# A REAL TIME ANDROID BASED FACE RECOGNITION SYSTEM FOR TIME AND ATTENDANCE MANAGEMENT IN CORPORATES

|  |  |  |
| --- | --- | --- |
| **SNO** | **PARTICULARS** | **PAGE NO** |
|  | **CERTIFICATE** |  |
|  | **ACKNOWLEDEMENT** |  |
|  | **ABSTRACT** |  |
| **1.** | **INTRODUCTION** |  |
| **2.** | **LITERATURE SURVEY** |  |
| **3.** | **SYSTEM DISCRIPTION**  3.1 Objective  3.2 Problem Definition  3.3 Existing system  3.4 Proposed system  3.5 Applications |  |
| **4.** | **SYSTEM ANALYSIS** |  |
| **5.** | **FEASIBILITY STUDY**  5.1 Economical feasibility  5.2 Technical feasibility  5.3 Social feasibility |  |
| **6.** | **SYSTEM DESIGN**  6.1 Modules Details  6.2 Architecture Diagram  6.3 Algorithm Details |  |
| **7.** | **UML DIAGRAMS**  7.1 Use case Diagrams  7.2 Class Diagrams  7.3 Sequence Diagram  7.4 Collaboration Diagram  7.5 Activity Diagram  7.6 E-R Diagram |  |
| **8.** | **DATA MODEL**  8.1 Data flow diagram |  |
| **9.** | **DATABASE DESIGN** |  |
| **10.** | **SYSTEM REQUIREMENTS** |  |
| **11.** | **SOFTWARE DISCRIPTION**  11.1 About .Net  11.2 About SQL Server |  |
| **12.** | **TESTING** |  |
| **13.** | **CONCLUSION** |  |
| **14.** | **BIBLIOGRAPHY** |  |
| **15.** | **APPENDICES**  15.1 Screen Shots  15.2 Sample Coding |  |

**ABSTRACT**

This paper presents an automated system for human face recognition in a real time background for an Organization to mark the attendance of their employees. So Smart Attendance using Real Time Face Recognition is a real world solution which comes with day to day activities of handling employees. The task is very difficult as the real time background subtraction in an image is still a challenge. To detect real time human face, Principal Component Analysis (PCA) is used to recognize the faces detected with a high accuracy rate. The matched face is then used to mark attendance of the employees once recognition is done, automatically attendance will be updated in an Excel Sheet along with his name, date and time. This project gives much more solutions with accurate results in user interactive manner rather than existing attendance and leave management systems.

**CHAPTER-1**

**INTRODUCTION**

Face feature extraction is considered to be a key requirement in many applications such as Biometrics, Facial recognition systems, video surveillance, Human computer interface etc. Therefore reliable face detection is required for the success of these applications. The task of human facial feature extraction is not easy. Human face varies from person to person. The race, gender, age and other physical characteristics of an individual have to be considered thereby creating a challenge in computer vision. Facial feature detection aims to detect and extract specific features such as eyes, mouth and nose.

The system proposed a real time system. It takes input image through a web camera continuously. The main camera and attendance identification display can be placed at the entrance of the organization to get better result. It is a way of identifying different conditions of attendance marking. The camera should be installed in a place with good light in the background and free of obstacles. Attendance is automatically calculated depends on presence of employees. An android application is implemented for finding the Monthly salary details and no of working days details.

**CHAPTER-2**

**LITERATURE SURVEY**

**Face Recognition using Principle Component Analysis**

**Description**:

The Principal Component Analysis (PCA) is one of the most successful techniques that have been used in image recognition and compression. PCA is a statistical method under the broad title of *factor analysis*. The purpose of PCA is to reduce the large dimensionality of the data space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables), which are needed to describe the data economically. This is the case when there is a strong correlation between observed variables. The jobs which PCA can do are prediction, redundancy removal, feature extraction, data compression, etc. Because PCA is a classical technique which can do something in the linear domain, applications having linear models are suitable, such as signal processing, image processing, system and control theory, communications, etc.

Face recognition has many applicable areas. Moreover, it can be categorized into face identification, face classification, or sex determination. The most useful applications contain crowd surveillance, video content indexing, personal identification (ex. driver’s licence), mug shots matching, entrance security, etc. The main idea of using PCA for face recognition is to express the large 1-D vector of pixels constructed from 2-D facial image into the compact principal components of the feature space. This can be called eigenspace projection. Eigenspace is calculated by identifying the eigenvectors of the covariance matrix derived from a set of facial images(vectors). The details are described in the following section.

**An Efficient LDA Algorithm for Face Recognition**

**Descriptions**: It has been demonstrated that the Linear Discriminant Analysis (LDA) approach outperforms the Principal Component Analysis (PCA) approach in face recognition tasks. Due to the high dimensionality of a image space, many LDA based approaches, however, first use the PCA to project an image into a lower dimensional space or socalled face space, and then perform the LDA to maximize the discriminatory power. In this paper, we propose a new, unified LDA/PCA algorithm for face recognition. The new algorithm maximizes the LDA criterion directly without a separate PCA step. This eliminates the possibility of losing discriminative information due to a separate PCA step. We discuss the connection between the new algorithm and the traditionalPCA+LDA approach. We also prove that the new algorithm is equivalent to the eigenface (PCA) approach in

a special case, where each person has only one sample in the training set. The feasibility of the new algorithm has been demonstrated by experimental results.

**Feature Extraction based Face Recognition, Gender and Age Classification**

**Descriptions**: Many group communications require a security infrastructure that ensures multiple levels-of access privilege fix group members. Access control in hierarchy is prevalent in multimedia applications, which consist of users that subscribe to different quality levels or different sets of data streams. In .this paper, we present a multi-group key management scheme that achieves such a hierarchical access control by employing an integrated key graph and by managing group kegs for all wars with various access privileges Compared with applying existing tree-based group keg management schemes directly to the hierarchical access control problem, the proposed scheme significantly reduces the communication, computation and storage overhead associated with key management and achieves better scalability when the number of access levels increases. In addition, the proposed key graph is suitable for both centralized and contributory environments.

