Kasdi Merbah University of OUARGLA

Faculty of New Information and Telecommunication Technologies

Department of: Electronic and Telecommunication

MASTER

Domain: Automation

Field: Automation and system

Submitted by: OULEDMOKHTER MOHAMMED TOUHAMI and OUALID DJELLAKH

Theme:

SMART ATTENDANCE RECORDING SYSTEM (SARS)

Evaluation Date:01/06/2025

Presented To The Jury:

Dr . CHERGUI ABDELHAKIM	MCB	President	UKM Ouagla
Dr . ROUABAH BOUBAKEUR	MCA	Supervisor	UKM Ouagla
Dr .TIDJANI SAYHIA	MCB	Examiner	UKM Ouagla

College year: 2024/2025

ACKNOWLEDGMENT

Above all, we thank Almighty God for giving us courage, will, patience, and health throughout these years, and thanks to whom this work has been possible.

we also wish to express our sincere gratitude to our supervisor, Dr. Roubeh Boubeker, for his follow-up throughout the process of developing this work "thesis", and who never ceased to provide us with advice and feedback.

Heartfelt gratitude to my jury members for their interest in our work. We are grateful to Mr. Ouargli Mohamed Riade for agreeing to review this work.

We dare not forget to thank all the people of the faculty at Kasdi Merbah Ouargla University.

Furthermore, we sincerely thank all those who contributed directly and indirectly to the success of this research, our thanks also go to all those who helped us, whether from near or far

ملخص

يقدم هذا المشروع نظام حضور ذكي يستخدم Raspberry Pi وتقنيات الرؤية الحاسوبية مثل YOLO و Face recognition لأتمتة تتبع حضور الطلاب . من خلال التعرف على وجوه الطلاب في الوقت الفعلي ,يقلل النظام من العمل اليدوي و الأخطاء. يهدف الى تحسين دقة امن عمليات الحضور , مما يساهم في انشاء حرم جامعي متقدم تقنيا

Summary

This project presents a smart attendance system that utilises Raspberry Pi and computer vision technologies such as YOLO and Face recognition to automate student attendance tracking. By recognising students' faces in real-time, the system reduces manual labour and errors. It aims to improve the accuracy and security of attendance processes, contributing to the creation of a technologically advanced campus.

Résumé

Ce projet présente un système de présence intelligent qui utilise Raspberry Pi et des technologies de vision par ordinateur telles que YOLO et la reconnaissance faciale pour automatiser le suivi de la présence des étudiants. En reconnaissant les visages des étudiants en temps réel, le système réduit le travail manuel et les erreurs. Il vise à améliorer la précision et la sécurité des processus de présence, contribuant ainsi à la création d'un campus technologiquement avancé.

LIST OF ABBREVIATIONS

BRIEF: Binary Robust Independent Elementary Features
CNN: Convolutional Neural Network
CPU: Central Processing Unit
CSI: Camera Serial Interface
DCT: Discrete Cosine Transform
DWT: Discrete Wavelet Transform
FPS: Frames Per Second
FREAK: Fast Retina Keypoint
GPU: Graphics Processing Unit
HOG: Histogram of Oriented Gradients
HTTPS: HyperText Transfer Protocol Secure
JPEG: Joint Photographic Experts Group
KNN: K-Nearest Neighbors
LBP: Local Binary Pattern
LAN: Local Area Network
LDA: Linear Discriminant Analysis
NMS: Non-Maximum Suppression
OPEN-CV: Open-source library for real-time computer vision.
PCA: Principal Component Analysis
RTSP: Real-Time Streaming Protocol
SSL: Secure Sockets Layer
SIFT: Scale-Invariant Feature Transform
SURF: Speeded-Up Robust Features
SVM: Support Vector Machine
SARS: Smart Attendance Recording System
SSD: Solid-State Drives
TCP: Transmission Control Protocol
TPU: Tensor Processing Unit
TLS: Transport Layer Security
TP: True Positive
TN: True Negative
FP: False Positive
FN: False Negative
RAM: Random Access Memory
ROM: Read-Only Memory
IoT: Internet of Things
IP: Internet Protocol
YOLO: You Only Look Once

CHAPTER 1: GENERAL INTRODUCTION

1.1 INTRODUCTION

Tracking attendance is an essential administrative function in educational institutions, crucial for assessing student engagement and academic success. Traditionally, attendance has been taken manually, either by calling names or circulating sheets, consuming valuable class time and introducing opportunities for errors, inefficiencies, and even proxy attendance. As class sizes increase and institutions embrace digital transformation, the demand for automated and smart systems becomes increasingly urgent.

This project tackles these issues by implementing a smart attendance system that utilizes facial recognition technology. By harnessing resources like Raspberry Pi, YOLO (You Only Look Once) for object detection, and HOG (Histogram of Oriented Gradients) for recognizing faces, the system can automatically identify students in real-time and register their attendance without any human input. This method greatly minimizes the time and effort required for attendance tracking while maintaining high levels of accuracy and reliability.

The system aims to foster the creation of smart campuses by integrating automation into everyday academic operations. By streamlining attendance management, the system enables educators to dedicate more time to teaching rather than routine administrative tasks, resulting in a more efficient and intelligent learning atmosphere.

1.2 PROBLEMATIC

Length of time and administrative effort

Significant time is consumed by teachers and administrators in the process of manually recording attendance, which reduces the time available for educational and developmental activities.

Susceptibility to human error and manipulation

Relying on verbal summons or signature sheets opens the door to errors (forgetting names, recording wrong attendance) and fraud (proxy attendance).

Increased administrative burdens in large environments

As the number of students and departments increases, it becomes increasingly difficult to manage and organize attendance records manually, overburdening the administrative system.

Lack of accuracy and reliability

Failure to ensure that attendance matches the student's identity negatively impacts the credibility of academic records and increases the risk of cheating and falsifying attendance.

Lack of live data and real-time analytics

Delayed access to attendance reports hinders the ability to make real-time decisions on identifying issues (such as student dropouts or low engagement).

Poor integration with learning management systems

Difficulty in seamlessly linking manual attendance schedules to LMS or academic databases affects the effectiveness and quality of reports.

Challenges in scaling, hybrid and remote learning models

Traditional methods lose their relevance when moving to virtual or blended environments, and do not guarantee reliable monitoring of remote student attendance.

1.3 PROPOSED SOLUTION

Use SARS (Smart Attendance Recording System) : The Smart Attendance Registration System (SARS) serves as an effective solution for managing attendance in corporations and large institutions. SARS is capable of processing a variety of data formats, including photos, videos, and other files, in a structured and efficient manner. Utilizing advanced technologies such as facial recognition, it facilitates automated attendance tracking, thereby conserving time and minimizing errors. This system proves particularly beneficial for environments such as universities and large organizations, where maintaining precise attendance records is crucial. Furthermore, SARS enables seamless sharing of attendance data across departments and ensures secure online storage. Its compatibility with diverse technologies and applicability in both in-person and online contexts make SARS a robust and adaptable tool for contemporary attendance management in the workplace.

1.4 CONSTRUCTION OF SYSTEM

The facial recognition system in this study:

- It is one of the modern technical solutions to automate the process of verifying identity and recording academic attendance.

- It contributes to reducing the phenomenon of cheating and identity theft in classrooms and exam halls.

How the system works:

- Upon activation, the system starts the process of verifying the identity of the responsible professor.

- The professor is asked to enter the pre-trained identification model on the professor's photos already stored in the database.

- The system automatically opens the camera to match the face with the photos using the trained model.

- If the professor is successfully recognized: The program interface opens.

- If not recognized: Access is blocked to maintain system security.

Steps to recognize students:

- After verifying the teacher's identity, the teacher is asked to enter the pre-trained recognition form on the students' photos and list of names in an Excel file.

- The camera is opened to scan the faces in front of it.

- Matching is done using the trained model to automatically record the students' attendance.

- After they are recognized, their attendance is automatically recorded in the attendance list

Dealing with suspicious cases:

- If an unknown face appears: The system puts a colored frame (red) around the face and warns the teacher of the presence of an unknown person.

- The system has the ability to detect forgery attempts (such as using printed photos or cell phones to impersonate another student).

- If a forgery attempt is detected: The system documents the offender's photo for later action.

1.5 PROJECT SCOPE

This project seeks to create an intelligent attendance tracking system that leverages facial recognition and anti-spoofing methods to automate and safeguard the attendance process within educational institutions. The system encompasses several essential features: verifying the identities of faculty members, compiling attendance records for students, logging attendance in real-time, and identifying fraudulent activities.

At the start of each session, the system authenticates the professor's identity through a trained facial recognition model before allowing access to the attendance management interface. For students, the system employs a pre-trained model and references a connected Excel file that holds names and university numbers to correlate biometric data with individual identities.

The system continuously captures a live video feed from a camera. When a recognized student is detected, their attendance and timestamp are logged. If the system identifies an unknown face, it highlights the individual with a red frame to alert the professor.

To bolster security, the system utilizes the YOLO object detection model to detect potential cheating attempts, including printed images or digital photos displayed on phones. Any suspicious behavior activates an alert and automatically records a time-stamped image of the incident in a specific log for future review.

This system is meant to operate locally in a classroom setting, utilizing hardware such as a Raspberry Pi and an attached camera. It is designed for implementation in schools, colleges, universities, and various other educational environments.

Nonetheless, this version does not encompass features like mobile access, cloud-based data storage, remote or online attendance tracking, or integration with external academic management systems. Additionally, advanced biometric functionalities such as emotion analysis or voice recognition are not included in this scope.

The anticipated result is a streamlined, secure, and fully automated attendance system that decreases manual effort, saves time, and helps eliminate common types of attendance fraud.

1.6 CONCLUSION

This chapter highlighted the necessity of implementing an automated attendance system while pointing out the limitations of traditional manual methods. The suggested SARS solution utilizes facial recognition and object detection to enhance both accuracy and efficiency. Additionally, the chapter established the project's framework and scope, paving the way for the upcoming chapter that will delve into the core technologies of Artificial Intelligence and Computer Vision that support the system.

CHAPTER 2: SMART VISION TECHNOLOGIES

2.1 INTRODUCTION

In recent years, the integration of modern technologies into everyday systems has created new possibilities for enhancing efficiency, automation, and decision-making. A key area in this technological transformation is Computer Vision, which empowers machines to analyses and understand visual data. With this ability, systems can identify objects, recognize patterns, and intelligently respond to images and video feeds. One of the primary applications of computer vision is facial recognition, which has swiftly emerged as a crucial tool in fields like security, identity verification, and attendance monitoring. These systems utilize sophisticated image processing methods to accurately and quickly identify and match human faces, rendering them highly beneficial across multiple industries. This chapter offers an in-depth exploration of fundamental concepts in computer vision and how they relate to facial recognition technologies. It analyses current attendance registration systems, the algorithms behind them, and assesses their advantages and limitations. Particular attention is devoted to the algorithm selected for this project, Additionally, the chapter investigates the wider implications of implementing automated facial recognition, discussing not only the technical advantages but also the ethical, financial, and operational challenges that such systems may encounter.

