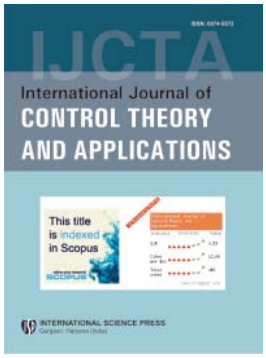

International Journal of Control Theory and Applications

ISSN : 0974–5572

Volume 9 • Number 40 • 2016

Smart Classroom Attendance System

Hari Baabu V[a] Senthil kumar G[a] Meru Prabhat[b] and Suhail Sayeed Bukhari[b]

[a]*Assistant Professor, Department of Software Engineering SRM University*

[b]*Student, Department of Software Engineering SRM University*

E-mail: haribaabu.v@ktr.srmuniv.ac.in, corresponding author

Abstract: The Smart Classroom Attendance system (SCAS) is a system which is developed to automate the attendance system in a classroom. The major goal is to lower the time and human efforts by integrating the technology and functions of "IoT" to work on targeted audience. On implementing this project both time and efforts will be eventually minimized. This project is mainly consisting of a pre-loaded Human Facial Database System of the Students residing on the classroom, in a micro-controller system. On starting of any class, when the faculty is ready to take the attendance of their students, instead of following the traditional, yet an orthodox system of taking attendance; a new method is introduced. Here, at the stating of the attendance by the faculty, all the students are required to face up to the cameras installed on the ceiling. Three cameras with a wide-angle of 130 degrees covering the whole classroom area will be used to take the picture. The picture taken will be used to match with the existing Facial Database of the class, and within minutes the attendance will be displayed to the faculty. All the attendance details will be stored on a separate record database for any future reference. This whole project consists of micro-controllers like Raspberry Pi, sensors and components like a wide-angled camera for capturing the photos. Along with the hardware, python programming will be used for building up the code and to enable the inter-networking between the gadgets.

Keywords: *OpenCV, IoT, Raspberry Pi, SCAS*

1. INTRODUCTION

The IoT (Internet of Things) is the main area of interest for a large number of people, organizations, groups etc. associated with both technical and non-technical sectors to ease and automating of their work. The main reason for this is that these devices are small, efficient and are available for a very low price. Day by day the performances of these devices are increasing with a significant amount without spoiling the image of budget efficient devices. These devices we agree are not to be used to carry out the heavy task but then they called internet of things, devices used to carry out small automated tasks with all-time connectivity with the end nodes.

Nowadays in this world, our face is not only a part of our body but a crucial part of our identification from School Identity Card to Passport. Many firms are developing new methods to produce efficient services for processing the image and focus on the frontal face. These companies range from Intel to Facebook with no end. Face detection is used in many places from photo sharing platforms to security. In our system, we have two different levels to process the image. First, we will identify the faces in the photo followed by matching it with the database to count the presence of students inside the classroom.

For the first phase, which is face detection we have developed a system which is small in size, though fast and efficient which aims for high simplicity and accuracy. It first identifies all the faces in the picture of students then extract out their frontal face. Second, we match all the extracted faces one by one with our predefined database, faces matched will be marked present and rest will be marked absent.

Finally, these results will be saved in an excel sheet and to be forward towards the respective target to update over the system. For all of these, we are using OpenCV [1], Yale face recognition algorithm [2] and python programming language.

2. LITERATURE SURVEY

This section of the research paper provides a brief overview of the methods and techniques available to carry out portions of our system such as face detection, face identification and other. Apart from the listing and briefing this section also holds the advantages and disadvantages of these technologies followed by our own method which is used to build our system. The techniques that we will discuss are principal component analysis with eigenfaces, modern neural networks, geometrical feature matching and hidden Markov model.

2.1. PCA with eigenfaces

Eigenfaces is a term used explicitly to point out a set of eigenvectors, used in solving the problems related to computer vision, image processing of predefined objects here human face recognition.

