

# Face Detection System for Attendance of Class' Students

Muhammad Fuzail<sup>1</sup>, Hafiz Muhammad Fahad Nouman<sup>2</sup>, Muhammad Omer Mushtaq<sup>3</sup>, Binish Raza<sup>4</sup>, Awais Tayyab<sup>5</sup>,  
Muhammad Waqas Talib<sup>6</sup>

<sup>1,2,3,4,5,6</sup>Department of Computer Science and Engineering, University of Engineering and Technology, Lahore, Pakistan  
m.fuzail@ymail.com<sup>1</sup>, engr.fahadnoman@gmail<sup>2</sup>, engr.omermushtaq@gmail.com<sup>3</sup>, binish155@yahoo.com<sup>4</sup>,  
awaisriaz333@yahoo.com<sup>5</sup>, waqastalib@hotmail.com<sup>6</sup>

**Abstract**– Face detection technology has widely attracted attention due to its enormous application value and market potential, such as face recognition and video surveillance system. Real-time face detection not only is one part of the automatic face recognition system but also is developing an independent research subject. So, there are many approaches to solve face detection. This paper introduces a new approach in automatic attendance management systems, extended with computer vision algorithms. We propose using real time face detection algorithms integrated on an existing Learning Management System (LMS), which automatically detects and registers students attending on a lecture. The system represents a supplemental tool for instructors, combining algorithms used in machine learning with adaptive methods used to track facial changes during a longer period of time. This new system aims to be less time consuming than traditional methods, at the same time being nonintrusive and not interfere with the regular teaching process. The tool promises to offer accurate results and a more detailed reporting system which shows student activity and attendance in a classroom.

**Keywords**– Computer Vision, Object Tracking, Face Recognition, Machine Learning and Teaching

## I. INTRODUCTION

A key factor of improving the quality of education is having students attend classes regularly. Traditionally students are stimulated to attend classes using attendance points which at the end of a semester constitute a part of a students' final grade. However, traditionally this presents additional effort from the teacher, who must make sure to correctly mark attending students, which at the same time wastes a considerable amount of time from the teaching process. Furthermore it can get much more complicated if one has to deal with large groups of students.

This paper introduces a new automatic attendance management marking system, without any interference with the regular teaching process. The system can be used also during exam sessions or other teaching activities where attendance is obligatory. This system eliminates classical student identification such as calling student names, or checking respective identification cards, which can not only interfere with the teaching process, but also can be stressful for students during exam sessions.

## II. RELATED WORK

Two researchers Visar Shehu and Agni Dika proposed in [1] a system which introduces an attendance marking system, which integrates computer vision and face recognition algorithms into the process of attendance management. The system is implemented using a non-intrusive digital camera installed on a classroom, which scans the room, detects and extracts all faces from the acquired images. After faces have been extracted, they are compared with an existing database of student images and upon successful recognition a student attendance list is generated and saved on a database. This paper addresses problems such as real time face detection on environments with multiple objects, face recognition algorithms as well as social and pedagogical issues with the applied techniques.

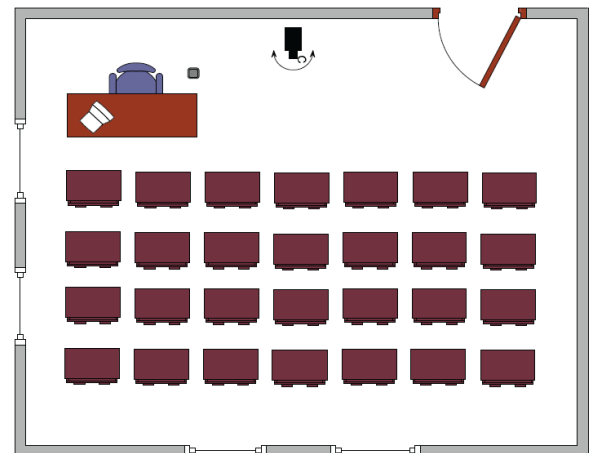


Fig. 1. Class Room Setup [1]

In [2] PAN Xiang described work process of a system as given below:

When a person wants to enter the access control system, he used the RFID card to swiping card by non-touch way. The system reads the information in the card and meanwhile the video camera is started to take photos of the person. Then the face can be detected in a short time. The identity information in the card is compared to the information from the database

and the corresponding face data will be obtained. If the identity information and the face data are all matched to the information from the database, the person will be passed. Else he can't enter. The manager can do the manage work such as query the records.

