**SMART ATTENDANCE MONITORING SYSTEM USING FACIAL RECOGNIZATION.**

### A PROJECT REPORT

#### Submitted by

NAVYA ARORA (19CBS1027), SPARSH PHUTELA (19CBS1016), MAHIN MODI (19CBS1076).

#### in partial fulfillment for the award of the degree of

**BACHELOR OF ENGINEERING**

###### IN

##### COMPUTER SCIENCE AND ENGINEERING WITH SPECIALIZATION BUISNESS SYSTEMS.



### Chandigarh University

##### MAY 2023

***Annexure-2***

## DECLARATION

We are the students of **‘Bachelor of Engineering in Computer Science and Business System**, **session: 2023**, Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Punjab, hereby declare that the work presented in this Project Work entitled ‘**SMART ATTENDANCE MONITORING SYSTEM USING FACIAL RECOGNIZATION’.** Itis the outcome of our own bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

This the method corresponds to the horizontal histograms, the adjacent element adjectives, and shapes, but differs in that it is calculated in a dense grid of evenly spaced cells and uses normal spatial brightness for improved accuracy [2]. First the face image is divided into smaller connected regions, called cells, and the horizontal histogram is calculated per computer over the pixels of a cell using an abstract mask such as Sobel masks.



**BONAFIDE CERTIFICATE**

Certified that this project report **“SMART ATTENDANCE MONITORING SYSTEM USING FACIAL RECOGNIZATION.”** is the bonafide work of “**NAVYA ARORA, SPARSH PHUTELA, MAHIN MODI”** who carried out the project work under my/our supervision.

<<Signature of the HoD>>

###### SIGNATURE

<<Name of the Head of the Department>>

###### HEAD OF THE DEPARTMENT

<<Department>>

<<Signature of the Supervisor>>

###### SIGNATURE

<<Name>>

###### SUPERVISOR

<<Academic Designation>>

<<Department>>

Submitted for the project viva-voce examination held on

INTERNAL EXAMINER EXTERNAL EXAMINER

*Annexure-3*

ABSTRACT

A systematic plan will enable institutions to have a growing growth. It helps students and teachers in every way to make good progress in fellowship, thereby reducing teacher time and effort. In real classrooms, the usual way of naming students and marking their presence / absence is a common practice. A presence monitoring system is proposed using prudence.

First, a website was created that contained facial images of students in a particular classroom. The information gained through the Convolutional Neural Network (CNN) is also fully utilized using transfer learning. The program is designed to improve student participation time within the classroom, regular communication with parents, avoid the presence of a representative and create detailed reports for future use.

The personal record system does not work and extra time is needed to record and calculate each student's attendance. So a plan is needed that will solve the problem of personal presence. While advancing in the digital age is accelerating every hour, biometric technology has begun to impact daily human life. The following system is based on face recognition to maintain the attendance record of students. The daily attendance of students is recorded subject wise which is stored..

## Table of Contents

|  |
| --- |
|  |
| **1.** | **INTRODUCTION\*** |
| 1. **LITERATURE SURVEY** 2. **PROBLEM FORMULATION** 3. **THEORY** 4. **OBJECTIVES** 5. **METHODOLOGY** 6. **CONCLUSIONS AND DISCUSSION** 7. **REFERENCES** | |

1. **INTRODUCTION**

Presence is defined as an act or condition of regular attendance or presence at a place or event. The presence of every student is maintained by schools and colleges. To maintain the attendance record with day to day activities is a challenging task. The conventional method of calling name of each student is time consuming and there is always a chance of proxy attendance. The personal record system does not work and extra time is needed to record and calculate each student's attendance. So a plan is needed that will solve the problem of personal presence. While advancing in the digital age is accelerating every hour, biometric technology has begun to impact daily human life. The following system is based on face recognition to maintain the attendance record of students. The daily attendance of students is recorded subject wise which is stored already by the administrator. As the time for corresponding subject arrives the system automatically starts taking snaps and then apply face detection and recognition technique to the given image and the recognize students are marked as present and their attendance update with corresponding time and subject id. We have used deep learning techniques to develop this system, histogram of oriented gradient method is used to detect faces in images and deep learning method is used to compute and compare feature facial of students to recognize them. Our system is capable to identify multiple faces in real time. Biometrics technology uses features such as fingerprints, facial irises, retina patterns, palm printers, voice, handwritten signatures, and more for verification. These methods of using mobile data are increasingly being seen as an effective alternative to common security measures such as passwords or identity cards. Biometric personal verification uses data taken from measurements. Such data is unique to a person and remains the same throughout a person's life. It is important to identify the right tools to use in commercial and scientific studies. Barcode students, Radio

# HARDWARE SPECIFICATIONS:

* Intel Core i3 3rd gen processor or latest.
* 512 MB disk space.
* 512 MB RAM.
* Any external or in-build camera with minimum pixel resolution 200 x 200 (300ppi or 150lpi) 4-megapixel cameras and up.

# SOFTWARE SPECIFICATIONS:

* Microsoft Windows XP or later / Ubuntu 12.0 LTS or later /MAC OS 10.1 or later.
* Python Interpreter (3.6).
* TensorFlow framework, Keras API.
* Python OpenCV2, scipy, qimage2ndarray, winGuiAuto, pypiwin32, sys, keyboard, pyttsx3, pillow library.

**OVERVIEW**

* + 1. ***Existing System***

The existing system mainly contains a visual register in which the administrator records the entry record of all students. Other technologies developed to replace this manual system include fingerprints, retina scan, voice recognition etc. The problem with the existing system is that the manual system is time consuming and advanced technology is too expensive to be used on a large scale in any way. organization.

* + 1. **Proposed System**

The following system eliminates the annoying task of keeping existing records by default. The supervisor completes the details of the teachers and students at the beginning of the semester. Class list is automatically generated. This information may be used for other seminars with minor modifications.



