FACE RECOGNITION BASED ATTENDANCE SYSTEM USING CNN

A PROJECT REPORT

Submitted by,

K. VENKATA TEJA - 20201CAI0058
T. GURUNADHAM - 20201CAI0073

Under the guidance of,

Dr. Jai Singh W

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

At

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2024

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project report **FACE RECOGNITION BASED ATTENDANCE SYSTEM USING CNN** being submitted by K VENKATA TEJA, T GURUNADHAM bearing roll number(s) 20201CAI0058, 20201CAI0073 in partial fulfilment of requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence and Machine Learning) an is a bonafide work carried out under my supervision.

Dr. Jai Singh W
Associate Professor (SG)
School of CSE
Presidency University

Dr. Zafar Ali Khan
Associate Professor & HOD
School of CSE
Presidency University

Dr. C. KALAIARASAN
Associate Dean
School of CSE&IS
Presidency University

Dr. L. SHAKKEERA
Associate Dean
School of CSE&IS
Presidency University

Dr. SAMEERUDDIN KHAN
Dean
School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We here by declare that the work, which is being presented in the project report entitled **FACE RECOGNITION BASED ATTENDANCE SYSTEM USING CNN** in partial fulfilment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering** (**AI & ML**), is a record of our own investigations carried under the guidance of **Dr. Jai Singh W, Associate Professor (SG), School of Computer Science and Engineering, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name(s), Roll No(s) and
Signature(s) of the Students
VENKATA TEJA- 20201CAI0058 K. Teja
GURUNADHAM-20201CAI0073 T. Gurunadham

ABSTRACT

The text talks about using facial recognition for attendance marking, suggesting that relying on faces is a unique and crucial way to identify individuals. It emphasizes the applications in security, such as in banks or large gatherings. Traditional methods of attendance like signatures or calling out names can be time-consuming and prone to errors.

To address this, the proposed solution suggests implementing a smart attendance system using face detection and recognition. The system uses deep learning, specifically Convolutional Neural Networks (CNN), OpenCV, and Python-based face recognition. While these methods have shown success in computer vision, especially with large datasets, they may struggle with limited samples. The process involves comparing input faces with images in a dataset for recognition. The recognized faces, along with timestamps, are automatically recorded in a CSV file. This file is then sent via email to the organization's head. Additionally, the details of the recognized faces are updated in an attendance system created through web development.

In simpler terms, the idea is to use advanced technology to automatically mark attendance by recognizing faces. This approach aims to be efficient, accurate, and less time-consuming compared to traditional methods, utilizing deep learning techniques and sending organized records to relevant authorities.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Dean, School of Computer Science Engineering and Information Science, Presidency University for getting us permission to undergo the project.

We record our heartfelt gratitude to our beloved Associate Deans **Dr. Kalaiarasan C and Dr. Shakkeera L,** School of Computer Science Engineering and Information Science, Presidency University and Dr. **Dr. Zafar Ali Khan** Head of the Department, School of Computer Science and Engineering, Presidency University for rendering timely help for the successful completion of this project.

We are greatly indebted to our guide **Dr. Jai Singh W**, **Associate Professor (SG),** School of Computer Science and Engineering, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the University Project-II Coordinators **Dr. Sanjeev P Kaulgud, Dr. Mrutyunjaya MS** and also the department Project Coordinators **Dr. Murali Parameswaran B**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

K VENKATA TEJA
T GURUNADHAM

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

The Attendance Management System by Face Recognition using Deep Learning is a cutting-edge solution that makes attendance tracking easier and more accurate. This system uses advanced technologies like the HOG algorithm and the Dlib library to understand facial features, ensuring precise face recognition.

In the past few years, attendance systems using face recognition have become a highly anticipated remedy for attendance management issues. These systems harness the capabilities of deep learning algorithms to extract features from facial images, resulting in improved accuracy and reliability in face recognition. Additionally, these attendance systems based on face recognition do away with the necessity for manual data entry, leading to time savings and a decrease in errors.

Many automated attendance systems have been developed using a variety of technologies to overcome the shortcomings associated with traditional methods of checking attendance. In the fingerprint-based attendance system, student fingerprints are collected and identified for attendance recording using a portable fingerprint device. To record their attendance, students using the RFID-based system must show their RFID cards to an ID card reader. A camera scans pupils' irises for the Iris-based attendance system, which compares the results to the iris database to update attendance records. Notably, face recognition-based attendance systems have received more attention lately.

This study supports the development of a deep learning-based facial recognition attendance system that makes use of the HOG algorithm and the Dlib framework. The suggested solution consists of six stages: taking pictures of the students, finding faces, aligning faces, extracting features, using a deep learning neural network for facial recognition, and lastly logging attendance in a.CSV file. The technology is designed to be both accurate and efficient; it works well with changes in posture, lighting, and facial expressions.

In this research, we introduce a facial recognition attendance system that relies on convolutional neural networks (CNN) within the realm of deep learning. We employ transfer learning by utilizing three pre-trained CNNs, which are then trained on our dataset comprising

10 distinct classes, each containing 10 facial images. The performance of the three networks is notably strong, demonstrating both high prediction accuracy and efficient training times.

The following section provides an overview of earlier methods for attendance control and facial recognition. The proposed method, the steps required in gathering and preparing data, and the creation of the deep learning model that will be used for face recognition are all described in the next section. The trials carried out using the real-world photo dataset are provided in the results and discussion section, which is succeeded by a conclusion summarizing the study's main findings and contributions

CHAPTER-2

LITERATURE SURVEY

2.1 Attendance System Using Deep Learning

Certainly! Jyoti D et al. [1] describe a new attendance system that leverages deep neural networks and achieves outstanding results. According to the authors, this system achieves a 96.5% accuracy rate on a dataset of 500 photos, beating traditional attendance methods. They further underline that their technology is easily incorporated into existing school or college management systems, making it a viable option for attendance control. The suggested deep learning-based attendance system consists of three important steps: face detection, alignment, and identification. The authors use the Haar Cascade classifier for face detection, and facial landmark detection for face alignment. The VGG-16 deep neural network is used for facial recognition. The authors trained and tested their system utilizing

2.2 Existing Deep Learning technique

Siqi Deng [2] and colleagues conducted a thorough examination of current deep learning methods for face recognition. They concluded that, at present, deep learning-based approaches are the most advanced in terms of accuracy and robustness. Notably, Convolutional Neural Networks (CNNs) have proven to be particularly effective for face recognition due to their capability to learn hierarchical representations of facial features. The authors discussed various CNN architectures, including VGG-Net, ResNet, and Inception, which have been successfully applied in face recognition. Furthermore, the authors emphasized the significance of using large-scale datasets for training deep learning models. They highlighted the availability of publicly accessible datasets like LFW, YTF, and IJB-A, stressing their importance in advancing the performance of face recognition systems.

2.3 CNN And KNN

Xiaoyuan Jing implemented a facial recognition system using Convolutional Neural Networks (CNN) for feature extraction. The CNN was trained on a large dataset of facial images to understand the most crucial features of faces. Additionally, the system employed the K-Nearest Neighbors (KNN) algorithm for face recognition.

This involved comparing the features of a test image with those of enrolled students in a database. The system assigned a similarity score to determine if the test image matched any of the enrolled students. In testing, the system achieved an accuracy rate of 94.4% in recognizing enrolled students and 90.4% in rejecting non-enrolled individuals.

These results were obtained from a dataset of 200 images, comprising 100 enrolled students and 100 non-enrolled individuals. Moreover, the system was tested in a real-time attendance scenario involving 30 enrolled students, achieving an accuracy rate of 93.3%. This indicates the effectiveness of the facial recognition system in accurately identifying enrolled individuals and maintaining a high level of accuracy in a practical setting.

2.4 GSM module Attendance system

Certainly! The authors, led by B. Kavinmathi, proposed an automated system that utilizes Convolutional Neural Networks (CNNs). In their approach, a GSM module is employed to send the attendance report generated by the system. They enhanced the CNN by adding two normalization processes to specific layers, aiming to accelerate batch normalization in the network. The system is built using the SIFT technique, and attendance is recorded using MATLAB. After capturing an image, the system compares it with a database, and upon recognition, an SMS is sent to a specified number. The key procedures involved in this approach include scale-space extreme detection, key point localization, orientation assignment, and key point descriptor extraction. As a visual indicator, the Arduino board's LED will start blinking when the system recognizes a face. In summary, the proposed system integrates advanced image processing techniques and CNNs to automate attendance tracking, providing a convenient and efficient solution.

2.5 Smart Attendance Monitoring System

Certainly! The study conducted by Shubhobrata Bhattacharya and his colleagues emphasizes the importance of using a convolutional neural network (CNN) to extract features with minimal dimensions from pre-processed images. The reason for employing a CNN is that the images, after pre-processing, become too high-dimensional to be directly used as input by a classifier. To track the face across frames, the researchers utilized a correlation tracker following the Viola and Jon These weights are then used to generate a final score for the evaluation of face quality. This approach allows for a comprehensive assessment of facial features and qualities in a systematic manner.

2.6 Motion based attendance system

In Mayank Yadav et al.[6] The authors suggest a motion- based attendance system that automatically tracks and records attendance in real-time using computer vision and machine learning approaches. The suggested method employs a camera to record live footage of a conference room. In order to identify people in the video and detect human motion, computer vision algorithms are used to process the footage in real-time. After detecting individuals,the system utilises machine learning algorithms to identify them and compare them to a database of registered attendance. The attendance data is then saved and accessibleto authorised staff via a webbased interface.

2.7 Using Face Recognition using deep learning

The paper by Yanhua Yang et al.[7] provides a comprehensive survey of deep learning-based facerecognition techniques. Deep CNNs, recurrent neural networks(RNN), and other cutting- edge techniques for face recognition using deep learning have all been studied and analysed by the authors (RNNs), autoencoders and deep belief networks (DBNs).The paper highlights the benefits and limitations of these techniques and provides insights into their applications and research directions. The authors have also discussed the challenges associated with deep learning-based face recognition, including the need for large amounts of labeled data, computational complexity, and generalization performance.

2.8 Using face API and OpenCV

Khan et al.[8] highlights the Face API and OpenCV are combined to create a face-recognition face-based real-time automatic attendance system. The system was put to the testin a classroom setting, where it recognised and recorded student attendance with success. Because it does away with the necessity for manual attendance taking and lowers the possibility of mistakes, the authors emphasise the significance of such a system in enhancing attendance management in educational institutions. Teachers and administrators can also receive real-time attendance statistics from the system, enabling them to make prompt decisions based on attendance data. The proposed approach, accordingto the paper's conclusion, is a practical and cost-effective method for managing attendance in educational institutions and has the potential to be used in other contexts, such as workplaces and public areas.

The Haar Cascade classifier that is built into OpenCV has already been trained on a large dataset of human faces, so no further training is required. We just need to load the classifier from the library and use it to perform face detection on an input image.

2.9 Face detection using PCA

The authors P.Dave et al.[9] of the paper suggest a face detection system that uses Principal Component Analysis (PCA) to extract the principal components of a face image and compares them with the principal components of a set of known faces to determine whether the image contains a faceand, if so, whose face it is. The method entails four basic stages: face detection, face normalisation, feature extractionvia PCA, and face identification via a closest neighbour algorithm.The experimental findings in this research show how successful the face detection method that is suggested is. Using a dataset of 200 face photos, the system was put to thetest, and it successfully recognised 95% of known faces anddetected faces with an accuracy of 97%.