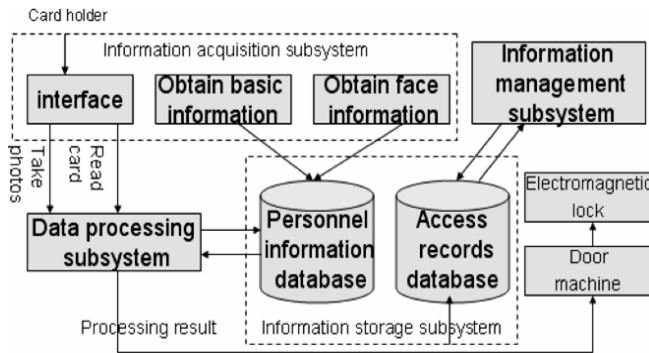


Fig. 2. Framework of access control system [2]

In [3] a technique based on ear is also introduced that is a photo of the subject's ear is taken and fed into the computer. Edge detection is carried out on this picture. From this detected edge, is separated a reference line with respect to which other features are identified. These extracted features are stored in a database in the form of a vector, each vector corresponding to a particular image in the database. The feature vector of the test image obtained is compared with those in the vector database, For creating and maintaining database for records of individuals and feature vectors, which are used for the purpose of comparison and decision making, linking of MATLAB and some data base using ODBC Drivers is carried out according to which a match is calculated. This match is compared with a predecided threshold value, which decides the identity of the person.

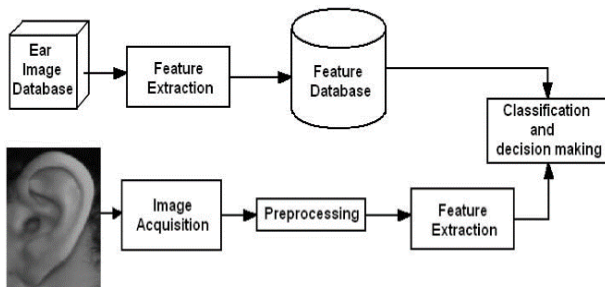


Fig. 3. Ear based attendance system framework [3]

The [4] proposed a framework for fast embedded face detection system based on three modules. One fast face detection method based on optimized AdaBoost algorithm with high speed and high detection rate, one SOC hardware framework to speed up detection operations and one software distribution strategy to optimize the memory sub-system.

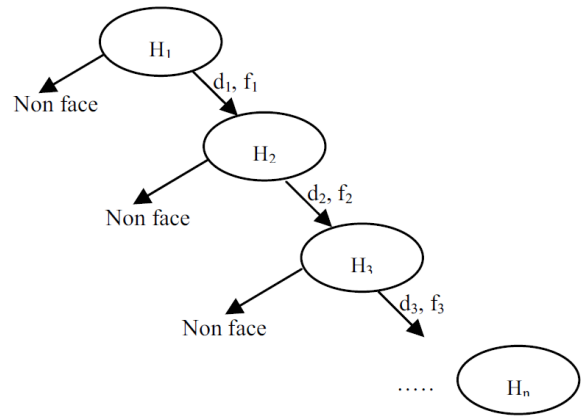


Fig. 4. Illustration of cascade architecture with n nodes

The [6] describes a Real Time System developed for Multi-face detection. As most of the system are based on software algorithms. This proposed system is based on hardware design to enhance the processing time. The different stages of this hardware design includes skin color detection, morphology, Fast connected-component labeling algorithm, Implementation of the Fast connected-component labeling algorithm, Lip feature extraction, Horizontal edge detection.

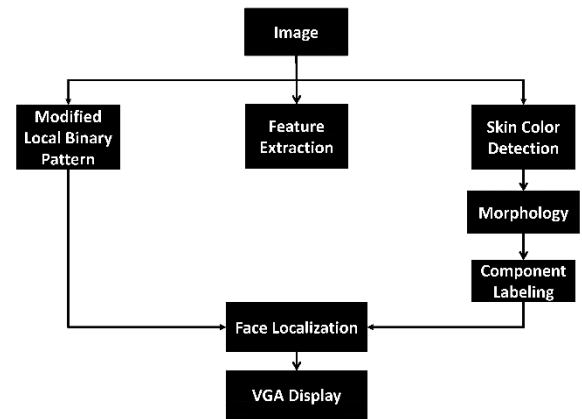


Fig. 5. Algorithm Flow

The [8] proposed a method of face and head detection simultaneously for real time surveillance system which employs four directional features (FDF) and linear discriminant analysis. FDF is one of the robust features to distinguish patterns. The FDF represents four directional (vertical, horizontal, and both diagonals) edge features of the input image. The proposed method achieved the performance of approximately over 10 fps for detection for implementation and so required a lot of improvements.