The system allows easy management using Face Detection, which is one of the most popular methods. The teacher needs to carry Digital Photography Devices to the classroom and take a picture of the class. The teacher then needs to log into the computer using his login details. A webpage provides a way to upload an image to a computer. The program has a copy of MATLAB installed on it. The system also contains a database that includes photos of all students and their personal details. The website also maintains accurate records of class teachers. After the image is uploaded to the program, the faces of the students are captured on the image using the MATLAB software. These images are then compared to student photos stored on the website using a face recognition algorithm and the attendance record is maintained. The computer then returns a copy of the html page to the teacher and to each student. But, what if only half of a student's face is visible or his or her face is blurred in the first image. The teacher can personally mark the presence of those students. This is

Smart attendance system.

Near area conversation is a type of short distance wi-fi conversation that takes region among gadgets, one energetic and the alternative passive. The gadgets are essentially inductor coils which can respond to an electromagnetic induction. The energetic device is utilized to produce an electromagnetic area of a given radius and strength. Which used to put into effect an attendance device. In a faculty placing for example, students can be given NFC tags that are uniquely programmed with their unique identity numbers. Upon attending the classes, the academics deliver the NFC readers and a scholar is needed to swipe their NFC tags near the reader, say the lecturers’ phone. This information is then transmitted to the faculty database to mark the attendance of the scholar.

In this system we have implemented an attendance system for a lecture, section or

laboratory by which lecturer or teaching assistant can record students’ attendance

. It saves time and effort, especially if it is a lecture with huge number of students.

Automated Attendance System has been envisioned for the purpose of reducing the

drawbacks in the traditional (manual) system.

This attendance system demonstrates the use of image processing techniques in classroom.

## Face Recognition

**Histogram of Oriented Gradient (HOG)**

How to calculate the appearance of gradient shapes in local parts of an image. This the method corresponds to the horizontal histograms, the adjacent element adjectives, and shapes, but differs in that it is calculated in a dense grid of evenly spaced cells and uses normal spatial brightness for improved accuracy [2]. First the face image is divided into smaller connected regions, called cells, and the horizontal histogram is calculated per computer over the pixels of a cell using an abstract mask such as Sobel masks. Each pixel in the cell will be the edge parameter and part of the gradient is attached to it, thus calculating for the barrels. Histogram channels are evenly distributed over 0–180 degrees or 0–360 degrees, depending on the ‘unsigned’ or ‘signed’ gradient. Histogram figures often compensate for the light. This process is performed by accumulating the amount of local histogram energy over larger connected regions and subsequently using the results to make all block cells normal. The combination of these histograms represents the final HOG definition.

Flexibility in scale and rotation can also be achieved by removing adjectives from only the key points (key points) in the image scale space following the normal rotation. The steps involved are:

* Extreme Scale-space discovery.
* Assignment assignment.
* Definition of the interpreter.An important idea behind the Histogram definitions of Oriented Gradient definitions is that the appearance of the object and the position within the image can be explained by the distribution of dynamic gradients or edge directions. The application of these definitions can be achieved by dividing the image into smaller connected circuits, called cells, with each cell comprising a histogram of the directional or orientation of the pixels within the cell. The combination of theseistograms then represents the adjective. For improved accuracy, local histograms can be simulated-standard by calculating the degree of stiffness in a large image area, called a block, and then using this number to make all the normal cells within the block. This tendency leads to better flexibility in light change or shade.

The HOG descriptor maintains a few key advantages over other descriptor methods. Since the HOG descriptor operates on localized cells, the method upholds invariance to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions. Moreover, as per the discovery of Dalal and Triggs, coarse spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of pedestrians to be ignored so long as they maintain a roughly upright position.

The final step in object recognition using Histogram of Oriented Gradient descriptors is to feed the descriptors into some recognition system based on supervised learning. The Support Vector Machine classifier is a binary classifier which looks for an optimal hyper plane as a decision function. Once trained on images containing some particular object, the SVM classifier can make decisions regarding the presence of an object, such as a human being, in additional test images

Error-correcting Output Coding (ECOC)

Error-Correct Extraction Code (ECOC) [1] is a compilation method designed for the problem of classification. In the case of multiple class divisions, the task is a single label decision from the 2 possible options. For example, in the digital detection function, we need to place a written digital map of each hand in one of the classes k = 10. ECOC is a meta method that combines multiple binary categories to solve a multi-stage problem. Uses SVM (Support Vector Machine) binary classifier.

In machine learning, vector support devices

(SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data and detect patterns, which are used for editing and retransmission analysis. Given a set of training examples, each marked as part of two phases, the SVM training algorithm creates a model that provides new examples in one phase or another, making it a viable binary line divider.

1. **LITERATURE REVIEW**

Plenty of studies has been performed to this point at the numerous to be had techniques for implementation of an powerful attendance tracking device. These techniques range in phrases of the types of input method used, the types of information processing hired and the controllers used to put into effect the structures. In this segment seeking out the numerous to be had solution with the advantages and disadvantages of every device. First device, “Attendance System Using NFC Technology with Embedded Camera on Mobile Device” (Bhise, Khichi, Korde,Lokare, 2015). Near area conversation is a type of short distance wi-fi conversation that takes region among gadgets, one energetic and the alternative passive. The gadgets are essentially inductor coils which can respond to an electromagnetic induction. The energetic device is utilized to produce an electromagnetic area of a given radius and strength. Which used to put into effect an attendance device. In a faculty placing for example, students can be given NFC tags that are uniquely programmed with their unique identity numbers. Upon attending the classes, the academics deliver the NFC readers and a scholar is needed to swipe their NFC tags near the reader, say the lecturers’ phone. This information is then transmitted to the faculty database to mark the attendance of the scholar. However this device is liable to impersonation in which one character can check in for a person else. The different associated structures that use biometrics (Fingerprint recognition RFID, etc) to identify end user are time control structures used in lots of colleges, establishments and schools. However, those device introduce further privateness concerns. These structures also are problem to bodily harm from their users. Therefore, they need extra upkeep costs. The concept proposed through us, eliminates bodily access from anyone to the computerized device.

**Literature Review Summary**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Year and citation** | **Article Title** | **Purpose of study** | **Tools / Software Used** | **Comparison of technique** | **Source (Journal**  **/Conference)** | **Findings** | **Data set (if used)** | **Evaluation parameters** |
| 201  9 | 2278-01  81 | To Built  automated  attendance | Python VS  Code | Face  Recognition Attendance | IJERT | - | - |  |
| system | system |

## PROBLEM FORMULATION

Attendance is an important part of daily classroom assessment. At the beginning and end of a class, it is usually viewed by the teacher, but it may seem that the teacher misses out on a comment by one or more students. The face recognition system is a face-to-face movement problem using face recognition technology based on high definition video surveillance and other information technology.

