Standalone Application Using Face Recognition

Harshit Srivastava

Department of Computer Engineering, VelTech Dr.RR & Dr.SR Technical University

Chennai, India

ABSTRACT --- There are so many techniques are in the environment of attendance of students are available examples RFID Cards and manually attendance. But in this all techniques proxy of attendance not to be stopped because of that ,introduce Facial Recognition system [1] is a field of identifying a person from a facial database or from a live web camera source specially used in security systems. The aim of this Application is to provide an automated system where attendance of any institute or an organization will be done by using face recognition techniques within less time and efficiently without proxy attendance.

Keywords--- *Face Detection*, *Face* recognition, Eigen face, Haar CascadeClassifier, Emgu CV.

I. INTRODUCTION

Human face recognition has drawn considerable attention from the researchers in recent years. An automatic face recognition system will find many applications in areas such as human-computer interfaces, model-based video coding and security control systems. In this application such as video database search, a person's face can appear in arbitrary backgrounds [2] with unknown size and orientation. Thus there is a need for face recognition systems to handle these uncertainties.

In order for face recognition system to work it has to know what a basic face looks like. Face recognition system is based on ability to first recognize faces [3], which are a technological feat in itself and then measure the various features of each face. If you look into mirror you can see that your face has certain distinguishable landmarks sometimes called as nodal points. Nodal points on human face like-distance between eyes, width of nose, depth of eye, cheekbones, jaw line, chin etc.

Propose of a method that takes the attendance using face recognition [4] based on continuous observation. In this paper, our purpose is to obtain the attendance, positions and images of students' faces, which are useful information saved in the database.

II. ARCHITECTURE

In this paper, our system consists of any one kind of cameras. One is the CCTV Camera on the ceiling at an angle from where we see the faces of the students of the class. The other is the Web Cam above the white board in front of the class to capture the images of student's faces. The procedure of our system consist the following steps (see below Figure 1):