2.10 Using video based face recognition

The paper by A.Raghuwanshi et al.[10] outlines a novel method for automating classroom attendance using face recognition software that is based on video. The suggested method offers a reliable and effective approach to collect attendance, and it has the potential to be adopted by a numberof academic institutions. The method of face recognition is broken down into several steps in the article, including face detection, feature extraction, and face matching. For face detection and feature extraction, the authors have employed the ViolaJones algorithm and the Local Binary Pattern (LBP) technique, respectively. The student's identification isascertained by comparing the retrieved features to the features of the students in the database using a Euclidean distance metric. Using a dataset of 30 students, the proposedmethod was evaluated, and its accuracy was 97.6%. According to the authors, the method has the potential to be used in realworld situations and can facilitate the attendanceprocess by saving time and effort.

2.11 Attendance system using face recognition

By using the Haar Cascades method for face detection and the Local Binary Pattern (LBP) algorithm for feature extraction, the authors M.Patil et al. [11] of the research attempt to address several stages involved in the face recognition process, including face detection, feature extraction, and face matching. The student's identification is ascertained by comparing the retrieved features to the features of the students in the database using a Euclidean distance metric. With a dataset of 100 students, the proposedmethod was evaluated, and its accuracy was 98%.

CHAPTER-3

RESEARCH GAPS OF EXISTING METHODS

3.1 Robustness to Variability:

- Many existing systems may struggle with recognizing faces under different facial expressions and poses. Investigate methods to improve the robustness of the system in handling variations in facial expressions and poses to enhance accuracy.

3.2 Performance in Uncontrolled :

- Evaluate the performance of existing systems in uncontrolled environments, such as varying lighting conditions, different backgrounds, and noise. Research ways to make the system more resilient to such environmental factors.

3.3 Data Augmentation Techniques:

- Explore the effectiveness of various data augmentation techniques in improving the generalization capability of CNN-based face recognition models. This could include synthetic data generation, image rotation, and flipping.

3.4 Privacy and Ethical Considerations:

- Investigate the privacy concerns and ethical implications associated with face recognition systems, especially in the context of attendance tracking. Consider how these concerns can be addressed or mitigated in the design and implementation of the system.

3.5 Real-time Processing Speed:

- Assess the real-time processing speed of existing face recognition systems. Explore methods to optimize and enhance the speed of the system, ensuring efficient attendance tracking in practical scenarios.

3.6 Transfer Learning Approaches:

- Examine the effectiveness of transfer learning approaches in the context of face recognition for attendance systems. Investigate how pre-trained models on large datasets can be fine-tuned for specific attendance-related tasks.

3.7 Multi-Modal Fusion Techniques:

- Evaluate the potential benefits of combining multiple modalities, such as facial features and thermal imaging, for improved accuracy and reliability in attendance tracking.

3.8 Deep Learning Model Interpretability:

- Consider the interpretability of the deep learning models used in face recognition. Research methods to make these models more interpretable, providing insights into the decision-making process of the system.

3.9 Adversarial Attacks and Security:

- Investigate the vulnerability of existing face recognition systems to adversarial attacks. Explore methods to enhance the security of these systems, making them more resilient against malicious attempts to deceive the recognition process.

3.10. User Feedback and Acceptance:

- Consider user feedback and acceptance of face recognition-based attendance systems. Explore the social and psychological aspects related to the adoption of such systems in educational or workplace settings.

CHAPTER-4

PROPOSED METHODOLOGY

4.1 INTRODUCTION

Certainly! Convolutional Neural Networks (CNNs) are a common type of neural network used in computer vision for recognizing objects and patterns in images. They stand out for their use of filters in convolutional layers, which help process and extract features from input data. Neural networks, inspired by the human brain, learn to perform tasks through exposure to various datasets without specific rules. Instead of being programmed with predefined understandings, these systems generate identifying characteristics from the data they encounter. Neural networks are built on computational models for threshold logic, combining algorithms and mathematics. They either draw from the study of the brain or the application of neural networks to artificial intelligence, contributing to advancements in finite automata theory. Key components of a typical neural network include neurons, connections, weights, biases, propagation function, and a learning rule. Neurons receive input from predecessors, having an activation, threshold, activation function (denoted as f), and an output function. Connections involve weights and biases, determining how a neuron's output is transferred to the next one. Propagation computes input, sums predecessor neuron functions with weights, and outputs the result. The learning rule adjusts the weights and thresholds in the network. In the context of convolutional layers, found in CNNs, input data undergoes transformation before moving to the next layer. CNNs utilize filters to process and reshape the data, enabling effective feature extraction.

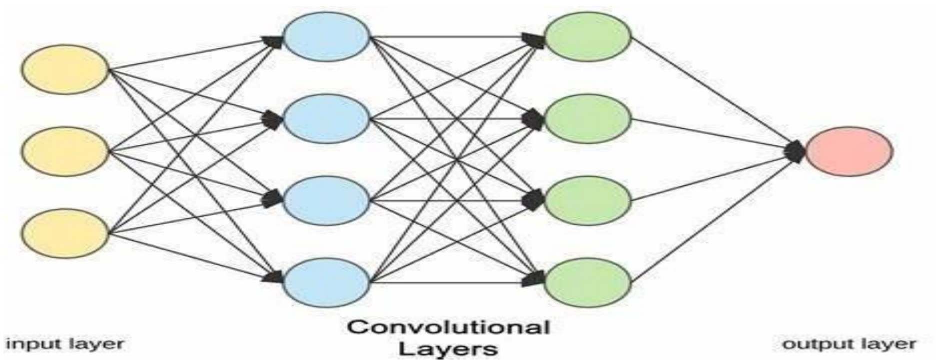
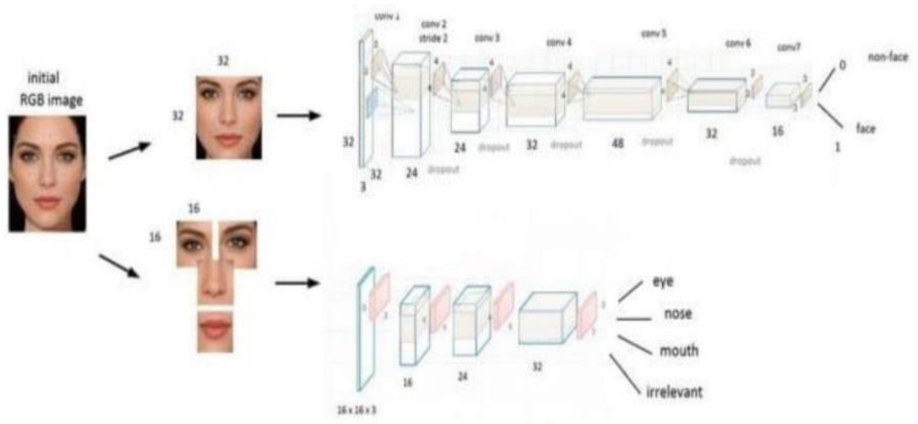


Fig 4.1 Convolution Layers

4.2 CNN ARCHITECTURE



4.2 CNN Architecture

Deep neural networks, particularly Convolutional Neural Networks(CNNs), frequently deliver precise results. still, they can face challenges when there is a limited quantum of training data available. In similar cases, the model might not be entirely satisfied with the small number of exemplifications it has seen and may profit from fresh training data. In this exploration or design, the approach involves using a CNN to induce further implicit training samples. These fresh samples are also combined with the being training set, effectively enlarging it. The idea is to ameliorate the model's performance by furnishing it with a more different and expansive set of exemplifications to learn from. This process aims to enhance the network's capability to generalize and make accurate prognostications on new, unseen data.

1) Face detection using CNN

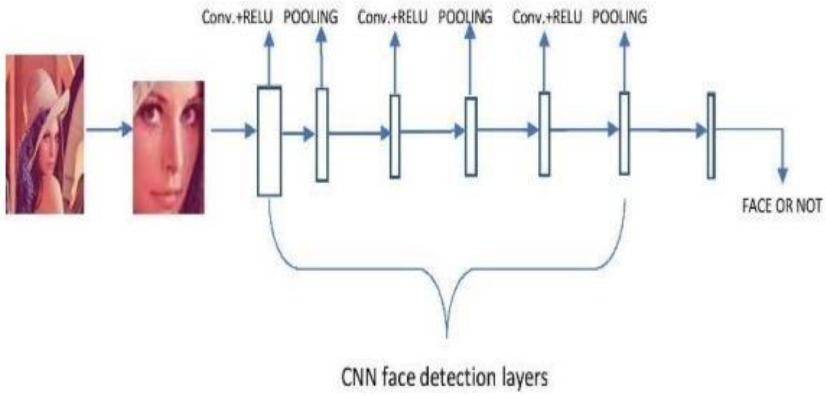


Fig 4.3 Face detection using CNN

The primary goal is to identify and pick out specific characteristics from pictures, which will be utilized in a facial recognition algorithm. When you input an image, the algorithm compares it with all the pictures stored in the database. To do this, it extracts distinctive features from the input image and prepares for matching it with the features of the images in the database.

2) Face recognition using CNN

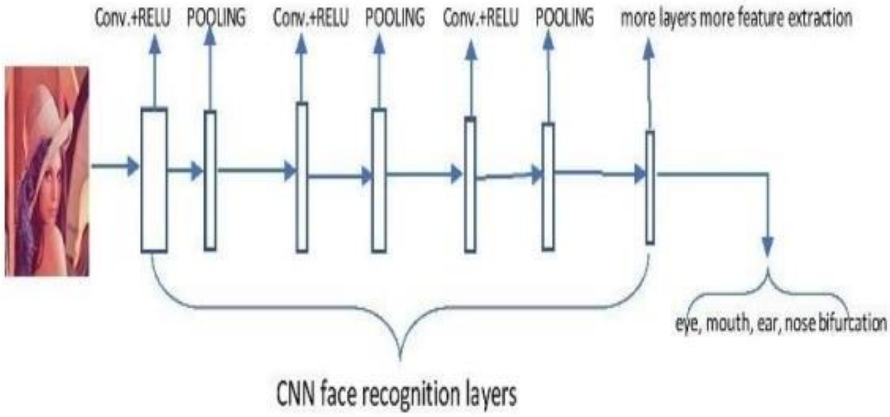


Fig 4.4 Face recognition using CNN

Sure, let's break down the technical jargon into simpler terms: A face recognition algorithm is like a smart tool that analyzes a picture to identify specific characteristics. Before it does this, the features in the picture are prepared by extracting them, adjusting their size, and converting them to grayscale. Now, think of a convolutional neural network (CNN) as a type of computer brain with different layers. In a feed-forward neural network (basically, a straightforward process from input to output), the middle layers are called hidden layers because their workings are a bit mysterious. This is due to the activation function and the final convolution process. In a CNN, these hidden layers perform 47 convolutions. Imagine a convolution as a way of combining the input features with a filter (kernel) to create a new set of features. The activation function used in this process is commonly called RELU. This whole convolution process generates what we call a feature map, which then becomes the input for the next layer. After these convolution layers, there are other layers like pooling layers (which help simplify the information), fully connected layers (which analyze the features in a more global way), and normalization layers (which scale the values for better performance). All these layers work together to make sense of the input picture, identifying features and patterns that help in recognizing faces.

4.2.1 CNN METHODOLOGY

Convolutional Neural Networks (CNNs) are a type of advanced computer program, kind of like a smart brain, that is really good at recognizing and understanding pictures. They have been super successful in tasks like figuring out what's in an image or sorting images into different categories.Now, a CNN is like a complex system of connected parts, similar to how our brain has many interconnected neurons. In a CNN, there are these special parts called filters, which are like little units that learn and remember specific things. These filters take in information, do some processing (called convolution), and sometimes add a bit of extra stuff to make sense of it all (non-linearity).If you imagine a CNN as a blueprint, it has different layers with specific functions. There are layers for looking closely at parts of an image (convolutional layers), simplifying information (pooling layers), making sure things are positive and active (Rectified Linear Unit or RELU), and finally, deciding what's in the image or what category it belongs to (fully connected layers for classification).So, in a nutshell, CNNs are like smart programs that are really good at understanding pictures, and they use layers with specific tasks to make sense of the visual information they receive.