2.2 ARTIFICIAL INTELLIGENCE AND COMPUTER VISION

Artificial Intelligence and Computer Vision Artificial Intelligence (AI) mimics human cognitive abilities in machines, enabling them to learn and make decisions. A prominent area within AI is computer vision, which aims to equip machines with the capability to comprehend and analyse visual information from their surroundings. Recent innovations have shown the impressive outcomes of combining deep learning with computer vision, resulting in enhanced performance in image classification and object detection tasks. For example, one study found that deep learning models achieved remarkable accuracy in medical imaging tasks such as feature extraction and segmentation, particularly in diagnosing chronic low back pain.[1] Another analysis pointed out that the combination of deep learning and computer vision technologies has produced significant advancements, outpacing conventional techniques.[2] Additionally, a comprehensive review of AI applications in diagnosing low back pain indicated that contemporary methods primarily rely on deep learning, achieving high precision in identifying anatomical structures.[1] The continuous progress in this domain is anticipated to improve the reliability and autonomy of computer vision systems, opening doors to innovative applications in various sectors, including healthcare.[2] Ultimately, the fusion of AI and computer vision is poised to transform how machines interpret visual data, leading to groundbreaking advancements in technology and its applications.[3]

2.3 FACIAL RECOGNITION SYSTEMS AND ATTENDANCE REGISTRATION

Facial recognition systems utilize advanced technology to identify and authenticate individuals by examining their facial characteristics. These systems are gaining popularity due to their effectiveness in various applications, such as security, surveillance, and attendance monitoring. Recent studies suggest that incorporating facial recognition technology into attendance systems can significantly enhance both accuracy and efficiency. For example, research has shown that attendance systems using facial recognition can simplify the tracking process, offering a more efficient and user-friendly alternative to conventional methods.[4] Furthermore, the adoption of deep learning techniques has improved the speed and accuracy of facial recognition, making it a viable solution for real-time applications.[5]

2.3.1 Examples of Attendance Registration Systems:

• **Smart Attendance System:** These systems utilize real-time facial recognition technology to monitor student attendance through the analysis of facial features. Recent studies have introduced a new framework that improves the speed and accuracy of recognition by leveraging advanced deep learning methods.[5]

• **Automated Attendance Tracking:** A recent study introduced a method that integrates multiple feature extraction algorithms to automate attendance monitoring by utilizing existing security cameras in educational settings, demonstrating accuracy and efficiency.[6]

• **Mobile Attendance Applications:** A mobile application was created that leverages cutting edge facial recognition technology, resulting in a remarkable accuracy rate and a substantial decrease in the time needed for attendance marking when compared to conventional methods.[7]

• **Web-Based Systems:** There are web-based systems that employ facial recognition and deep learning algorithms to facilitate dynamic attendance processes, demonstrating high accuracy rates in user testing.[8]

2.4 ALGORITHMS USED IN RECOGNITION SYSTEMS WITH ADVANTAGE AND DISADVANTAGE:

Algorithms are crucial in the fields of computer vision, image processing, and machine learning, facilitating functions such as object detection, feature extraction, and pattern recognition. A thorough understanding of their principles and limitations is key to developing effective intelligent systems. This review examines sixteen commonly used algorithms, organized into categories including object detection (YOLO, Viola–Jones), feature descriptors (HOG, LBP), dimensionality reduction techniques (PCA, LDA), signal processing methods (Gabor Filters, DWT), deep learning architectures (CNNs), and classical classifiers (SVM, K-NN), highlighting their advantages and disadvantages.

2.4.1 Object Detection Framework (Viola-Jones, YOLO (Deep Learning-based)):

• **Viola–Jones Detector:** The Viola–Jones framework, introduced in 2001, is an efficient machine-learning approach designed for real-time face detection. It employs Haar-like features and utilizes integral images for quick processing.[9], [10] During the training phase, AdaBoost identifies the most effective features to create weak classifiers, which are organized in a cascading structure to swiftly filter out unlikely areas and concentrate on potential faces. This streamlined architecture allows it to achieve real-time performance, handling about 15 frames per second on early hardware with only grayscale input.[9], [10]

➢ **Advantages:**

✔ **Real-time performance:** The Viola–Jones framework is designed for high-speed performance, employing efficient feature evaluation and a cascade structure to enable real time face detection, achieving approximately 15 frames per second on a 700 MHz CPU.[10] This makes it ideal for fast applications that do not rely on GPU support.

✓ **Simplicity and effectiveness:** The Viola–Jones algorithm employs straightforward Haar features and a clear AdaBoost training method, which makes it both user-friendly and efficient. Its rapid performance and minimal resource demands render it perfect for embedded systems and educational applications.[9]

✓ **Proven in practice:** The Viola–Jones detectors have played a crucial role in vision libraries such as Open-CV and have significantly shaped contemporary detection techniques. Their effectiveness, ease of use, and enduring influence have established them as a standard in the field for more than twenty years.[9]

➢ **Disadvantages:**

✗ **Strict conditions:** The Viola–Jones method performs optimally on frontal, well-illuminated, and unobstructed faces; however, it faces difficulties with variations in pose and lighting, resulting in decreased accuracy under challenging conditions such as side profiles or shadows.[9]

✗ **fixed set of features:** The Viola–Jones framework utilizes fundamental Haar features, which often fail to adequately capture intricate details or complex shapes. As a result, this can result in an increased number of false positives and reduced accuracy when compared to contemporary methods that leverage learned features.[9]

✗ **Cost of scaling and computation:** Although the Viola–Jones method employs a cascade approach to quickly eliminate negative detections, it still analyses multiple image scales, which can slow down processing for large or high-resolution images. In comparison to contemporary deep learning detectors, it is less efficient and continues to be constrained by computational and environmental factors.[9]

• **YOLO (You Only Look Once):** is a real-time object detection technique that approaches detection as a unified task. It converts image pixels into bounding boxes and class labels in a single step.[11]

➢ **Advantages:**

✔ **Real-time performance:** YOLO operates at impressive speeds. The original model achieves 45 frames per second (FPS), while the optimized version runs at 155 FPS, making it ideal for real-time applications.[11]

✔ **Sees the Whole Image:** YOLO examines the whole image both during training and testing, allowing it to learn contextual details and minimize background false positives.[11]

✔ **Excels in New Environments:** YOLO is capable of accurately detecting objects in unfamiliar environments, such as artwork, outperforming several other methods.[11]

➢ **Disadvantages:**

✗ **Has Trouble Locating Small or Close Objects:** YOLO may have difficulty accurately detecting small or close objects due to its grid-based image division.[11]

✗ **Limited by Grid System:** Each grid cell is capable of detecting only a limited number of objects, resulting in challenges for YOLO when trying to identify multiple objects in close proximity.[11]

✗ **Rough characteristics:** Because of several down sampling layers, YOLO may rely on lower-resolution features, which can affect precise localization.[11]

2.4.2 Feature Descriptors (HOG, LBP):

• **HOG (The Histogram of Oriented Gradients):** Is a well-known image feature descriptor that was introduced in 2005. It is utilized to characterize the shape and texture of objects by examining the orientations of edges (gradients) within small sections of an image. The process involves dividing the image into cells, typically measuring 8×8 pixels, calculating gradient histograms for these cells, and normalizing groups of cells (known as blocks) to mitigate the effects of lighting changes.[12], [13] The resulting feature is a composite of these normalized histograms. HOG is extensively applied in applications such as pedestrian and facial detection because it effectively captures local shapes while maintaining resilience against variations in lighting.[13], [14]

➢ **Advantages:**

✓ **Captures Shape and Texture Well:** HOG effectively represents edge directions to capture an object's local shape and texture.[14] It extracts useful gradient information, and when combined with SVM, has shown superior performance in tasks like pedestrian detection compared to older methods like wavelets and SIFT.[13]

✓ **Handles Lighting and Small Changes Well:** HOG uses gradient strength and block normalization to stay effective under changes in brightness and contrast. It also tolerates slight distortions and pose shifts, making it reliable in varied but controlled conditions.[14]

✓ **Demonstrated Effectiveness in Detection Tasks:** HOG has shown remarkable success in areas such as pedestrian and face detection, with Dalal & Triggs attaining nearly flawless results in 2005.[13] Even now, HOG continues to serve as a robust handcrafted benchmark and is occasionally integrated with CNNs to enhance performance.

➢ **Disadvantages:**

✗ **Limited Invariance:** The method does not inherently maintain invariance to scale or rotation, necessitating scans at various scales. It faces challenges when confronted with significant changes in pose, occlusions, or severe deformations.[14]

✗ **Setting Sensitivity:** The performance is significantly influenced by the selection of hyper parameters such as cell size, block size, and normalization.[14] Making suboptimal choices can lead to a considerable decrease in accuracy.

✗ **Surpassed by Deep Learning:** In contemporary computer vision, approaches such as CNNs and Transformers have primarily taken the place of HOG. These models autonomously acquire optimal features from data, consistently demonstrating superior accuracy on intricate tasks.[14] Although HOG is still effective in certain controlled scenarios, it typically falls short in performance on difficult datasets when compared to deep learning techniques.[15]

• **LBP (local Binary pattern):** is a texture descriptor utilized in image analysis, particularly in face recognition and object detection tasks. It evaluates each pixel in relation to its surrounding Neighbors, assigning binary values according to the differences in intensity. These binary patterns are then transformed into decimal values to create a histogram that captures the texture of the image.[16]

➢ **Advantages:**

✓ **Robustness to Illumination modifications:** LBP is invariant to monotonic modifications in illumination, making it powerful in varying light conditions.[16]

✓ **Computational efficiency:** The LBP algorithm is easy and computationally efficient, which allows for real-time programs in facial recognition.[17]

✓ **Powerful Texture analysis:** LBP is adept at taking pictures' texture info, making it appropriate for numerous photographs processing obligations, consisting of object recognition and classification.[18]

➢ **Disadvantages:**

✗ **Sensitivity to Noise:** LBP may be susceptible to noise, potentially leading to faulty texture representations in photos with large noise levels[19]

✗ **Limited Global Understanding:** LBP only looks at small local areas, so it misses the bigger picture, which can reduce its accuracy in complex recognition tasks.[17]

✗ **High Feature Size:** LBP can produce very large feature sets, making computation slower and increasing the risk of over-fitting in machine learning models.[20]

2.4.3 Dimensionality Reduction And Feature Extraction (PCA/Eigenfaces, LDA/Fisher faces):

• **PCA (Principal Component Analysis):** This method is commonly employed to minimize the number of variables in a dataset while retaining the majority of significant information. It identifies new directions, known as principal components, that encapsulate the greatest variance within the data. These components are uncorrelated and arranged based on the amount of variance they account for. By selecting only the most important components, PCA streamlines the data while maintaining its essential patterns.[21]

➢ **Advantages:**

✓ **Efficient Dimensionality Reduction:** PCA is adept at minimizing the number of variables while retaining the majority of the original data's variance and statistical information.[21]

✓ **Enhanced Computational Efficiency:** Analysing data in a reduced-dimensional space can greatly accelerate the performance of subsequent machine learning algorithms.[22]

✓ **Effective Eigenfaces Representation:** The Eigenfaces approach offers a more efficient way to represent facial images in terms of processing time and storage, as opposed to utilizing raw pixel data.[23]

➢ **Disadvantages:**

✗ **Information Loss:** When principal components are discarded, some information is inevitably lost. The degree of this loss is contingent on the variance contributed by the components that are not retained. [21]

✗ **Linearity Assumption:** PCA operates under the assumption that the data's inherent structure can be represented adequately through linear combinations. Consequently, it may not perform well when the data exhibits substantial non-linear relationships.[21]

✗ **Variance Maximization:** PCA focuses on maximizing the total variance present in the data, which may not align with the dimensions that optimally distinguish between different classes in a classification scenario. It overlooks the significance of class labels.[22]

• **LDA (Linear Discriminant Analysis / Fisherfaces):** Also known as Fisher's Linear Discriminant (FLD) or Discriminant Function Analysis (DFA), this supervised dimensionality reduction technique is primarily utilized for multi-class classification tasks. Unlike unsupervised methods such as PCA, LDA leverages class label information explicitly during the process of dimensionality reduction. Its objective is to uncover a lower dimensional subspace composed of linear combinations of the original features that maximizes the separation between different classes. This is achieved by identifying projection directions that increase the distance between class means (between-class scatter, SB) while minimizing the variance within each class (within-class scatter, SW).[24]