Principal component analysis converts a cluster of data provided from the correlated variables into a set of values of uncorrelated variables which is then termed as principal components. The resultant elements may be present in less or equal to the initially provided variables [3]. The key process in the PCA is to convert ID vector of 2D image say *a*b* into *a* 1**n* vector with a length equals = *a*b*. Then, following the same with test image PCA matches the vector to ientify the faces.

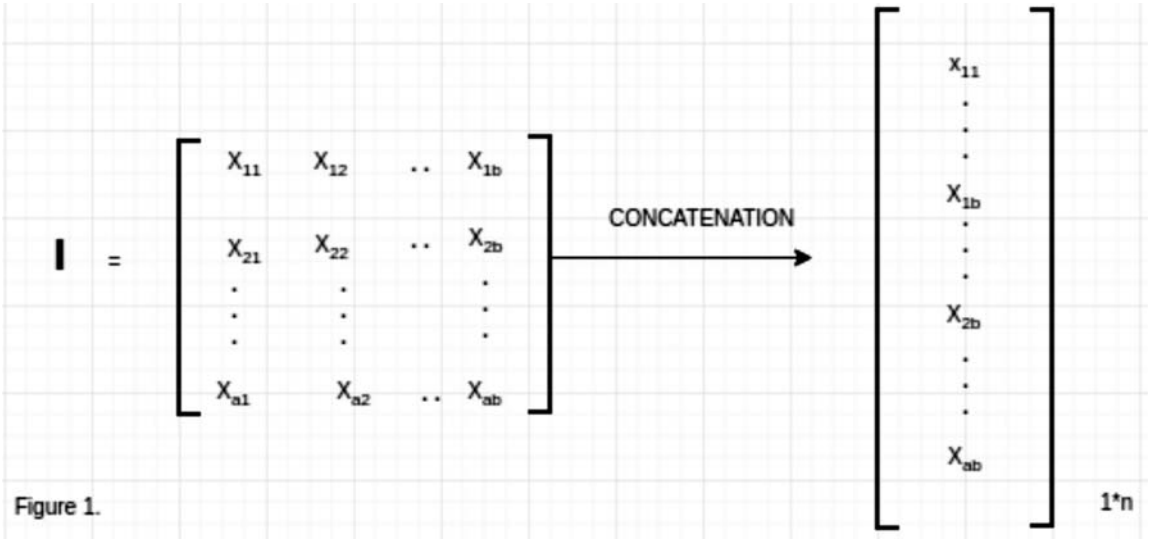

Figure 1: Concatenation process of PCA matrix

2.2. Neural Networks

The neural network or more precisely multi-layer perceptrons consist of an input layer, output layer, and a hidden layer. Each of them retains one or more than one neuron linked with the neurons from the last and next layer [1].

The major advantage we see in a neural network is its non-linearity in the given set of the network. WISARD was one among few artificial neural network developed initially that have to contain a distinguished network for every single element present [4]. To get the maximum efficiency and the accuracy we must check how our neural network is prepared. We have provided a number of patterns to build the neural network according to the need of the hour, such as, for detection of a face we use multilayer perceptron [5] and convolution neural network [6], whereas if we need verification of faces we apply [7] a multi-resolution pyramid structure.

Apart from traditional some authors provide a mixed hybrid neural network fitting according to their requirements. Reference [6] uses a combination of the local image sampling, a self-organizing map (SOM) and a convolution neural network. Again [8] holds a probabilistic decision-based neural network (PDBNN). The PDBNN works in three segments of the processes a) face detector, b) eye localizer and c) face recognizer. Both [6][8], were tested in databases such as ORL and provided 96.2% and 96% accuracy respectively.

