**Attendance System using Face Recognition**

**Abstract**

This research aims at providing a system to automatically record the students’ attendance during lecture hours in a hall or room using facial recognition technology instead of the traditional manual methods.

The objective behind this research is to thoroughly study the field if pattern recognition (facial recognition) which is very important and is used in various applications like identification and detection. **Introduction**

Humans have been using physical characteristics such as face, voice, gait, etc. to recognize each other for thousands of years. With new advances in technology, biometrics has become an emerging technology for recognizing individuals using their biological traits. Now, biometrics is becoming part of day to day life, where in a person is recognized by his/her personal biological characteristics. Examples of different Biometric systems include Fingerprint recognition, Face recognition, Iris recognition, Retina recognition, Hand geometry, recognition, Signature recognition, among others. Face recognition, in particular has received a considerable attention in recent years both from the industry and the research community. The objective of our project is to create a C# code that can be used to identify people using their face images.

Using the face recognition technique we use in the attendance management system. With the help of the face recognition we make the attendance of the student.

**Literature Review**

**Biometrics:**

Biometrics is the automated recognition of individuals based on their behavioural or physiological characteristics .The physiological characteristics are related to the shape of the body. The most common example is fingerprint. Other examples include face recognition, hand geometry and iris recognition. The behavioural characteristics are related to the behaviour of a person. Signature is one example of these characteristics which is still widely used today. ***Modern approaches are the study of keystroke dynamics and voice.***

With the rapid development in the field of pattern recognition and its uses in different areas e.g. (signature recognition, facial recognition), arises the importance of the utilization of this technology in different areas in large organizations. This is mainly because these applications help the top-management take decisions that improve the performance and effectiveness of the organization. On the other hand, for an organization to be effective, it needs accurate and fast means of recording the performance of the people inside this organization. Biometric recognition has the potential to become an irreplaceable part of many identification systems used for evaluating the performance of those people working within the organization. Although biometric technologies are being applied in many fields it has not yet delivered its promise of guaranteeing automatic human recognition. This research is the first of its kind to attempt to provide an automated attendance system that recognizes students using face recognition technology through an image/video stream to record their attendance in lectures or sections and evaluating their performance accordingly.

Problem Definition:-

Every time a lecture, section starts the lecturer or teaching .This is a lengthy process and takes a lot of time and effort, especially if it is a lecture with a huge number of students. It also causes a lot of disturbance and interruption when an exam is held. Moreover the attendance sheet is subjected to damage and loss while being passed on between different students or teaching staff. And when the number of students enrolled in a certain course is huge, the doctors tend to call a couple of student names at random which is not a fair student evaluation process either. Finally, these attendance records are used by the staff to monitor the students’ 3 attendance rates. This process could be easy and effective with a small number of students but on the other hand, dealing with the records of a large number of students often leads to human errors.

Aim

Aim:-

Create the attendance system which uses the face recognition technique which is record the attendance of the student.

Existing System:-

**Following are the Existing system for the attendance system**,

1. **Attendance take by manually**:-
   1. This is old version method for the attendance system where the teacher/staff/representative take the attendance by call the person name and tick in the attendance register. If it’s present then tick as present or absent.
2. **Computerized attendance system**:-
   1. This is same method that proposed in above session. Instead of tick in the attendance register the teacher/staff/representative click in front of the student name in the computer record.

Drawback of the existing system:-

1. Since this process is manually human error can occurs.
2. The existing system is time consuming.
3. Less security.

**Proposed System**

The Proposed system overcomes the problem of the existing system. This project uses the face recognition technique using this student record the attendance.

In the proposed system when student come to the class or lecture system application is start.

It works only is standing in front of the system (Computer application) the application capture the image and send the processing side.

The processing side the application recognize the face of the student.

Finally the application mark as student present. If the face is not recognizing the application make as absent.