**Face Recognition for Smart Interactions**

**Descriptions:**

The face recognition efforts in the research group ”Computer Vision for Human-Computer Interaction” (CV-HCI) at the Universit¨at Karlsruhe (TH) concentrate on the development of a fast and robust face recognition algorithm and fully automatic face recognition systems that can be deployed for real-life smart interaction applications. Our face recognition algorithm is based on appearances of local facial regions that are represented with discrete cosine transform coefficients. Many fully automatic face recognition systems have been developed based on this algorithm. Among these systems two of the portable ones - an open-set verification system, and a system for interactive person retrieval in multimedia - will be shown as interactive demos. Moreover, demo videos will be shown for the other systems.

This paper, we give a short overview of the systems that will be presented.

**CHAPTER-3**

**SYSTEM DESCRIPTIONS**

**3.1 OBJECTIVE:**

The main objective of this project is to create a real time android based face recognition system for time and attendance management in corporate.

* 1. **PROBLEM DEFINITION:**

# The problems found in our existing system are Localization measured manually, Functions of this system based RFID. It is easy to miss use and Manual control of appliance is difficult to monitor. To rectify this problem we propse a new system called a real time android based face recognition system for time and attendance management in corporate.

**3.3 EXISTING SYSTEM:**

* In this system Radio Frequency Identification (RFID) technology can be used to improve the localization of mobile robots and persons in their environment.
* The mobile robot is equipped with an active RFID reader and some tags are placed in the room to provide RF Beacons in order that the robot can localize itself with the known tag geographical locations.
* Remote monitoring system or manual calculation system can be implemented to calculate the attendance.

**Disadvantages:**

* Localization measured manually.
* Functions of this system based RFID. It is easy to miss use.
* Manual control of appliance is difficult to monitor.

**3.4 PROPOSED SYSTEM:**

The system proposed a real time system. It takes input image through a web camera continuously.

The main camera and attendance identification display can be placed at the entrance of the organization to get better result. It is a way of identifying different conditions of attendance marking.

The camera should be installed in a place with good light in the background and free of obstacles. Attendance is automatically calculated depends on presence of employees.

Android applications is implemented for finding the Monthly salary details and no of working days details

**Advantages:**

* Time Management
* Accurate Salary Prediction

**3.5 APPLICATIONS:**

* Organizations
* Governments
* Organizations

**CHAPTER-4**

**SYSTEM ANALYSIS**

**REQUIREMENTS & SPECIFICATION:**

**What is SRS?**

Software Requirement Specification (SRS) is the starting point of the software developing activity. As system grows more complex it became evident that the goal of the entire system cannot be easily comprehended. Hence the needs for the requirement phase Specification. The software project is initiated by the client needs. The SRS is the means of translating the ideas of the minds of clients (the input) into a formal document (the output of the requirement phase.)

The SRS phase consists of two basic activities:

**Problem/Requirement Analysis:**

The process is order and more nebulous of the two, deals with understand the problem, the goal and constraints.

**Requirement Specification:**

Here, the focus is on specifying what has been found giving analysis such as representation, specification languages and tools, and checking the specifications are addressed during this activity.

The Requirement phase terminates with the production of the validate SRS document. Producing the SRS document is the basic goal of this phase.

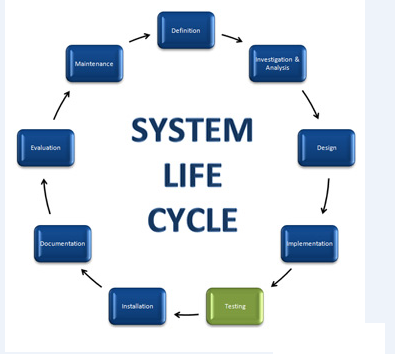
ROLE OF SRS:

The purpose of the Software Requirement Specification is to reduce the communication gap between the clients and the developers. Software Requirement Specification is the medium though which the client and user needs are accurately specified. It forms the basis of software development. A good SRS should satisfy all the parties involved in the system.

The purpose of this document is to describe all external requirements for Project Control System. It also describes the interfaces for the system.

**SCOPE:**

This document is the only one that describes the requirements of the system. It is meant for the use by the developers, and will also by the basis for validating the final delivered system. Any changes made to the requirements in the future will have to go through a formal change approval process. The developer is responsible for asking for clarifications, where necessary, and will not make any alterations without the permission of the client.



**CHAPTER-5**

**FEASIBILITY STUDY**

An analysis of the ability to complete a project successfully, taking into account legal, economic, technological, scheduling and other factors. Rather than just diving into a project and hoping for the best, a feasibility study allows project managers to investigate the possible negative and positive outcomes of a project before investing too much time and money.

**5.1 TECHINICAL FEASIBILITY:**

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at this point in time, not too many detailed design of the system, making it difficult to access issues like performance, costs on (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis.

Understand the different technologies involved in the proposed system:

Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system.

Find out whether the organization currently possesses the required technologies:

Is the required technology available with the organization?

If so is the capacity sufficient?

For instance –

“Will the current printer be able to handle the new reports and forms required for the new system?”

**5.2 ECONOMIC FEASIBILITY:**

It refers to the benefits or Outcomes we are deriving from the product as compared to the total cost we are spending for developing the product. If the benefits are more or less the same as the older system, then it is not feasible to develop the product.

In the present system, the development of new product greatly enhances the accuracy of the system and cuts short the delay in the processing of Birth and Death application. The errors can be greatly reduced and at the same time providing a great level of security. Here we don’t need any additional equipment except memory of required capacity.

No need for spending money on client for maintenance because the database used is web enabled database.

**5.3 SOCIAL FEASIBILITY:**

Proposed projects are beneficial only if they can be turned into information systems that will meet the organizations operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? Here are questions that will help test the operational feasibility of a project:

Is there sufficient support for the project from management from users? If the current system is well liked and used to the extent that persons will not be able to see reasons for change, there may be resistance.