The concept of face recognition is to give the computer system the ability to detect and see a person's face quickly and accurately in photos or videos. Many algorithms and techniques have been developed to improve the performance of facial recognition. Recently, in-depth reading has been extensively tested in computer vision systems. The human brain can detect and detect multiple faces automatically and instantly. But when it comes to computers, it is extremely

In biometrics, human features are compared to existing data. Face features are extracted and applied using algorithms, such as successfully and some changes were made to improve existing algorithm models. Detectors and face detectors can be used in many different programs including crime detection, security systems, identity verification etc. Face recognition the process usually consists of two phases:

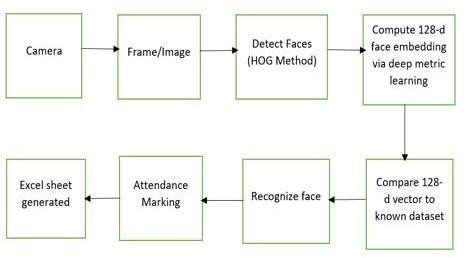
* *Face Recognition* - where the inserted image is searched to find any face, and then an image processing cleans the face image so that it can be easily seen.
* *Face detection* - where the detected and processed face is compared to a known face website to determine who the person is.

## RESEARCH OBJECTIVES

The main objective of the project is to establish a better future for the coming generation. For our teachers & for our students, we want better future & our project is one of the best medium for it.

Our program aims to create an effective classroom approach using face recognition techniques and our vision. The proposed program will be able to mark attendees at the event with a face id. It will see faces with a webcam and see faces. Upon recognition, it will mark the presence of a known student and update the attendance record.

## BLOCK DIAGRAM of FACE RECOGONITION PROCESS



1. **THEORY ABOUT THE WHOLE PROJECT**

In recent years, a number of face recognition based attendance management system have introduced in order to improve the performance of students in different organization. In [4] Jomon Joseph, K. P. Zacharia proposed a system using image processing, PCA, Eigen faces, Microcontroller, based on Matlab. Their system works only with front face images and there is need of a suitable method which works with the orientation of the system. Ajinkya Patil with their fellows in [5] proposed a face recognition approach for attendance marking using Viola jones algorithm, Haar cascades are used to detect faces in images and recognition performs through Eigen face method. Another approach of making attendance system easy and secure, in [6] the author proposed a system with the help of artificial neural networks, they used PCA to extract face images and testing and training were achieved by neural networks, their system performs in various orientation. A 3D face recognition approach for attendance management system was proposed by MuthuKalyani.K, VeeraMuthu.A [7] has proposed, they marked attendance with monthly progress of each student. There is need for an alternative algorithm which can enhance the recognition on oriented faces. Efficient Attendance Management system is designed with the help of PCA algorithm [8], the have achieved accuracy up to 83% but their system performance decreases due to slightly changes in light condition. An eigen face approach along with PCA algorithm for marking face recognition attendance system have introduced by author in [9], they mention comparison of different face recognition algorithm in their paper. Overall it was good approach to maintain record of attendance.

## LIBRARIES USED IN PROJECT:

* 1. **TKINTER:**

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.

Running python -m tkinter from the command line should open a window demonstrating a simple Tk interface, letting you know that tkinter is properly installed on your system, and also showing what version of Tcl/Tk is installed, so you can read the Tcl/Tk documentation specific to that version.

Tkinter supports a range of Tcl/Tk versions, built either with or without thread support. The official Python binary release bundles Tcl/Tk 8.6 threaded. See the source code for the \_tkinter module for more information about supported versions.

Tkinter is not a thin wrapper, but adds a fair amount of its own logic to make the experience more pythonic. This documentation will concentrate on these additions and changes, and refer to the official Tcl/Tk documentation for details that are unchanged.

Tkinter provides various controls, such as buttons, labels and text boxes used in a GUI application. These controls are commonly called widgets. They are:

1. Button
2. Canvas
3. Checkbutton
4. Entry
5. Frame
6. Label
7. Listbox
8. Menubutton
9. Menu
10. Message
11. RadioButton
12. Scale
13. Scrollbar
14. Text
15. TopLevel
16. SpinBox
17. PanedWindow
18. LabelFrame
19. tkMessageBox

All Tkinter widgets have access to specific geometry management methods, which have the purpose of organizing widgets throughout the parent widget area. Tkinter exposes the following geometry manager classes: pack, grid, and place.

The ***pack()*** Method − This geometry manager organizes widgets in blocks before placing them in the parent widget.

The ***grid()*** Method − This geometry manager organizes widgets in a table- like structure in the parent widget.

The ***place()*** Method − This geometry manager organizes widgets by placing them in a specific position in the parent widget.

Tkinter allows you to develop desktop applications. It’s a very good tool for GUI programming in Python.

Tkinter is a good choice because of the following reasons:

* 1. Easy to learn.
  2. Use very little code to make a functional desktop application.
  3. Layered design.
  4. Portable across all operating systems including Windows, macOS, and Linux.

## OpenCV/cv2.imread() method:-

OpenCV-Python is a library of Python bindings designed to solve computer vision problems.

cv2.imread() method loads an image from the specified file. If the image cannot be read (because of missing file, improper permissions, unsupported or invalid format) then this method returns an empty matrix.

OpenCV-Python makes use of Numpy, which is a highly optimized library for numerical operations with a MATLAB-style syntax. All the OpenCV array structures are converted to and from Numpy arrays. This also makes it easier to integrate with other libraries that use Numpy such as SciPy and Matplotlib.

OpenCV is an open source initiative, all are welcome to make contributions to the library, documentation, and tutorials. If you find any mistake in this tutorial (from a small spelling mistake to an egregious error in code or concept), feel free to correct it by cloning OpenCV in GitHub and submitting a pull request. OpenCV developers will check your pull request, give you important feedback and (once it passes the approval of the reviewer) it will be merged into OpenCV.