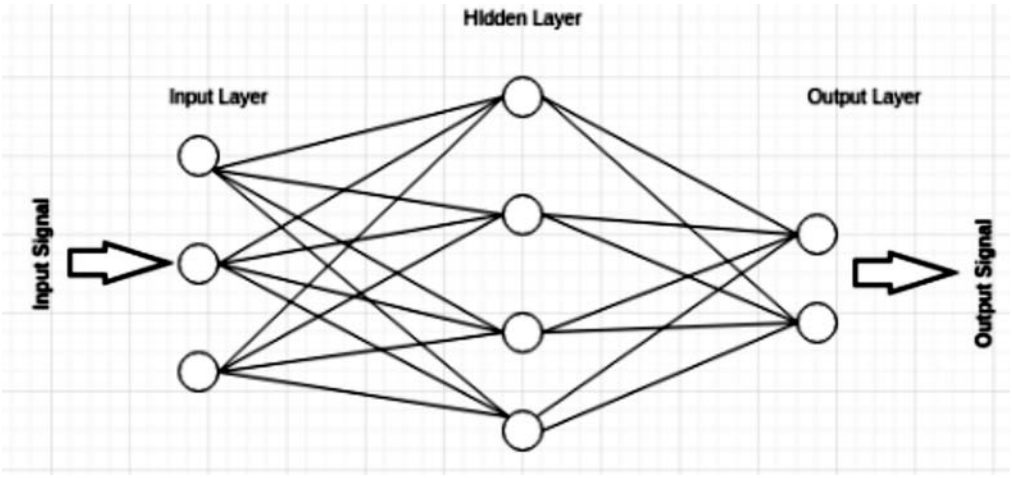

Figure 2: Neural Network Architecture

2.3. Geometrical feature matching

Geometrical feature matching is a technique in which we specify some points on the face such as eyes, nose, mouth and other parts on and around the face. We also have an option to pass the size along with the components of the face [9]. This technique works in two different parts.

Initially, it detects the edges, direction or the region of specifies points or components on the face, followed by building feature vectors from them. To do this we can pass a number of filters, out of them Canny filter to detect eyes or mouth or both region in the image, using transforms method such as Hough transform to detect eyes [10].

After this the second part work on grayscale differences of the components that we passed on the face. It then divides up face image to regions having both important components and unimportant components as proposed by LBP method [11]. Every single region will hold a central pixel and then we identify the neighbor pixel, which will then decide the value of neighbor of central pixel gained through grayscale to 1 or 0. These feature vector with the help of Cascaded Adaboost [12] and Chain AdaBoost, integrate with Haar-like feature detects the important components to identify the frontal face.

2.4. Hidden Markov model

The HMM is a technique which is widely used for many applications from image manipulation to speech application.

The HMMs can only operate on 1-D observation sequences and the images we handle are 2-D, so our images must be converted either to 1D temporal or spatial sequences [13].

The HMM so far have been used by many people and introduced many methods to handle the facial recognition such as [14] have used the HMM to identify the human face, here the faces were divided into components such as eyes, nose, mouth and other can also be termed as regions on the face. These all components or region can be correlated with the states of a hidden Markov model.

The accuracy of HMM too have found to be more than 90% as shown in [15] a pseudo 2D HMM is said to get a 95% of accuracy in recognizing the faces in their preliminary experiments.

3. STEPS

3.1. Acquiring the image

This is the first step in our project where we will take a picture of all the students present inside the classroom with the help of camera installed on the ceiling. The picture will be then saved in the raspberry pi storage where our python file and haarcascade_frontalface_default.xml file will be placed already.

3.2. Face detection

This is the very next process after acquiring the image. In this, our python executing the file with the image and haarcascade_frontalface_default.xml file initiate the process of face detection. The next steps will be done using the OpenCV inscribe with python executing the file. On the execution of them python file, the image will be loaded using cv2.imread(), followed by converting the image from BGR color to the GRAY scale using cv2. cvtColor(). After these function, we will pass the values of scaleFactor, minNeighbors and the minimum size of the pixels for our faces.

3.4. Individual face extraction

Post detecting all the faces successfully our second half of python file initiates. In this set, we take four variables which then act as the coordinates of the face. These coordinates will give us our roi (region of interest), with the help of cv2.imwrite() we save the roi as a separate image to do post processing on the face alone. Apart from all this we also provide a total number of faces find to know how many students are present in general.