Are the current business methods acceptable to the user? If they are not, Users may welcome a change that will bring about a more operational and useful systems.

Have the user been involved in the planning and development of the project? Early involvement reduces the chances of resistance to the system and in General and increases the likelihood of successful project.

Since the proposed system was to help reduce the hardships encountered in the existing manual system, the new system was considered to be operational feasible.

**CHAPTER-6**

**SYSTEM DESIGN**

**6.1 MODULES DETAILS:**

**Modules:**

* + Face Detection
  + Face Recognition
  + Face Authentication & Attendance Maintenance
  + Salary Calculation
  + Android App Creation

**Modules descriptions:**

**FACE DETECTION:**

The Principal Component Analysis (PCA) is one of the most successful techniques that have been used in image recognition and compression. PCA is a statistical method under the broad title of factor analysis. The purpose of PCA is to reduce the large dimensionality of the data space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables), which are needed to describe the data economically. This is the case when there is a strong correlation between observed variables.

The jobs which PCA can do are prediction, redundancy removal, feature extraction, data compression, etc. Because PCA is a classical technique which can do something in the linear domain, applications having linear models are suitable, such as signal processing, image processing, system and control theory, communications, etc. Face recognition has many applicable areas. Moreover, it can be categorized into face identification, face classification, or sex determination.

**FACE RECOGNITION:**

In this project, you will implement a face recognition system using the Principal Component Analysis (PCA) algorithm. Automatic face recognition systems try to ﬁnd the identity of a given face image according to their memory. The memory of a face recognizer is generally simulated by a training set. In this project, our training set consists of the features extracted from known face images of diﬀerent persons. Thus, the task of the face recognizer is to ﬁnd the most similar feature vector among the training set to the feature vector of a given test image. Here, we want to recognize the identity of a person where an image of that person (test image) is given to the system.

You will use PCA as a feature extraction algorithm in this project. In the training phase, you should extract feature vectors for each image in the training set. Let ΩA be a training image of person A which has a pixel resolution of M £ N (M rows, N columns). In order to extract PCA features of ΩA, you will ﬁrst convert the image into a pixel vector ÁA by concatenating each of the M rows into a single vector. The length (or, dimensionality) of the vector ÁA will be M £ N. In this project, you will use the PCA algorithm as a dimensionality reduction technique which transforms the vector ÁA to a vector !A which has a dimensionality d where d ¿ M £ N. For each training image Ωi , you should calculate and store these feature vectors .

**ATTENDANCE MAINTENANCE:**

When face recognition success mean, our system automatically put attendance for that particular person. And also it maintains a proper attendance details for each person.

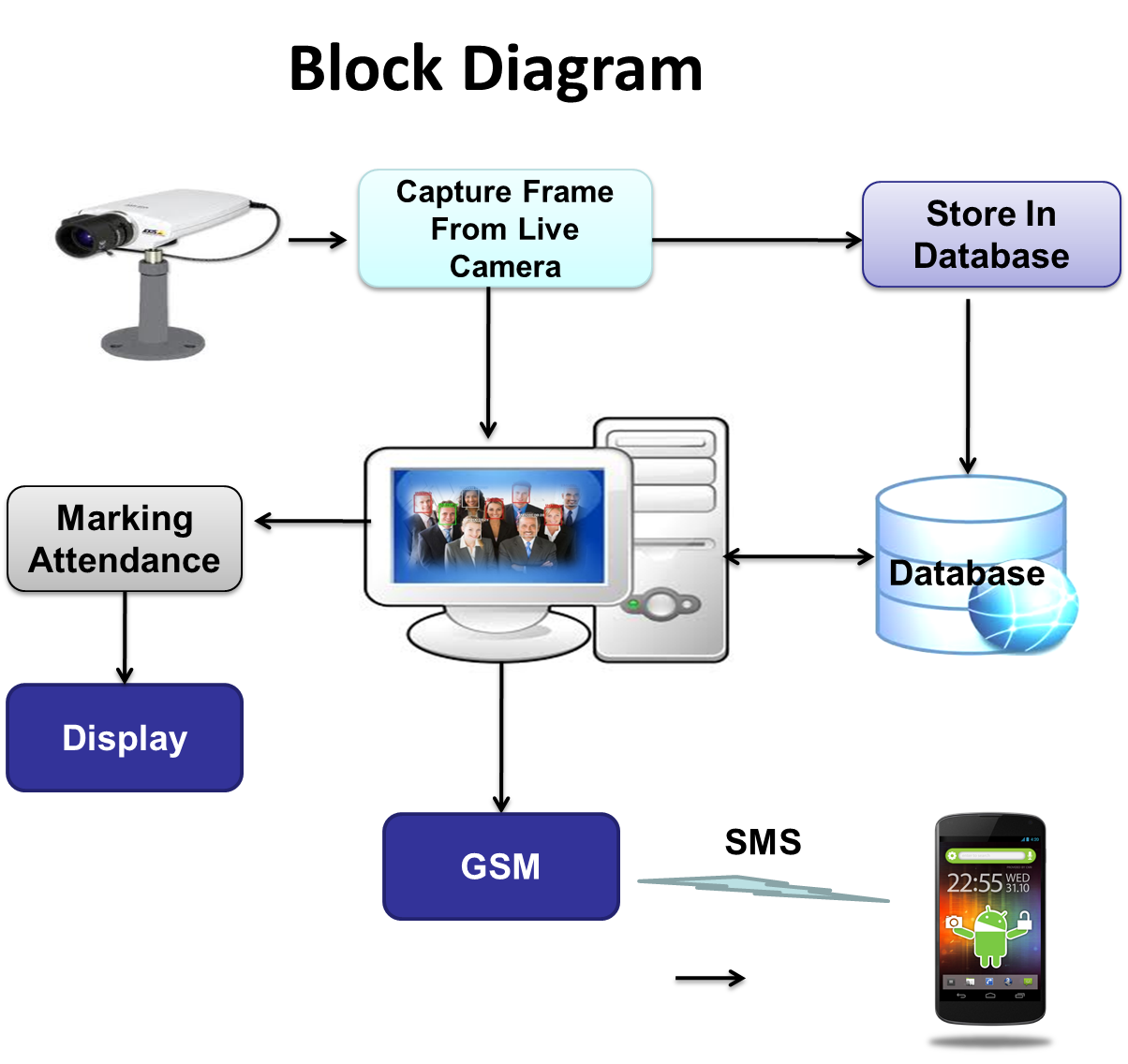
**SALARY CALCULATION:**

In this modules used for calculate a accurate salary for each person depend upon their attendance record. In salary calculation depend upon person working hours.

