



# Face Detection and Recognition With and Without MS

Y.Bhargavi Devi <sup>1</sup>

<sup>1</sup> M.Tech student, Dept of CSE  
Dr. Samuel George Institute of  
Engineering & Technology  
Markapur, Prakasam Dt ,  
Andhra Pradesh, India.

D.Kumar <sup>2</sup>

<sup>2</sup> Associate Professor, Dept of CSE  
Dr. Samuel George Institute of  
Engineering & Technology  
Markapur, Prakasam Dt ,  
Andhra Pradesh, India.

P.Pedda Sadhu Naik <sup>3</sup>

<sup>3</sup> Assistant Professor &HOD , Dept of CSE  
Dr. Samuel George Institute of  
Engineering & Technology  
Markapur, Prakasam Dt ,  
Andhra Pradesh, India.

**Abstract:** Our project deals with concepts in object (face) detection and recognition in real world using open source computer vision (openCV) with Eigen faces algorithm. The objective of object (face) recognition involves the extraction of different features of the human face from the face image for discriminating it from other persons. It is the problem of searching a face in reference database to find the matches as a given face. The purpose is to find a face that has highest similarity with a given face in the database. To implement the face detection and recognition using Microsoft access and without Microsoft access by manually browsing for the images in the pc to display the details of the detected images ex: name, reg no, year of study, branch The scope of the project is to detect and recognize the faces either by using ms access database method or without using ms access method and to display the details of the recognized image (face).

**Key Words:** visual studio2010, Open CV, Emgu CV, Eigen faces algorithm.

## I. INTRODUCTION

The main objective of this project is to make simple, easy and fast Detection and recognition of a image (face) and to display the details of the detected face.

Face recognition is the process of putting a label to a known face. There are many techniques for a computer to learn to recognize a known face and one of them is Eigen faces algorithm. We can use this in some simple applications. Face recognition generally involve four main steps to be followed:

- **Face detection:** It is the process of locating a face region in an image. This step does not care who the person is, just that it is a human face. It simply displays a rectangle around the faces.
- **Face preprocessing:** It is the process of adjusting the face image to look more clear and similar to other faces by adjusting the image color , contrast etc.
- **Collect and learn faces:** It is the process of saving many preprocessed faces (for each person that should be recognized), and then learning how to recognize them. The faces that are learned can be saved as an image or an xml format.

- **Face recognition:** It is the process that checks which of the collected people are most similar to the face in the camera and gives the specific details according to the program.

The same is applied to objects too.

At present face recognition is used trough out the world in many different applications. Face recognition is now widely combined with augmented reality, optical flow and many other to increase its functionality. We are trying to implement it and make a sample application through it.

## II. PROPOSED WORK

The proposed system is implemented in Microsoft visual studio 2010 by using EIGEN FACES algorithms with OPENCV libraries. The system mainly involves in four stages face detection, face preprocessing, collect and learn faces and face recognition. And the system displays the details of the specific recognized person i.e., name, reg no, year of study, branch .

The main advantage of the proposed frame work is it detect the faces and recognizes the face and display the details of the detected face instantly. This system is very flexible when implemented in educational institutions and research centers. So it can recognizes the people.

### 1.1. Block diagram of system architecture:

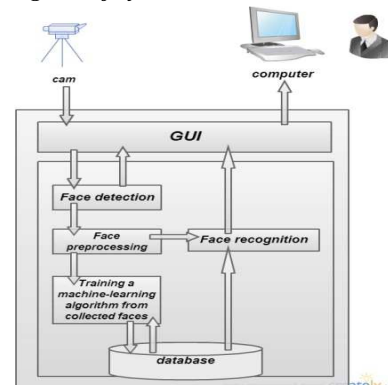


Fig 1.1: SYSTEM ARCHITECTURE



# International Journal of Ethics in Engineering & Management Education

Website: [www.ijeee.in](http://www.ijeee.in) (ISSN: 2348-4748, Volume 2, Issue 3, March 2015)

## III. RELATED WORK

Face recognition is an easy task for humans. By using open cv , Eigen faces algorithm we implemented the face recognition.

**3.1 Open CV:** Open CV is an open source computer vision library. The library is written in C and C++ and runs under Linux, Windows and Mac OS X. There is active development on interfaces for Python, Ruby, Mat lab, and other languages.