OpenCV, which is an image and video processing library with bindings in C

++, C, Python, and Java. OpenCV is used for all sorts of image and video analysis, like facial recognition and detection, license plate reading, photo editing, advanced robotic vision, optical character recognition, and a whole lot more.

OpenCV allows performing multiple inputs and outputs simultaneously through multiple windows. The waitKey method is used to inform OpenCV the time duration over which a window can be kept open. This method takes time as an argument in milliseconds. If 0 is provided as the argument, the user should close the window manually. It can be noted that the window has a name of its own. Therefore, when there comes a window into play, it must be named. Here, the window name is ‘Image Window’. destroyAllWindows method is used to force all the open windows to close at once.

The basic 10 OPENCV Methods are:

* 1. How to Read an Image
  2. How to write an Image
  3. Getting image Information
  4. IMage Rotation
  5. Image Transpose
  6. CapturingVideo Using OPENCV
  7. How to Crop Images
  8. How to Blur Images
  9. Image Pyramid
  10. Edge detection in OPENCV

## 3. CSV

The so-called CSV (Comma Separated Values) format is the most common import and export format for spreadsheets and databases. CSV format was used for many years prior to attempts to describe the format in a standardized way in RFC 4180. The lack of a well-defined standard means that subtle differences often exist in the data produced and consumed by different applications. These differences can make it annoying to process CSV files from multiple sources. Still, while the delimiters and quoting characters vary, the overall format is similar enough that it is possible to write a single module which can efficiently manipulate such data, hiding the details of reading and writing the data from the programmer.

The csv module implements classes to read and write tabular data in CSV format. It allows programmers to say, “write this data in the format preferred by Excel,” or “read data from this file which was generated by Excel,” without knowing the precise details of the CSV format used by Excel. Programmers can also describe the CSV formats understood by other applications or define their own special-purpose CSV formats.

CSV files are normally created by programs that handle large amounts of data. They are a convenient way to export data from spreadsheets and databases as well as import or use it in other programs. For example, you might export the results of a data mining program to a CSV file and then import that into a spreadsheet to analyze the data, generate graphs for a presentation, or prepare a report for publication.

CSV files are very easy to work with programmatically. Any language that supports text file input and string manipulation (like Python) can work with CSV files directly.

The csv library provides functionality to both read from and write to CSV files. Designed to work out of the box with Excel-generated CSV files, it is easily adapted to work with a variety of CSV formats. The csv library contains objects and other code to read, write, and process data from and to CSV files.

Reading from a CSV file is done using the reader object. The CSV file is opened as a text file with Python’s built-in open() function, which returns a file object. This is then passed to the reader, which does the heavy lifting.

The reader object can handle different styles of CSV files by specifying additional parameters, some of which are shown below:

***delimiter*** specifies the character used to separate each field. The default is the comma (',').

***quotechar*** specifies the character used to surround fields that contain the delimiter character. The default is a double quote (' " ').

***escapechar*** specifies the character used to escape the delimiter character, in case quotes aren’t used. The default is no escape character.

The quotechar optional parameter tells the writer which character to use to quote fields when writing. Whether quoting is used or not, however, is determined by the quoting optional parameter:

1. If quoting is set to csv.QUOTE\_MINIMAL, then .writerow() will quote fields only if they contain the delimiter or the quotechar. This is the default case.
2. If quoting is set to csv.QUOTE\_ALL, then .writerow() will quote all fields.
3. If quoting is set to csv.QUOTE\_NONNUMERIC, then .writerow() will quote all fields containing text data and convert all numeric fields to the float data type.
4. If quoting is set to csv.QUOTE\_NONE, then .writerow() will escape delimiters instead of quoting them. In this case, you also must provide a value for the escapechar optional parameter.

Reading CSV files is possible in pandas as well. It is highly recommended if we have a lot of data to analyze.

Pandas is an open-source Python library that provides high performance data analysis tools and easy to use data structures. Pandas is available for all Python installations, but it is a key part of the Anaconda distribution and works extremely well in Jupyter notebooks to share data, code, analysis results, visualizations, and narrative text.

1. Pandas recognized that the first line of the CSV contained column names, and used them automatically.
2. Pandas is also using zero-based integer indices in the DataFrame. That’s because we didn’t tell it what our index should be.
3. If you look at the data types of our columns , you’ll see pandas has properly converted the Salary and Sick Days remaining columns to numbers, but the Hire Date column is still a String.

## OS

This module provides a portable way of using operating system dependent functionality. If you just want to read or write a file see open(), if you want to manipulate paths, see the os.path module, and if you want to read all the lines in all the files on the command line see the fileinput module. For creating temporary files and directories see the tempfile module, and for high-level file and directory handling see the shutil module.

The design of all built-in operating system dependent modules of Python is such that as long as the same functionality is available, it uses the same interface; for example, the function os.stat(path) returns stat information about path in the same format (which happens to have originated with the POSIX interface).

Extensions peculiar to a particular operating system are also available through the os module, but using them is of course a threat to portability.

All functions accepting path or file names accept both bytes and string objects, and result in an object of the same type, if a path or file name is returned.

On VxWorks, os.popen, os.fork, os.execv and os.spawn\*p\* are not supported.

If the PYTHONUTF8 environment variable is not set at all, then the interpreter defaults to using the current locale settings, unless the current locale is identified as a legacy ASCII-based locale (as described for PYTHONCOERCECLOCALE), and locale coercion is either disabled or fails. In such legacy locales, the interpreter will default to enabling UTF-8 mode unless explicitly instructed not to do so.

The filesystem encoding and error handler are configured at Python startup by the PyConfig\_Read() function: see filesystem\_encoding and filesystem\_errors members of PyConfig.

Changed in version 3.1: On some systems, conversion using the file system encoding may fail. In this case, Python uses the surrogateescape encoding error handler, which means that undecodable bytes are replaced by a Unicode character U+DCxx on decoding, and these are again translated to the original byte on encoding.

The file system encoding must guarantee to successfully decode all bytes below 128. If the file system encoding fails to provide this guarantee, API functions can raise UnicodeError.