**ANDROID APP CREATION:**

In this module create an android application for known salary information. If the person or employee wants to know their salary information mean, they simply send a SMS to control section. In control section, When SMS received form employee means it automatically send information to that particular employee.

**6.2 ARCHITECTURE DIAGRAM:**

****

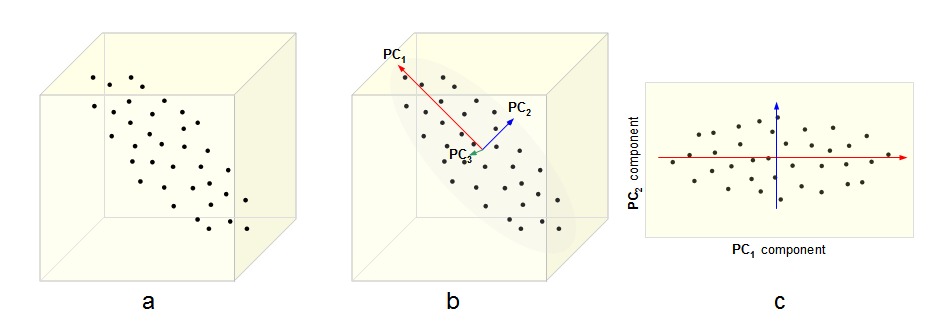
**6.3: ALGORITHM DETAILS:**

**Algorithm used:**

1. PCA

**Algorithm Explanations:**

* **Objective:**
  + To discover or to reduce the dimensionality of the data set.
  + To identify new meaningful underlying variables.
* Facial recognition system is a computer application for automatically identifying or verifying a person from a digital image or a video frame from a video source. One of the ways to do this is by comparing selected facial features from the image and a facial database.
* The increasing interest in the evaluation of biometric systems security has led to the creation of numerous and very diverse initiatives focused on this major field of research.
* To ensure the actual presence of a real legitimate trait in contrast to a fake self-manufactured synthetic or reconstructed sample is a significant problem in biometric authentication.

****

* + Data is projected into a lower dimensional space
  + preserving the directions that are most significant
  + Not necessarily orthogonal to the original ones!

**CHAPTER-7**

**UML DIAGRAMS**

**Introduction:**

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer’s requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

There are various kinds of methods in software design:

They are as follows:

* Use case Diagram
* Class Diagram
* Sequence Diagram
* Collaboration Diagram
* Activity Diagram
* E-R Diagram

**7.1. Use case diagrams:**

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what’s called an actor.Use case diagram can be useful for getting an overall view of the system and clarifying that can do and more importantly what they can’t do.Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

* The purpose is to show the interactions between the use case and actor.
* To represent the system requirements from user’s perspective.
* An actor could be the end-user of the system or an external system.

****

**7.2. Class diagram:**

Class is nothing but a structure that contains both variables and methods. The Class Diagram shows a set of classes, interfaces, and collaborations and their relating ships. There is most common diagram in modeling the object oriented systems and are used to give the static view of a system. It shows the dependency between the classes that can be used in our system.

* The interactions between the modules or classes of our projects are shown below. Each block contains Class Name, Variables and Methods.

****

**7.3. Sequence diagram:**

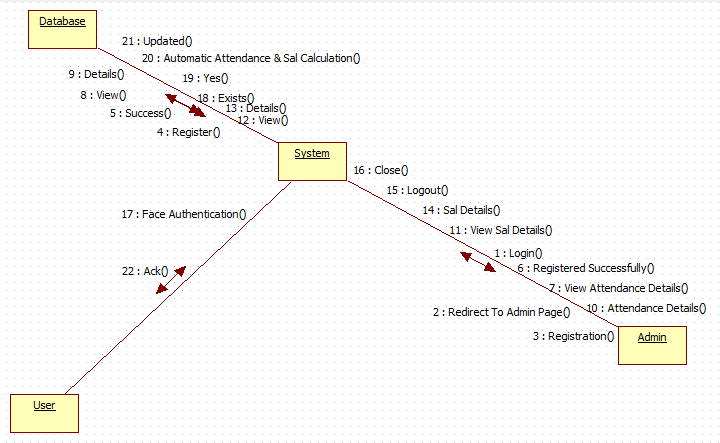
Sequence diagram and collaboration diagram are called INTERACTION DIAGRAMS. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them.

A sequence diagram is an introduction that empathizes the time ordering of messages. Graphically a sequence diagram is a table that shows objects arranged along the X-axis and messages ordered in increasing time along the Y-axis

****

**7.4. Collaboration diagram:**

A collaboration diagram is an introduction diagram that emphasizes the structural organization of the objects that send and receive messages. Graphically a collaboration diagram is a collection of vertices and arcs.

****

**CHAPTER-8**

**DATA MODEL**

**Descriptions:**

The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labeled arrows and transformation are represented by circles also called as bubbles. DFD is presented in a hierarchical fashion i.e. the first data flow model represents the system as a whole. Subsequent DFD refine the context diagram (level 0 DFD), providing increasing details with each subsequent level.

The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst performs an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of the data as it moves through the process that embodies the applications.