Open CV is an open source computer vision library. The library is written in C and C++ and runs under Linux, Windows and Mac OS X. There is active development on interfaces for Python, Ruby, Mat lab, and other languages. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. **Computer vision (CV)** is the transformation of data from a still or video camera into either a decision or a new representation. OpenCV includes many different applications like 2D and 3D feature toolkits, Facial recognition, Gesture recognition, Human computer Interaction, Motion Tracking, Augmented Reality. Many of this application are being used in present days like in Google Glass, Samsung S4 mobile and more.

### 3.2 Uses of Open CV:

Open CV was designed for computational efficiency and with a strong focus on real-time applications.

### 3.3 Computer vision (CV) :

It is the transformation of data from a still or video camera into either a decision or a new representation. Open CV includes many different application like 2D and 3D feature toolkits ,Facial recognition , Gesture recognition, Human computer Interaction, Motion Tracking, Augmented Reality. Many of this application are being used in present days like in Google Glass, Samsung S4 mobile and more.

### 3.4: visual studio:

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silver light. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building GUI applications, web designer, class designer, and database designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source-control systems (like Subversion) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for

other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

Visual Studio supports different programming languages and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include C,[6] C++ and C++/CLI (via Visual C++), VB.NET (via Visual Basic .NET), C# (via Visual C#), and F# (as of Visual Studio 2010[7]). Support for other languages such as M, Python, and Ruby among others is available via language services installed separately. It also supports XML/XSLT, HTML/XHTML, JavaScript and CSS.

Microsoft provides "Express" editions of its Visual Studio at no cost. Commercial versions of Visual Studio along with select past versions are available for free to students via Microsoft's Dream Spark program.

### 3.5: Emgu CV:

It is the name given to a set of eigenvectors when they are used in the computer vision problem of human face recognition.[1] The approach of using eigenfaces for recognition was developed by Sirovich and Kirby (1987) and used by Matthew Turk and Alex Pentland in face classification.[2] The eigenvectors are derived from the covariance matrix of the probability distribution over the high-dimensional vector space of face images. The eigenfaces themselves form a basis set of all images used to construct the covariance matrix. This produces dimension reduction by allowing the smaller set of basis images to represent the original training images. Classification can be achieved by comparing how faces are represented by the basis set.

## IV. EXPERIMENTAL RESULTS

**4.1.**Face Detection is to identify an object as a face and locate it in the input image. The image of the face captured by webcam with the help of Processing draw a rectangular box on face which is detected by open cv.

**4.2.**After that we use the face preprocessing is the process of adjusting the face image to look more clear and similar to other faces by adjusting the image color, contrast etc.

Collect and learn faces It is the process of saving many preprocessed faces (for each person that should be recognized), and then learning how to recognize them. The faces that are learned can be saved as an image or an xml format.

**4.3.**By using Face Recognition It is to decide if this face is someone KNOWN are or UNKNOWN, basing on the database of faces it uses to validate the input face.

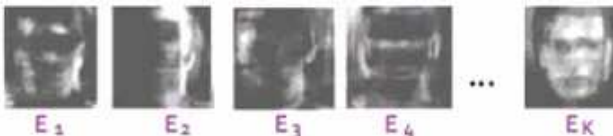
- We used Principal Component Analysis based on Eigenfaces.
- PCA based on Eigenface method is the easiest and simplest of efficient face recognition algorithm.
- PCA Eigenface has an average efficiency is up to 70%.

- ( There is no face recognition with 100% efficiency )
- In this method there is a set of face images that make up a training set. And an unknown face image; the input image. All the images must be of same size.
- PCA converts each image into vector matrices and works on these vector form.
- PCA is used to generate K Eigenfaces for training set of M images, where  $K < M$ , thereby reducing the number values from M to K needed to identify an unknown face.
- Since Eigenfaces visually depict the major “features” of training set.
- Each face can be said to have been made up of “proportions” of all these K “features” of all training set.
- E.g. face image 1 = (34% of E1)+(14% of E2)+....+(1% of Ek)
- This proportions are represented as weight vector, showing “how much of which eigenface makes up this image”
- Weight vector 1 = [0.34,0.14,.....0.01]
- To represent an image in terms of these eigenfaces with assigned weights.
- For recognition their weight vectors are compared.
- Advantage of this technique is that it reduce the data required to identify the individual to 1/1000th of the data presented.