This mapping may be used to modify the environment as well as query the environment. putenv() will be called automatically when the mapping is modified.

On Unix, keys and values use sys.getfilesystemencoding() and 'surrogateescape' error handler. Use environb if you would like to use a different encoding.

These functions and data items provide information and operate on the current process and user.

os.ctermid()

Return the filename corresponding to the controlling terminal of the process.

Availability: Unix.

os.environ

A mapping object where keys and values are strings that represent the process environment. For example, environ['HOME'] is the pathname of your home directory (on some platforms), and is equivalent to getenv("HOME") in C.

This mapping is captured the first time the os module is imported, typically during Python startup as part of processing site.py. Changes to the environment made after this time are not reflected in os.environ, except for changes made by modifying os.environ directly.

environb

Bytes version of environ: a mapping object where both keys and values are bytes objects representing the process environment. environ and environb are synchronized (modifying environb updates environ, and vice versa).

environb is only available if supports\_bytes\_environ is True.

New in version 3.2.

Changed in version 3.9: Updated to support PEP 584’s merge (|) and update (|=) operators.

os.chdir(path) os.fchdir(fd) os.getcwd()

These functions are described in Files and Directories.

os.fsencode(filename)

Encode path-like filename to the filesystem encoding and error handler; return bytes unchanged.

os.fspath(path)

Return the file system representation of the path.

If str or bytes is passed in, it is returned unchanged. Otherwise fspath () is called and its value is returned as long as it is a str or bytes object. In all other cases, TypeError is raised.

New in version 3.6.

class os.PathLike

An abstract base class for objects representing a file system path, e.g. pathlib.PurePath.

New in version 3.6.

abstractmethod fspath ()

Return the file system path representation of the object.

The method should only return a str or bytes object, with the preference being for str.

os.getenv(key, default=None)

Return the value of the environment variable key if it exists, or default if it doesn’t. key, default and the result are str. Note that since getenv() uses os.environ, the mapping of getenv() is similarly also captured on import, and the function may not reflect future environment changes.

On Unix, keys and values are decoded with sys.getfilesystemencoding() and 'surrogateescape' error handler. Use os.getenvb() if you would like to use a different encoding.

Availability: most flavors of Unix, Windows.

os.getenvb(key, default=None)

Return the value of the environment variable key if it exists, or default if it doesn’t. key, default and the result are bytes. Note that since getenvb() uses os.environb, the mapping of getenvb() is similarly also captured on import, and the function may not reflect future environment changes.

getenvb() is only available if supports\_bytes\_environ is True.

Availability: most flavors of Unix.

New in version 3.2.

os.get\_exec\_path(env=None)

Returns the list of directories that will be searched for a named executable, similar to a shell, when launching a process. env, when specified, should be an environment variable dictionary to lookup the PATH in. By default, when env is None, environ is used.

New in version 3.2.

os.getegid()

Return the effective group id of the current process. This corresponds to the “set id” bit on the file being executed in the current process.

Availability: Unix.

os.geteuid()

Return the current process’s effective user id.

Availability: Unix.

os.getgid()

Return the real group id of the current process.

Availability: Unix.

os.getgrouplist(user, group)

Return list of group ids that user belongs to. If group is not in the list, it is included; typically, group is specified as the group ID field from the password record for user.

Availability: Unix.

New in version 3.3.

os.getgroups()¶

Return list of supplemental group ids associated with the current process.

Availability: Unix.

os.getlogin()

Return the name of the user logged in on the controlling terminal of the process. For most purposes, it is more useful to use getpass.getuser() since the latter checks the environment variables LOGNAME or USERNAME to find out who the user is, and falls back to pwd.getpwuid(os.getuid())[0] to get the login name of the current real user id.

Availability: Unix, Windows.

os.getpgid(pid)

Return the process group id of the process with process id pid. If pid is 0, the process group id of the current process is returned.

Availability: Unix.

os.getpgrp()

Return the id of the current process group.

Availability: Unix.

os.getpid()

Return the current process id.

os.getppid()

Return the parent’s process id. When the parent process has exited, on Unix the id returned is the one of the init process (1), on Windows it is still the same id, which may be already reused by another process.

Availability: Unix, Windows.

Changed in version 3.2: Added support for Windows.

os.getpriority(which, who)

Get program scheduling priority. The value which is one of PRIO\_PROCESS, PRIO\_PGRP, or PRIO\_USER, and who is interpreted relative to which (a process identifier for PRIO\_PROCESS, process group identifier for PRIO\_PGRP, and a user ID for PRIO\_USER). A zero value for who denotes (respectively) the calling process, the process group of the calling process, or the real user ID of the calling process.

Availability: Unix.

New in version 3.3.

os.PRIO\_PROCESS os.PRIO\_PGRP os.PRIO\_USER

Parameters for the getpriority() and setpriority() functions.

Availability: Unix.

New in version 3.3.

os.getresuid()

Return a tuple (ruid, euid, suid) denoting the current process’s real, effective, and saved user ids.

Availability: Unix.

New in version 3.2.

os.getresgid()

Return a tuple (rgid, egid, sgid) denoting the current process’s real, effective, and saved group ids.

Availability: Unix.

os.getuid()

Return the current process’s real user id.

Availability: Unix.

os.initgroups(username, gid)

Call the system initgroups() to initialize the group access list with all of the groups of which the specified username is a member, plus the specified group id.

Availability: Unix.

New in version 3.2.

os.putenv(key, value)

Set the environment variable named key to the string value. Such changes to the environment affect subprocesses started with os.system(), popen() or fork() and execv().

Assignments to items in os.environ are automatically translated into corresponding calls to putenv(); however, calls to putenv() don’t update os.environ, so it is actually preferable to assign to items of os.environ. This also applies to getenv() and getenvb(), which respectively use os.environ and os.environb in their implementations.

## NUMPY

NumPy is an open-source Python library that facilitates efficient numerical operations on large quantities of data. There are a few functions that exist in NumPy that we use on pandas DataFrames. Eventually, you will be familiar For us, the most important part about NumPy is that pandas is built on top of it. So, NumPy is a dependency of Pandas.