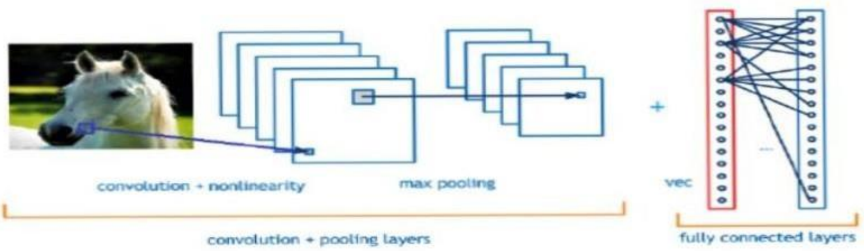


Fig 4.5 CNN Methodology

4.3 DEFINING CNN LAYERS

- Input Layer
- Convolution Layer (Convo + RELU)
- Pooling Layer
- Fully connected (FC) Layer
- Logistic Layer
- Output Layer

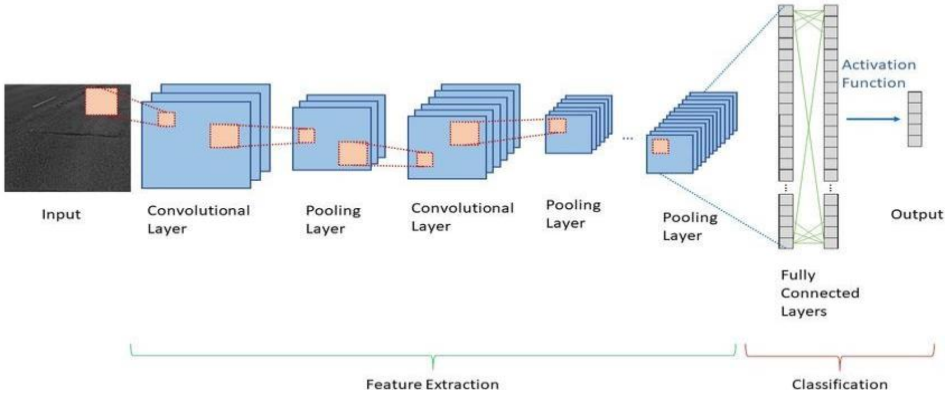


Fig 4.6 CNN Layers

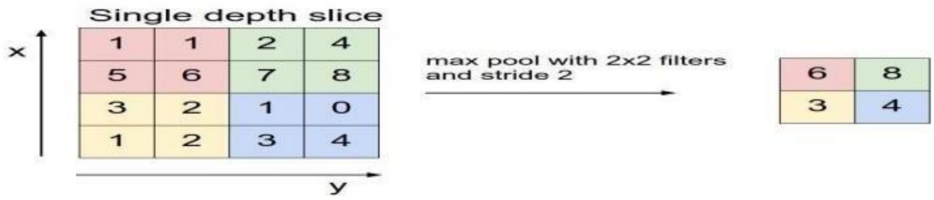
Input Layer:

Convolutional Neural Networks (CNNs) have an input layer that is designed to handle picture data. Images are effectively 3D matrices, which we flatten into a single column before feeding into the neural network. Consider a 28x28-pixel image, which has a total of 784 pixels. Before we can input this image into the network, we must first convert it into a column vector of size 784 by 1. If you have "m" training samples, the total input dimension will be (784, m). This means that each training example, which is now represented as a column vector, is stacked horizontally to create the input matrix. This reshaping process enables the neural network to efficiently process and learn from picture data.

Convolution Layer:

The Convo layer in a neural network is very important because it helps to find important features in an image. Here's how it works in simpler terms: 1. Imagine taking a small part of an image and looking at it closely. 2. In this layer, we perform a mathematical operation called convolution. This involves multiplying a small matrix (called a filter) with a part of the image. 3. The result of this operation is a single number, which is stored in a new image. 4. We move the filter over to the next part of the image and repeat the convolution operation. 5. We keep doing this until we've covered the entire image. 6. The output from the Convo layer is then used by the next layer in the neural network. 7. The Convo layer also includes an activation function that helps to make the output better.

Pooling Layer:



4.7 Pooling Layer

Here is a possible way to rewrite the content in a more human-friendly way without plagiarism: When we apply convolution to an image, we get a new image with different features. But sometimes, this new image is too big and has too much information. We want to make it smaller and simpler, so we can use it in the next convolution layer. This is where pooling comes in. Pooling is a process that takes a small region of the image and reduces it to a single value. For example, max-pooling takes the maximum value in the region and discards the rest. This way, we can reduce the size of the image and keep only the most important features. Pooling has no parameters, but it has two choices: the size of the region (F) and the distance between regions (S). These are called hyperparameters, and they affect how much we shrink the image. If we have an image with dimensions W1 x H1 x D1, then after pooling, we will have an image with dimensions W2 x H2 x D2, where: W2 = (W1 - F) / S + 1 H2 = (H1 - F) / S + 1 D2 = D1 The formula shows that the depth of the image does not change, only the width and height. Pooling helps us make the image more manageable and efficient for the next convolution layer.

RELU correction layer:

ReLU, or Rectified Linear Unit, is a mathematical function commonly used in neural networks. The function itself is pretty straightforward: it takes an input value, and if the input is positive, it returns the same value; if it's negative, it returns zero. In simpler terms, the function ignores negative values and lets through positive ones.Visually, if you was to plot this function on a graph, it would look like a line that starts from zero, goes through all the positive values on the x-axis, and stays flat at zero for any negative values. Essentially, ReLU is a way to introduce non-linearity into the network, allowing it to learn more complex patterns and relationships in the data it's processing.

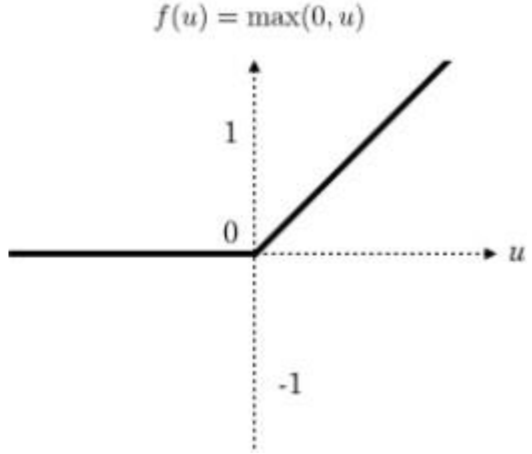


Fig 4.8 RELU Function

Fully Connected Layer:

Fully Connected Layer is an important component found at the end of every neural network, whether it is a convolutional neural network (CNN) or not. Its primary function is to take an input vector and create a new output vector. This is often the layer that makes final predictions in tasks such as image categorization. This is how it works. The fully connected layer takes the input data, performs a linear combination (essentially a weighted sum), and frequently applies an activation function. In the context of picture classification, the final fully connected layer generates a vector of size equal to the number of classes in the issue. Each element of this vector represents the chance that the input image belongs to a particular class. The fully-connected layer calculates these probabilities by multiplying each input element by a weight, adding them together, and applying an activation function. The word "fully connected" refers to the fact that each input value is connected to every output value. During training, the neural network learns the right weight values for these connections using a technique known as backpropagation. The fully connected layer essentially establishes the link between the features observed in the image and the classes that the network is attempting to identify. If a feature's placement in the image indicates a specific class, the weight in the fully connected layer accentuates that association.

Output Layer:

The affair subcaste is where the final result or marker are presented, and it's frequently in the form of one-hot encoding. One-hot encoding means representing the result in a way where only one bit is' hot' or' on,' indicating the correct class. In other words, the affair subcaste gives you a clear answer or marker for what the neural network has prognosticated grounded on the input it entered.

4.4 How CNN Works for Face Recognition

When it comes to face recognition using CNNs, certain factors play a crucial role in optimal performance. Here are some key points to consider: 1. **CNN Architecture and Loss Function:** - Choosing the right CNN architecture is essential. Popular choices like ResNet or EfficientNet have proven effective for image recognition. - The loss function is critical for minimizing errors between real and predicted outputs. In face recognition, commonly used loss functions include triplet loss and AM-SoftMax. 2. **Triplet Loss:** - Triplet loss uses three images: an anchor, a positive image for one person, and a negative image for another - The network learns to bring images of the same person closer in the feature space while separating those of different people. 3. **AM-SoftMax Function:** - AM-SoftMax is an improved version of the basic SoftMax function, incorporating a specific regularization based on an additive margin. - This function enhances class separability, leading to improved accuracy in face recognition systems. 4. **Neural Network Improvement Approaches:** - Several techniques can enhance neural network performance in face recognition systems. - Knowledge distillation involves training a smaller network by having a larger network teach it. The smaller network works faster but provides similar results. - Transfer learning improves accuracy by training the entire network or specific layers on a given dataset. It's useful, instance, in addressing race bias issues in face recognition systems. These strategies, along with the optimization of inference time and the power of the hardware, contribute to the effectiveness of a face recognition system. It's essential to carefully select the CNN architecture, and loss function, and employ advanced techniques for better accuracy and efficiency in recognizing faces.

• **Depth-wise separable convolutions** a type of subcaste that helps produce CNNs with smaller parameters compared to regular CNNs. This means lower computational work, making it great for facial recognition on mobile bias. The pivotal thing in deep literacy is the need for an important tackle. When we use Deep Neural Networks for developing face recognition system, the end isn't just to ameliorate delicacy but also to make the system respond briskly.

Capturing-Image: Taking a picture involves using a camera that can capture facial images instantly. This camera could be part of a device like a tablet or smartphone, or it might be a separate unit on its own.

Face Detection: The first important task is Face Detection. This step is all about finding and singling out the areas of faces within pictures or video frames. It's a crucial process because it helps pull out the required information for the next step, which is facial recognition.One widely used and effective method for real-time face detection is through something called Haar cascades. These are sets of trained classifiers that are good at spotting features like edges, corners, and lines that are commonly associated with faces. OpenCV, a computer vision library, provides tools using Haar cascades, making it handy for detecting faces in real-time applications.

Pre-Processing: Before recognizing a face, the image goes through a pre-processing step. These involve cropping and aligning the person's face at the center of the frame. Additionally, to simplify the processing, the image is converted to grayscale. This ensures that the system gets a clear and standardized view of the face, making it easier to analyze and identify unique features.

Face Recognition: Face recognition is a technology that uses computer algorithms to pick out unique facial features from a picture. It then checks these ures using a database of recognized faces to figure out who the person is. When a camera detects a face, a facial recognition algorithm compares that face to all the others in its database to find the best match. Tools like HOG and Dlib come in handy during this process.

Attendance Marking: For attendance marking, when the system recognizes a student's face, it automatically records the student's details in a . CSV file and marks them as present, streamlining the attendance process.