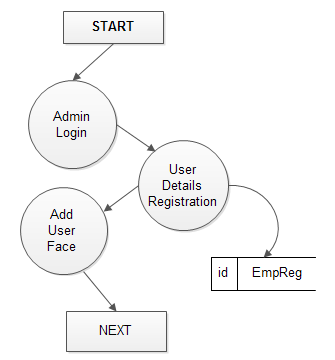
A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

**Rules for DFD:**

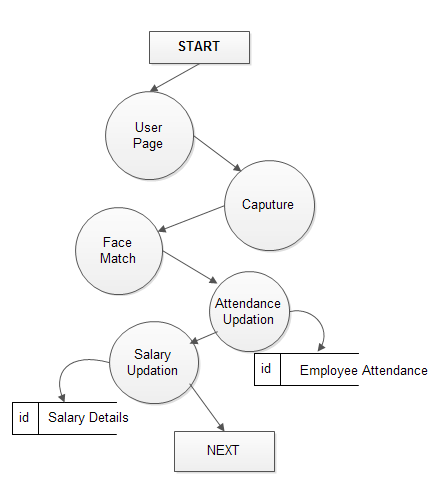
* Fix the scope of the system by means of context diagrams.
* Organize the DFD so that the main sequence of the actions
* Reads left to right and top to bottom.
* Identify all inputs and outputs.
* Identify and label each process internal to the system with Rounded circles.
* A process is required for all the data transformation and Transfers. Therefore, never connect a data store to a data Source or the destinations or another data store with just a Data flow arrow.
* Do not indicate hardware and ignore control information.
* Make sure the names of the processes accurately convey everything the process is done.
* There must not be unnamed process.
* Indicate external sources and destinations of the data, with Squares.
* Number each occurrence of repeated external entities.
* Identify all data flows for each process step, except simple Record retrievals.
* Label data flow on each arrow.
* Use details flow on each arrow.
* Use the details flow arrow to indicate data movements.

**8.1 DATA FLOW DIAGRAMS:**

**Level 0:**

****

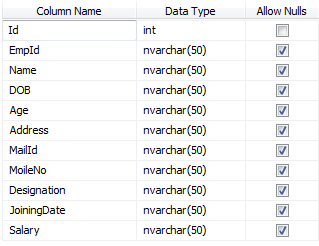
**Level 1:**

****

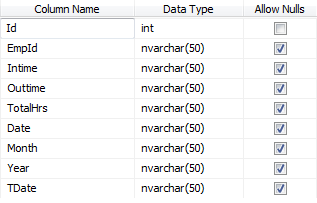
**CHAPTER-9**

**DATABASE DESIGN**

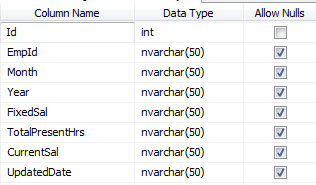
**Employee Registration Details:**

****

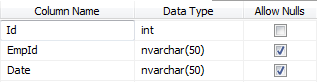
**Employee Attendance Details:**

****

**Employee Salary Details:**

****

**SMS Status Details:**

****

**CHAPTER-10**

**SYSTEM REQUIREMNTS**

**Software Requirements:**

* Operating system :- Windows7
* Front End :- Microsoft Visual Studio .Net 2010
* Coding Language :- C#
* Backend :- SQL Server 2008

**Hardware Requirements:**

* Processor : Pentium Dual Core 2.00GHZ
* Hard disk : 40 GB
* Mouse : Logitech.
* RAM : 2GB(minimum)
* Keyboard : 110 keys enhanced.

**Hardware Modules:**

* GSM Module
* Web Cam

**CHAPTER-11**

**SOFTWARE DESCRIPRIONS**

11.1 ABOUT .NET

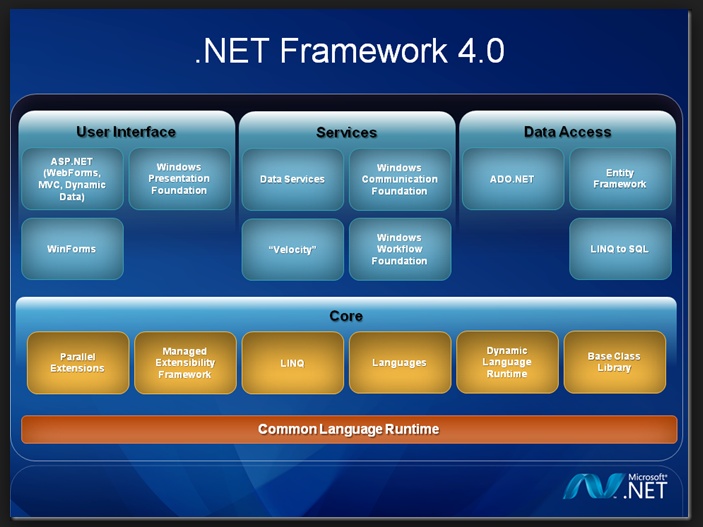
**Overview of the .NET Framework**

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet/Intranet. The .NET Framework is designed to fulfill the following objectives:

1. To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
2. To provide a code-execution environment that minimizes software deployment and versioning conflicts.
3. To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.
4. To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
5. To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.
6. To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.
7. . NET Support Remoting application.

The .NET Framework has two main components:

* The common language runtime.
* **.** NET Framework class library.



Overview of ADO.NET

ADO.NET provides consistent access to data sources such as Microsoft SQL Server, as well as data sources exposed via OLE DB and XML. Data-sharing consumer applications can use ADO.NET to connect to these data sources and retrieve, manipulate, and update data.

ADO.NET cleanly factors data access from data manipulation into discrete components that can be used separately or in tandem. ADO.NET includes .NET data providers for connecting to a database, executing commands, and retrieving results. Those results are either processed directly, or placed in an ADO.NET **Dataset** object in order to be exposed to the user in an ad-hoc manner, combined with data from multiple sources, or remote between tiers. The ADO.NET **Dataset** object can also be used independently of a .NET data provider to manage data local to the application or sourced from XML.