3.5. Face recognition

This is the center of our project and for the base of python file, we have used Yale face recognition algorithm [], with other alteration we have made to make the program work according to our need. This program as previous start execution by using the haarcascade_frontalface_default.xml file. The libraries used in this program are cv, os (to make the Linux command work), Numpy and PIL from the image. The most import attribute used here from cv is “face” which is used to build the face images and to identify the test dataset from the training dataset. The first half of the program will train our predefined database which will be done very time before matching the faces to seek for any update in our database. The predefined images will be named as the name of student followed by his/her roll number to identify them as any face matches to it. On successfully training the dataset the program will search for the faces extracted from the 3rd step and then will start to match them with our training data.

3.6. Managing the results

This final step will handle the output from step IV. As soon as the previous step provides the data after matching of faces i.e. faces matched will be given as present. Now this step will check for all the students who are present and will remove from the original list contains all the students to get the name and roll number of absentees. On doing this these results will be saved in an excel sheet marking all the students present and absent according to the results obtained followed by date and time the attendance took place for future records.

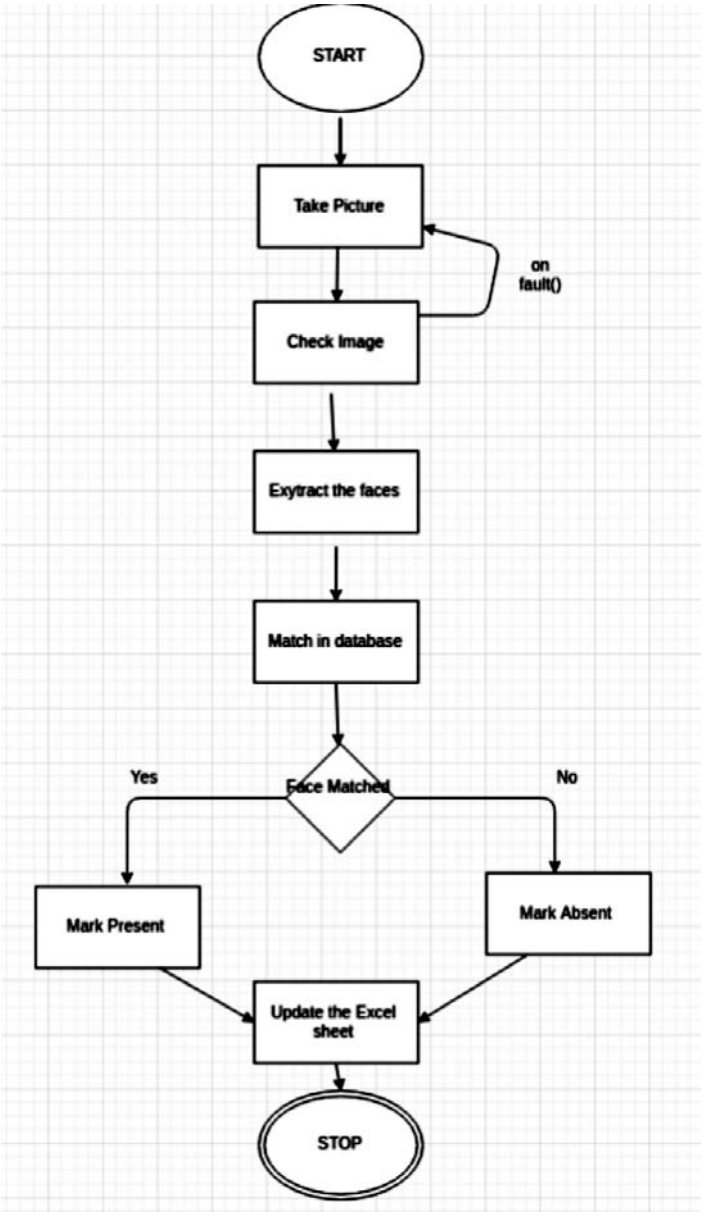

Figure 3: Data Flow Diagram of SCAS

4. RESULT / EXECUTION OF PROJECT

We have implemented this on two datasets one is ours and another is Yale Database.