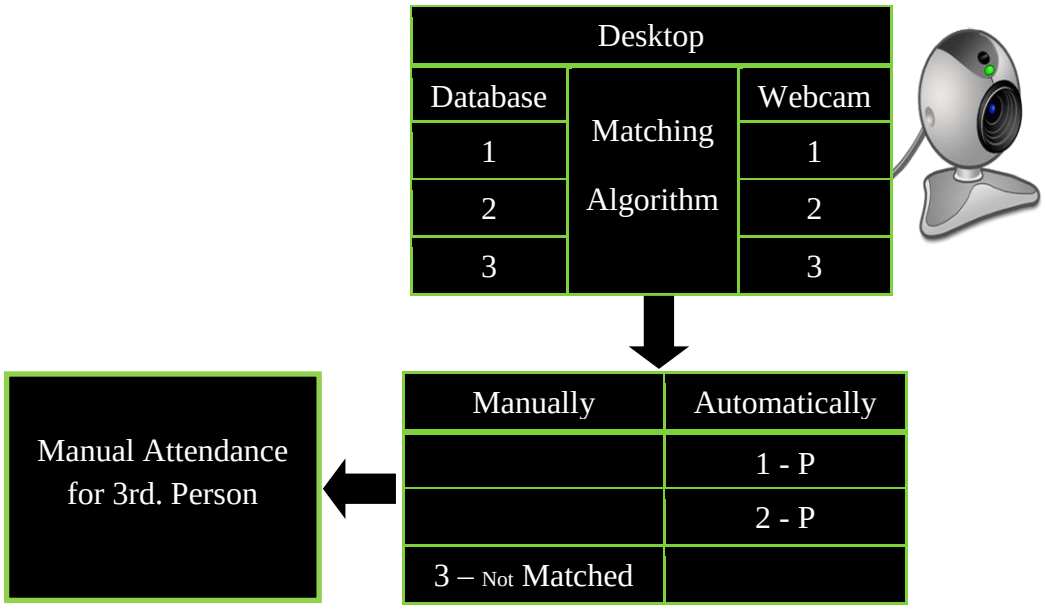

Fig 1: Architecture of the system

1. When application run on desktop automatically the Web Cam / CCTV Camera on.
2. They taking per sec one images up to ten images by coding like that
3. Then face can be detected by the Web Cam / CCTV Camera in images automatically.
4. The detected face by Cam and crop it and then saved into the Webcam folder.
5. Database having the images of the students.
6. We apply the algorithm for matching the database to the folder automatically by coding.
7. Those detected faces are repeatedly in the images and match with database the application give automatically attendance to him.
8. Those images not match in database they give error message on desktop to Instructor.
9. They give not match person list to them and Instructor take manual attendance to them only.

III. RELATED WORK

The first attempts to use face recognition began in the 1960's with a semi-automated system. Marks were made on photographs to locate the major features; it used features such as eyes, ears, noses, and mouths. Then distances and ratios were computed from these marks to a common reference point and compared to reference data. In recent decade, a number of algorithms for face recognition have been proposed, but most of these works deal with only single image of a face at a time. By taking images throw web cam /CCTV Camera we observe the face information, our approach can solve the problem of the face detection, and improve the accuracy of face recognition using techniques [5].

IV. IMPLEMENTATION DETAILS

EMGU

Emgu CV [6] is a cross platform .NET wrapper to the OpenCV image processing library. Allowing OpenCVfunctions to be called from .NET compatible languages such as C#, VB, VC++, Iron Python etc.We are using in our application these third party libraries to run c program into .NET. Essential library of EMGU**:**

using Emgu.CV;

using Emgu.Util;

using Emgu.CV.Structure;

using System.Drawing;

If these libraries are not found when running the program they give error message to the user.

Face Detection

haarcascade_frontalface_alt_tree.xml: It is a cascade file in XML used to obtain Haar cascade [7] for the frontal face in the image. It is used in the EMGU CV library.

StudentAttendence.xls: It records the attendance of the detected face according to the system time in excel sheet.

StudentAttendence.doc: It is same as the above file; the only differences that it saves the records in document format which can be easily printed for the detailed information.

Face Recognition

example.m: It is the first page to be shown to the user. It calls the other files in this module.

It takes input training dataset and also inputs the test dataset.

CreateDatabase.m: This module is in excel sheet used to create database for the face images in the training dataset in a sequence of increasing numbers as the face images in the dataset are in number format.

EigenfaceCore.m: This module in the face recognition stage calculates the eigenface value using PCA and then applying the LDA algorithm on the result of PCA.

EigenObjectRecognizer Works [8]: There are 3 different constructors for the EigenObjectRecognizer class. Each constructor takes an array of Grayscale images, each one of these images muct be the same size and it is suggested that the histograms are equalised. The reason histogram equalisation is suggested is that it can produce more desirable results when lighting within the image changes. [9] This is common when using web cameras as they tend to depend on natural lighting that is not always uniform.

EigenObjectRecognizer (Image<Gray, Byte> [], MCvTermCriteria)

The simplest constructer takes an array of images, as this recogniser takes only images as its training (web image) data it will return an image of its closest match. This can be useful, especially if you wish to use the data within the closet match to compare against that of the input image.

EigenObjectRecognizer (Image<Gray, Byte > [], String [], MCvTermCriteria)

This constructor takes an array of images and an array of strings, both equal in size. The results from this recogniser will be a string. This string can be used as an identifier to control the process flow of a program.

EigenObjectRecognizer (Image<Gray, Byte > [], String [], Double, MCvTermCriteria)

This is the constructor we use for face recognition [10]. Like the previous constructor it takes an array of images and an array of strings of the same size. Returned is a string identifier. An additional Double is used to control an eigenDistance [11] Threshold; the suggested value is 0 - ~1000. In practice this is not always true in the source you will notice a value of 5000 is actually used. The eigenDistanceThreshold sets how likely an examined image will be treated as unrecognized object. If the threshold is < 0, the recognizer will always treat as the examined image as one of the known objects.

