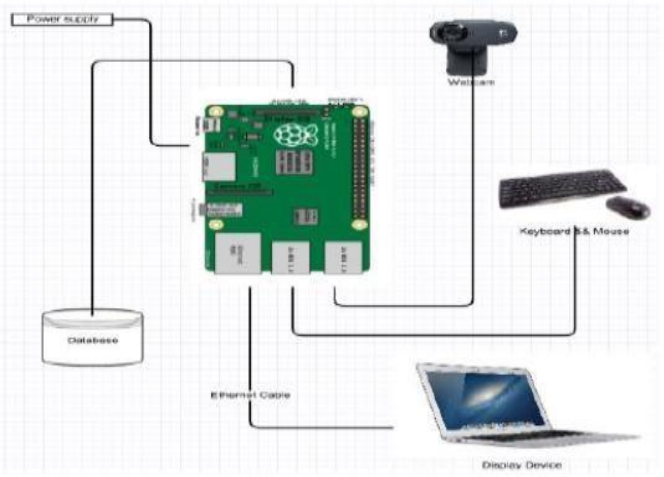

Fig -6: Implementation

This algorithm includes Haar feature selection process.

All human faces share some similar properties. These regularities may be matched using Haar Features.

A few properties common to human faces:

a) The eye region is darker than the upper-cheeks.
b) The nose bridge region is brighter than the eyes.

Composition of properties forming matchable facial features:

a) Location and size: eyes, mouth, bridge of nose
b) Value: oriented gradients of pixel intensities[4]

ii. Histogram of oriented gradients (HOG)

Histogram of oriented gradients (HOG) is a feature descriptor used to detect objects in computer vision and image processing. The HOG descriptor technique counts occurrences of gradient orientation in localized portions of an image - detection window, or region of interest (ROI).

Implementation of the HOG descriptor algorithm is as follows:

1. Divide the image into small connected regions called cells, and for each cell compute a histogram of gradient directions or edge orientations for the pixels within the cell.
2. Discretize each cell into angular bins according to the gradient orientation.
3. Each cell's pixel contributes weighted gradient to its corresponding angular bin.
4. Groups of adjacent cells are considered as spatial regions called blocks. The grouping of cells into a block is the basis for grouping and normalization of histograms.
5. Normalized group of histograms represents the block histogram. The set of these block histograms represents the descriptor [3].

B) Numpy

NumPy is the fundamental package for scientific computing with Python. NumPy is an acronym for"Numeric Python" or "Numerical Python". It is an open source extension module for Python, which provides fast precompiled functions for mathematical and numerical routines. Furthermore, NumPy enriches the programming language Python with powerful data structures for efficient computation of multi-dimensional arrays and matrices. The implementation is even aiming at huge matrices and arrays. Besides that the module supplies a large library of high-level mathematical functions to operate on these matrices and arrays.

It contains other things like:

a) A powerful N-dimensional array object
b) Sophisticated (broadcasting) functions
c) Tools for integrating C/C++ and Fortran code
d) Useful linear algebra, Fourier transforms, and random number capabilities.

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

In our project we need to convert images into multi-dimensional or 2D-array representation, and also conversions from gray scale to color images which can be done easily by Numpy.

5. CONCLUSION

We came to realize that there are extensive variety of methods, for example, biometric, RFID based and so on which are tedious and non-productive. So to defeat this above framework is the better and solid arrangement from each keen of time and security. Hence we have accomplished to build up a solid and productive participation framework to actualize an image handling algorithm to identify faces in classroom and to

perceive the confronts precisely to check the attendance.

6. SCOPE and FUTURE WORK

The same project can be utilized for several security applications where authentication is needed to access the privileges of the respective system. It can be used in recognizing guilty parties involving in unauthorized business. Face recognition algorithm can be improved with respect to the utilization of resources so that the project can recognize more number of faces at a time which can make the system far better. Many variants of the project can be developed and utilized for home security and personal or organizational benefits. We can also trace a particular student in an organization quickly with the help of this system.

ACKNOWLEDGEMENT

Author would like to thank P. Purna Sekhar, Assistant Professor of Electronics and Communication Engineering for the technical support during system experimentation.

REFERENCES

[1] Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on (Vol. 1, pp. I-511). IEEE.

[2] Rohit, C., Baburao, P., Vinayak, F., &Sankalp, S.(2015). attendance management system using face recognition. International Journal for Innovative Research in Science and Technology, 1(11), 55-58.

[3] NirmalyaKar, MrinalKantiDebbarma, AshimSaha, and DwijenRudra Pal, "Implementation of Automated Attendance System using Face Recognition", International Journal of Computer and Communication Engineering, Vol. 1, No. 2, July 2012.

[4]Benfano Soewito, Ford Lumban Gaol," Attendance System on Android Smartphone", 2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC).

[5] AparnaBehara, M.V.Raghunadh, "Real Time Face Recognition System for time and attendance applications", International Journal of Electrical, Electronic and Data Communication, ISSN 2320-2084, Volume-1, Issue-4.

[6] KAWAGUCHI, Y., SHOJI, T., Weijane, L. I. N., KAKUSHO, K., & MINOH, M. (2005). Face recognition-based lecture attendance system. In the 3rd AEARU Workshop on Network Education (pp. 70-75).

[7] Soundrapandiyan Rajkumar, J. Prakash, "Automated attendance using Raspberry pi", International Journal of Pharmacy and Technology, Sep 2016.