Due to our small training set, we were able to gain ~80% accuracy but on implementing it on Yale database we have achieved ~100% accuracy. As we already have a system that can extract out the faces from any given picture we tested Yale database in our second phase i.e. after extraction of faces, here we intentionally deleted few of the test images from the yale database and we got the accurate result by getting output only for the faces left in database as present. This can be directly comparable to out proposed system.

This is our small database to test our system where we got both positive and negative result with an average of ~80% accuracy.

Figure 4: Training dataset of students

This is the first image for test and all the faces in the image got extracted successfully.

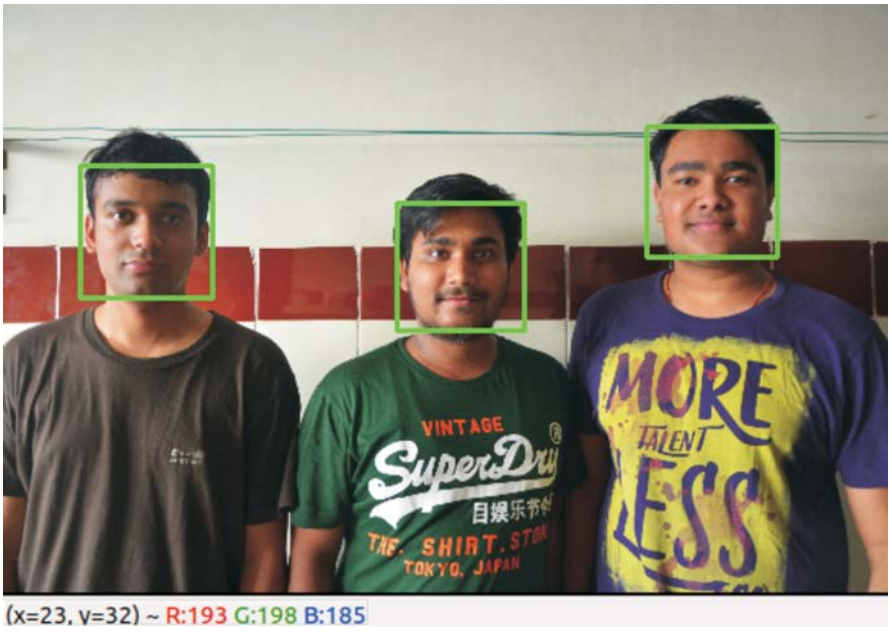

Figure 5: Test Image No.1

```
init done
opengl support available
The  Student 3 is persent
The  Student 2 is persent
The  Student 1 is persent
('This output is asved for further reference at ', 'Sun Mar 26 23:26:38 2017')
```

Figure 6: Result for test image No.1

Now taking another image in which only two students are present.

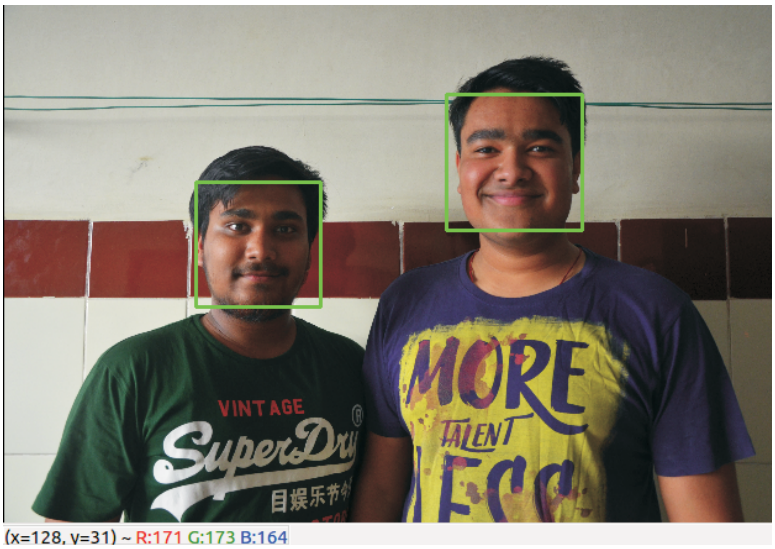

Figure 7: Test image no.2 with absence of first student

```
init done
opengl support available
The  Student 3 is persent
The  Student 2 is persent
('This output is asved for further reference at ', 'Sun Mar 26 23:33:57 2017')
```