➢ **Advantages:**

✓ **Addresses Multicollinearity:** LDA effectively addresses correlations among original features by projecting data into a lower-dimensional space.[24]

✓ **Computational Efficiency:** LDA is relatively straightforward and computationally efficient when compared to more complex non-linear dimensionality reduction or classification techniques.[24]

✓ **Resilience to Lighting:** The Fisher face approach is generally more resilient to changes in lighting conditions than Eigenfaces, in part because it can eliminate dimensions (such as the initial principal components in the PCA+LDA method) that mainly represent variations in illumination.[23]

➢ **Disadvantages:**

✗ **Outlier Sensitivity:** The presence of outliers can greatly influence the computation of class means and scatter matrices, which may result in less effective projections [25]

✗ **Assumption of Linearity:** LDA determines linear decision boundaries and projections, and its effectiveness can diminish if the ideal separation between classes is significantly non linear. To address non-linearity, Kernel LDA extensions are available.[26]

✗ **Complexity:** The Fisher face approach (which combines LDA or PCA with LDA) is typically more computationally intensive to implement compared to the simpler Eigenface (PCA) method.[23]

2.4.4 Signal/Image Processing Techniques (Gabor Filters, DWT, DCT):

• **Gabor Filters:** Gabor filters are commonly utilized linear filters in image processing and computer vision for applications such as texture analysis, edge detection, and feature extraction. They integrate a Gaussian function with a sinusoidal wave, enabling accurate localization in both spatial and frequency domains. Often employed in banks with different scales and orientations, Gabor filters capture a variety of image features, resulting in a representation referred to as "Gabor space".[27]

➢ **Advantage:**

✓ **High Precision:** Gabor filters effectively identify both the locations and frequencies of patterns in an image.[27]

✓ **Adjustable Settings:** They can be tailored to recognize features in particular directions and sizes.[27]

✓ **Reliable Performance:** Gabor filters demonstrate effectiveness even when lighting or contrast fluctuates.[28] Enhanced variations manage noise more effectively while maintaining image details.[29], [30]

➢ **Disadvantages:**

✗ **High Computation Cost:** Using multiple Gabor filters can be slow and requires a lot of processing power.[27]

✗ **Redundant Data:** The filters may generate comparable results, resulting in repeated or overlapping information.[27]

✗ **Excessive Features:** The output may become quite large, necessitating additional measures to minimize the data size.[31]

• **DWT (The Discrete Wavelet Transform):** is a signal processing technique that decomposes data into various frequency components through a constrained set of scales and shifts. In contrast to the Fourier Transform, the Discrete Wavelet Transform (DWT) examines both time (or space) and frequency, providing multi-resolution analysis. It employs low-pass and high-pass filters to differentiate a signal into its approximation (low frequency) and detail (high-frequency) components.[32] This decomposition is carried out iteratively to achieve greater resolution.[33] When applied to images, the DWT is utilized in two dimensions, generating four subbands: LL, LH, HL, and HH. A more resource-efficient variant, known as the lifting scheme, minimizes memory use and processing requirements. [34]

➢ **Advantages:**

✓ **Quick and Efficient:** The Discrete Wavelet Transform (DWT) can be computed rapidly with optimized algorithms, particularly utilizing the lifting scheme, making it ideal for hardware applications while also conserving memory.[34]

✓ **No Blocky Effects:** DWT keeps images smooth and clear, even when compressed a lot, unlike JPEG which can create blocky areas.[35]

✓ **Accurate Signal Recovery:** If designed properly, DWT can perfectly rebuild the original signal from its wavelet coefficients.[36]

➢ **Disadvantages:**

✗ **Absence of Phase Information:** DWT primarily measures the strength of features, but it does not distinctly indicate their position (phase).[37]

✗ **Less Efficient Than DCT:** DWT may require more computational resources than DCT, particularly when using intricate filters or multiple layers.[38]

✗ **Potential Overflow:** The use of certain filter types, such as IIR, may lead to uncontrolled output growth, resulting in potential issues.[39]

• **DCT(Discrete Cosine Transform):** One widely utilized technique in signal and image processing, particularly for lossy compression such as JPEG, is the Discrete Cosine Transform (DCT).[40] This method transforms data into a combination of cosine waves at varying frequencies and is optimized for real numbers, enhancing its efficiency.[41] When applied to images, the 2D DCT processes the data by first analysing the rows and then the columns.[42] In the JPEG format, this transformation is performed on individual 8×8 pixel blocks, with each block compressed independently.[43]

➢ **Advantages:**

✓ **Extensively Utilized:** DCT serves as the foundation for JPEG and is also employed in video formats such as MPEG,[44] ensuring broad compatibility with a majority of systems.[40]

✓ **Good Balance:** It offers a great mix of compression efficiency and reasonable processing cost.[40]

✓ **Only Real Numbers:** It operates exclusively with real values, resulting in greater simplicity and speed compared to transformations like the DFT. [42]

➢ **Disadvantages:**

✗ **Narrow Focus:** DCT examines patterns within small segments, overlooking larger image structures.[45]

✗ **Intentional Loss:** Although the transformation can be lossless, compression typically removes data, particularly finer details.[40]

✗ **Noise Sensitivity:** Compression can lead to the loss or distortion of high-frequency details, making noise more prominent.[42]

2.4.5 Deep Learning Models (CNNs):

• **CNN (Convolutional Neural Network):** It is a deep learning model designed to process input images by assigning significance (through learnable weights and biases) to different features or objects within the image, enabling it to distinguish between them. The preprocessing needed for CNNs is significantly less intensive than that required for traditional classification algorithms. Unlike earlier methods that rely on manually crafted filters, CNNs can automatically learn these filters and characteristics through sufficient training.[46]

➢ **Advantages:**

✓ **Location Flexibility:** CNNs can recognize patterns anywhere in an image, making them less sensitive to where things appear.[47]

✓ **Fewer Parameters:** They use shared weights, so they need less memory and are less likely to overfit.[47]

✓ **Highly Versatile:** CNNs work well not just in image tasks, but also in videos, language, and medical fields.[47]

➢ **Disadvantages:**

✗ **Susceptibility to Adversarial Attacks:** Minor, unnoticeable alterations to input data can cause CNNs to produce inaccurate predictions, which raises security issues in essential areas such as autonomous vehicles.[48]

✗ **Interpretability Challenges:** Due to their intricate structures, CNNs are frequently viewed as "black boxes," making it difficult to comprehend and clarify their decision-making processes.[47]

✗ **Overfitting Risks:** In the absence of appropriate regularization methods, CNNs may overfit the training data, resulting in diminished performance with new, unseen data.[47]

2.4.6 Classical Machine Learning Classifiers (SVM, K-NN):

• **SVM (Support Vector Machines):** are supervised learning algorithms used for classification and regression tasks. They create a hyperplane, or multiple hyperplanes, in a high-dimensional space to facilitate these analyses. The optimal hyperplane is the one that maximizes the margin, or distance, between the two classes.[49]

➢ **Advantages:**

✓ **Great in High Dimensions:** SVMs work well when there are many features but few samples, like in text or gene data.[49]

✓ **Flexible Kernels:** They can use different kernel types to handle complex data patterns.[49]

✓ **Good Generalization:** SVMs are less likely to overfit, especially with high-dimensional data.[49]

➢ **Disadvantages**:

✗ **Needs Careful Tuning:** SVM performance depends heavily on picking the right settings; poor choices can hurt accuracy.[50]

✗ **Slow with Big Data:** Training can be time-consuming and resource-heavy for large datasets.[50]

✗ **Struggles with Noisy Data:** If data is messy or classes overlap, SVMs may make more mistakes.[50]

• **K-NN (K-Nearest Neighbors):** is a non-parametric technique utilized for both classification and regression tasks. In either scenario, the input comprises the k nearest training samples within the feature space. The output varies based on the application: in classification, the result is determined by a majority vote among the neighbors, while in regression, the output is the average value of those neighboring samples.[51]

➢ **Advantages:**

✓ **Easy to Understand:** K-NN is simple and easy to implement, making it ideal for those new to machine learning.[52]

✓ **No Training Required:** As a lazy learning algorithm, K-NN eliminates the need for a training phase, allowing for efficient use of time and computational resources.[52]

✓ **Versatile for Different Tasks:** K-NN can be applied to both classification and regression tasks, offering flexibility across a range of applications.[52]

➢ **Disadvantages:**

✗ **Slow Predictions:** K-NN evaluates every data point when making predictions, which can lead to slowness with large datasets.[52]

✗ **Vulnerable to Irrelevant Information:** Features that are inconsequential or data that is not properly scaled can negatively impact its accuracy due to its dependence on distance measurements.[52]

✗ **Challenges in High-Dimensional Spaces:** In datasets with numerous features, the effectiveness of K-NN diminishes as distance measures become less reliable.[52]

2.5 ALGORITHM USED IN THIS PROJECT

Category	Technique/Model	Library	Usage in Code
Object Detection Frameworks	YOLOv8	Ultralytics	Phone and person detection in run_detection to capture violations during exams.
Feature Descriptors	Histogram of Oriented Gradients (HOG)	Face recognition (dlib)	Face detection in start_camera1 and run_camera for identifying individuals.
Dimensionality Reduction & Feature Extraction	128-Dimensional Face Embeddings		Face encoding in select_folder1 and run_camera for matching faces to known individuals.
Signal & Image Processing Techniques	Color Space Conversion	Opencv-python	Converting BGR to RGB in start_camera1 and run_camera for face recognition compatibility.
	Image Resizing	Pillow	Resizing images for GUI display in create_main_page, create_second_page, etc.
	Circular Cropping		Cropping images to circular shapes for profile pictures in crop_to_circle.
	Bounding Box Drawing & Text Annotation	Opencv-python	Drawing rectangles and labels in start_camera1, run_camera, and run_detection for visualization.
Deep Learning Models	ResNet-based Face Recognition Model	Face recognition	Generating face embeddings for recognition in select_folder1, start_camera1, and run_camera.

Table 1: Algorithm Used in This Project

Each algorithm was chosen not only for its theoretical performance but also for its practical advantages in real-world applications, striking a balance between speed, accuracy, and hardware efficiency to satisfy the requirements of a smart attendance and monitoring system.

2.6 FUNDAMENTAL OPERATION OF EACH ALGORITHM

After carefully investigating and evaluating different types of algorithms typically employed in facial and object recognition, we identified a selection of algorithms that align well with the hardware specifications and performance limitations of this beta version of the application. Our aim was to achieve an optimal balance between accuracy, speed, and resource efficiency, which is particularly crucial for deployment on consumer-grade devices with constrained processing capabilities. In the next section, we will explore the foundational principles and operational functions of the chosen algorithms.

• **YOLOv8:** Analyzes an image using a convolutional neural network (CNN) to simultaneously predict bounding boxes, class probabilities, and objectness scores in a single pass, applying non-maximum suppression (NMS) to enhance detection accuracy.[53]

• **HOG:** Calculates gradient orientations within localized image patches to extract edge and shape details. The image is segmented into small cells, where histograms of gradient directions are generated and normalized across blocks to create a strong feature descriptor. These features are then utilized for object detection (such as faces) by comparing them to a trained model.[13]

• **Face Embeddings (128-Dimensional Vectors):** A pre-trained deep convolutional neural network based on ResNet analyses a facial image to produce a 128-dimensional vector that signifies distinctive facial traits. This network transforms high-dimensional image data into a compact, fixed-size embedding that encapsulates identity-related features.[54]

• **Color Space Conversion:** Transforms pixel values across different color spaces (such as converting BGR to RGB) by applying a mathematical operation to each channel. Specifically, for the BGR to RGB conversion, it exchanges the red and blue channels (transforming [B, G, R] into [R, G, B]).[55]

• **Image Resizing:** Resizes an image to a specified target dimension using interpolation methods, such as Lanczos, to calculate pixel values at new coordinates while maintaining visual quality.[55]

• **ResNet-based Face Recognition Model:** Employs a Residual Network (ResNet) equipped with skip connections to extract hierarchical features from facial images. The network generates a 128-dimensional embedding by processing the image through convolutional layers, pooling layers, and fully connected layers, optimizing the model to reduce the distance between embeddings of images belonging to the same individual.[56]

2.7 OBSTACLES ENCOUNTERED BY AUTOMATED RECOGNITION SYSTEMS

Automated recognition systems, particularly those that utilize facial recognition and object detection, provide significant advantages for surveillance, security, and automation. Nonetheless, they present a range of challenges that need to be tackled to ensure their ethical, effective, and responsible implementation. This chapter examines the key categories of these challenges in real world applications.