NumPy arrays are unique in that they are more flexible than normal Python lists. They are called ndarrays since they can have any number (n) of dimensions (d). They hold a collection of items of any one data type and can be either a vector (one-dimensional) or a matrix (multi- dimensional). NumPy arrays allow for fast element access and efficient data manipulation.

Many operations can be performed on NumPy arrays which makes them very helpful for manipulating data:

Selecting array elements

Slicing arrays

Reshaping arrays

Splitting arrays

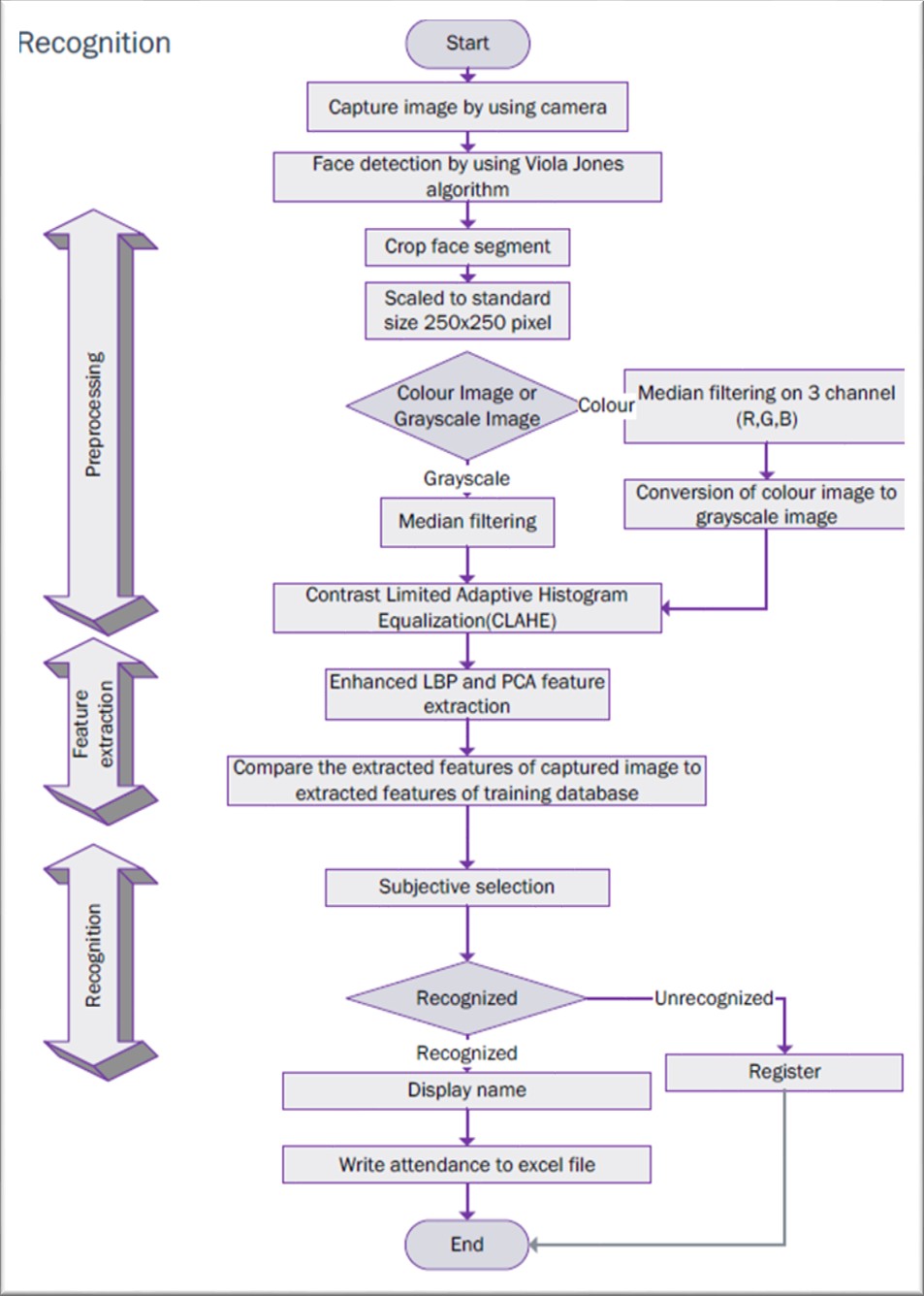
Combining arrays

Numerical operations (min, max, mean, etc)

Mathematical operations can be performed on all values in a ndarray at one time rather than having to loop through values, as is necessary with a Python list. This is very helpful in many scenarios. Say you own a toy store and decide to decrease the price of all toys by €2 for a weekend sale. With the toy prices stored in an ndarray, you can easily facilitate this operation.

Just as the ndarray is the foundation of the NumPy library, the Series is the core object of the pandas library. A pandas Series is very similar to a one-dimensional NumPy array, but it has additional functionality that allows values in the Series to be indexed using labels. A NumPy array does not have the flexibility to do this. This labeling is useful when you are storing pieces of data that have other data associated with them. Say you want to store the ages of students in an online course to eventually figure out the average student age. If stored in a NumPy array, you could only access these ages with the internal ndarray indices 0,1,2. .. With a Series object, the indices of values are set to 0,1,2... by default, but you can customize the indices to be other values such as student names so an age can be accessed using a name. Customized indices of a Series are established by sending values into the Series constructor, as you will see below.

There are different ways to fill a DataFrame such as with a CSV file, a SQL query, a Python list, or a dictionary. Here we have created a DataFrame using a Python list of lists. Each nested list represents the data in one row of the DataFrame. We use the keyword columns to pass in the list of our custom column names.



**FIGURE 2.3: BLOCK DIAGRAM OF PROJECT**

The project is completely built in VS code with certain python libraries implemented in it.

The approach performs face recognition-based student attendance system. This method is also similar to others and begins with the input of an image either loaded from memory or from camera. Then it pre-processes the facial features and extracts it followed by subjective selecting and then the recognition of the facial images from known database. Both LBP and PCA feature extraction methods are studied in detail and computed in this approach in order to make comparisons. LBP is enhanced in this approach to reduce the illumination effect. An algorithm to combine enhanced LBP and PCA is also designed for subjective selection in order to increase the accuracy.