Figure 8: Result for test image no.2

5. ADVANTAGES

1. This project covers a vast audience and can be used in schools, colleges and places where taking attendance is a need.
2. This project as being fully automated reduces the human effort to a remarkable level.
3. Rate of error and false positive attendance from students is ~0.
4. The project as whole is very cost effective comparing with the result it is providing.
5. The hardware involved in this consumes negligible space and bundled with all technologies such as remote login, secure channel for flow of information and other.

6. DISADVANTAGES

1. The person who is responsible to take attendance must check for enough light in the room to ensure good quality images.
2. Students must look at the camera when the photo is being taken.
3. Security majors must be taken to stop any unauthorized usage as the hardware is always present on site.

7. CONCLUSION

This project has achieved ~100% accuracy and has proven good efficiency. Installing of this project in any organization will reduce the human workload and provide more accuracy towards the attendance. This system will be very easy to install and can be operated with minimal knowledge of computing. The remote login will provide a better maintenance from technical department of the same institution.

8. ACKNOWLEDGMENT

This research was guided by Mr. Haribaabu.v. We thank our colleagues Ishaan Prakash, Pulkit Singhal and Aman from SRM University who provided images which are used in this paper. It was a pleasure working with them. We also want to appreciate the support and guidance provided by various staff members of our university.

REFERENCES

[1] OpenCV documentation

[2] Yale face algorithm

[3] SERBIAN JOURNAL OF ELECTRICAL ENGINEERING Vol. 9, No. 1, February 2012, 121-130

[4] T.J. Stonham, "Practical face recognition and verification with WISARD," Aspects of Face Processing, pp. 426-441, 1984.

[5] K.K. Sung and T. Poggio, "Learning human face detection in cluttered scenes," Computer Analysis of Image and patterns, pp. 432-439, 1995.

[6] S. Lawrence, C.L. Giles, A.C. Tsoi, and A.D. Back, "Face recognition: A convolutional neural-network approach," IEEE Trans. Neural Networks, vol. 8, pp. 98-113, 1997.

[7] J. Weng, J.S. Huang, and N. Ahuja, "Learning recognition and segmentation of 3D objects from 2D images," Proc. IEEE Int'l Conf. Computer Vision, pp. 121-128, 1993.

[8] S.H. Lin, S.Y. Kung, and L.J. Lin, "Face recognition/detection by probabilistic decision-based neural network," IEEE Trans. Neural Networks, vol. 8, pp. 114-132, 1997.

[9] Toan Thanh Do and Thai Hoang Le, "Facial Feature Extraction Using Geometric Feature and Independent Component Analysis"

[10] Kawaguchi,T.; Hidaka,D. and Rizon,M, "Detection of eyes from human faces by Hough transform and separability filter", IEEE International Conference on Image Processing, Volume 1, pp. 49 – 52, 2000.

[11] Shu Liao, Wei Fan, Albert C. S. Chung and Dit-Yan Yeung, "Facial Expression Recognition Using Advanced Local Binary Patterns, Tsallis Entropies And Global Appearance Features", IEEE International Conference on Image Processing, pp. 665-668, 2006.

[12] M. Jones and P. Viola, "Face Recognition Using Boosted Local Features", IEEE International Conference on Computer Vision, 2003.

[13] A. S. Tolba, A.H. El-Baz, and A.A. El-Harby, "Face Recognition: A Literature Review", International Journal of Signal Processing 2;2 2006.

[14] F. Samaria and F. Fallside, "Face identification and feature extraction using hidden markov models," Image Processing: Theory and Application, G. Vernazza, ed., Elsevier, 1993.

[15] F. Samaria and A.C. Harter, "Parameterisation of a stochastic model for human face identification," Proc. Second IEEE Workshop Applications of Computer Vision, 1994.