• **Cost-Related Challenges:** Setting up a dependable automated recognition system typically necessitates high-resolution cameras, robust computing resources, and sophisticated software, which can lead to significant initial setup and maintenance expenses. Furthermore, there may be extra costs related to staff training, system enhancements, and adhering to legal regulations.[57]

• **Ethical and Privacy Issues:** The implementation of facial recognition technology presents significant concerns regarding individual privacy, particularly in surveillance contexts. There is a potential for misuse or unauthorized monitoring, which could infringe upon personal rights and liberties. Ongoing ethical discussions focus on issues of consent, data storage practices, and the sharing and safeguarding of biometric information.[57]

• **Technical Challenges:** Face recognition systems must contend with a variety of real-world factors that can adversely affect their performance. These factors include changes in lighting, different facial expressions, head angles, obstructions (such as hats or glasses), low image quality, and the effects of aging. Furthermore, the need for real-time processing and large-scale recognition requires the use of efficient algorithms and reliable data management.[57]

2.8 CONCLUSION

This chapter examined the fundamental concepts of AI and computer vision, as well as the essential algorithms employed in facial recognition-based attendance systems. After evaluating their strengths and weaknesses, we identified the most appropriate techniques, such as YOLOv8 and HOG, for implementation in this project. Building upon this theoretical foundation, the next chapter outlines the overall architecture of the system and describes how these technologies are incorporated into the practical functioning of the FRAS system.

CHAPTER 3: SYSTEM STRUCTURE AND MECHANISM OF ACTION

3.1 INTRODUCTION

In this section, we explore the structural design and operational mechanisms of the Smart Attendance Recording System. We begin with a thorough examination of the system’s architecture, pinpointing the essential modules responsible for facial recognition and automated attendance tracking. Key components such as the camera, image processing unit, facial recognition algorithms, and data storage systems are intricately discussed. Furthermore, we analyze the system's workflow, detailing each step from facial detection to automatic attendance logging. We also investigate the integration of software components, user interfaces, and hardware elements, offering a comprehensive view of their collaboration in achieving seamless functionality. This analysis provides valuable insights into the system's technical foundation, emphasizing the efficient processes that facilitate real-time operations and accurate attendance recording.

3.2 SYSTEM STRUCTURE AND MECHANISM OF ACTION

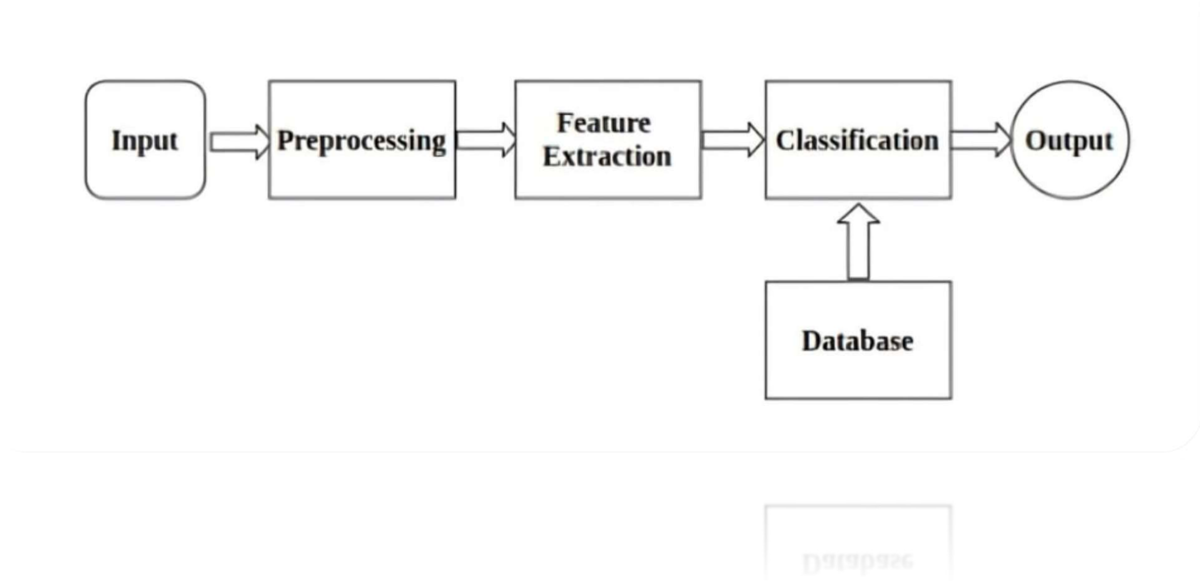

Figure 1: Structure of Recognize System

3.2.1 Equipment:

If we are talking about facial recognition systems in particular, then a webcam or phone cameras are enough to perform the task of collecting data. As for recognition systems in general, we need to classify them by the nature of the system, these are the most important dating equipment:

➢ **Input Devices:**

• **Cameras:** These devices record visual data, including images and videos, to facilitate object, gesture, or motion recognition.[58]

• **Microphones:** These capture sound waves, serving as input for speech or audio recognition systems.[59]

• **Sensors:** Various kinds are utilized, such as infrared, ultrasonic, and biometric sensors, which recognize physical attributes like movement, pressure, and fingerprints.[59]

➢ **Preprocessing and Conditioning Equipment:**

• **Analog-to-Digital Converters (ADC):** These devices transform analog signals (originating from microphones or sensors) into digital data for subsequent processing.[60]

• **Signal Conditioners:** These are utilized to filter, amplify, or modify raw signals prior to digitization, enhancing the quality of the input.[61]

• **Lighting Systems (for visual recognition):** These systems provide uniform lighting to minimize errors due to shadows or glare.[62]

➢ **Computing Units:**

• **CPU (Central Processing Unit):** A versatile processor designed for executing logical operations and managing basic recognition models.[63]

• **GPU (Graphics Processing Unit):** Tailored for parallel processing, crucial for deep learning recognition systems because of their significant computational requirements.[63]

• **TPU (Tensor Processing Unit):** Created by Google, this unit is specifically optimized for computations related to TensorFlow-based neural networks. Embedded.[63]

➢ **Memory and Storage:**

• **RAM (Random Access Memory):** This is a temporary memory utilized for processing real time input data.[64]

• **ROM / Flash Memory:** Employed in embedded systems for the storage of models and firmware.[64]

• **Solid-State Drives (SSD):** Provides rapid access storage for pretrained models, recognition logs, and recorded inputs.[64]

➢ **Networking and Communication Components:**

• **Ethernet / Wi-Fi Modules:** Allow data exchange between input devices and processors, or with remote systems.[65]

• **IoT Gateways:** Consolidate data from dispersed sensors and relay it to recognition modules. [65]

➢ **Energy and Cooling Systems:**

• **Power Supply Units (PSU):** Deliver consistent power to all hardware elements.[66]

• **Batteries (for portable systems):** Guarantee continuous operation in mobile or remote settings.[67]

• **Cooling Systems (Fans, Heatsinks, Liquid Coolers):** Prevent overheating of CPUs, GPUs, and embedded processors, particularly in high-performance systems.[68]

The equipment list in general recognition systems is organizer based on the functional roles that each component serves within the recognition pipeline.

3.2.2 Database:

The database within the recognition system serves as a central repository for reference data, where various types of data such as images, sound, and text are stored.... etc. at this point, we will discuss the process of collecting data for the database.

• **Define the Purpose and Scope:** Start by clearly defining the goals of your recognition system. Specify the particular functions it will carry out, such as facial recognition, object detection, or speech recognition. Set the boundaries by identifying the types of data needed, the number of classes or categories to include, and the variety required in your dataset.[69]

• **Choose Input Methods:** Select the relevant data types (modalities) your system will handle, such as images, audio, video, or sensor data. Ensure that your choices are consistent with the system's goals and the specific requirements of the recognition task.[70]

• **Gather Unprocessed Data:** Collect raw data that accurately represents the diversity and variability typically found in real-world situations. For example, in the context of face recognition, gather images under various lighting conditions, angles, and expressions to guarantee robustness.[71]

• **Data Classification:** Precisely label your gathered data to support supervised learning. Utilize annotation tools to highlight important features or areas within your dataset, ensuring both consistency and accuracy throughout.[72]

• **Prepare the Data:** Prepare and refine your data to improve its quality and appropriateness for training. This process may include normalization, noise reduction, resizing, or converting formats, depending on the type of data.[73]

• **Arrange and Archive the Database:** Arrange your dataset in a systematic manner to ensure easy access and effective management. Sort the data into directories according to classes or categories, and uphold consistent naming conventions.[74]

• **Safeguard Quality, Privacy, and Ethical Standards:** Establish procedures to uphold data quality and comply with ethical guidelines. This involves securing informed consent, anonymizing sensitive data, and prioritizing data security.[75]

• **Divide the Data for Utilization:** Split your dataset into training, validation, and testing subsets. This division enables efficient model training, hyper-parameter optimization, and performance assessment.[70]

3.3 WORKING OF ALL ATTENDANCE REGISTRATION SYSTEMS USING FACIAL RECOGNITION

The automated attendance system employs advanced real-time facial recognition technology to simplify the student attendance recording process. By integrating OpenCV, Haar Cascade classifiers for facial detection, and the LBPH algorithm for recognition, the system accurately identifies individuals through their facial characteristics. This innovation removes the necessity for manual attendance recording, boosts security, and enhances efficiency by automatically updating an attendance database with recognized faces. Furthermore, the system is tailored to operate efficiently in actual classroom settings with low hardware demands.[76]

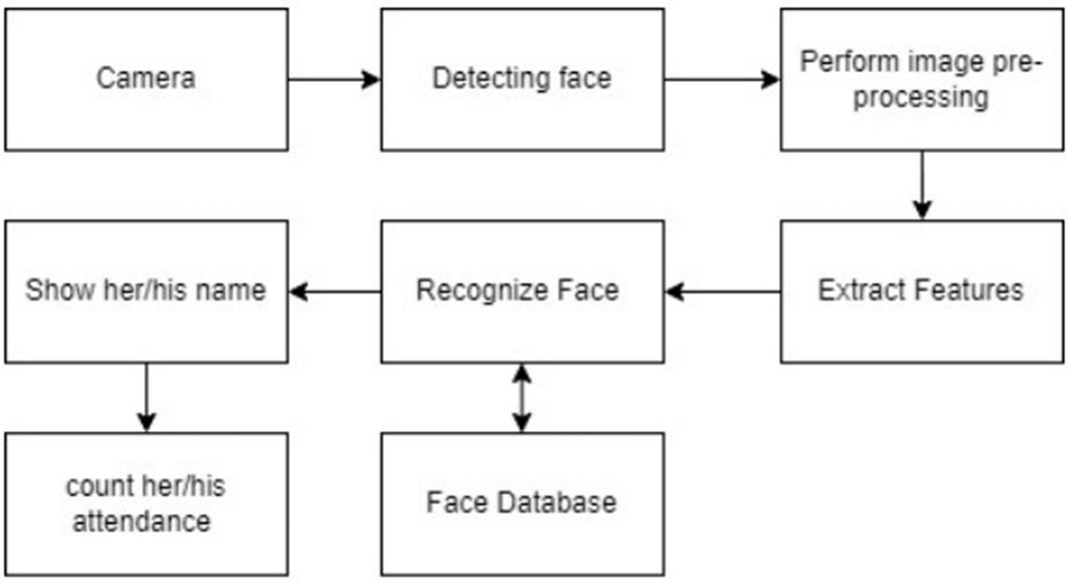

Figure 2: Block Diagram of Real-time Face Recognition Attendance System [76]