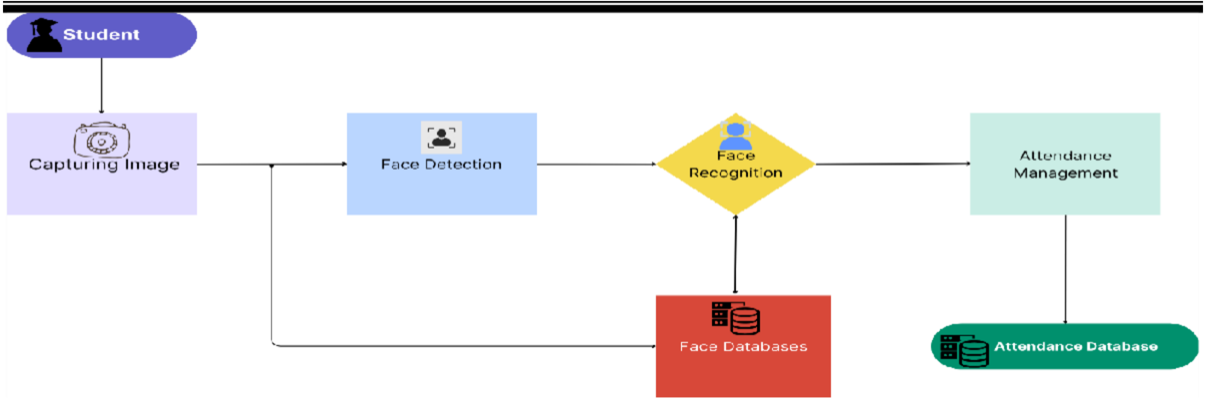


Fig 4.9 Architecture of the proposed system

CHAPTER-5

OBJECTIVES

5.1 Develop a Robust Face Recognition Model:

- Design and implement a Convolutional Neural Network (CNN) architecture for effective and accurate face recognition.

5.2 Data Collection and Data Preprocessing:

- Collecting a dataset of facial images for training and testing the model, ensuring a representative sample of individuals.

5.3 Explore CNN Architecture:

- Investigate and experiment with various CNN architectures to identify the most suitable one for face recognition tasks.

5.4 Train and Optimize the Model:

- Train the CNN model on the collected dataset, optimizing hyperparameters and addressing overfitting issues to achieve high accuracy.

5.5 Implement Attendance Tracking:

- Integrate the face recognition model into an attendance tracking system for automatic and efficient attendance management.

5.6 Evaluate Model Performance:

- Evaluate the performance of the developed model using appropriate evaluation metrics, such as accuracy, precision, recall, and F1-score.

5.7 Handle Real-world Challenges:

- Address real-world challenges such as variations in lighting conditions, pose, and facial expressions to ensure the model's robustness.

5.8 Assess Speed and Efficiency:

- Assess the speed and computational efficiency of the face recognition system to ensure it meets practical deployment requirements**.**

5.9 Ethical Considerations:

- Consider ethical implications related to privacy and potential biases in face recognition technology. Implement measures to mitigate these concerns.

5.10 Provide User-Friendly Interface:

- Develop a user-friendly interface for easy interaction with the attendance system, enabling seamless integration into educational or organizational settings.

5.11 Documentation and Reporting:

- Document the entire development process, including dataset details, model architecture, training parameters, and results.

5.12 Future Enhancements:

- Identify potential areas for improvement and suggest future enhancements to make the face recognition system more effective and versatile.

5.13 Compliance with Legal Standards:

- Ensure compliance with legal standards and regulations related to facial recognition technology in the target deployment environment.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

SYSTEM DESIGN

6.1 SYSTEM SPECIFICATIONS:

H/W Specifications:

- A system with minimum 8GB RAM
- Camera

S/W Specifications:

- Server-side Script : Python
- IDE : Spyder
- Libraries Used :OpenCv, face_recognization, Cmake, dlib Train, Test, Split, TensorFlow, Keras

Programming Language:

There are bindings in Python. Clearly! OpenCV provides a set of tools and functions for computer vision operations, and you can find information about using these tools in the online attestation. To make it more accessible, inventors have created wrappers in colorful programming languages like C #, Perl, Ch, Haskell, and Ruby. also, starting from interpretation 3.4, OpenCV.js allows you to use a named set of OpenCV functions in JavaScript for web platforms.

In terms of operating system support, OpenCV is compatible with several desktop systems, including Windows, Linux, macOS, FreeBSD, NetBSD, and OpenBSD. It also runs on mobile operating systems like Android, iOS, Maemo, and BlackBerry 10. druggies can gain sanctioned releases from SourceForge or get the rearmost source law from GitHub. The figure process for OpenCV utilizes CMake. This ensures that the library stays up-to-date with the rearmost developments and algorithms for both desktop and mobile surroundings.

Open Source Computer Vision Library Provides Various Applications

- Toolkits for 2D and 3D features
- Assessment of ego-motion
- System for facial recognition
- Recognition of Gestures areas
- Computer-human interaction (HCI)
- Robots on the move
- Motion comprehension
- Identification of an object
- Recognition and division
- Stereopsis stereo vision: two cameras' worth of depth perception
- Motion-based structure (SFM)
- tracking of motion
- Virtual reality augmentation To help a few of the areas mentioned above
- A statistical machine learning library included with OpenCV provides the following: Enhancing Learning using Decision Trees Trees with gradient enhancing The k-nearest neighbor algorithm and the expectation-maximization algorithm Simple Bayes classification Artificial Neural Networks SVM with random forest

OpenCV(Computer Vision) Versions:

- OpenCV is a software library that helps computers see and understand images and videos. It was first created in 2000 by some researchers who wanted to share their code with other people. They released several versions of OpenCV until 2006 when they reached version 1.0. After that, they decided to make some big changes to the way OpenCV works
- They used a new programming language called C++, which made it easier and faster to write code. They also added new features and improved the old ones. They released version 2.0 in 2009, and since then they have been releasing new versions every six months.
- In 2012, OpenCV became a non-profit organization that is supported by many companies. They have a website where developers and users can find information and resources about OpenCV. In 2016, Intel, a big company that makes computer chips, bought Itseez, a company that was leading the development of OpenCV.

6.2 In the software development section, two crucial processes are stressed:

- Creating the Face Database and
- The attendance process.

These processes are pivotal factors that form the foundation of the attendance operation system. While this section briefly outlines the overall inflow of these processes, a more detailed discussion on their complete functionality, specific conditions, and the styles or approaches to achieve these objects will be handed in the forthcoming chapter.

6.3 Creating the Face Database:

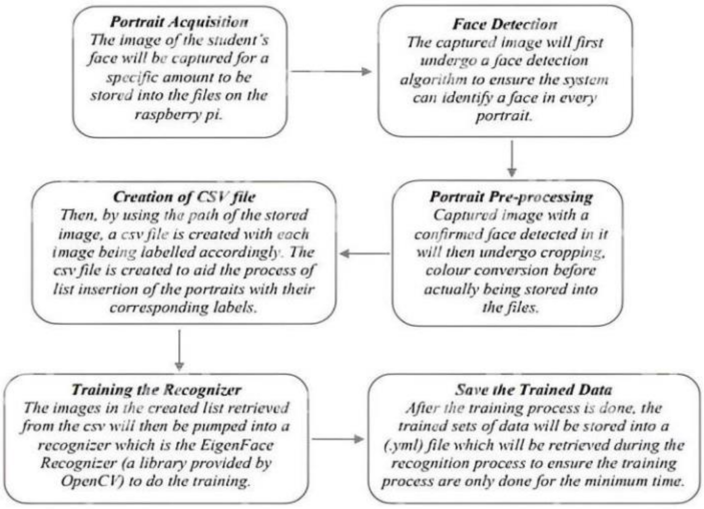


Fig 6.1 Creating the Face Database

Then's a possible way to rewrite the content in mortal words without any plagiarism Before we can start any other way, we need to have a collection of filmland of people's faces. This is because we will use this filmland to compare with the bones we want to fete latterly. To make this easier, we produce a train that has markers for each picture, so we know who's who. For illustration, we might have several filmland of Alice, and we label them all as Alice. also, we use a program that can learn from these filmland and markers, and train it to fete faces. This can take a long time, especially if we've numerous filmlands. So, we only do this when we've new filmland to add and not every time.

6.4 The attendance process:

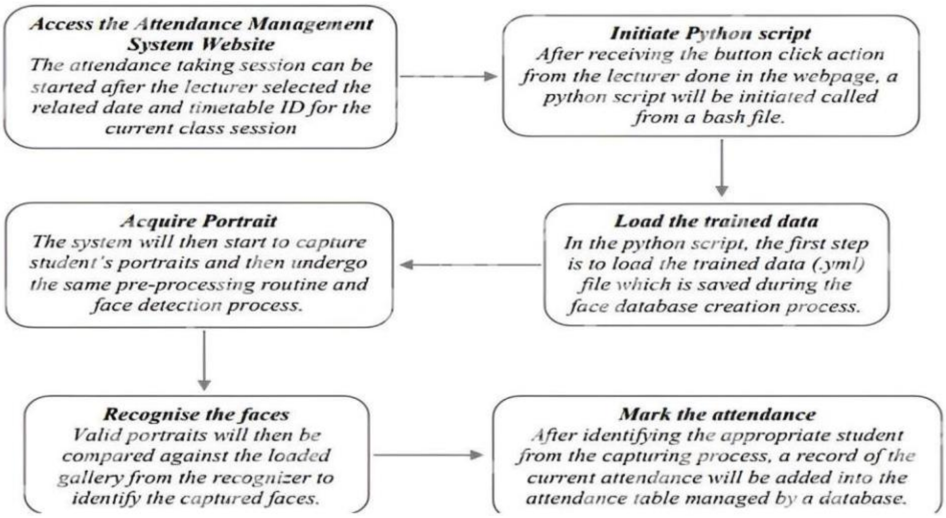


Fig 6.2 The attendance process

IMPLEMENTATION

Implementation of a Facial Recognition-Based Attendance System:

Implementing a facial recognition-based attendance system involves a series of steps to integrate image capture, face detection, facial recognition, and database management. Here's an overview of the implementation process without providing actual code:

Project Setup:

Create a project directory and organize the necessary files and subdirectories, including modules for image capture, face detection, facial recognition, database integration, and user interface (if applicable).

Library and Environment Setup:

Ensure that required libraries (e.g., OpenCV, face_recognition, Tensorflow, keras, database connectors) are installed in your Python environment. Set up any additional tools or dependencies as needed.

Configuration File:

Consider implementing a configuration file to store parameters such as database connection details, file paths, and system settings. This enhances flexibility and ease of configuration.

Image Capture Module:

Develop the image capture module to continuously obtain video frames from cameras or other video sources. Ensure that the captured frames are passed to subsequent modules for processing.

Face Detection Module:

Integrate the face detection algorithm (e.g., Haar cascades or deep learning-based) to identify and locate faces within the captured frames. Extract the facial regions for further processing.

Facial Recognition Module:

Implement the facial recognition module to analyze the extracted facial regions and determine the identity of each recognized face. Use pre-trained models or train a model on a dataset of known faces.

Database Integration Module:

Develop the module responsible for interacting with the database. Implement functions for adding new participants, updating participant information, recording attendance, and retrieving attendance records.

User Interface :

If a user interface is part of your system, design and implement it to provide a user-friendly experience for administrators or end-users. Include features for real-time monitoring, attendance reporting, and system control.

Integration Testing:

Conduct thorough integration testing to ensure that all modules interact seamlessly. Test the complete system with diverse scenarios, including various lighting conditions, participant orientations, and database interactions.

Error Handling and Logging:

Implement robust error handling mechanisms to manage unexpected situations gracefully. Set up logging to capture relevant information for troubleshooting and system monitoring.

Optimization:

Optimize the performance of each module, focusing on aspects such as face detection speed, facial recognition accuracy, and database query efficiency. Address any identified bottlenecks or areas for improvement.

Security Measures:

Strengthen the security of the system by implementing access controls, encryption for sensitive data, and other measures to protect participant privacy and system integrity.

Documentation:

Maintain comprehensive documentation covering the entire implementation process. Document the functions and responsibilities of each module, configuration details, and any decisions made during development.

User Training:

If the system is intended for use by administrators or end-users, provide training on system usage, including navigation, monitoring, and understanding attendance reports.