Our system has two users,

* Admin
* Students

**Face Recognition Process**

There are four steps in face recognition process:

1. **Acquiring a sample:** In a complete, full implemented biometric system, a sensor takes an observation. The sensor might be a camera and the observation is a snapshot picture. In our system, a sensor will be ignored, and a 2D face picture “observation” will supplied manually.
2. **Extracting Features:** For this step, the relevant data is extracted from the predefined captured sample. This is can be done by the use of software where many algorithms are available. The outcome of this step is a biometric template which is a reduced set of data that represents the unique features of the enrolled user's face.
3. **Comparison Templates:** This depends on the application at hand. For identification purposes, this step will be a comparison between a given picture for the subject and all the biometric templates stored on a database. For verification, the biometric template of the claimed identity will be retrieved (either from a database or a storage medium presented by the subject) and this will be compared to a given picture.
4. **Declaring a Match:** The face recognition system will return a candidate match list of potential matches. In this case, the intervention of a human operator will be required in order to select the best fit from the candidate list. An illustrative analogy is that of a walk-through metal detector, where if a person causes the 7 detector to beep, a human operator steps in and checks the person manually or with a hand-held detector.

**Face Recognition Techniques:**

All available face recognition techniques can be classified into four categories based on the way they represent face;

1. Appearance based which uses holistic texture features.
2. Model based which employ shape and texture of the face, along with 3D depth

information.

1. Template based face recognition.
2. Techniques using Neural Networks.

Classification of face recognition method

**PRINCIPLE COMPONENT ANALYSIS (PCA)**

PCA is a way of identifying patterns in data and expressing the data in such a way to highlight their similarities and differences [6]. The purpose of PCA is to reduce the large dimensionality of the data space (observed variables) to smaller intrinsic dimensionality of feature space (independent variables) which are needed to describe the data economically.

**Mathematical Theory**

To perform a PCA on a set of data, the following steps are needed:

**Step 1:** Get some data.

**Step 2**: Subtract the mean: take the average across each dimension. This produces a data set whose mean is zero.

**Step 3:** calculate the covariance matrix.

The aim of finding the covariance is to see if there is any relationship between the data dimensions. Covariance is always measured between two dimensions matrix. If we have a data set with more than one matrix of dimensions 2, there result are more than one covariance measurements that can be calculated. The most practical way to get all the possible covariance values between the different dimensions is to calculate them all and put them in one big 2D matrix. The definition for the covariance matrix for a set of data with n dimensions is:





**Validating Face Recognition**

In order to verify face recognition are functioning correctly, multiple users who are in the database as well as those who are not will need to test the system. The training library and recognition algorithms will also be modified as needed. One note is that we must make sure that the system rejects the users who are not in the database. When testing the algorithms that we will be using for feature extraction and classification, we will use a confusion matrix to determine the likeliness that a certain input sample is one of the users. In this matrix, the columns will represent what was detected, and the rows will represent the actual user. For example:



In this figure, there were 10 samples from each user who were in the database, as well as 10 from those who are not. User 1 was correctly identified every time, while user 2 was mistakenly identified as user 1 once and as not in the database once. With a good algorithm, our goal is to make this matrix s\*I where s is the number of samples from each user, and I is the n-by-n identity matrix, where n is the number of users.

**Hardware and Software Requirement**

**HARDWARE**

* 1.66 GHz Pentium Processor or Intel compatible processor.
* 2GB RAM.
* 80 GB free hard disk space.
* Web Camera

**SOFTWARE**

1. Visual Studio 2008(.Net framework)
2. .net
3. MS SQL Server 2005

**Testing Technology**

System testing is a critical phase implementation. Testing of the system involves hardware devise and debugging of the computer programs and testing information processing procedures. Testing can be done with text data, which attempts to stimulate all possible conditions that may arise during processing. If structured programming Methodologies have been adopted during coding the testing proceeds from higher level to lower level of program module until the entire program is tested as unit. The testing methods adopted during the testing of the system were unit testing and integrated testing.

**UNIT TESTING:**

Unit testing focuses first on the modules, independently of one another, to locate errors. This enables the tester to detect errors in coding and logical errors that is contained within that module alone. Those resulting from the interaction between modules are initially avoided.

**INTEGRATION TESTING:**

Integration testing is a systematic technique for constructing the program structure while at the same time to uncover the errors associated with interfacing. The objective is to take unit-tested module and build a program structure that has been detected by designing. It also tests to find the discrepancies between the system and its original objectives. Subordinate stubs are replaced one at time actual module. Tests were conducted at each module was integrated. On completion of each set another stub was replaced with the real module.

**FUNCTIONAL TESTING:**

Functional testing is a technique in which all the functionalities of the program are tested to check whether all the functions that where proposed during the planning phase are full filled.

This is also to check that if all the functions proposed are working properly.