➢ **Overview of the System:**

The suggested automated attendance system utilizes computer vision technology to effectively track student attendance within a classroom environment. Its architecture consists of a web camera and a personal computer; the camera captures live facial images, while the computer runs face detection and recognition algorithms. The system is developed using PyCharm and leverages the OpenCV library for implementation. For face detection, it employs the Haar cascade classifier, whereas the Local Binary Pattern Histogram (LBPH) algorithm is utilized for face recognition, known for its effectiveness in diverse lighting situations and different facial angles. The system functions continuously, observing students and automatically logging attendance when identification is successful.[76]

➢ **Face Detection:**

Face detection is performed utilizing the Haar cascade classifier, a reliable machine learning technique designed to recognize human facial features in grayscale images. Initially, each frame captured by the camera is converted to grayscale prior to being processed by the classifier. The classifier examines the image and locates areas that correspond to human faces by aligning with predefined Haar features. Once a face is successfully detected, the facial region is marked and sent to the recognition module for further processing. This approach is selected for its efficiency and effectiveness in real-time detection applications.[76]

➢ **Face Recognition:**

The system employs the Local Binary Pattern Histogram (LBPH) algorithm for facial recognition. This method works by examining the local texture of facial images. For each pixel, the algorithm compares it to its surrounding pixels to create a binary pattern. These patterns are then used to generate a histogram that captures the distinct texture of a face. Subsequently, these histograms are compared to pre-existing ones in the database to identify individuals. If the distance between the computed histograms is below a specified threshold, the system verifies the identity of the person. LBPH is chosen for its resilience to changes in lighting, variations in facial expressions, and its ability to operate effectively in real-world conditions.[76]

➢ **Attendance Log:**

After a face is successfully identified, the system efficiently logs the attendance of the recognized individual. This attendance is stored in a well-organized database that keeps track of identified faces along with their respective timestamps. The system records student attendance at two key points: at the start and at the conclusion of the class session, ensuring reliable monitoring of student presence throughout the lecture. This automation minimizes the manual workload for instructors and improves the accuracy of attendance tracking.[76]

➢ **Dealing with Unfamiliar Faces:**

When a detected face does not correspond to any records in the trained database, the system labels it as "Unknown." These unidentified facial images can be stored separately for later examination or for potential inclusion in future training datasets. This feature enhances system security by identifying unauthorized individuals or students who have not yet registered, while also aiding in the ongoing refinement of the recognition database.[76]

➢ **Assessment of Performance:**

The performance of the face recognition system is assessed through a confusion matrix. In an experiment with 20 facial images, the system successfully identified 8 trained faces while correctly rejecting 8 nontrained faces. However, it did have 2 false positives and 2 false negatives. This assessment reflects a commendable level of accuracy, particularly given the real-time requirements of the application and the use of limited hardware resources. The effective combination of the LBPH algorithm and Haar cascade detection demonstrates its capability to provide a reliable and efficient attendance solution for educational institutions.[76]

3.4 CONCLUSION

This chapter explored the essential framework and workflow of the attendance system utilizing facial recognition technology, highlighting its main hardware components, database architecture, and operational logic. With this foundational setup established, the following chapter will focus on the SARS system, detailing's its implementation, features, and real-time functionalities that transform the theoretical framework into practical use.

CHAPTER 4: ATTENDANCE SYSTEM

4.1 INTRODUCTION

In response to the increasing trend of digital transformation within educational institutions, the Smart Attendance Recording System (SARS) offers an intelligent and innovative solution for managing student attendance through artificial intelligence (AI), specifically facial recognition technology. SARS is designed to overcome the common challenges faced by educators during traditional attendance-taking, particularly in large lecture halls where manual methods are time-consuming and inefficient. Issues such as wasted instructional time, inaccuracy, cheating through proxy attendance, student distraction, and the disorganization of paper records all hinder the learning environment. SARS addresses these problems by providing a fast, accurate, and automated system that records attendance only when students are physically present in front of the camera, thus eliminating the possibility of cheating. The system securely stores data in Excel files on the teacher’s computer, ensuring easy access and organization. Its sophisticated design also enables it to distinguish between authorized students and unauthorized individuals. By streamlining the attendance process, SARS allows teachers to concentrate on delivering lessons, thereby enhancing the overall quality and efficiency of education.

4.2 THE MECHANISM OF ACTION OF THE SYSTEM SARS

When the program is started, a window appears asking the user (professor) to select the form that has been trained on the images of previously registered professors. This action aims to verify the user's identity and prevent any unauthorized use of the system.

4.2.1 Verification of the identity of the professor:

After selecting the model trained on the professors photos, the system automatically activates the camera and the model starts recognizing the professor If the professor is successfully selected, the program opens and provides access to manage attendance records. If the verification fails, access is blocked and an alert message is displayed to protect the data and ensure the security of the system.

offers flexibility, as the face dictionary can be readily updated by incorporating new encoding or recalculating averages, eliminating the need to retrain the entire deep learning network.

ƒ **Face encoding:**

Face coding is the process of converting a numerical vector resulting from the feature Extraction stage into a static structure used for comparison and classification. In our application, this process is represented by the following points:

• Reception and verification of data

The encoding extraction function is executed right after the face has been cropped from the frame. It initially verifies the presence of a pre-detected face by utilizing face locations, ensuring that the input is valid and includes at least one face.

• Numerical vector generation (Embedding)

The facial image is input into a pre-trained deep learning model from the face recognition library built on dlib. This model analyzes the image and generates a facial embedding, which is a vector of fixed size (usually 128). Each component of this embedding captures a unique numerical attribute, representing a specific facet of the individual's facial features.

• Normalization (Normalization) L2 normalization is typically performed on the output vector from the network by dividing it by ^. This procedure standardizes the scale of all vectors, facilitating more consistent and straightforward comparisons through distance or similarity measures.

• Representation and structure

Every face is converted into a unique numerical representation known as a vector, which is then stored in the system alongside the individual's name. When multiple images of the same person are captured, the system merges these vectors and computes an average. This process results in a more precise and consistent depiction of the person's face, minimizing inaccuracies due to variations in lighting, angle, or expression.

• Use in recognition

When a new face is presented, its encoding is extracted in the same format as the stored data and stored in temporary variables. These new vectors are subsequently compared to the existing ones using Euclidean distance. This distance is then transformed into a similarity ratio, with a threshold of 0.6 applied to determine whether the face is recognized or deemed unknown.

• Scientific advantages

Face encoding provides several significant benefits that enhance the system's overall performance. It increases stability through normalization and averaging methods, which minimize sensitivity to varying lighting conditions and camera angles. By transforming facial data into compact 128-dimensional vectors, face encoding also boosts storage and comparison efficiency, making it much quicker and smaller than working with complete pixel images. Furthermore, it enables straightforward scalability, as adding new faces is as simple as storing their encoded vectors without the need to retrain the deep learning model. In summary, face encoding is crucial for ensuring speed and accuracy throughout all phases of the system: recording, recognition, and evaluation.

ƒ **Face Comparison:**

The process of face comparison is carried out after the completion of the stages of sampling, extraction and coding of faces, with the aim of identifying the new face based on the stored encoding. The current application is based on the following methodology:

• Preparation of encoding

The system keeps an encoding dictionary that associates each individual's name (as the key) with a distinct, precomputed face encoding vector (as the value). Before any comparisons are conducted, a new face encoding is generated from the input image using the same method used during the initial recording. This process involves cropping the face, extracting the feature vector with the Deep model, and normalizing the results to maintain consistency and accuracy in matching.

• Calculation of Euclidean distances

For every recognized encoding, the system computes the Euclidean distance between the new face vector and the stored face vector:

2‖knownv−newv‖=d

This distance acts as an inverse measure of similarity; the closer the value is to zero, the higher the resemblance between the two faces.

• Converting distance to similarity ratio

To improve interpret ability, the distance is converted into a relative similarity score ranging from 0 to 1 using the equation:

d−1=similarity

This normalized value simplifies the assessment of how closely a match adheres to a predefined threshold.

• Choosing the best match

All similarity ratios are gathered in a data structure that is sorted in descending order by value. The top pair, which includes the individual's name and the associated similarity ratio, is identified as the best match.

• Threshold verification (Threshholding)

The model evaluates the highest similarity ratio against a defined threshold of 0.6. If the similarity ratio meets or exceeds this threshold, the individual is recognized as known, and their name, along with a confidence ratio, is presented. On the other hand, if the similarity ratio falls below 0.6, the face is categorized as "unknown" to ensure accurate recognition and reduce the chances of false positives

• View the result

The names of the recognized individuals, particularly when multiple faces are detected in the frame, are paired with their respective confidence scores. This information is then presented in a user-friendly format, such as: "Identified: Ahmed (82%), Leila (75%)".

• Performance characteristics

Calculating distances between 128-dimensional vectors is quick and efficient, allowing for almost immediate responses. The system is also easily extensible; new faces can be incorporated simply by saving their encoding, eliminating the need to retrain the model or alter the algorithm. Additionally, the recognition threshold can be adjusted flexibly to cater to specific accuracy or security needs, raising it to decrease false positives or lowering it to enhance sensitivity

4.4 DATASET OF SARS

The "smart attendance registration system" dataset serves as a visual resource for developing and training a facial recognition model as part of the smart attendance project at the University. It comprises facial images of seven students, with 40 photographs per student taken under varied conditions, including different angles, facial expressions, and lighting scenarios, to improve the model's generalization and recognition capabilities. All images are stored in JPG format and were captured using a high-resolution camera in a controlled on-campus setting to maintain data quality and consistency. This dataset is intended solely for academic use, with consent obtained from all participants, while fully prioritizing their privacy and rights. The data is divided into two sets: 80%

for training and 20% for testing, enabling an accurate and objective evaluation of the model's performance.

4.5 EXPERIMENTAL METRICS:

The performance of the face recognition model within the intelligent attendance registration system was evaluated using a set of quantitative experimental metrics that reflect the effectiveness of the model in accurately recognizing students ' faces. The following have been adopted:

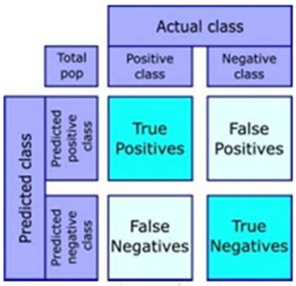

Figure 9: The confusion matrix

The confusion matrix is an essential tool for evaluating the performance of classification models in machine learning, by comparing the prediction results with real values. It consists of four cases:

TP: stands for the number of samples that were correctly predicted to be positive.

FP: stands for the number of samples that were falsely pre ducted to be positive.

FN: stands for the number of samples that were falsely predicted to be negative.

TN: stands for the number of samples that were correctly predicted to be negative

Recall is defined as the percentage of correctly predicted disease-affected leaves relative to the total positive instances of the test case.[77]

$$\frac{\quad}{+} =$$

Precision The ratio of correctly predicted disease-affected leaves to all positively predicted leaves by the model is known as precision and can be defined as follows.[78]

$$\frac{\quad}{+} =$$

F1-score is a metric that combines precision and recall in a single metric.[79]

$$\frac{\times}{+} \times 2 = \quad 1$$

An artificial intelligence model was trained using a set of images with the aim of carrying out the task of identifying students, and has undergone a comprehensive evaluation process to measure its performance on test data. This assessment includes a set of indicators and criteria such as accuracy, error rate, as well as graphs showing the evolution of the model's performance during various training periods. These results help to understand the effectiveness of the model and identify aspects that need to be improved or adjusted.