# Project Analysis: -

## 

## Life-long Learning

With the implementation of this project, we gained skills on the commands of LabView specifically Vision Assistant and Acquisition based modules. Understanding of Machine Vision Algorithm for face detection and reading manual of LabView enhances our skills on the LabView. Furthermore, the project management skills we gained by dividing the project into different phases and time slot not only developed our project management skills but also increased our time management skills.

## Impact of Engineering Solutions

This project saves time for the lecture by taking attendance on its own and update the attendance record. So that, additional time can be given to the topic for better understanding. Furthermore, it

will be easy to sort the attendance according to the student roll no. and name while updating t the student report for mentioning his/her attendance record. With the help of simple processing.Contemporary Issues Addressed.

Real-time image processing and detection are involved in this step. The student’s face is detected and the student is recognized using a webcam to record live video of the student. Because the image is captured in real time, image distortion can occur if a student is not fully facing the camera. Face landmark estimation [is used to detect the pose of the face, which solves the problem. There are 68 distinct landmarks on every face. The top of the chin, the outside edge of each eye, the inner edge of each eyebrow, and other landmarks can be found. landmarks present on the faceBy rotating, scaling, and shearing the face image, these landmarks can be used to center it. This image can now be used to calculate face encodings, which are then compared to encodings already stored in the database, and thestudent is identified as such. Attendance. Attendance List as observed using Google SheetsThe recognized faces are then marked present on a CSV file, which can be generated and assessed in a soft copy format on Excel, followingthe recognition process.

Manual attendance systemManual attendance systems are traditional systems in which a teacher or lecturer takes students’ attendance by calling names or signing an attendance sheet. Such attendance systems rely entirely on students acting in a fair and consistent manner. Although it is a low-cost system, it is extremely vulnerable to human error or manipulation. A student may be mistakenly marked present by the teacher if another student answers it on a roll-call, or a student can forge signatures on the sheet, resulting in ’proxy attendance’

The present system of attendance marking i.e., manually calling out the roll call by the faculty have quite satisfactorily served the purpose. With the change in the educational system with the introduction of new technologies in classroom such as virtual classroom, the traditional way of taking attendance may not be viable anymore.

NumPy stands for ‘Numerical Python’ or ‘Numeric Python’. It is an open source module of Python which provides fast mathematical computation on arrays and matrices. Since, arrays and matrices are an essential part of the Machine Learning ecosystem, NumPy along with Machine Learning modules like Scikit-learn, Pandas, Matplotlib, TensorFlow, etc. complete the Python Machine Learning Ecosystem.

NumPy provides the essential multi-dimensional array-oriented computing functionalities designed for high-level mathematical functions and scientific computation. Numpy can be imported into the notebook using

>>> import numpy as np

NumPy’s main object is the homogeneous multidimensional array. It is a table with same type elements, i.e, integers or string or characters (homogeneous), usually integers. In NumPy, dimensions are called axes. The number of axes is called the rank.

Some of the important attributes of a NumPy object are:

Ndim: displays the dimension of the array

Shape: returns a tuple of integers indicating the size of the array Size: returns the total number of elements in the NumPy array Dtype: returns the type of elements in the array, i.e., int64, character Itemsize: returns the size in bytes of each item

Reshape: Reshapes the NumPy array

NumPy array elements can be accessed using indexing. Below are some of the useful examples:

A[2:5] will print items 2 to 4. Index in NumPy arrays starts from 0 A[2::2] will print items 2 to end skipping 2 items

A[::-1] will print the array in the reverse order A[1:] will print from row 1 to end

In python, a vector can be represented in many ways, the simplest being a regular python list of numbers. Since Machine Learning requires lots of scientific calculations, it is much better to use NumPy’s ndarray, which provides a lot of convenient and optimized implementations of essential mathematical operations on vectors.

Vectorized operations perform faster than matrix manipulation operations performed using loops in python. For example, to carry out a 100 \* 100 matrix multiplication, vector operations using NumPy are two orders of magnitude faster than performing it using loops.

Basically, there are 2 rules of Broadcasting to remember:

* + 1. For the arrays that do not have the same rank, then a 1 will be prepended to the smaller ranking arrays until their ranks match. For example, when adding arrays A and B of sizes (3,3) and (,3) [rank 2 and rank 1], 1 will be prepended to the dimension of array B to make it (1,3) [rank=2]. The two sets are compatible when their dimensions are equal or either one of the dimension is 1.
    2. When either of the dimensions compared is one, the other is used. In other words, dimensions with size 1 are stretched or “copied” to match the other. For example, upon adding a 2D array A of shape (3,3) to a 2D ndarray B of shape (1, 3). NumPy will apply the above rule of broadcasting. It shall stretch the array B and replicate the first row 3 times to make array B of dimensions (3,3) and perform the operation.

IMPORTANT POINTS ON NUMPY:-

* + - 1. When we have to work on Numerical data, we prefer the numpy module.
      2. the powerful tool of numpy is Arrays.
      3. Numpy is memory efficient.
      4. Numpy has a better performance when number of rows is 50K or less.
      5. Indexing of numpy Arrays is very fast.
      6. Numpy is capable of providing multi-dimensional arrays whereas Pandas cannot do that.

## 6. DATETIME

The datetime module supplies classes for manipulating dates and times.

While date and time arithmetic is supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation.

Date and time objects may be categorized as “aware” or “naive” depending on whether or not they include timezone information.

With sufficient knowledge of applicable algorithmic and political time adjustments, such as time zone and daylight saving time information, an aware object can locate itself relative to other aware objects. An aware object represents a specific moment in time that is not open to interpretation. 1

A naive object does not contain enough information to unambiguously locate itself relative to other date/time objects. Whether a naive object represents Coordinated Universal Time (UTC), local time, or time in some other timezone is purely up to the program, just like it is up to the program whether a particular number represents metres, miles, or mass. Naive objects are easy to understand and to work with, at the cost of ignoring some aspects of reality.