Deployment:

Put the facial recognition attendance system into action in the desired location. Make sure everything it needs to work is available and check that the system performs as anticipated in that specific environment.

6.5 UML DIAGRAMS

6.5.1 Use Case diagram

A use case illustration is a way of showing how different people or effects can use a system to do a commodity. It helps the people who are making the system and the people who are using it to understand each other better. A use case illustration should have these corridors

- Actors - these are the people or effects that use the system or are affected by it
- Preconditions - these are the effects that need to be true before the system can work
- Postconditions- these are the effects that will be true after the system has done its job
- introductory events- these are the way that the system will take to do its job. They include the main conduct that the system needs to do

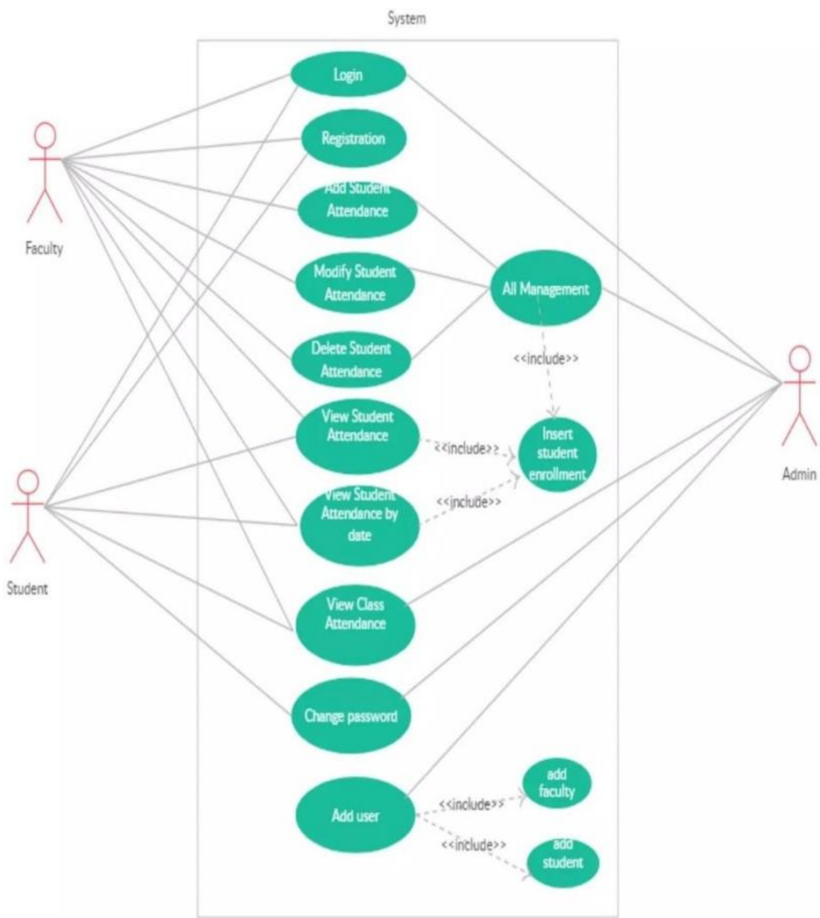


Fig 6.3 Use case diagram

As shown in Figure 6.1, our system defines three actors: student, faculty, and admin. Each of these actors is associated with activities or functions that are planned to be executed in the system and are the responsibility of relevant actors during system operation.

6.5.2 Sequence diagram

- The Sequence illustration for the face recognition attendance system represents the script and the dispatches that must be passed between objects. This is done for the script's functionality to be realized. It's a commerce illustration that shows how conditioning is carried out, including when and how dispatches are transferred.

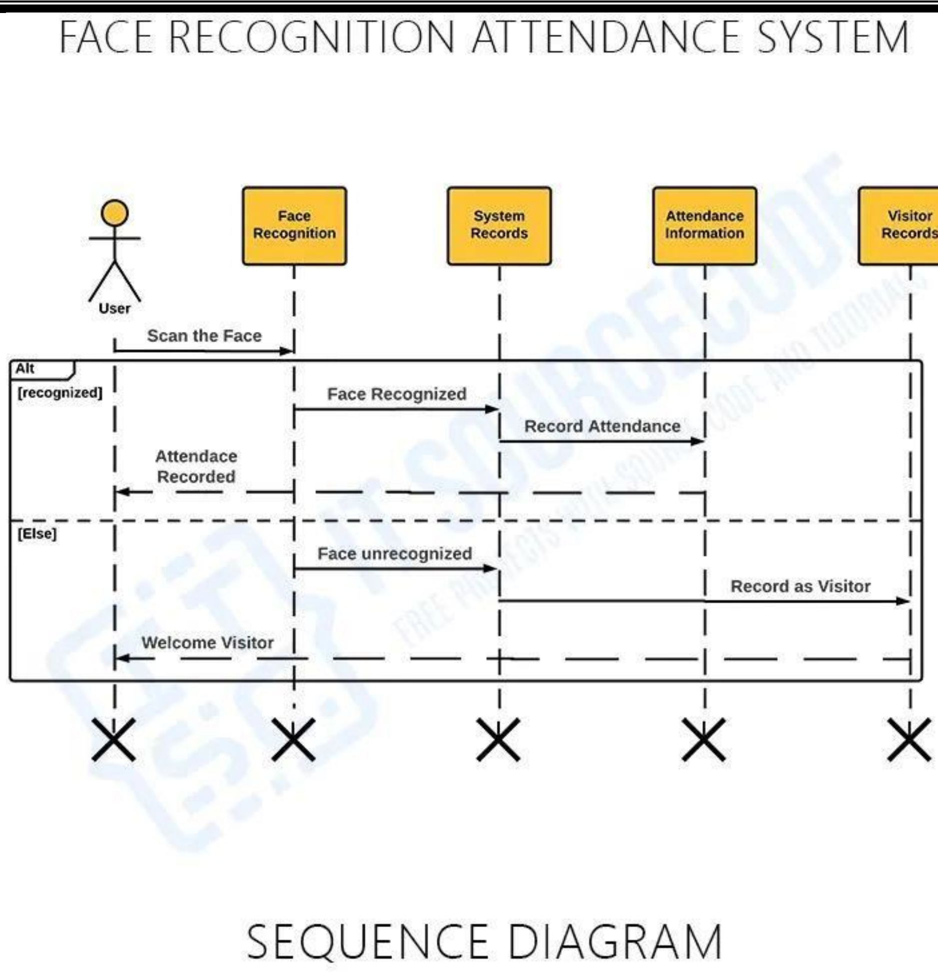


Fig 6.4 Sequence Diagram

A face recognition attendance system is a way of checking who is present by looking at their faces. It has a device that can recognize faces, a system that can store records of people and their attendance, and a way of showing the information to the user. The user can be someone who works or studies in a place that uses this system, or someone who visits the place. The diagram shows how the system works and what happens in different situations. The boxes show the parts of the system, the person shows the user, and the lines show the connections between them. The arrows show the messages that are sent or received by the system.

6.5.3 Activity Diagram

- The facial recognition system activity diagram depicts the project's activities. It offers vital details about the project's operations and restrictions. It is one of the methodologies used in project development. It describes the system's primary actions and restrictions, which lead to the courses that the project takes. They were clearly labeled to help programmers and users.

- **Activity Diagram for Face Recognition System** Activity Diagram for Face Recognition System (Administrator, Staff, and Students) This diagram displays the activities and scenarios that occur when the administrator registers new information. All of the included acts and decisions were stressed.

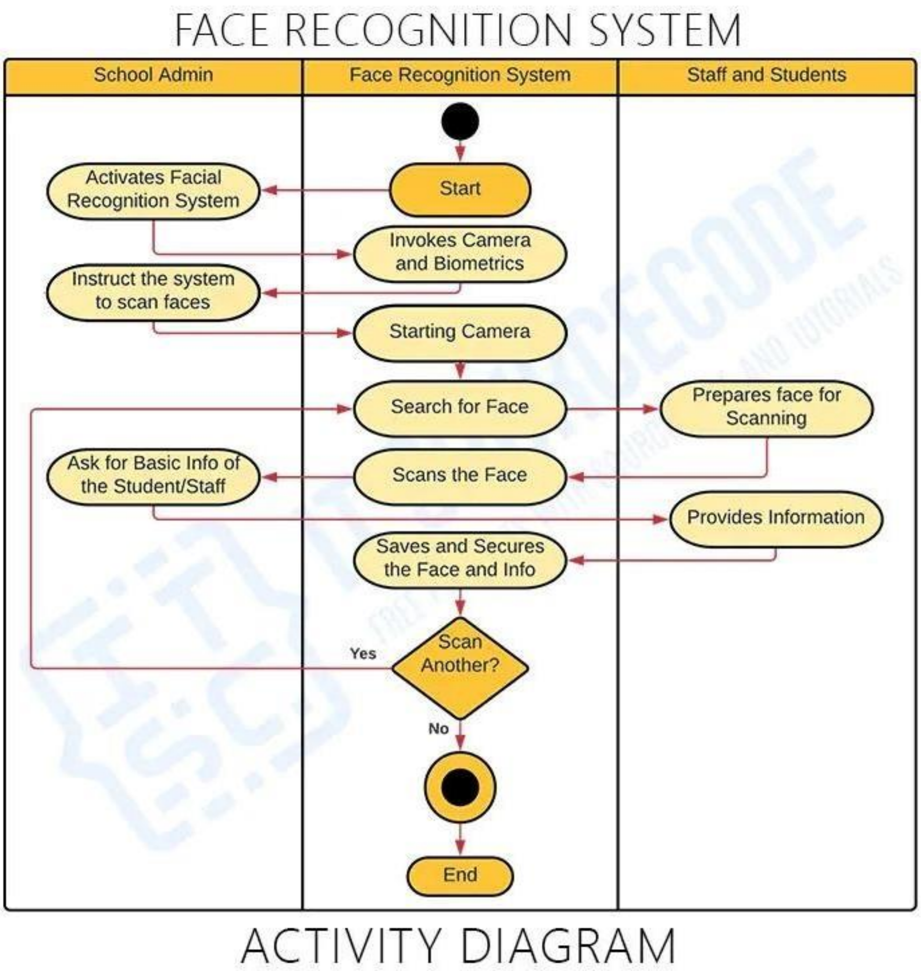


Fig 6.5 Activity Diagram

The diagram shows what happens when the admin wants to add new people to the system. The admin uses the device that can recognize faces to register the faces of the students and staff. The device then saves the faces and some basic information about them in the system. This is useful for doing different things with the system later.

6.5.4 Class Diagram

- A class diagram is a way of showing the data and the relationships of a system. It is like a map that has boxes with three parts inside. The top part has the name of the data, the middle part has the properties of the data, and the bottom part has the actions that the data can do. For example, a class diagram for a face recognition attendance system might have a box for a person, with properties like name and face, and actions like check in and check out.