The variable MCvTermCritera is the termination criteria for the training of the Eigen Recognizer. As the Eigen Recognizer [12] is a form of Neural Network you can set when you want to stop finding a perfect solution or when the Neural Network is said to have converged. We do this a Neural Network may never be able to find a perfect solution to a problem and convergence will never happen. If you set this value to high then you will receive more errors, set it to low and you will end up with a continuous loop.

- MCvTermCriteria (Int32, Double)

Eigen faces Algorithm [13]:

Stage 1: Subtract the Mean of the data from each variable (our adjusted data)

Stage 2: Calculate and form a covariance Matrix

Stage 3: Calculate Eigenvectors and Eigen values [14] from the covariance Matrix

Stage 4: Chose a Feature Vector (a fancy name for a matrix of vectors)

Stage 5: [15] Multiply the transposed Feature Vectors by the transposed adjusted data

How the Application run:

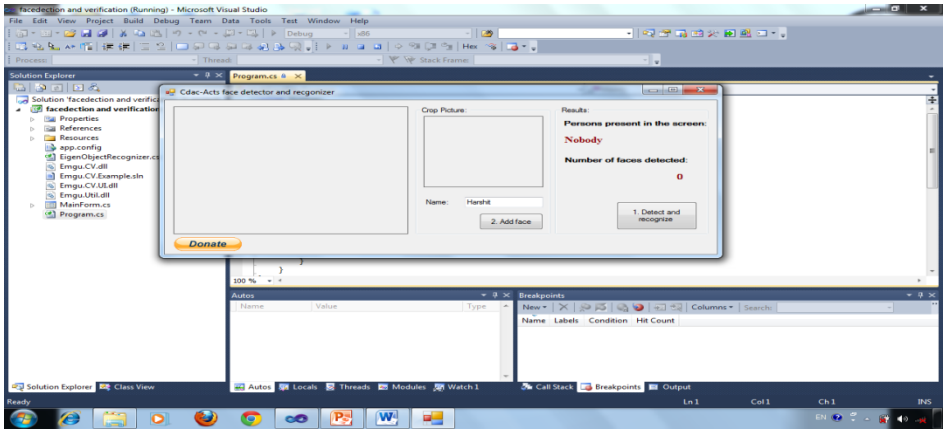

Fig 2. When Application running

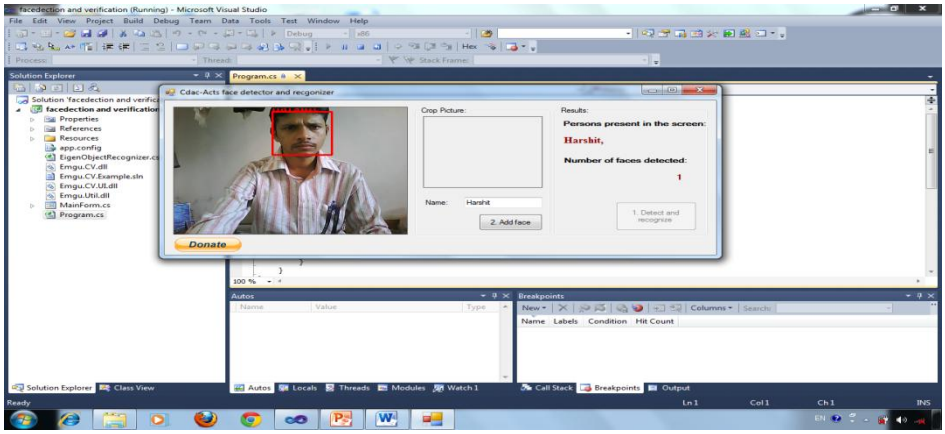

Fig 3. When face detected by the web cam

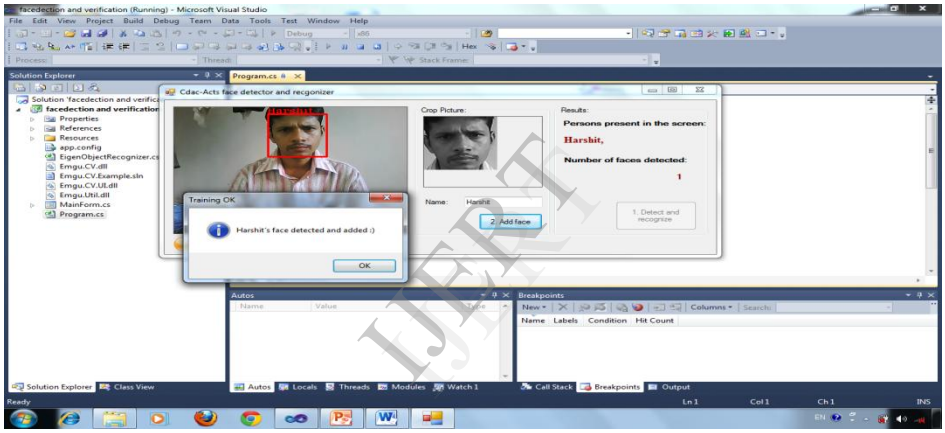

Fig 4. Save it and match with database and give name of the person

V. CONCLUSION AND FUTURE WORK

This paper introduces the efficient and accurate method of attendance without proxy in the classroom environment that can replace the old manual methods as well as RFID card reader. This method is secure enough, reliable and available for use. No need for specialized hardware for installing the system in the classroom. It can be constructed using a camera and computer. There is a need to use some algorithms that can recognize the faces in to improve the system performance.

In future work we can use as a client-server application. In this server database fully copy / replica in client side if any condition if server down is no matter because we have same database in client side.