This is further done in two phases:

* One before the integration to see if all the unit components work properly
* Second to see if they still work properly after they have been integrated to check if some functional compatibility issues arise.

**PERFORMANCE TESTING:**

**Expected Result**

* The client should be able to connect to the server properly without any problems.
* The connection establishment between the mobile device and the server should take minimal time.
* The mobile device should be able receive data from the server uninterruptedly.
* Information provided by the application should be correct and as per the user’s need.
* **Observation**
* Connection can be established easily provided that the server is on.
* The connection with the server takes time as it uses Internet connection.
* Receiving data from the server takes time.
* Information coming from the database is correct.

**LOAD / STRESS TESTING :**

* **Expected Result**
  + Response time should be unaffected irrespective of the no of users.
  + The introduction of the newer clients should not make the server to work hap hazardously.
  + Continuous use of the server by different clients should not result into the server getting slowed down.
  + Response time should not be degraded if there is congestion in network.
* **Observation**
* The speed of transmission was fine even when the newer clients were getting added. The response of the server was satisfying even with the introduction of newer client.

**IDENTIFICATION OF THE PROJECT FROM THE STUDY**

**FEASIBILITY STUDY :**

The very first phase in any system developing life cycle is preliminary investigation. The feasibility study is a major part of this phase. A measure of how beneficial or practical the development of any information system would be to the organization is the feasibility study.

The feasibility of the development software can be studied in terms of the following aspects:

1.Operational Feasibility.

2.Technical Feasibility.

3.Economical feasibility.

4.Motivational Feasibility.

5.Legal Feasibility

**OPERATIONAL FEASIBILITY :**

The site will reduce the time consumed to maintain manual records and is not tiresome and cumbersome to maintain the records. Hence operational feasibility is assured.

**TECHNICAL FEASIBILITY :**

* At least 166 MHz Pentium Processor or Intel compatible processor.
* At least 16 MB RAM.
* 14.4 kbps or higher modem.
* A video graphics card.
* A mouse or other pointing device.
* At least 3 MB free hard disk space.
* Microsoft Internet Explorer 4.0 or higher.

**ECONOMICAL FEASIBILTY :**

Once the hardware and software requirements get fulfilled, there is no need for the user of our system to spend for any additional overhead.

For the user, the web site will be economically feasible in the following aspects:

* The web site will reduce a lot of paper work. Hence the cost will be reduced.
* Our web site will reduce the time that is wasted in manual processes.
* The storage and handling problems of the registers will be solved.

**LEGAL FEASIBILITY :**

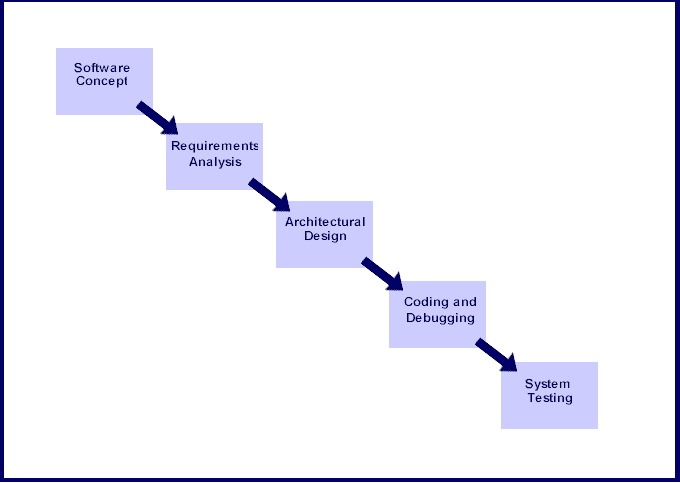
The licensed copy of the required software is quite cheap and easy to get. So from legal point of view the proposed system is legally feasible.

**Methodology**

**Steps Towards Implementation**

**System Development Life Cycle:**

The System Development Life Cycle is the process of developing information systems through investigation, analysis, design, implementation, and maintenance.  The System Development Life Cycle (SDLC) is also known as Information Systems Development or Application Development.



**🡺 Steps involved in the System Development Life Cycle :**

Below are the steps involved in the System Development Life Cycle.  Each phase within the overall cycle may be made up of several steps.

**Step 1:** **Software Concept**

The first step is to identify a need for the new system.  This will include determining whether a business problem or opportunity exists, conducting a feasibility study to determine if the proposed solution is cost effective, and developing a project plan.