Class	Precision	Recall	F1-Score	Support	Loss
abdeldjalil	1	1	1	5	0
cherif	1	0.71	0.83	7	0.000706
imad	1	1	1	10	0
mohamad	1	1	1	9	0
momn	1	1	1	6	0
oualid	0.67	1	0.8	2	0
salah	0.8	1	0.89	4	0

Table 2: Evaluation of the model table

➢ **Accuracy**

The model demonstrated outstanding performance across most categories, with the three metrics Precision, Recall, and F1-Score reaching the perfect score of 1.0 for the majority of students, including Abdeljalil, Imad, Mohammad, and Momn. This highlights the model's strong capability to accurately classify faces without errors. However, some variances in performance were observed in certain categories. For example, Cherif achieved flawless precision (1.0), but the recall fell to 0.71, suggesting that not all images in this category were identified, resulting in an F1 score of 0.83. Oualid exhibited a notably lower accuracy of 0.67, indicating the presence of false positives in this category. In Salah's case, while precision slightly decreased to 0.8, recall remained high at 1.0, signifying that all images of this category were accurately recognized, though some incorrect images were also misclassified as belonging to this category.

➢ **Support**

This indicates the number of images utilized for assessing each category. We observe a variation in the number of photos across different categories, which could impact the accuracy of the evaluations. For instance, Imad was assessed using 10 images, whereas Oualid and Salah were evaluated with only two or four images each; this disparity may account for the inconsistencies in their performance metrics.

➢ **Loss**

The loss column shows that the majority of categories recorded a loss value of zero (0), indicating the model's success in matching predicted outputs with actual results. The sole exception is the cherif category, which registered a minimal loss of 0.000706, reinforcing the notion that the model exhibits a strong overall fit.

The results indicate that the model has excellent accuracy in recognizing the majority of categories, which confirms its effectiveness in a semi-realistic environment. However, uneven performance stands out in some cases, which may be attributed to:

The small number of samples for some categories.

The similarity of faces among some students.

Lighting or different capture angles

4.5.1 Analysis of the confusion matrix

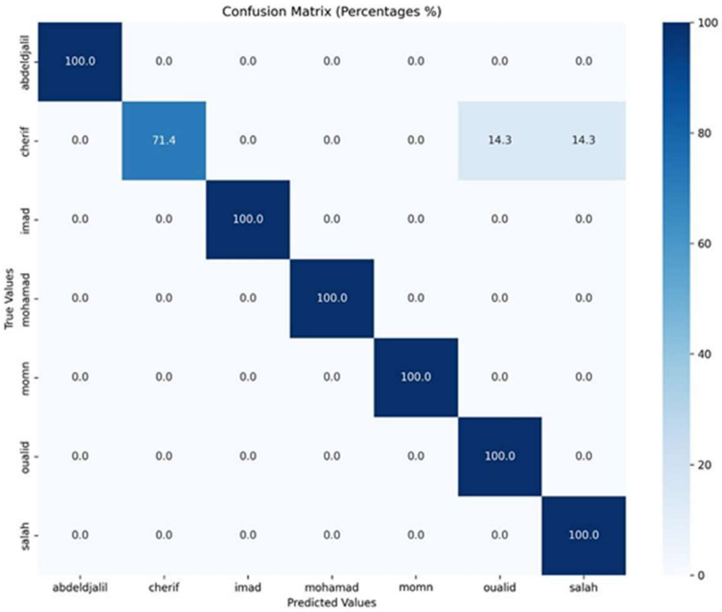

Figure 10: confusion matrix of our data

● **Performance of ideal classes**

The model classified the following categories with 100% accuracy, without any prediction errors: Abdel-Jalil, Imad, Mohamad, Momn, Oualid, and Salah. This indicates that the model was able to correctly identify all images belonging to these categories, demonstrating its high efficiency and effectiveness in handling these specific cases.

● **Which category has classification errors**

The only category that did not receive a perfect rating is Cherif, with 71.4% of the images correctly identified. The remaining images were misclassified, with 14.3% mistakenly categorized as Oualid and another 14.3% as Salah. This trend indicates there may be a visual or technical resemblance between the Cherif images and those in the other two categories, potentially leading to the model's errors. Factors that might have contributed to this confusion include similar lighting conditions during photography, comparable facial poses, and an insufficient number of training samples for the Cherif category

The confusion matrix aligns with the earlier tabular results, showcasing the model's exceptional accuracy in most instances, though it does exhibit a minor difficulty in differentiating between 'cherif' and a few other categories.

4.6 CONNECT THE SYSTEM WITH RASPBERRY PI AND CAMERA

Since large and spacious halls need full coverage to efficiently monitor all attendees, the importance of using a camera comes with the Raspberry Pi device. This solution enables the system to monitor large areas without the need to connect the cameras directly to the computer, providing flexibility and ease of installation and operation.

The Raspberry Pi is a small-sized, low-cost computer developed to encourage learning programming and electronics, and today it is widely used in Internet of Things (IoT) projects, control systems, and intelligent applications.

Despite its small size (almost the size of a credit card), the Raspberry Pi has advanced computing capabilities, as it contains a processor, RAM, USB ports, network ports (Ethernet or Wi-Fi), and a dedicated camera port (CSI).

Thanks to these features, the Raspberry Pi is an ideal platform for taking photos, operating cameras, collecting data, and sending it to centralized systems via wireless networks. It also features the ease of programming using languages such as Python, and its support for various operating systems such as Raspberry Pi OS (modified linux), which makes it suitable for connecting peripherals (such as cameras) to larger systems.

4.6.1 Mechanism of action

The camera is installed at a high location inside the hall, such as the top of the wall or the middle of the ceiling, to ensure comprehensive coverage of as many students as possible. The camera connects directly to the Raspberry Pi via the USB port or via a dedicated camera module (such as Camera Module v2 or HQ Camera).

The Raspberry Pi processes raw images or videos (such as converting the signal into data packets). After the system is turned on, the Raspberry Pi transmits the captured data wirelessly (via Wi-Fi network or via wireless Ethernet) to the main computer running the face recognition system.

The role of Raspberry Pi

Capture and conversion: capture photos or video from the camera and its initial processing if necessary (such as data compression to reduce the size).

Wireless transmission: sending data directly to the computer via network protocols (such as Web Socket or TCP/IP).

By processing a few simple operations, the Raspberry Pi relieves the load on the main computer

➢ **Advantages of using Raspberry Pi and camera**

Flexible installation: the possibility of installing cameras anywhere without being limited by the length of USB wires.

Extend coverage: more than one camera connected to several Raspberry Pi devices can be added to cover huge halls.

Autonomy: In the event of any malfunction in the camera or in the connection, the main computer is not directly affected.

Low cost: Raspberry Pi is a low-cost device compared to traditional advanced monitoring systems.

Easy maintenance: The camera can be easily maintained or repositioned without modifying the network infrastructure.

➢ **Technologies used in communication**

The connection is most often made via a local network (LAN) using Wi-Fi.

Light communication protocols are used such as:

WebSocket: for live and continuous connection (Live Stream).

HTTP with sending fragmented photos (Snapshots) at regular intervals.

RTSP or MJPEG Stream to display live video if the system needs live streaming.

Additional considerations

The stability of the wireless connection must be ensured to ensure that the tires are not lost during transportation.

It is preferable to use Raspberry Pi 4 or a newer version to ensure proper processing speed for modern Wi-Fi networks (Wi-Fi 5 or Wi-Fi 6).

The connection between the Raspberry Pi and the computer can be secured via encryption protocols such as SSL/TLS to ensure data protection during transfer.

The integration of the Raspberry Pi with the camera provides a smart and economical solution to expand the coverage of the attendance recording system in large halls in a flexible, secure and highly efficient way.

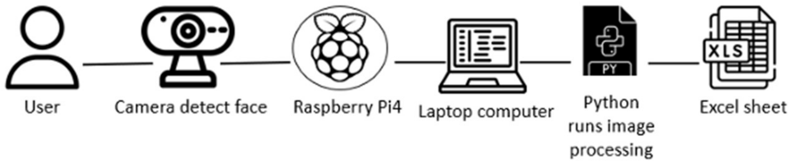

Figure 11: The attendance registration system with Raspberry Pi

4.7 CHALLENGES

4.7.1 Technical Challenges

● Improved Accuracy of Facial Recognition

Facial recognition technology faces significant accuracy challenges that require extensive engineering to overcome:

• **Lighting Variations:** The system must function reliably across diverse lighting conditions, from bright daylight to dimly lit lecture halls. We implemented adaptive brightness normalization algorithms that dynamically adjust image processing parameters based on ambient lighting conditions. This involved developing a multi-stage preprocessing pipeline with histogram equalization and gamma correction to enhance facial features in suboptimal lighting.

• **Occlusions:** Masks, glasses, and other face coverings significantly impaired recognition accuracy.

• **Pose Variations:** Students rarely present their faces in perfect alignment with cameras.

• **Appearance Changes:** Students' appearances naturally change over time (haircuts, facial hair, weight changes).

● **Real-time System Performance**

Real-time performance presented numerous challenges requiring optimization at multiple levels:

• **Computational Efficiency:** Initial processing times exceeded acceptable thresholds for classroom use. We addressed this through code optimization and algorithm selection, prioritizing computationally efficient methods that maintain accuracy. This included implementing parallel processing pipelines that distribute workloads across available CPU/GPU resources.

• **Frame Processing Optimization:** Rather than processing every video frame, we implemented an intelligent frame selection algorithm that analyzes only key frames determined by motion detection and quality assessment. This reduced computational load while maintaining system effectiveness.

• **Hardware Limitations:** Standard classroom computers often lack dedicated GPUs.

4.7.2 Software Challenges

● **Embedding Software Libraries**

Integration of multiple software components presented significant challenges:

• **Library Compatibility Issues:** The face recognition, OpenCV, and Tkinter libraries often had conflicting dependencies and version requirements. We created a containerized environment with precisely defined dependencies to ensure consistent behavior across different deployment environments.

• **Memory Management:** Face recognition libraries consumed excessive memory during continuous operation.

• **Thread Synchronization:** Concurrent processes for UI updates, video capture, and recognition processing required careful synchronization. deadlocks during operation.

● **Data Management**

Managing student data while ensuring privacy and accessibility presented complex challenges:

• **Efficient Storage:** Facial templates and associated metadata required optimized storage solutions.

• **Data Correlation:** Accurately linking facial data with student records required robust data structures.

• **Privacy Protection:** Student biometric data requires stringent protection.

• **Data Life cycle Management:** Policies for data retention and deletion require automated enforcement.

4.7.3 User-related Challenges

• Resistance to change by employees, especially the elderly, those unfamiliar with technology

• The time required for training in the use of the serious system

• Human errors in entering data or using the system

4.7.4 Data security challenges

• The risk of hacking the system and unauthorized access to personal data

• Challenges of data storage and protection from unauthorized access

• The difficulty of balancing ease of use with strict security requirements

4.8 FUTURE DEVELOPMENTS OF THE SMART ATTENDANCE REGISTRATION SYSTEM

ƒ **Integration with e-learning platforms**

The attendance system will evolve to become more than just a tool for tracking attendance, as it will be fully integrated within the broader educational system:

• **Integration with learning management systems (LMS):** development of connectors (APIs) for popular platforms such as Moodle, which will enable bidirectional data flow. This integration will allow automatic correlation of attendance patterns with interaction indicators, such as assignment delivery and participation in discussions, producing an overall perception of the level of student participation.