The datetime module exports the following constants:

datetime.MINYEAR

The smallest year number allowed in a date or datetime object. MINYEAR is 1.

datetime.MAXYEAR

The largest year number allowed in a date or datetime object. MAXYEAR is 9999.

class datetime.date

An idealized naive date, assuming the current Gregorian calendar always was, and always will be, in effect. Attributes: year, month, and day.

class datetime.time

An idealized time, independent of any particular day, assuming that every day has exactly 24\*60\*60 seconds. (There is no notion of “leap seconds” here.) Attributes: hour, minute, second, microsecond, and tzinfo.

class datetime.datetime

A combination of a date and a time. Attributes: year, month, day, hour, minute, second, microsecond, and tzinfo.

class datetime.timedelta

A duration expressing the difference between two date, time, or datetime instances to microsecond resolution.

class datetime.tzinfo

An abstract base class for time zone information objects. These are used by the datetime and time classes to provide a customizable notion of time adjustment (for example, to account for time zone and/or daylight saving time).

class datetime.timezone

A class that implements the tzinfo abstract base class as a fixed offset from the UTC.

New in version 3.2.

Objects of these types are immutable.

The date, datetime, time, and timezone types share these common features:

Objects of these types are immutable.

Objects of these types are hashable, meaning that they can be used as dictionary keys.

Objects of these types support efficient pickling via the pickle module.

## METHODOLOGY

Having a video sequence as the input to the system the details of face detection, facial features extraction, normalization of facial features and quality score assignment are described in the following subsections:

* *Face Detection:* For better face-to-face facial recognition, we have used the face tracking method. What we did was first find the face using Viola and Jones' view as described, and then use the compilation tracker from the *dlib* library to track the face from frame to frame. This method also saves the calculation power as we do not need to find a face after switching to a new real-time video sequence frame. This helps to produce face-log i.e., a short representation of the title of the title in video sequence.
* *Parameters:* As people move around and look at different directions in front of the real-time camera, there may be a variety of headsets aimed at different angles. But because of biometrics, it is important to have a slightly modified face as a feature throughout the face-log. It is therefore important to include this feature in facial quality testing. We determined the position of the head using three angles: Roll, Yaw, Pitch. All of these angles are usually in the middle -90 to + 90. The roller and the voice are adjusted in a coordinated manner during the production of the facial log, so the only thing that worries us is the yaw angle. We use the detection of landmarks on the face, counting the tips of the nose tip and the point between the eyebrows.

## CONCLUSIONS & DISCUSSIONS

Automated attendance monitoring machine was the assignment selected with the aid of using us with the aid of using maintaining in view of the demand’s of day after day needs and wants of the society. The advancements in generation lead us to suppose out of the container and give you a few concept that might be destiny changing. Education is the maximum crucial issue which anybody need to collect as it's miles the idea for a higher life-style and could definitely alleviate the popular of a living community. What our schooling machine lacks is the involvement of college students withinside the schools, schools and universities. Instead of attending lectures and studying they prefer staying away from class and hold engaged in the use of those gadgets. Low attendance manner that the scholars and now no longer there to collect the information which they are supposed get and is of enormous importance for them and may cause them to a higher destiny

**7.FUTURE WORK**

Facial recognition solutions are expected to be present in 1.3 billion devices by 2024. Powered by AI,

facial recognition software in mobile phones is already being used by companies like iProov and Mastercard to authenticate payments and other high-end authentication tasks.

A facial recognition attendance system can be employed to recognize employees and confirm or refuse access upon entry.

This is especially important if your company has pricey inventory or sensitive data. As a result, other employees or persons outside of your organization cannot access a company's data.

Facial recognition is here to stay and rather than seeing facial recognition as a threat to our personal privacy, we should instead be embracing the many benefits that facial recognition provides.

Whilst there are isolated cases of facial recognition being used inappropriately, there are now thousands of use cases that show that,

when deployed appropriately and with the consent of people using the software, facial recognition is helping to create a safer environment, providing outstanding security and enhancing customer experiences across a wide range of settings.

Attendance Management keeps track of your employee hours. It is the system you use to document the time your employees work and the time they take off.

Attendance Management can be done by recording employee hours on paper, using spreadsheets, punching time cards, or using online attendance software for your company.

Output in MS Excel We get the output as given below. After that we can derive the results in appropriate format using different function in the spreadsheet .

We can get the following parameters by using this format as output as shown in the figure. This function is performed using the Spreadsheet Link

Ex toolbox of the PYTHON. •

If a person is present, a ‘1’ is passed on to the particular field of the student • The date and time are also passed on to the sheet.

We can include any number of students’ data using this system and provided we use a better quality of an image capturing device.

In the next section we describe how we integrate all these function y using the

Graphical User Interface (GUI). This gives an easy-to-use interface to the users.

**CONCLUSION**

In this system we have implemented an attendance system for a lecture, section or laboratory by which lecturer or teaching assistant can record students’ attendance. It saves time and effort, especially if it is a lecture with huge number of students.

Automated Attendance System has been envisioned for the purpose of reducing the drawbacks in the traditional (manual) system.

This attendance system demonstrates the use of image processing techniquesin classroom.

This system can not only merely help in the attendance system, but also improve the goodwill of an institution.

The User Interface of it is very friendly and can be easily used by anyone. It also decreases the amount of time taken to write details and other modules.

All the details about students, teachers, and their other tasks can only be seen by the verified users.

An automatic attendance management system provides accurate time tracking, which helps to minimise the costly and inevitable errors in the manual data entry processes.

This kind of accurate data helps provide accurate payroll and performance data.

The cost and time saved are even larger because the data acquired from the face recognition attendance system is accurate in real-time.

Because the overall process is automated, human intervention is limited.

The objective of the school attendance software is to reduce the time that is consumed when attendance is taken manually.

Unlike the manual process, an online system easily helps management to analyze student's attendance details as per requirement.

## 

## 8.REFERENCES

1. Face verification using error correcting output codes*. In: Computer Vision and Pattern Recognition (CVPR01), IEEE Press. pp. 755-760.*
2. Dalal, N. and Triggs, B*.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR). pp. 886–* [*893. (2005)*](https://en.wikipedia.org/wiki/General-purpose_programming_language)
3. Viola, Jones: Robust Real-time Object Detection*, IJCV 2001*
4. Nirmalya Kar and Ashim Saha ; *Study of implementing automated attendance system using face recognition technique*