Overview of ASP.NET

ASP.NET is a programming framework built on the common language runtime that can be used on a server to build powerful Web applications. ASP.NET offers several important advantages over previous Web development models:

* **Enhanced Performance:** ASP.NET is compiled common language runtime code running on the server. Unlike its interpreted predecessors, ASP.NET can take advantage of early binding, just-in-time compilation, native optimization, and caching services right out of the box. This amounts to dramatically better performance before you ever write a line of code.
* **World-Class Tool Support.** The ASP.NET framework is complemented by a rich toolbox and designer in the Visual Studio integrated development environment. WYSIWYG editing, drag-and-drop server controls, and automatic deployment are just a few of the features this powerful tool provides.
* **Power and Flexibility.** Because ASP.NET is based on the common language runtime, the power and flexibility of that entire platform is available to Web application developers.
* **Simplicity.** ASP.NET makes it easy to perform common tasks, from simple form submission and client authentication to deployment and site configuration. For example, the ASP.NET page framework allows you to build user interfaces that cleanly separate application logic from presentation code and to handle events in a simple, Visual Basic - like forms processing model. Additionally, the common language runtime simplifies development, with managed code services such as automatic reference counting and garbage collection.
* **Manageability.** ASP.NET employs a text-based, hierarchical configuration system, which simplifies applying settings to your server environment and Web applications. An ASP.NET Framework application is deployed to a server simply by copying the necessary files to the server. No server restart is required, even to deploy or replace running compiled code.
* **Scalability and Availability.** ASP.NET has been designed with scalability in mind, with features specifically tailored to improve performance in clustered and multiprocessor environments. Further, processes are closely monitored and managed by the ASP.NET runtime, so that if one misbehaves (leaks, deadlocks), a new process can be created in its place, which helps keep your application constantly available to handle requests.
* **Customizability and Extensibility.** ASP.NET delivers a well-factored architecture that allows developers to "plug-in" their code at the appropriate level. In fact, it is possible to extend or replace any subcomponent of the ASP.NET runtime with your own custom- written component. Implementing custom authentication or state services has never been easier.
* **Security.** With built in Windows authentication and per-application configuration, you can be assured that your applications are secure.

#### Language Support

The Microsoft .NET Platform currently offers built-in support for three languages: C#, Visual Basic.Net, and JScript.

**What is ASP.NET Web Forms?**

The ASP.NET Web Forms page framework is a scalable common language runtime programming model that can be used on the server to dynamically generate Web pages. It provides the ability to create and use reusable UI controls that can encapsulate common functionality and thus reduce the amount of code that a page developer has to write.

* The ability for developers to cleanly structure their page logic in an orderly fashion (not "spaghetti code").
* The ability for development tools to provide strong WYSIWYG design support for pages (existing ASP code is opaque to tools).

**Advantages of ASP.NET web form & Server Controls**

1. ASP.NET Web Forms provide an easy and powerful way to build dynamic Web UI.
2. ASP.NET Web Forms pages can target any browser client (there are no script library or cookie requirements).
3. ASP.NET Web Forms pages provide syntax compatibility with existing ASP pages.
4. ASP.NET server controls provide an easy way to encapsulate common functionality.
5. ASP.NET ships with 45 built-in server controls. Developers can also use controls built by third parties.
6. ASP.NET server controls can automatically project both up level and down-level HTML.
7. ASP.NET templates provide an easy way to customize the look and feel of list server controls.
8. ASP.NET validation controls provide an easy way to do declarative client or server data validation.

**11.2 ABOUT SQL SERVER:**

Microsoft SQL Server is a Structured Query Language (SQL) based, client/server relational database. Each of these terms describes a fundamental part of the architecture of SQL Server.

**DATABASE**

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

A database typically has two components: the files holding the physical database and the database management system (DBMS) software that applications use to access data. The DBMS is responsible for enforcing the database structure, including:

* Maintaining the relationships between data in the database.
* Ensuring that data is stored correctly and that the rules defining data relationships are not violated.
* Recovering all data to a point of known consistency in case of system failures.

**Structured Query Language (SQL)**

To work with data in a database, you must use a set of commands and statements (language) defined by the DBMS software. There are several different languages that can be used with relational databases; the most common is SQL. Both the American National Standards Institute (ANSI) and the International Standards Organization (ISO) have defined standards for SQL. Most modern DBMS products support the Entry Level of SQL-92, the latest SQL standard (published in 1992).

### SQL Server Features

Microsoft SQL Server supports a set of features that result in the following benefits:

### Ease of installation, deployment, and use:

SQL Server includes a set of administrative and development tools that improve your ability to install, deploy, manage, and use SQL Server across several sites.

### Scalability:

The same database engine can be used across platforms ranging from laptop computers running Microsoft Windows® 95/98 to large, multiprocessor servers running Microsoft Windows NT®, Enterprise Edition.

### Data warehousing:

SQL Server includes tools for extracting and analyzing summary data for online analytical processing (OLAP). SQL Server also includes tools for visually designing databases and analyzing data using English-based questions.

### System integration with other server software:

SQL Server integrates with e-mail, the Internet, and Windows.

### Databases:

A database in Microsoft SQL Server consists of a collection of tables that contain data, and other objects, such as views, indexes, stored procedures, and triggers, defined to support activities performed with the data. The data stored in a database is usually related to a particular subject or process, such as inventory information for a manufacturing warehouse.

SQL Server can support many databases, and each database can store either interrelated data or data unrelated to that in the other databases. For example, a server can have one database that stores personnel data and another that stores product-related data. Alternatively, one database can store current customer order data, and another; related database can store historical customer orders that are used for yearly reporting. Before you create a database, it is important to understand the parts of a database and how to design these parts to ensure that the database performs well after it is implemented.

**CHAPTER-12**

**TESTING**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design, and code generation. Once source code has been generated, software must be tested to uncover as many errors as possible before delivery to the customer. A test case is one that has high probability of finding a yet undiscovered error. A successful test is one that uncovers as yet undiscovered error.

**The basic types of testing are:**

* White Box Testing
* Black Box Testing

**White Box Testing**

This is a code testing strategy and it checks for the correctness of the every statement in the program. To follow this testing strategy, there should be cases that result in the execution of every instruction in the program or module. That is every path in the program is tested.