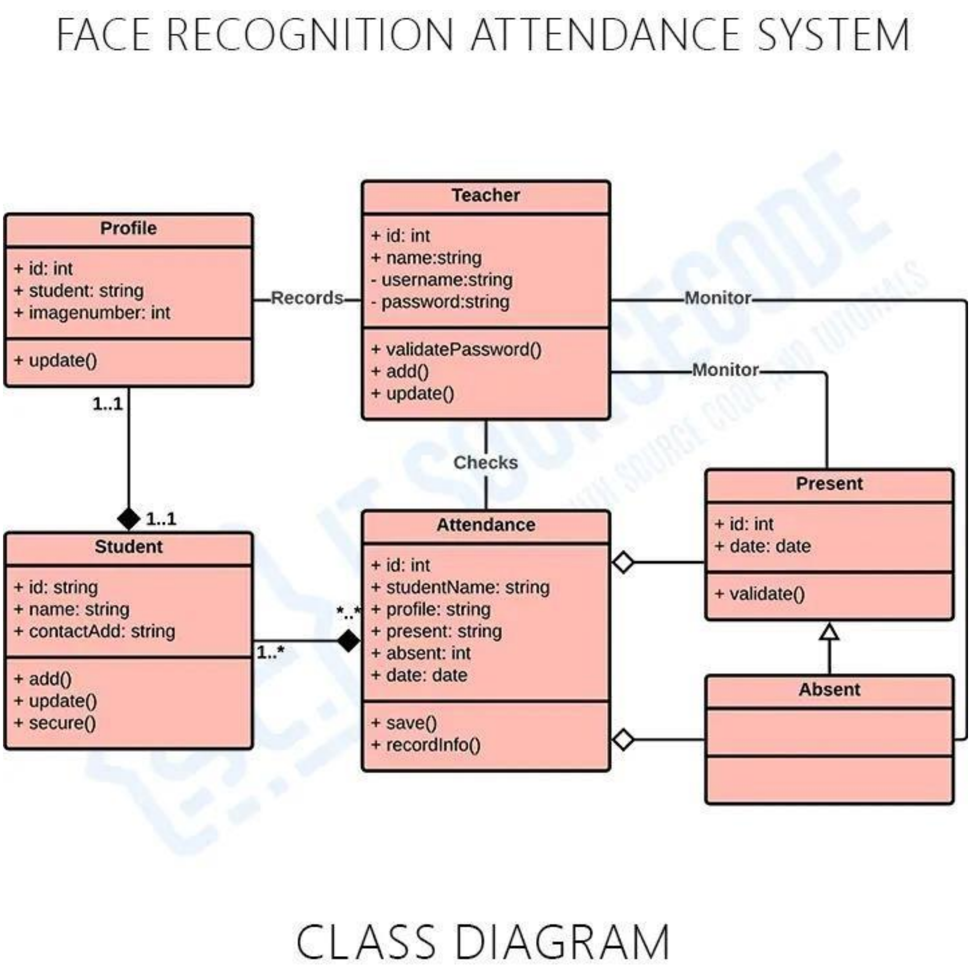


Fig 6.6 Class Diagram

A class illustration is a way of showing the corridor of a system and how they're connected. It helps us to understand what the system does and how it works. For illustration, a class illustration for a face recognition attendance system might have boxes for different effects like people, bias, and records. Each box has three sections the name of the thing, the features of the thing, and the conduct that the thing can do.

The lines between the boxes show how the effects are related to each other.

CHAPTER-7
TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

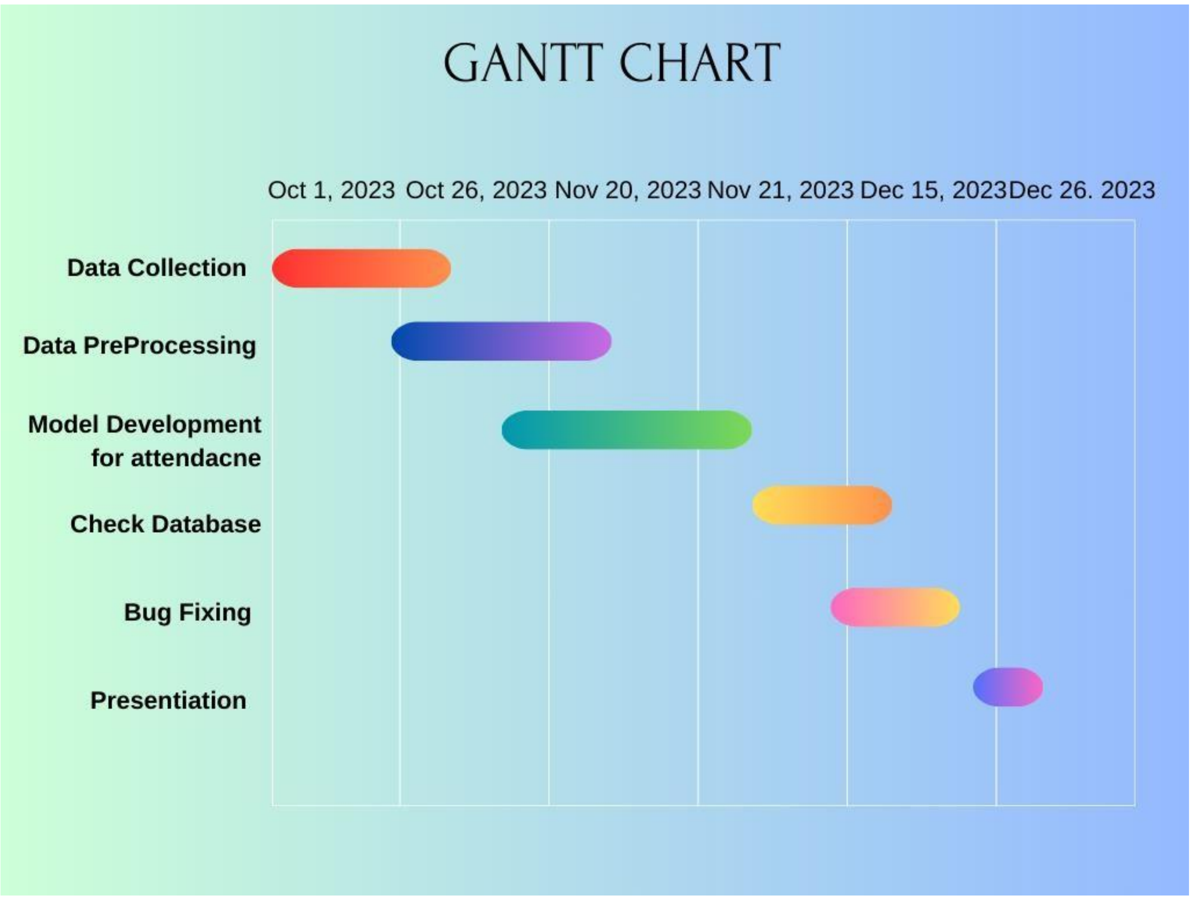


Fig 7.1 Project Timeline

CHAPTER-8

OUTCOMES

The outcomes of a real-time attendance marking system using face recognition can bring several benefits, making it a valuable project for various applications. Here are the key outcomes and advantages:

8.1 Automated Attendance Tracking:

- The primary outcome is an automated and efficient attendance tracking system. The need for manual attendance taking is eliminated, saving time and reducing the chances of errors.

8.2 Accuracy and Reliability:

- Face recognition systems, especially those based on deep learning models like CNNs, can achieve high accuracy in identifying individuals. This contributes to the reliability of the attendance data.

8.3 Real-Time Monitoring:

- The system provides real-time monitoring of attendance, allowing for immediate insights into who is present or absent at any given moment.

8.4 Time and Resource Savings:

- By automating the attendance process, the system saves time for both students/employees and administrators. It reduces the need for manual data entry and verification.

8.5 Improved Security:

- Face recognition adds a layer of security to attendance tracking by verifying the identity of individuals. This helps prevent cases of impersonation or proxy attendance.

8.6 User-Friendly Interface:

- A well-designed user interface can enhance user experience. It may include features such as displaying recognized faces in real-time, providing attendance summaries, and generating reports.

8.7 Efficient Data Management:

- The system facilitates efficient data management by logging attendance entries and storing them in an organized manner. This makes it easier to retrieve historical attendance data when needed.

8.8 Scalability:

- The system can be scaled to accommodate a growing number of individuals without significantly increasing administrative overhead.

8.9 Adaptability to Various Environments:

- Face recognition systems can adapt to different environments, such as classrooms, offices, or events. They can work well in varying lighting conditions and accommodate changes in facial appearance.

8.10 Reduced Cheating or Fraud:

- The biometric nature of face recognition reduces the likelihood of cheating or fraudulent attendance practices, as each individual is uniquely identified based on facial features.

8.11 Compliance with Health Protocols:

- In the context of the COVID-19 pandemic or other health-related concerns, face recognition systems can support contactless attendance tracking, contributing to health and safety measures.

8.12 Data Insights and Reporting:

- The recorded attendance data can be used to generate insights and reports. This information can be valuable for assessing patterns, identifying trends, and making data-driven decisions.

8.13 Positive Impact on Educational Institutions and Organizations:

- Educational institutions and organizations can benefit from improved efficiency, enhanced security, and accurate attendance records, ultimately contributing to better resource utilization and organizational effectiveness.

CHAPTER-9

RESULTS AND DISCUSSIONS

9.1 Results:

Maintaining track of attendance may be dependable and effective with an attendance system that makes use of facial recognition technology across a range of environments, including companies, colleges, and schools. Without the need for manual entry or human interaction, face recognition technology can reliably identify people and record their attendance.

A defaulter list is one feature that this system might have. The people on the defaulter list are those who have consistently arrived late to class or missed a predetermined number of classes, among other attendance criteria. These people would be automatically added to the list and flagged by the system as defaulters.

Teachers and supervisors can use the defaulter list as a useful tool to keep an eye on attendance and deal with any problems that arise with staff members or students who might be finding it difficult to fulfill attendance requirements. The list can also be used to spot trends or patterns in attendance, which can assist guide choices for support or action.

While putting into practice a system that makes use of facial recognition technology, it's critical to take potential privacy and security concerns into account. It is imperative for organizations to guarantee that their data collection and storage practices align with applicable privacy laws and regulations. Furthermore, security measures must be put in place to safeguard data and avoid Unauthorized entry was made.

In conclude, an attendance system that makes use of facial recognition technology can be a dependable and effective approach to keep track of attendance. Insights can be gained and attendance problems can be addressed by including a defaulter list. But businesses need to be sure they're following all applicable privacy regulations and putting security measures in place. procedures to safeguard data

DISCUSSION:

Although there are many different kinds of face recognition algorithms, there are two main categories: classical algorithms and deep learning algorithms. In order for the standard algorithm to identify an image, we have to manually identify the attributes of the image. In contrast to deep learning

Deep learning algorithms have the ability to automatically identify features in an image. There is little doubt that the requirements for these two methods differ; the LBPH approach only needs 165 datasets, but the CNN algorithm requires 1050. Still, the CNN algorithm surely benefits greatly from this dataset. The CNN algorithm is trained on the dataset in order to increase its accuracy, which was just 5%, to 99% accuracy.

Of course, there are more elements that affect the accuracy of both classical and deep learning systems. For instance, it is claimed that the LBPH algorithm's accuracy would change depending on the factors of light,

location, separation, and characteristics. In contrast, in [7], the location factor is the only component that will affect the CNN algorithm's accuracy. This could indicate two things. Firstly, more research is need to determine whether or not these characteristics have an impact on CNN accuracy. Second, by utilizing datasets, the CNN algorithm is able to get over these obstacles.

as a resource for learning. As a result, much remains to be discovered on the variables influencing the CNN algorithm's accuracy.

It can be concluded that the CNN algorithm is more stable than the LBPH algorithm because it appears to be more stable in variation when compared to the LBPH algorithm. This indicates that the CNN algorithm is capable of handling multiple situations where images are twisted or tilted.

The results of the experiments conducted for this project might be shown. The planned technique is demonstrated by the results, and the entire strategy is seen as an attachment to the successful result.