The [10] for a sequence of images usually in videos, face detection problem has been solved in two main approaches. The first way is to detect faces for every frame without using temporal information. The other is to detect a face in the first frame and then track the face through the sequence. This paper presents an improved face detection system in video sequence based on the first way. To reduce the effect of

illumination changing caused by automatic focus of camera, we propose an adaptive selection of skin color models to receive more reasonable skin detection. Unreasonable skin regions are discarded by facial geometric conditions of human faces. Reasonable ones are prevented by replacing them with elliptic skin regions. Then we get the most potential ones considered as face candidates. We propose a modified LBP considering not only local spatial textures but also principal local shapes. A histogram of modified LBP coefficients is considered as facial representation. A combination of template matching and appearance-based methods is used for classification step. LBP histogram matching and eHMMs are combined into a hierarchical classifier to identify if face candidates are human faces.

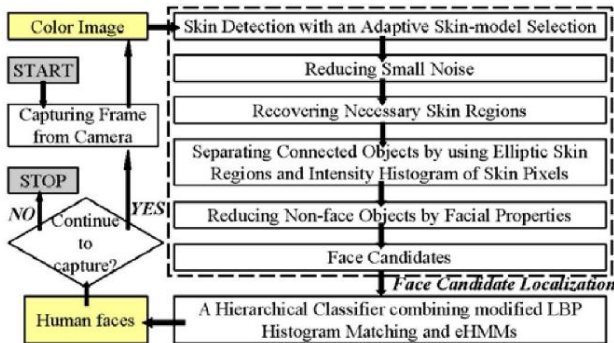


Fig. 6. Face Detection System

### III. PROPOSED SYSTEM'S ARCHITECTURE

The system designed is part of an in-house built learning management suite (Libri) [11]. It is constructed in many modules:

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

The entire process is described in the pseudo code shown in Fig 7:

```

scan room with digital camera
for each detected object
    transfer the object in server
next object

for each face in server
    if face in students database
        record student as enrolled
        save face in students database
    else
        mark face as unrecognized
next face

```

Fig. 7. System Basic Algorithm

The required infrastructure in classroom is a rotating camera positioned centrally in the front of the classroom. Using this setup, the camera is capable to capture frontal images from students A different approach would be to use a camera at the entrance of the classroom, which would individually detect faces for everyone entering the classroom.

This way, the face detector would have much less work to do, but there would be only one chance to capture a good frame. A frontal camera in the middle of the classroom can take as many pictures as necessary. Physically the system is integrated on the existing South East European Universities infrastructure. To function, the system requires each classroom to have at least one internet connected computer. This computer communicates with the LMS server, where captured images are transferred. Fig. 8 depicts the physical architecture of this system.

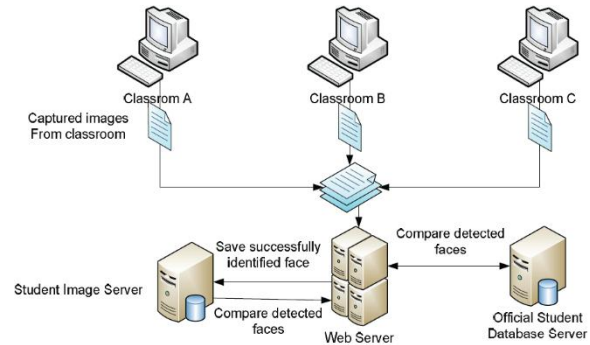


Fig. 8. Physical Architecture of the System

#### A) Image Capturing

Images are captured using a module that is a digital camera whose link is integrated to the application that is developed using the proposed idea. After an image is captured, using web services transfers the image on server 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 continuously takes pictures on a given interval (by default each five minutes), until all faces detected are successfully identified or until the system is told to stop. This means that in some cases, e.g., when a face cannot be successfully identified, the camera keeps taking pictures until the class finishes.

#### 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 detection 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. Our university, upon enrollment takes pictures from every student, and those images 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 (Pyfaces - <http://pyfaces.blogspot.com/>) 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 is 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 automatically by our system.

To speed up the face recognition process we only compare images captured in a classroom, with the database of students enrolled for that course only. This ensures that we process only a small subset of images available on our central data base.