The test cases should make sure that independent paths within a module are executed at least once as and when required.

* Exercise all logical decision on their true or false side.
* Execute all loops at their boundaries and within their operational bounds.

The testing strategy, on the face of it, sounds exhaustive. If every statement in the program is checked for its validity, there doesn’t seem to be much scope of error.

**Black Box Testing**

To perform Black Box Testing, the analyst examines the specification taking the program or module. It checks for the basic functionality and how it should perform on the various conditions when submitted for processing. By examining the result, the analyst can report whether the program performs according to the specified requirements.

**Testing strategy**

Software testing consists of series of tests, which are implemented sequentially. These tests are:

* Test Plan
* Unit testing
* Integration testing
* System testing

**1. Test Plan**

Test plan specifies the objective if testing for completion criteria, system integration plan, and methods to be used on modules and particular test cases to be used. The four types of test that a software product must satisfy are:

* Function Test specifies operating conditions, input values and expected result.
* Performance Test verify response time, execution time and throughout primary and secondary memory utilization links.
* Stress Test is to determine the limitations of the system.
* Structural Test is concerned with examining the internal processing logic of a software system.

**2. Unit Testing**

Instead of testing the system as a whole, unit testing focuses on the modules that make up the system. Each module is taken up individually and testing for the correctness in the coding and logic. Errors resulting from interaction of modules are initially avoided.

**The advantages of unit testing are:**

* Size of a module is quite small and errors can easily be located.
* Confusing interaction of multiple errors in widely different parts of software eliminated.
* Module level testing can be exhaustive.

The obvious assumption made when unit test is pursued is that, individual modules can be isolated from the system module interacts with other modules in the system and, to isolate module, the analyst must stimulate these interactions.

That is the analyst must create driver modules to call the procedures in the module and stub functions for the module to call. The cost involved in creation of this stimulated environment, may or at times is prohibited.

**3. Integration Testing**

It tests for the errors resulting from integration modules. One specific target of integration testing is the interface: whether parameters match on both sides as to type, permissible ranges and meaning. Analyst tries to find areas where modules have been designed with different specifications.

This is the top-level testing. In this all modules tested separately would be put together and tested for producing the ultimate result of the system. The main emphasis during this testing will be on the interface between the modules. By applying various business rules generated as a part of test cases, we were able to ease certain design level complexities.

**4. System Testing**

The main objective of the system testing is to find out the discrepancies between the developed system and its original objective, current specifications and the system documentation. It also verifies for the compatibility of the system with the operational environment. The following system testing tasks are essential.

1. **Peak Load Testing**

* It is done to determine whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand.

1. **Stress Testing**:

* This is performed to determine the capacity of the system stored to transactions. Stress testing executes a system in a manner that demands resources in abnormal quantity, frequency, or volume.

1. **Performs Time Testing**:

* It is done to determine the length of time used by the system to process a transaction or user request.

1. **Recovery Testing**:

* It is done to determine the ability of the software to recover from failure. The recovery may be automatic or may need human intervention. In either case the mean time to repair is evaluated to determine whether it is within acceptable limits.

1. **Procedure Testing**:

* It determines the clarity of the documentation operation and use of system. Asking the user to do exactly what the manual requests to perform.

**CHAPTER-13**

**CONCLUSION**

It can be concluded from the above discussion that a reliable, secure, fast and an efficient system has been developed replacing a manual and unreliable system. This system can be implemented for better results regarding the management of attendance and leaves. This system will save time, reduce the amount of work the administration has to do and will replace the stationery material with electronic apparatus. Hence a system with expected results has been developed but there is still some room for improvement. Under future development of face recognition, it should be capable of detecting any faces under any light conditions.

**CHAPTER-14**

**BIBLIOGRAPHY**

**Reference Books:**

[1] S. Prabhakar, S. Pankanti, and A. K. Jain, “Biometric recognition:

Security and privacy concerns,” *IEEE Security Privacy*, vol. 1, no. 2,

pp. 33–42, Mar./Apr. 2003.

[2] T. Matsumoto, “Artificial irises: Importance of vulnerability analysis,”

in *Proc. AWB*, 2004.

[3] J. Galbally, C. McCool, J. Fierrez, S. Marcel, and J. Ortega-Garcia, “On

the vulnerability of face verification systems to hill-climbing attacks,”

*Pattern Recognit.*, vol. 43, no. 3, pp. 1027–1038, 2010.

[4] A. K. Jain, K. Nandakumar, and A. Nagar, “Biometric template security,”

*EURASIP J. Adv. Signal Process.*, vol. 2008, pp. 113–129, Jan. 2008.

[5] J. Galbally, F. Alonso-Fernandez, J. Fierrez, and J. Ortega-Garcia,

“A high performance fingerprint liveness detection method based on

quality related features,” *Future Generat. Comput. Syst.*, vol. 28, no. 1,

pp. 311–321, 2012.

[6] K. A. Nixon, V. Aimale, and R. K. Rowe, “Spoof detection schemes,”

*Handbook of Biometrics*. New York, NY, USA: Springer-Verlag, 2008,

pp. 403–423.