• **Automatic grade synchronization:** The next stage of development includes instantaneous synchronization of grades between attendance records and assessment platforms. When a student completes a test or submits an assignment, the system will automatically update their comprehensive academic profile, allowing an integrated view of their participation and performance.

• **Virtual attendance tracking:** With the increasing hybrid education, we will apply mechanisms to monitor attendance in virtual sessions. The system will track the duration of the login, the frequency of interaction, and the metrics of participation in virtual classes, and then combine this data with the actual attendance records to form an integrated participation file.

• **Tracking of continuing education modules (CEU):** In professional development and certification courses, the system will automatically calculate the approved attendance hours and generate official documents that comply with the requirements of accredited bodies.

ƒ Smart reporting and advanced analytics

Advanced analytics will turn raw attendance data into actionable educational insights:

• **Predictive performance models:** Develop machine learning algorithms that link attendance patterns and academic performance. These models will identify attendance behaviors associated with poor performance, such as frequent absences in the morning or certain subjects, for each student individually.

• **Interactive visualization tools:** upcoming updates will include customization dashboards that allow teachers and administrators to visualize attendance trends via heat maps, time patterns, and associative charts. These visualizations reveal patterns such as the most absent classes or the change in attendance during the semester.

• **Comparative analysis of study groups:** The system will automatically provide comparative reports between current payments and historical data, indicating deviations with statistical significance that may indicate problems in the curriculum or in the effectiveness of teaching.

• **Assessment of the quality of participation:** In addition to bilateral tracking (present/absent), we are developing indicators to measure the quality of participation based on sitting position, attention indicators, and frequency of interaction within the classroom.

ƒ **Intelligent points database and performance tracking**

The system will evolve into a comprehensive academic performance management tool:

• **Instant integration with the Grade record:** the next stage includes the creation of a unified academic record that is automatically updated with each attendance or assessment, eliminating the need for manual input and providing instant feedback to students about their current status.

• **Calculation of performance based on custom weights:** development of customizable algorithms for calculating the final grade based on variable weights, including attendance, quality of participation, delivery of tasks, and performance in tests, reducing the administrative burden on the professor.

• **Identify performance trends:** The system will automatically monitor significant changes in the student's performance, highlighting sudden declines or improvements that require the professor's attention or academic intervention.

• **Collect performance evidence:** In performance-based courses, the system will automatically collect digital evidence (such as recordings or reports) and create achievement files related to learning objectives and competency frameworks.

ƒ **Smart Alerts and academic interventions**

• **Early warning system:** development of an advanced predictive algorithm aimed at identifying students at risk of low performance, depending on attendance patterns, task completion, and assessment results. The system automatically issues smart alerts when students exceed certain limits, such as frequent absences or non-delivery of assignments, allowing early intervention and appropriate support before the problem worsens.

• **Automatic communication workflow:** The system will include a gradual escalation mechanism in communicating with absent students, starting with automated reminders, then personal messages from the teacher, and finally referring the case to academic counseling in case of continued absence.

• **Modeling of success paths:** by analyzing historical data of thousands of students, the system will recognize common patterns among excellent students and recommend specific behavioral actions for struggling students, based on those proven paths to success.

ƒ Corporate integration and administrative dashboard

The integration at the enterprise level will bring a shift in the management vision towards the effectiveness of Education:

• **Cross-departmental analytics:** The central dashboard will aggregate data from various departments and programs, enabling administrators to identify broad organizational patterns, such as courses with the highest withdrawal rates or teachers with the highest interaction rates.

• **Optimize resource allocation:** By linking attendance and performance data to class scheduling, the system will recommend an optimal distribution of rooms, class sizes, and lecture schedules to enhance student engagement and success.

• **Accreditation and compliance reports:** Attendance and participation reports will be generated automatically in formats compatible with the requirements of accreditation bodies, using templates customization by the relevant entity.

• **Organization effectiveness indicators:** The system will calculate key performance indicators related to student interaction, progress, and graduation rates, providing managers with real-time dashboards that show how effective the organization is in achieving its goals.

GENERAL CONCLUSION

In conclusion, our project “Smart Attendance System” is an excellent demonstration of how AI and computer vision can be applied to address real-world challenges in educational institutions. One of the important issues we were aiming to address was the traditional way of taking attendance, which is time-consuming and prone to manipulation. Initially, we focused on theoretical research. We delved into concepts such as artificial intelligence, facial recognition algorithms, and the functioning of time and attendance systems. Then, we developed a practical system and tested it in a real university environment. Our tests revealed that the system can accurately identify students, enabling automated and secure attendance tracking while minimizing the risk of cheating. The system performed effectively in real classroom settings, where we used a Raspberry Pi for ease of installation and portability. The results were impressive, achieving high scores in metrics such as precision and recall, indicating its reliable performance. However, we faced several challenges. Testing all the algorithms proved difficult, and we needed a more diverse set of facial data for training. In addition, factors such as different lighting conditions and different head orientations added additional complexity. These challenges provide valuable insights for future improvements. We can incorporate more data, explore advanced algorithms, and integrate the system with other learning tools. Ultimately, this project represents an important step towards implementing smart attendance systems in schools. Its promising potential suggests that it could see wider application in the future.

Reference

[1] F. D'Antoni et al., 'Artificial Intelligence and Computer Vision in Low Back Pain: A Systematic Review', Int. J. Environ. Res. Public. Health, vol. 18, no. 20, p. 10909, Oct. 2021, doi: 10.3390/ijerph182010909.

[2] B. Liu, L. Yu, C. Che, Q. Lin, H. Hu, and X. Zhao, 'Integration and Performance Analysis of Artificial Intelligence and Computer Vision Based on Deep Learning Algorithms', 2023, doi: 10.48550/ARXIV.2312.12872.

[3] S. Atallah, 'Artificial Intelligence and Computer Vision', pp. 407–417, 2021, doi: 10.1007/978 3-030-49100-0_31.

[4] D. R. Nijgal, S. George, and P. Subramanian, 'Evaluating the Effectiveness of a Facial Recognition-Based Attendance Management System in a Real-World Setting', 2023 14th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT, pp. 1–5, Jul. 2023, doi: 10.1109/ICCCNT56998.2023.10306966.

[5] G. Rajesh, N. Sairam, and M. Praveen, 'Smart Attendance Registration and Management System through AI', 2023 2nd Int. Conf. Autom. Comput. Renew. Syst. ICACRS, pp. 1234–1238, Dec. 2023, doi: 10.1109/ICACRS58579.2023.10404725.

[6] A. S. Lateef and M. Y. Kamil, 'Effortless Student Attendance: A Smart Human-Computer Interactive System Using Real Time Facial Recognition', -Najah Univ. J. Res. - Nat. Sci., vol. 39, no. 2, p. 2377, Aug. 2025, doi: 10.35552/anujr.a.39.2.2377.

[7] E. Guaichico, M. Pusdá-Chulde, M. Ortega-Bustamante, P. Granda, and I. García-Santillán, 'Mobile app for real-time academic attendance registration based on MobileFaceNet Convolutional neural network', Data Metadata, vol. 4, p. 193, Feb. 2025, doi: 10.56294/dm2025193.

[8] 'Web-based University Classroom Attendance System Based on Deep Learning Face Recognition', KSII Trans. Internet Inf. Syst., vol. 16, no. 2, Feb. 2022, doi: 10.3837/tiis.2022.02.008.

[9] S. A. Wijaya et al., 'Trends and Impact of the Viola-Jones Algorithm: A Bibliometric Analysis of Face Detection Research (2001-2024)'.

[10] P. Viola and M. Jones, 'Rapid object detection using a boosted cascade of simple features', in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA: IEEE Comput. Soc, 2001, p. I-511-I–518. doi: 10.1109/CVPR.2001.990517.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, 'You Only Look Once: Unified, Real Time Object Detection', May 09, 2016, arXiv: arXiv:1506.02640. doi: 10.48550/arXiv.1506.02640.

[12] A. Singh, 'HOG Feature Descriptor: Feature Engineering for Images', Analytics Vidhya. Accessed: Apr. 29, 2025. [Online]. Available: https://www.analyticsvidhya.com/blog/2019/09/feature-engineering-images-introduction-hog feature-descriptor/

[13] N. Dalal and B. Triggs, 'Histograms of Oriented Gradients for Human Detection', in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA: IEEE, 2005, pp. 886–893. doi: 10.1109/CVPR.2005.177.

[14] 'A Systematic Review of CNN Architectures, Databases, Performance Metrics, and Applications in Face Recognition'. Accessed: Apr. 29, 2025. [Online]. Available: https://www.mdpi.com/2078-2489/16/2/107

[15] A. Suleiman, Y.-H. Chen, J. Emer, and V. Sze, 'Towards Closing the Energy Gap Between HOG and CNN Features for Embedded Vision', Mar. 17, 2017, arXiv: arXiv:1703.05853. doi: 10.48550/arXiv.1703.05853.

[16] S. Lan, J. Li, S. Hu, H. Fan, and Z. Pan, 'A neighbourhood feature-based local binary pattern for texture classification', Vis. Comput., vol. 40, no. 5, pp. 3385–3409, May 2024, doi: 10.1007/s00371-023-03041-3.

[17] CSE Department, SGBAU, Amravati, Maharashra, India Sipna , College of Engineering & Technology, Amravati ,Infront of Nemani Godown,Badnera Road, Amravati, (M.S), India. and A. A. Bardekar, 'Emotional Influence of Indian Classical RAAG on Face using the Local Binary Pattern (LBP) Method: An Empirical Study', in INTERNATIONAL JOURNAL OF COMPUTING ALGORITHM, Jun. 2012, pp. 23–27. doi: 10.20894/IJCOA.101.001.001.005.

[18] 'Local binary pattern encoding schemes for computed tomography image segmentation: An experimental and comparative study | Semantic Scholar'. Accessed: Apr. 29, 2025. [Online]. Available: https://www.semanticscholar.org/paper/Local-binary-pattern-encoding-schemes-for computed-Lo-Wu/8a9aa474ad1f30be3c56062db0cd33c6058c68b8?_gl=1%2A1w6h70p %2A_gcl_au%2ANDQ4OTc3ODg2LjE3NDU4MzYwMTA.

[19] H. Chen, Z. Lin, and Q. Xu, 'Deep learning-based recognition model of football player's technical action behavior using PCA–LBP algorithm', Sci. Rep., vol. 15, no. 1, p. 13788, Apr. 2025, doi: 10.1038/s41598-025-94732-5.

[20] L. Leonardo, Y. Yohannes, and E. Hartati, 'Klasifikasi Sampah Daur Ulang Menggunakan Support Vector Machine Dengan Fitur Local Binary Pattern', in Jurnal Algoritme, Oct. 2020, pp. 78–90. doi: 10.35957/algoritme.v1i1.440.

[21] 'Principal Component Analysis(PCA) | GeeksforGeeks'. Accessed: May 01, 2025. [Online]. Available: https://www.geeksforgeeks.org/principal-component-analysis-pca/

[22] S. Karamizadeh, S. Abdullah, A. Manaf, M. Zamani, and A. Hooman, 'An Overview of Principal Component Analysis', J. Signal Inf. Process., Aug. 2013, doi: 10.4236/jsip.2013.43B031.

[23] K. S. M. Reddy, 'Comparison of Various Face Recognition Algorithms', vol. 4, no. 2, 2017.

[24] 'What Is Linear Discriminant Analysis? | IBM'. Accessed: May 01, 2025. [Online]. Available: https://www.ibm.com/think/topics/linear-discriminant-analysis

[25] L. Qu and Y. Pei, ‘A Comprehensive Review on Discriminant Analysis for Addressing Challenges of Class-Level Limitations, Small Sample Size, and Robustness’, Processes, vol. 12, no. 7, Art. no. 7, Jul. 2024, doi: 10.3390/pr12071382.