Training set images



PCA transforms and supply with a lower-dimensional picture



- This transformation is defined in such a way that the first principal component shows the most dominant “directions/features” of the dataset and next component in turn shows the next most possible dominant “directions/features”.
- Since Principal Component show the “directions” of data, and each preceding component shows less “directions” and more “noise”, Only Few First principal components are selected whereas the rest of the last components are discarded.
- Each variable in the original dataset can be represented in terms of these K principal components.

- PCA can be done by eigenvalue decomposition of a data covariance matrix.

**4.4 START SCREEN:**

The below figure is the home screen of the application

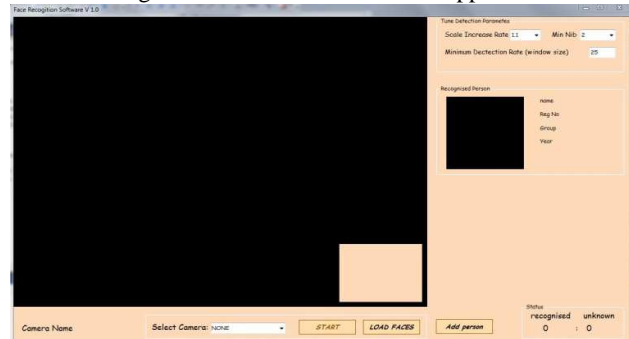


FIG 4.1 – START SCREEN WITH MS ACCESS

The camera can be started by clicking on “START ” button at the below and the option is given to start multiple cameras by using the “select camera” option located side to the START button

**4.5 IMAGE BROWSING:**

At this stage the image is loaded in to the application by browsing it by using the “LOAD IMAGE” option at the bottom



FIG 4.2 – IMAGE BROWSING WITH MS ACCESS

**4.6 ENTERING THE DETAILS FOR LOADED IMAGE:**

After loading the image to the application .we have to enter the details of the loaded image(face) ex: reguister number, first name, last name, year of study, branch on the right hand side.



FIG 4.3 – ENTERING DETAILS WITH MS ACCESS

And clicking on “save to db” the image get saved in the database with the specified details.

**4.7 Face detection phase:**

The face is detected in this stage and we can increase the scale rate which in turn results in good performance and we can also select the face style like straight wise, side wise



FIG 4.4 – FACE DETECTION WITH MS ACCESS

And the detected face is shown at the black box on right hand side and the image is saved to database. And we can save the images(faces) in different angles with the help of the face style drop down menu.

**4.8 Face recognition:**

At this stage the face is recognized and the relevant information regarding to the image are displayed.

Ex: The image(face) name, registration number and year of study



FIG 4.5 – FACE RECOGNITION WITH MS ACCESS

The green box around the face is the area of detected and recognized face and the relevant details are displayed in the right hand side

**4.9 The application without Microsoft access database: Home screen**

The figure below is the home screen of the application with two options recognition and adds faces

By clicking on the “recognition” button the application display the details of the face that’s been detected

And by clicking on the “add faces” button we can add the faces of the person.

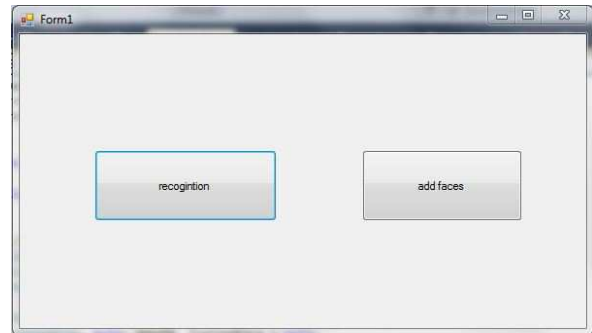


FIG 2.6 – START SCREEN WITHOUT MS ACCESS

**4.10 Adding faces:**

After clicking on the “add faces” button the camera gets started and we can add the faces .



FIG 4.7 – ADD FACES WITHOUT MS ACCESS

**4.11 Entering the details:**

After adding the faces the details of the face has to enter manually and after by clicking on “add person” .The image get saved in a folder and the details related to that images are saved to a text document.