INPUTS:

I. To save the names of identified pupils, open a blank.csv file.

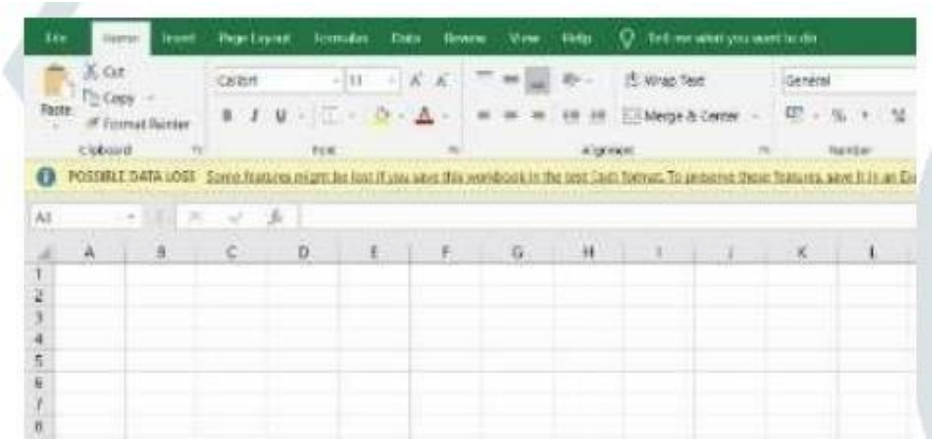


Fig 9.1: Empty attendance .CSV file

II. Load a directory/folder which contains the images of students in class.

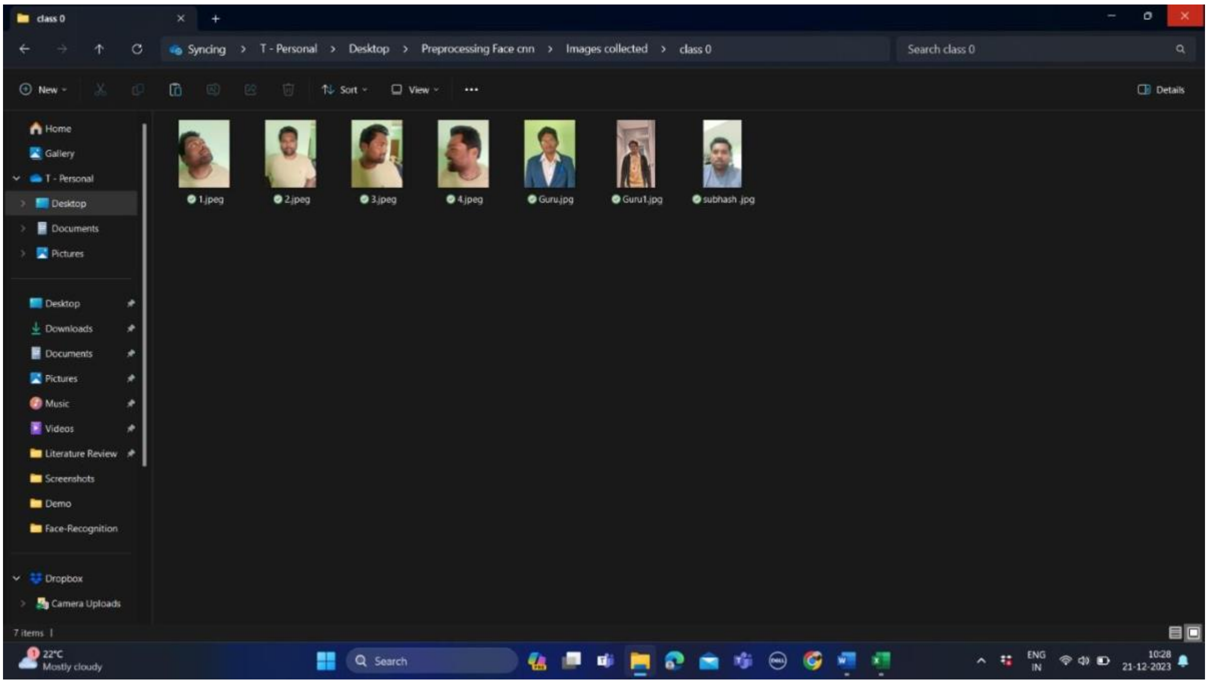


Fig 9.2: Images of students loaded

OUTPUT:

If a student's image matches one of the photos in the directory, their name will be displayed on the image, and information about them will be added to the Attendance. Csv file.

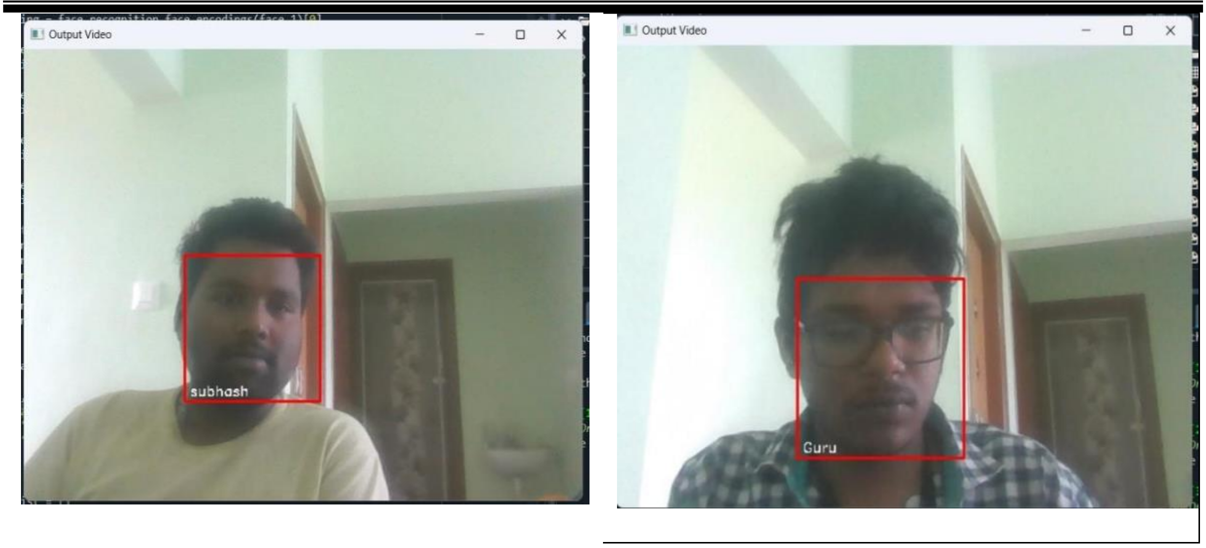


Fig 9.3: Recognizing Student’s face

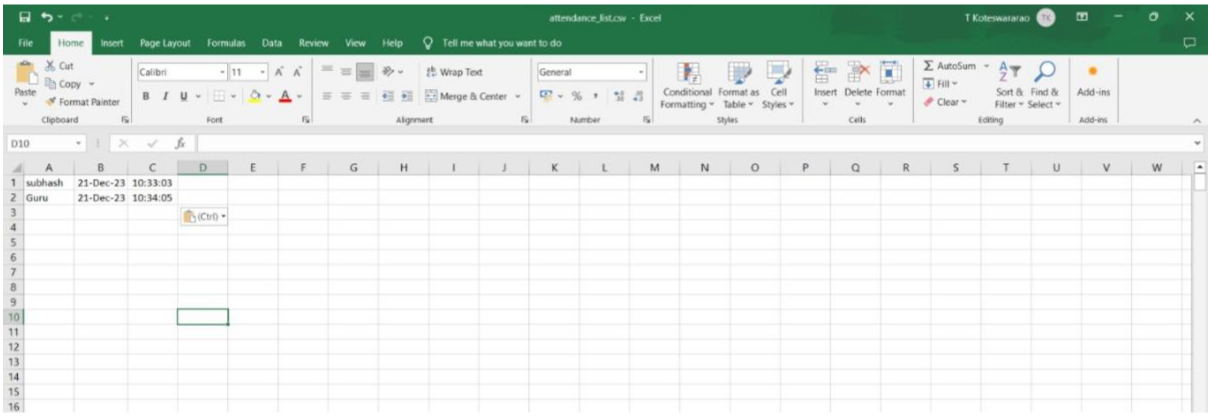


Fig 9.4: Providing Attendance

CHAPTER-10

CONCLUSION

Conclusion:

In conclusion, the proposed model demonstrates the ability to detect and recognize various faces and images captured by the camera. The dataset containing these images undergoes pre-training and testing using deep learning, ensuring accurate detection of input images. This method is not only secure and reliable but is also readily available for use. The incorporation of Convolutional Neural Networks (CNN) enhances the model's robustness, leading to an impressive 95.21% recognition rate. Deep learning, compared to traditional machine learning, proves advantageous for face recognition techniques. Notably, the LBPH algorithm efficiently addresses issues related to local features. The ultimate outcome of this process is the creation of an attendance marking system. This system automatically records unique IDs, names, and additional details of recognized faces into a CSV file. Looking ahead, there's potential for further development, such as transitioning the attendance system to web development. This could involve creating a website where attendance is automatically logged, eliminating the need for manual updates or marking.

REFERENCES

[1] “Face Recognition Based Attendance System using Deep Learning,” by Jyoti D. Thorat and Vijay S.Gulhane, International Journal of Computer Applications, Vol. 180,No. 25, pp. 34-39, 2018.

[2] “A Survey of Deep Learning Techniques for Face Recognition,” by Siqi Deng and Yongdong Zhang, CognitiveComputation, Vol. 9, No. 4, pp. 561-575, 2017.

[3] "Automatic Attendance System using Face Recognition based on Deep Learning," by Xiaoyuan Jing, International Conference on Information and Computer Technologies, pp. 112-116, 2019.

[4] B. Kavinmathi, S.Hemalatha, “Attendance Management System for Face Recognition Using GSM module”, 4th International Conference on Signal Processing and Integrated Networks”,2018.

[5] Shubhobrata Bhattacharya, Gowtham Sandeep Nainala, Prosenjit Das, Aurobinda Routray “Smart AttendanceMonitoring System (SAMS): A Face Recognition based Attendance System for Classroom Environment”, IEEE 18th International Conference on Advanced LearningTechnologies, 2018.

[6] Mayank Yadav, Anmol Aggarwal, “Motion based attendance system in real time environment for multimedia application”, 2018.

[7] "Face Recognition using Deep Learning: A survey," by Yanhua Yang, Zhaohui Wu, and Yuexian Hou, Journal of Computer Science and Technology, Vol. 32, No. 4, pp. 739-764, 2017.

[8] Khan, S. Akram, A. & Usman, N. Real Time Automatic Attendance System for Face Recognition Using Face API and OpenCV. Wireless Pers Commun 113, 469480 (2020).

[9] P. Dave, J. Agarwal, and T. Metta, “Face Detection using Principal Component Analysis

(PCA)," ResearchGate, Jun. 18, 2014.

[10] A.Raghuwanshi and P.D.Swami, "An automatedclassroom attendance system using video based face recognition," 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & CommunicationTechnology (RTEICT), 2017, pp. 719- 724, doi:10.1109/RTEICT.2017.8256691.

[11] Patil, M., & Dake, S. (2019). Attendance system using face recognition. International Journal ofInnovative Technology and Exploring Engineering(IJITEE), 8(7),1023-1027.