This process may involve end users who come up with an idea for improving their work. Ideally, the process occurs in tandem with a review of the organization's strategic plan to ensure that IT is being used to help the organization achieve its strategic objectives.  Management may need to approve concept ideas before any money is budgeted for its development.

**Step 2:** **Requirements Analysis**

Requirements analysis is the process of analyzing the information needs of the end users, the organizational environment, and any system presently being used, developing the functional requirements of a system that can meet the needs of the users.  Also, the requirements should be recorded in a document, email, user interface storyboard, executable prototype, or some other form.  The requirements documentation should be referred to throughout the rest of the system development process to ensure the developing project aligns with user needs and requirements.

Professionals must involve end users in this process to ensure that the new system will function adequately and meets their needs and expectations.

**Step 3:** **Architectural Design**

After the requirements have been determined, the necessary specifications for the hardware, software, people, and data resources, and the information products that will satisfy the functional requirements of the proposed system

can be determined.  The design will serve as a blueprint for the system and helps detect problems before these errors or problems are built into the final system. Professionals create the system design, but must review their work with the users to ensure the design meets users' needs.

**Step 4:** **Coding and Debugging** Coding and debugging is the act of creating the final system.  This step is done by software developer.

**Step 5:** **System Testing** The system must be tested to evaluate its actual functionality in relation to expected or intended functionality.  Some other issues to consider during this stage would be converting old data into the new system and training employees to use the new system.  End users will be key in determining whether the developed system meets the intended requirements, and the extent to which the system is actually used.

**Step 6:** **Maintenance**

Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

There are various software process models like:-

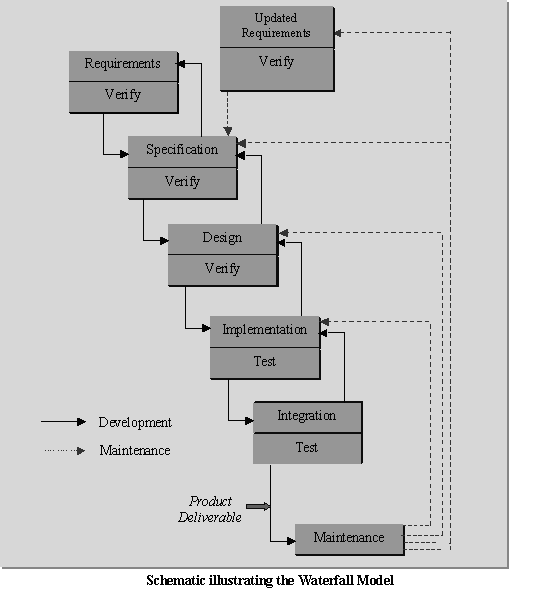
* Prototyping Model
* RAD Model
* The Spiral Model
* The Waterfall Model
* The Iterative Model

Of all these process models we’ve used the Iterative model(The Linear Sequential Model) for the development of our project.

**The Iterative model**

The waterfall model derives its name due to the cascading effect from one phase to the other as is illustrated in Figure1.1. In this model each phase well defined starting and ending point, with identifiable deliveries to the next phase.

This model is sometimes referred to as the linear sequential model or the software life cycle.



The model consists of six distinct stages, namely:

1.      In the **requirements analysis** phase

(a)    The problem is specified along with the desired service objectives (goals)

(b)   The constraints are identified

2.      In the **specification phase** the system specification is produced from the detailed definitions of (a) and (b) above. This document should clearly define the product function.

3.      In the system and software **design phase**, the system specifications are translated into a software representation. The software engineer at this stage is concerned with:

        Data structure

        Software architecture

        Algorithmic detail

        Interface representations

The hardware requirements are also determined at this stage along with a picture of the overall system architecture. By the end of this stage should the software engineer should be able to identify the relationship between the hardware, software and the associated interfaces. Any faults in the specification should ideally not be passed ‘down stream.

4.      In the **implementation and testing**phase stage the designs are translated into the software domain

        Detailed documentation from the design phase can significantly reduce the coding effort.

        Testing at this stage focuses on making sure that any errors are identified and that the software meets its required specification.

5.      In the **integration and system testing** phase all the program units are integrated and tested to ensure that the complete system meets the software requirements. After this stage the software is delivered to the customer [**Deliverable – The software product is delivered to the client for acceptance testing**.]