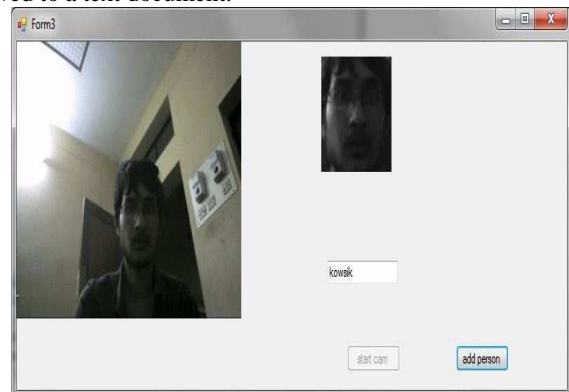


FIG 4.8 – ENTERING DETAILS WITHOUT MS ACCESS



# International Journal of Ethics in Engineering & Management Education

Website: [www.ijeee.in](http://www.ijeee.in) (ISSN: 2348-4748, Volume 2, Issue 3, March 2015)

## V. CONCLUSIONS AND FUTURE SCOPE

### Conclusion

This project mainly focuses on the face detection and recognizing to achieve the accuracy, performance and simplicity in detecting the faces .By using vs2010,open cv,eigen face recognition we implemented face detection and recognition ,emgu cv is used as an interface for open cv .In this project we implemented face recognition with eigen face recognition algorithm .it is work in two conditions if there is no ms we can run this project it is easy to perform. And this application can be implemented in educational and research institutions. And the application can be implemented in different devices.

### Future work

Till now we had done face detection and recognition of a particular face by displaying their details. And in future we will going to extend the system with an enhanced search option where we can make search of a specific person by giving their name or register number. And we will also implement the fisher faces algorithm along with eigen faces algorithm to improve the accuracy of the system.

## REFERENCES

- [1]. Pavan Kulkarni and Aditya Khatawkar "Face Detection and Tracking using OpenCV" in The SIJ Transactions on Computer Networks & Communication Engineering (CNCE), Vol. 1, No. 3, July-August 2013.
- [2]. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection, P.N. Belhumeur, J. Hespanha and D. Kriegman, Proceedings of the IEEE Transactions.
- [3]. Learning OpenCV: Computer Vision with the OpenCV Library, G Bradski and A. Kaehler, pp. 186-190, O'Reilly Media.
- [4]. Mastering OpenCV with Practical Computer Vision, Daniel Lélis Baggio, Shervin Emami, David Millán Escrivá, Khvedchenia Ievgen, Naureen Mahmood, Jason Saragih, Roy Shilkrot ,PACKT publishing.
- [5]. <http://en.wikipedia.org/wiki/OpenCV>
- [6]. [http://www.emgu.com/wiki/index.php/Main\\_Page](http://www.emgu.com/wiki/index.php/Main_Page)

## BIOGRAPHY



**Y. Bhargavi Devi** was born in Markapur, Prakasam Dt, Andhrapradesh, India. She received B.tech in C.S.E, from JNTUK Kakinada [Dr.Samel George Institute of engineering college from Markapur], Andhra pradesh, India. Presently She is pursuing M.tech in C.S.E from Dr. Samuel George institute of engineering & technology, Markapur,

Prakasam Dt , Andhra pradesh, India .she attended various national conferences.



**D. Kumar** received B. Tech(CSIT) Degree from JNT University in 2006 an M.Tech (CSE) Degree from JNTUK,Kakinada in 2011. He has 09 years of teaching experience. He joined as Assistant Professor in Dr.Samuel George Institute of Engineering& Technology, Markapur, India in 2006. Presently he is working as Associate

Professor in CSE Dept. His Interested research areas are Image Processing and Computer Networks. He attended Various National and International Workshops and Conferences.



**P. Pedda Sadhu Naik** received B.Tech (CSE) Degree from JNT University in 2003 and M.Tech (CS) Degree from JNTUCE Anapatur in 2007.He is doing Ph D from JNTUK, Kakinada in Computer Science under the guidance of Dr T. Venu Gopal. He has 10 years of teaching experience. He joined as Assistant Professor in Dr. Samuel

George Institute of Engineering & Technology, Markapur, India in 2003. Presently he is working as Associate Professor and Head of the Department of CSE. His Interested research areas are Image Processing and Soft Computing. He has life member of ISTE and IAENG. He organized various National level Technical Symposiums as Convener.He attended Various National and International Workshops and Coherences on image processing.