VI. REFERENCES

[1] Guillaume Dave, Xing Chao, Kishore Sriadibhatla "Face Recognition in Mobile Phones".

[2] C# Implementation of a Face Detection System using Template Matching and Skin colour Information.

[3] Case Study Bangladesh Voter Registration Duplicate Search System.

[4] Nirmalaya Kar,Mrinal Kanti Debbarma,Ashim Saha "Study of Implementing Automated Attendance System Using Face Recognition Technique".

[5] M.Turk and A. Pentland, "Eigen faces for Recognition", Journal of Cognitive Neuroscience, March 1991.

[6] http://www.emgu.com/wiki/index.php/Main_Page

[7] Menezes, P., Barreto, J.C. and Dias, J. Face tracking based on Haar-like features and eigenfaces. 5th IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon, Portugal, July 5-7, 2004.

[8] Zhao, W., Chellappa, R., Phillips, P. J., Rosenfeld, A., 2003, Face recognition: A literature survey, ACM Computing Surveys (CSUR), V. 35, Issue 4, pp. 399-458.

[9] Belhumeur, V., Hespanda, J., Kiregeman, D., 1997, "Eigenfaces vs. fisherfaces: recognition using class specific liear projection", IEEE Trans.on PAMI, V. 19, pp. 711-720.

[10] Y. Li, S. Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In IEEE International Conference on Automatic Face and Gesture Recognition, March 2000.

[11] Aditya kelkar,"Face recognition using Eigen faces Approach"

[12] Dimitri Pissarenko, "Eigenface-based facial recognition"

[13] Ming-Hsuan Yang, "Recent Advances in Face Recognition"

[14] J. Zhu and Y. L. Yu, "Face Recognition with Eigenfaces," IEEE International Conference on Industrial Technology, pp. 434 -438, Dec. 1994

[15] Y. Cui, J. S. Jin, S. Luo, M. Park, and S. S. L. Au, "Automated Pattern Recognition and Defect Inspection System," in proc. 5th International Conference on Computer Vision and Graphical Image, vol. 59, pp. 768 – 773, May 1992.