## IV. PRIVACY CONCERNS

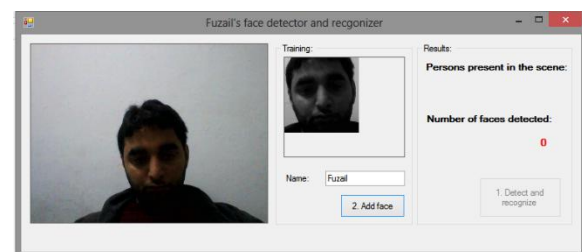
As mentioned previously, we store image data on server. This approach can always raise privacy concerns, regarding the safety of those images and access level to that server. Whenever one has to deal with this type of sensitive data, he must always offer means to protect data from unauthorized access. In some countries, especially in EU [4], these privacy concerns are considered very seriously with regulations in place in their corresponding legislatures.

Since this system is mainly done to explore the advantages and the feasibility of using face recognition in attendance management, it still does not offer any privacy protection.

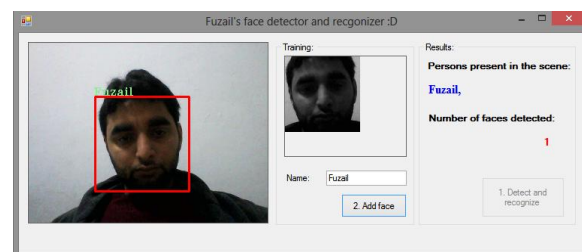
However, if the system proves to be usable one has to implement some type of secure means to ensure privacy. In password security one of the most popular approaches in this case is to encrypt data using a one way function. For us this means to encrypt images gotten from the camera and compare those images with existing (also encrypted) images on server. However, since biometric data is noisy, this approach is not feasible. Another approach can be to use one way encryption using a private/public key approach. This way only the person in possession of the correct key is able to decrypt the data.

## V. SOFTWARE APPLICATION

This section gives a description of a software application that implements the proposed idea. In the training the image is captured and saved in the data base. And from then the system is able to understand the face identity that is shown in the second image (Fig. 9) that is detection and recognition.



Training



Detection and Recognition

Fig. 9: Description of a software application



## VI. CONCLUSION

An automatic attendance management system is a necessary tool for any LMS. Most of the existing systems are time consuming and require for a semi manual work from the teacher or students. This approach aims to solve the issues by integrating face recognition in the process. Even though this system still lacks the ability to identify each student present on class, there is still much more room for improvement. Since we implement a modular approach we can improve different modules until we reach an acceptable detection and identification rate. Another issue that has to be taken in consideration in the future is a method to ensure users privacy. Whenever an image is stored on our servers, it must be impossible for a person to use that image.

## REFERENCES

- [1]. Visar Shehu, Agni Dika, "Using Real Time Computer Vision Algorithms in Automatic Attendance Management Systems" ITI 2010 32nd Int. Conf. on Information Technology Interfaces, June 21-24, 2010, Cavtat, Croatia.
- [2]. PAN Xiang, "Research and Implementation of Access Control System based on RFID and FNNFace, 2012 International Conference on Intelligent Systems Design and Engineering Application 12 2012 2012.400 Recognition", 978-0-7695-4608-7/11 \$26.00 © 2011 IEEE
- [3]. Mr.Jitendra B. Jawale Ear Based Attendance Monitoring System Proceedings of ICETECT 2011, 978-1-4244-7926-9/11/\$26.00 ©2011 IEEE
- [4]. Jian Xiao, Gugang Gao, Chen Hu, Haidong Feng "A Novel Framework for Fast Embedded Face Detection System", 1-4244-1132-7/07/\$25.00 © 2007 IEEE
- [5]. Jing Bian, Wei Du, "Research of Face Detection System Based on Skin-Tone Feature", 978-1-61284-722-1/11/\$26.00 ©2011 IEEE
- [6]. Nai-Jian Wang "A Real-time Multi-face Detection System Implemented on FPGA", 2012 IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2012) November 4-7, 2012
- [7]. Xianghua Fan, Fuyou Zhang, Haixia Wang, Xiao Lu, "The System of Face Detection Based on OpenCV", 978-1-4577-2074-1/12/\$26.00\_c 2012 IEEE
- [8]. Yohei Ishii, "Face and Head Detection for a Real-Time Surveillance System", Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04) 1051-4651/04 \$ 20.00 IEEE
- [9]. Rahul Chandrasekaran, "Security Assurance Using Face Recognition & Detection System Based On Neural Networks", The 2005 IEEE International Conference on Neural Networks and Brain ICNN&B'05
- [10]. Phuong-Trinh Pham-Ngoc, "Multi-face Detection System in Video Sequence", 1-4244-0427-4/06/\$20.00 ©2006 IEEE
- [11]. Lejla Abazi-Bexheti. Development of a Learning Content Management System. WSEAS Transactions on Information Science and Applications.