6.      The **maintenance** phase the usually the longest stage of the software. In this phase the software is updated to:

        Meet the changing customer needs

        Adapted to accommodate changes in the external environment

        Correct errors and oversights previously undetected in the testing phases

        Enhancing the efficiency of the software

Observe that feed back loops allow for corrections to be incorporated into the model. For example a problem/update in the design phase requires a ‘revisit’ to the specifications phase. When changes are made at any phase, the relevant documentation should be updated to reflect that change.

**Advantages of the Iterative Model:-**

        Testing is inherent to every phase of the Iterative model

        It is an enforced disciplined approach

       It is documentation driven, that is, documentation is produced at every stage

**Disadvantages of the Iterative Model:-**

The waterfall model is the oldest and the most widely used paradigm. However, many projects rarely follow its sequential flow. This is due to the inherent problems associated with its rigid format. Namely:

        It only incorporates iteration indirectly, thus changes may cause considerable confusion as the project progresses.

        As The client usually only has a vague idea of exactly what is required from the software product, this IM has difficulty accommodating the natural uncertainty that exists at the beginning of the project.

        The customer only sees a working version of the product after it has been coded. This may result in disaster any undetected problems are precipitated to this stage.

**Microsoft SQL Server 2005**

Business today demands a different kind of data management solution. Performance scalability, and reliability are essential, but businesses now expect more from their key IT investment.

SQL Server 2005 exceeds dependability requirements and provides innovative capabilities that increase employee effectiveness, integrate heterogeneous IT ecosystems,and maximize capital and operating budgets. SQL Server 2005 provides the enterprise data management platform your organization needs to adapt quickly in a fast changing environment.

Benchmarked for scalability, speed, and performance, SQL Server 2005 is a fully enterprise-class database product, providing core support for Extensible Markup Language (XML) and Internet queries.

**Easy-to-use Business Intelligence(BI) Tools**

Through rich data analysis and data mining capabilities that integrate with familiar applications such as Microsoft Office, SQL Server 2005 enables you to provide all of your employees with critical, timely business information

tailored to their specific information needs. Every copy of SQL Server 2005 ships with a suite of BI services.

**Self-Tuning and Management Capabilities**

Revolutionary self-tuning and dynamic self-configuring features optimize database performance, while management tools automate standard activities. Graphical tools and performance, wizards simplify setup, database design, and performance monitoring, allowing database administrators to focus on meeting strategic business needs.

**Data Management Application and Services**

Unlike its competitors, SQL Server 2005 provides a powerful and comprehensive data management platform. Every software license includes extensive management and development tools, a powerful extraction, transformation, and loading (ETL) tool, business intelligence and analysis services such as Notification Service. The result is the best overall business value available.

Enterprise Edition includes the complete set of SQL Server data management and analysis features are and is uniquely characterized by several features that makes it the most scalable and available edition of SQL Server 2005 .It scales to the performance levels required to support the largest Web sites, Enterprise Online Transaction Processing (OLTP) system and Data Warehousing systems. Its support for failover clustering also makes it ideal for any mission critical line-of-business application.

**Estimation & Planning**

**IMPLEMENTATION PLAN**

The following table gives the project plan for the Phase 1 & 2 of our project:

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Description** | **Effort in person weeks** | **Deliverable** |
| Phase 1 |  |  |  |
| P1-01 | Requirement Analysis | 2 weeks | Requirement Gathering |
| P1-02 | Existing System Study & Literature | 3 weeks | Existing System Study & Literature |
| P1-03 | Technology Selection | 2 weeks | MATLAB |
| P1-04 | Modular Specifications | 2 weeks | Module Description |
| P1-05 | Design & Modelling | 4 weeks | Analysis Report |
|  | **Total** | **13 weeks** |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Description** | **Effort in person weeks** | **Deliverable** |
| Phase 2 |  |  |  |
| P2-01 | Detailed Design | 2 weeks | LLD / DLD Document |
| P2-02 | UI and user interactions design | Included in above | UI document |
| P2-03 | Coding & Implementation | 12 weeks | Code Release |
| P2-04 | Testing & Bug fixing | 2 weeks | Test Report |
| P2-05 | Performance Evaluation | 4 weeks | Analysis Report |
| P2-06 | Release | Included in above | System Release |
|  | **Total** | **20 weeks** | Deployment efforts are extra |