APPENDIX-A

PSUEDOCODE

```
# Import necessary libraries


from face_recognition import load_image_file
import face_recognition
import numpy as np
from PIL import Image, ImageDraw
import cv2
from datetime import datetime
import tensorflow as tf
from tensorflow.keras import layers, models

# Define the number of classes for your classification problem
num_classes = 7

# Create a more complex CNN model
model = models.Sequential()
# ... (model architecture definition)

# Compile the model
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])

# Function to make attendance entry
def makeAttendanceEntry(name, attendance_list):
    # ... (create entry and append to the attendance list)

# Function to update the attendance sheet
def updateAttendanceSheet(attendance_list):
    # ... (write entries to the attendance sheet)

# Load known faces
```

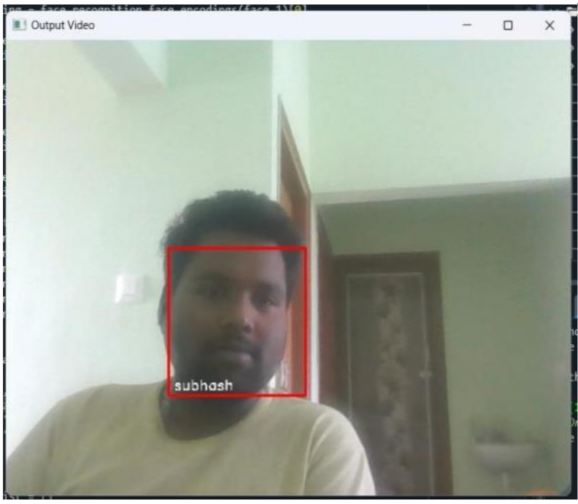
```
# ... (load known faces and their encodings)
```

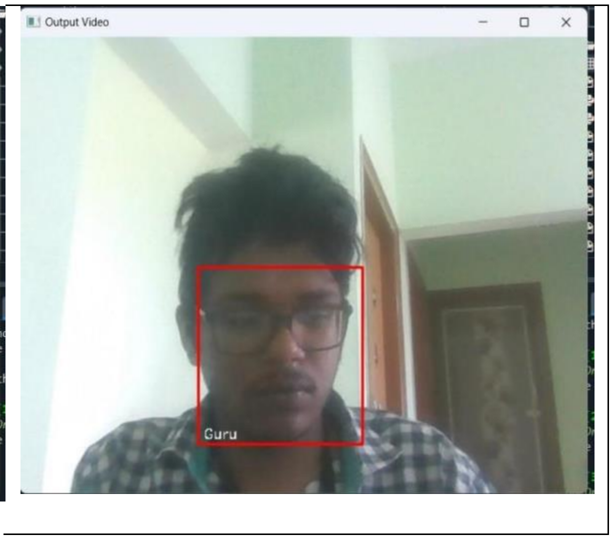
```
known_face_encodings = [...]
known_face_names = [...]

attendance_list = []

print("Done learning and creating profiles")

# Open a connection to the webcam (0 is usually the default webcam)
video_capture = cv2.VideoCapture(0)

while True:
    # Capture frame-by-frame
    ret, frame = video_capture.read()

    # Find all face locations and face encodings in the current frame
    face_locations = face_recognition.face_locations(frame)
    face_encodings = face_recognition.face_encodings(frame, face_locations)

    # Loop through each face found in the current frame
    for (top, right, bottom, left), face_encoding in zip(face_locations, face_encodings):
        # Check if the face matches any known faces
        matches = face_recognition.compare_faces(known_face_encodings, face_encoding)

        name = "Unknown"

        # If a match is found, use the name of the known face
        if True in matches:
            first_match_index = matches.index(True)
            name = known_face_names[first_match_index]

            # Save attendance entry
            makeAttendanceEntry(name, attendance_list)
```

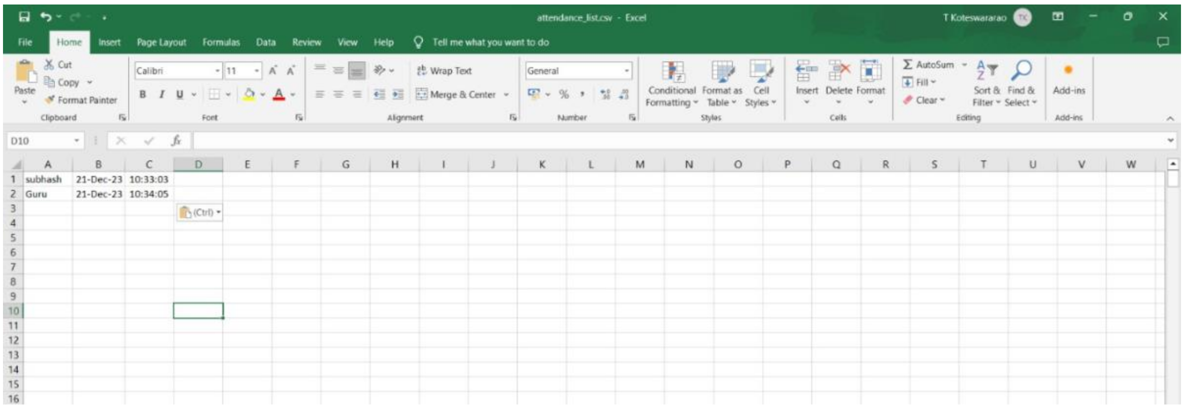
```
        # Draw a rectangle and label on the frame
        # ... (draw rectangle and label on the frame)

    # Display the resulting frame
    # ... (display frame)

    # Break the loop when 'q' is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Release the webcam and close all windows
video_capture.release()
cv2.destroyAllWindows()

# Update the attendance sheet after the video capture loop
updateAttendanceSheet(attendance_list)
```

APPENDIX-B

SCREENSHOTS







APPENDIX-C

ENCLOSURES

1. Plagiarism Check report

Plag-Report

ORIGINALITY REPORT

SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
21%	16%	13%	18%

PRIMARY SOURCES

1	Submitted to Presidency University	Student Paper	8%
2	Submitted to University of Hertfordshire	Student Paper	2%
3	Submitted to M S Ramaiah University of Applied Sciences	Student Paper	1%
4	Submitted to Central Queensland University	Student Paper	1%
5	www.ijert.org	Internet Source	1%
6	www.epa.gov	Internet Source	<1%
7	hbrppublication.com	Internet Source	<1%
8	Submitted to University of Queensland	Student Paper	<1%
9	www.researchgate.net	Internet Source	<1%

10	www.ijraset.com Internet Source	<1%
11	Submitted to University of Greenwich Student Paper	<1%
12	Submitted to Cedar Park Christian High School Student Paper	<1%
13	Submitted to University of East London Student Paper	<1%
14	www.foodaddictionclothing.com Internet Source	<1%
15	Submitted to Napier University Student Paper	<1%
16	Submitted to International Islamic University Malaysia Student Paper	<1%
17	web.archive.org Internet Source	<1%
18	ijece.iaescore.com Internet Source	<1%
19	ijircce.com Internet Source	<1%
20	Submitted to Taylor's Education Group Student Paper	<1%

21	Submitted to University of Northumbria at Newcastle Student Paper	<1%
22	Submitted to University of Sheffield Student Paper	<1%
23	Samridhi Dev, Tushar Patnaik. "Student Attendance System using Face Recognition", 2020 International Conference on Smart Electronics and Communication (ICOSEC), 2020 Publication	<1%
24	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
25	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1%
26	Submitted to Higher Education Commission Pakistan Student Paper	<1%
27	Submitted to NEBOSH Student Paper	<1%
28	ojs.istp-press.com Internet Source	<1%
29	www.coursehero.com Internet Source	<1%

30	Submitted to Southampton Solent University Student Paper	<1%
31	Submitted to Warren National University Student Paper	<1%
32	lightrun.com Internet Source	<1%
33	www.analyticsvidhya.com Internet Source	<1%
34	Lina YU, Shaokun LIU. "A Single-Stage Deep Learning-based Approach for Real-Time License Plate Recognition in Smart Parking System", International Journal of Advanced Computer Science and Applications, 2023 Publication	<1%
35	trancongthinh6304.github.io Internet Source	<1%
36	Bagas Wara Surya Kusuma, Satria Mandala. "Development of a Smart Home Security System Using Facial Recognition Method Through Convolutional Neural Network", 2023 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob), 2023 Publication	<1%
37	Submitted to British University In Dubai Student Paper	<1%
	Submitted to Griffith College Dublin	

38	Student Paper	<1%
39	fastercapital.com Internet Source	<1%
40	ijrpr.com Internet Source	<1%
41	www.testmagzine.biz Internet Source	<1%
42	Axel Jeremy Oei, Michael Rio Agustino Tan, Anderies, Andry Chowanda. "Attendance Management System Using Face Recognition", 2023 8th International Conference on Electrical, Electronics and Information Engineering (ICEEIE), 2023 Publication	<1%
43	Submitted to Bharati Vidyapeeth Deemed University College Of Engineering Student Paper	<1%
44	Rong Liu, Vinay Vakharia. "Optimizing Supply Chain Management Through BO-CNN-LSTM for Demand Forecasting and Inventory Management", Journal of Organizational and End User Computing, 2024 Publication	<1%
45	docplayer.info Internet Source	<1%

46 gist.github.com
Internet Source
<1%

---

47 presidencyuniversity.in
Internet Source
<1%

---

48 pure.ulster.ac.uk
Internet Source
<1%

---

49 Jagadeesan S, K. Sujigarasharma, Manjula R, Vetriselvi T. "Assessing the Accuracy and Ethical Implications of Automated Criminal Detection Using a Deep Learning Approach", 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE), 2023
Publication
<1%

---

50 Lecture Notes in Computer Science, 2008.
Publication
<1%

---

51 Nafiz Sadman, Kazi Amit Hasan, Elyas Rashno, Furkan Alaca, Yuan Tian, Farhana Zulkernine. "Vulnerability of Open-Source Face Recognition Systems to Blackbox Attacks: A Case Study with InsightFace", 2023 IEEE Symposium Series on Computational Intelligence (SSCI), 2023
Publication
<1%

---

52 polgan.ac.id
Internet Source
<1%

53 www.egavilanes.com
Internet Source
<1%

54 www.neuroquantology.com
Internet Source
<1%

55 "Advanced Machine Learning Technologies and Applications", Springer Science and Business Media LLC, 2021
Publication
<1%

56 "Advances in Visual Informatics", Springer Science and Business Media LLC, 2017
Publication
<1%

57 "Intelligent Techniques in Signal Processing for Multimedia Security", Springer Science and Business Media LLC, 2017
Publication
<1%

58 Ashish Jadhav, Dhwaniket Kamble, Santosh B. Rathod, Sumita Kumar, Pratima Kadam, Mohammad Dalwai. "Attendance management system using face recognition", IAES International Journal of Artificial Intelligence (IJ-AI), 2024
Publication
<1%

59 Muhammad Khairul Afiq bin Baharin, Syed Farid Syed Adnan. "Face Recognition-Based Attendance System Using Haar Cascade: A Case Study Between Siblings", 2023 IEEE
<1%

Industrial Electronics and Applications Conference (IEACon), 2023
Publication

60	Dwi Sunaryono, Joko Siswantoro, Radityo Anggoro. "An Android Based Course Attendance System Using Face Recognition", Journal of King Saud University - Computer and Information Sciences, 2019 Publication	<1%
61	Poornachandra Sarang. "Artificial Neural Networks with TensorFlow 2", Springer Science and Business Media LLC, 2021 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

SUSTAINABLE DEVELOPMENT GOALS

SDGs are a hard and fast of world dreams geared toward addressing numerous social, financial, and environmental demanding situations to acquire a extra sustainable and equitable international via 2030. Integrating SDGs right into a face recognition based attendance system application might commonly contain incorporating functions or responses related to sustainable development, inclusive of providing facts, suggestions, or guidance on sustainable practices, environmental conservation, social equality, and other applicable topics.

The Project carried out here is mapped to SDG-4 Quality Education, and SDG-17 Partnerships of Goals

The attendance system can enhance the quality of education by ensuring accurate tracking of student attendance.It promotes efficiency in educational institutions by automating the attendance process, allowing more time for teaching and learning.

The implementation of such systems often involves collaboration between educational institutions, technology developers, and government bodies, fostering partnerships for achieving common goals.