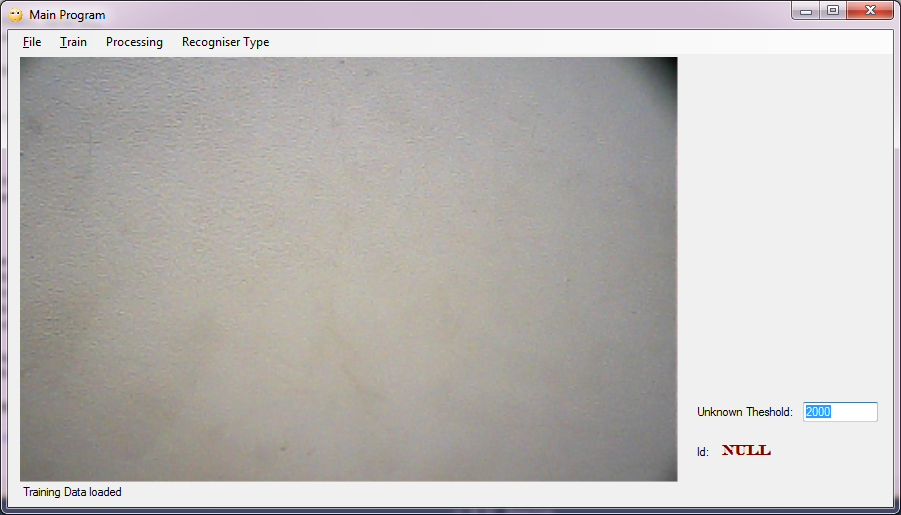
**Websites:**

* <http://www.google.com>
* <http://asp.net-tutorials.com/>

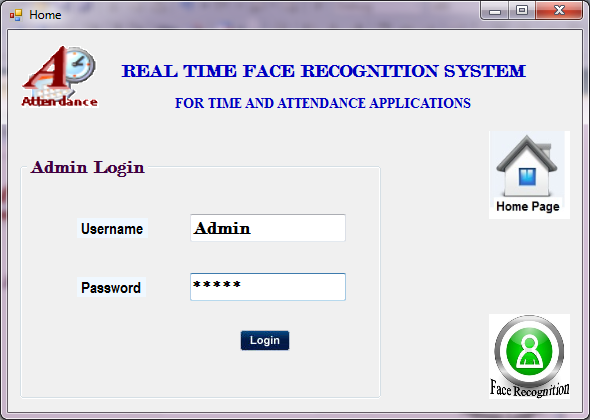
**XV. APPENDIX**

**15.1 SCREEN SHOTS**

**Home Page:**

****

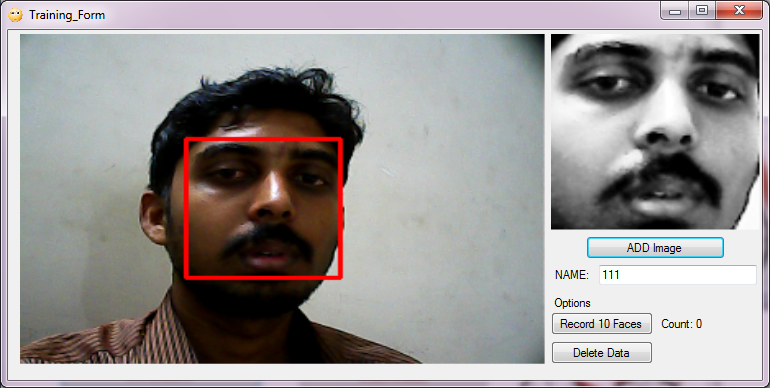
**Admin Login:**

****

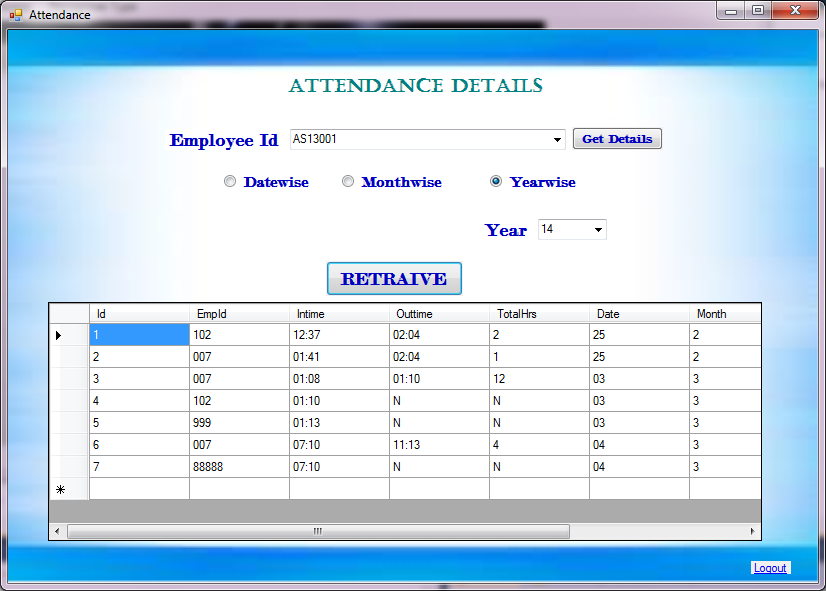
**Registration:**

****

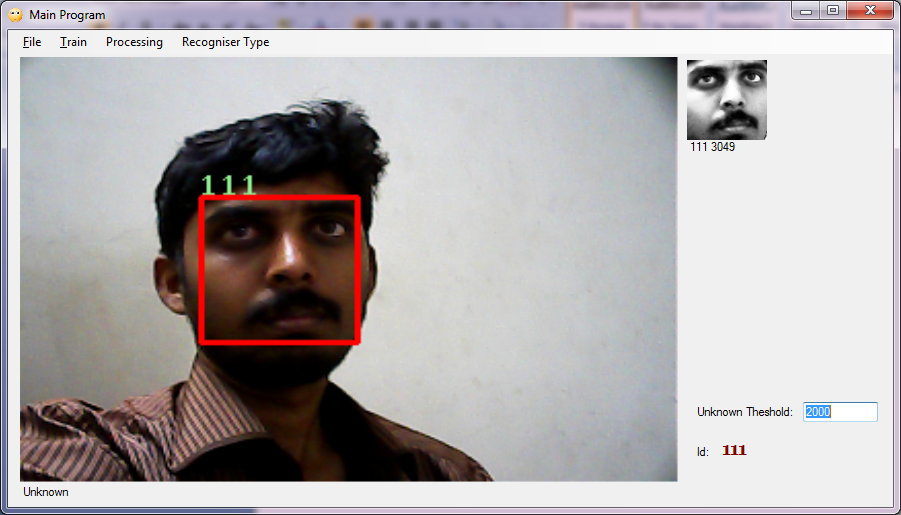
**Face Registration:**

****

**View Attendance Details:**

****

**Face Authentication:**

****