[26] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, ‘Linear discriminant analysis: A detailed tutorial’, AI Commun., vol. 30, no. 2, pp. 169–190, May 2017, doi: 10.3233/AIC 170729.

[27] J. R. Movellan, ‘Tutorial on Gabor Filters’.

[28] ‘GaborNet: investigating the importance of color space, scale and orientation for image classification - PMC’. Accessed: May 03, 2025. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC9044268/

[29] ‘A Note on Advantages of the Fuzzy Gabor Filter in Object and Text Detection’. Accessed: May 03, 2025. [Online]. Available: https://www.mdpi.com/2073-8994/13/4/678

[30] J. Yang, L. Liu, T. Jiang, and Y. Fan, ‘A modified Gabor filter design method for fingerprint image enhancement’, Pattern Recognit. Lett., vol. 24, no. 12, pp. 1805–1817, Aug. 2003, doi: 10.1016/S0167-8655(03)00005-9.

[31] M. Rai and P. Rivas, A Review of Convolutional Neural Networks and Gabor Filters in Object Recognition. 2020, p. 1567. doi: 10.1109/CSCI51800.2020.00289.

[32] ‘Two-Dimensional Discrete Wavelet Transform - an overview | ScienceDirect Topics’. Accessed: May 09, 2025. [Online]. Available: https://www.sciencedirect.com/topics/computer science/two-dimensional-discrete-wavelet-transform

[33] S. Gurumurthy, Image Compression Using Discrete Wavelet Transform. 2012.

[34] ‘(PDF) Discrete wavelet transform core for image processing applications’. Accessed: May 03, 2025. [Online]. Available: https://www.researchgate.net/publication/238054042_Discrete_wavelet_transform_core_for_image_processing_applications

[35] P. Xiao, ‘Image Compression by Wavelet Transform.’.

[36] ‘Image Processing: Discrete Wavelet Transform | Free Essay Example for Students’. Accessed: May 03, 2025. [Online]. Available: https://aithor.com/essay-examples/image processing-discrete-wavelet-transform

[37] ‘CT and MRI Image Fusion using Wavelet Transform and Log Gabor filter | Biores Scientia’. Accessed: May 03, 2025. [Online]. Available: https://bioresscientia.com/article/ct and-mri-image-fusion-using-wavelet-transform-and-log-gabor-filter

[38] A. Katharotiya, S. Patel, and M. Goyani, ‘Comparative Analysis between DCT & DWT Techniques of Image Compression’, vol. 1, 2011.

[39] ‘(PDF) Image Compression Using Discrete Wavelet Transform’. Accessed: May 03, 2025. [Online]. Available: https://www.researchgate.net/publication/282843786_Image_Compression_Using_Discrete_Wa velet_Transform

[40] Assistant Professor, Department of Electronics & Communication Engineering et al., 'Image Compression Using Discrete Cosine Transform & Discrete Wavelet Transform', Int. J. Web Technol., vol. 002, no. 001, pp. 25–27, Jun. 2013, doi: 10.20894/IJWT.104.002.001.007.

[41] 'DCT | PDF | Data Compression | Signal Processing'. Accessed: May 03, 2025. [Online]. Available: https://www.scribd.com/document/86644004/DCT

[42] I. A. Ismaili, S. A. Khowaja, and W. J. Soomro, 'Image Compression, Comparison between Discrete Cosine Transform and Fast Fourier Transform and the problems associated with DCT'.

[43] 'JETIRAU06057.pdf'. Accessed: May 03, 2025. [Online]. Available: https://www.jetir.org/papers/JETIRAU06057.pdf

[44] A. B. Watson, 'Image Compression Using the Discrete Cosine Transform', 1994.

[45] S. S. M.E, V. R. Vijayakuymar, and R. Anuja, 'A Survey on Various Compression Methods for Medical Images', Int. J. Intell. Syst. Appl., vol. 4, no. 3, pp. 13–19, Apr. 2012, doi: 10.5815/ijisa.2012.03.02.

[46] Manav, 'Convolutional Neural Networks (CNN) in Deep Learning', Analytics Vidhya. Accessed: May 03, 2025. [Online]. Available: https://www.analyticsvidhya.com/blog/2021/05/convolutional-neural-networks-cnn/

[47] M. Uniyal, 'Convolutional Neural Network (CNN) in Machine Learning', Applied AI Blog. Accessed: May 03, 2025. [Online]. Available: https://www.appliedaicourse.com/blog/convolutional-neural-network-cnn-in-machine-learning/

[48] Y. Deng, T. Zhang, G. Lou, X. Zheng, J. Jin, and Q.-L. Han, 'Deep Learning-Based Autonomous Driving Systems: A Survey of Attacks and Defenses', Apr. 10, 2021, arXiv: arXiv:2104.01789. doi: 10.48550/arXiv.2104.01789.

[49] T. Evgeniou and M. Pontil, Support Vector Machines: Theory and Applications, vol. 2049. 2001, p. 257. doi: 10.1007/3-540-44673-7_12.

[50] T. DigitalDefynd, '10 Pros & Cons of Support Vector Machines [2025]', DigitalDefynd. Accessed: May 03, 2025. [Online]. Available: https://digitaldefynd.com/IQ/pros-cons-of support-vector-machines/

[51] V. B. S. Prasath et al., 'Distance and Similarity Measures Effect on the Performance of K Nearest Neighbor Classifier -- A Review', Big Data, vol. 7, no. 4, pp. 221–248, Dec. 2019, doi: 10.1089/big.2018.0175.

[52] T. A. AI, 'K-Nearest Neighbor(KNN) Algorithm in Machine Learning', Applied AI Blog. Accessed: May 03, 2025. [Online]. Available: https://www.appliedaicourse.com/blog/knn algorithm-in-machine-learning/

[53] Ultralytics, 'YOLOv8'. Accessed: May 04, 2025. [Online]. Available: https://docs.ultralytics.com/models/yolov8

[54] F. Schroff, D. Kalenichenko, and J. Philbin, 'FaceNet: A unified embedding for face recognition and clustering', in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA: IEEE, Jun. 2015, pp. 815–823. doi: 10.1109/CVPR.2015.7298682.

[55] 'Results for "digital image processing"'. Accessed: May 04, 2025. [Online]. Available: https://www.pearson.com/en-us/search.html?aq=digital%20image%20processing

[56] K. He, X. Zhang, S. Ren, and J. Sun, 'Deep Residual Learning for Image Recognition', in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[57] C. Busch, 'Challenges for automated face recognition systems', Nat. Rev. Electr. Eng., vol. 1, no. 11, pp. 748–757, Sep. 2024, doi: 10.1038/s44287-024-00094-x.

[58] C. Sathwik, C. Vardhan, B. Abhiram, S. Shaik, and A. RaviKumar, Camera and Voice Control-Based Human–Computer Interaction Using Machine Learning. 2024, p. 187. doi: 10.1007/978-981-99-8451-0_15.

[59] A. Misra, G. Essl, and M. Rohs, 'Microphone as Sensor in Mobile Phone Performance', Jan. 2008.

[60] 'Signal Conditioning for a Sigma-Delta ADC in Industrial Multichannel Data Acquisition Systems | Analog Devices'. Accessed: May 10, 2025. [Online]. Available: https://www.analog.com/en/resources/technical-articles/2023/03/07/18/55/signal-conditioning sigma-delta-adc-industrial-multichannel-data-acquisition-systems.html

[61] M. A. G. Ruiz, 'Design of an Intelligent Front-End Signal Conditioning Circuit for IR Sensors', IEEE Trans. Nucl. Sci., Jan. 2000, doi: 10.1109/TNS.2007.913468.

[62] A. Lucena, J. Perotti, A. Eckhoff, and P. Medelius, 'Advanced Signal Conditioners for Data Acquisition Systems', KSC-12301, Mar. 2004. Accessed: May 10, 2025. [Online]. Available: https://ntrs.nasa.gov/citations/20110016669

[63] G. Nikolic, B. Dimitrijević, T. Nikolic, and M. K. Stojčev, A Survey of Three Types of Processing Units: CPU, GPU and TPU. 2022, p. 6. doi: 10.1109/ICEST55168.2022.9828625.

[64] V. W. Gbedawo, G. A. Owusu, C. K. Ankah, and M. I. Daabo, 'An Overview of Computer Memory Systems and Emerging Trends', Am. J. Electr. Comput. Eng., vol. 7, no. 2, Art. no. 2, Oct. 2023, doi: 10.11648/j.ajece.20230702.11.

[65] S. Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, Internet of Things (IoT) Communication Protocols : Review. 2017. doi: 10.1109/ICITECH.2017.8079928.

[66] 'Stationary Battery Thermal Management: Analysis of Active Cooling Designs'. Accessed: May 10, 2025. [Online]. Available: https://www.mdpi.com/2313-0105/8/3/23

[67] P. Fu, L. Zhao, X. Wang, J. Sun, and Z. Xin, ‘A Review of Cooling Technologies in Lithium-Ion Power Battery Thermal Management Systems for New Energy Vehicles’, Processes, vol. 11, no. 12, Art. no. 12, Dec. 2023, doi: 10.3390/pr11123450.

[68] ‘Research on Composite Liquid Cooling Technology for the Thermal Management System of Power Batteries’. Accessed: May 10, 2025. [Online]. Available: https://www.mdpi.com/2032 6653/16/2/74

[69] D. A. Borges Oliveira, L. G. Ribeiro Pereira, T. Bresolin, R. E. Pontes Ferreira, and J. R. Reboucas Dorea, ‘A review of deep learning algorithms for computer vision systems in livestock’, Livest. Sci., vol. 253, p. 104700, Nov. 2021, doi: 10.1016/j.livsci.2021.104700.

[70] ‘Deep Learning’. Accessed: May 07, 2025. [Online]. Available: https://www.deeplearningbook.org/

[71] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, ‘Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments’.

[72] ‘GitHub - HumanSignal/labelImg: LabelImg is now part of the Label Studio community. The popular image annotation tool created by Tzutalin is no longer actively being developed, but you can check out Label Studio, the open source data labeling tool for images, text, hypertext, audio, video and time-series data.’ Accessed: May 07, 2025. [Online]. Available: https://github.com/HumanSignal/labelImg

[73] ‘Speech and Language Processing’. Accessed: May 07, 2025. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/

[74] M. Fonder and M. Van Droogenbroeck, ‘Mid-Air: A Multi-Modal Dataset for Extremely Low Altitude Drone Flights’, in 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA: IEEE, Jun. 2019, pp. 553–562. doi: 10.1109/CVPRW.2019.00081.

[75] S. Lo Piano, ‘Ethical principles in machine learning and artificial intelligence: cases from the field and possible ways forward’, Humanit. Soc. Sci. Commun., vol. 7, no. 1, pp. 1–7, Jun. 2020, doi: 10.1057/s41599-020-0501-9.

[76] A. S. Shaikat et al., ‘Computer Vision Based Automated Attendance System Using Face Recognition’, in Proceedings of the International Conference on Industrial Engineering and Operations Management, Dhaka, Bangladesh: IEOM Society International, Dec. 2023. doi: 10.46254/BA06.20230105.

[77] Md. M. Islam et al., ‘DeepCrop: Deep learning-based crop disease prediction with web application’, J. Agric. Food Res., vol. 14, p. 100764, Dec. 2023, doi: 10.1016/j.jafr.2023.100764.

[78] F. Adnan, M. J. Awan, A. Mahmoud, H. Nobanee, A. Yasin, and A. M. Zain, ‘EfficientNetB3-Adaptive Augmented Deep Learning (AADL) for Multi-Class Plant Disease Classification’, IEEE Access, vol. 11, pp. 85426–85440, 2023, doi: 10.1109/ACCESS.2023.3303131.

[79] A. H. Ali, A. Youssef, M. Abdelal, and M. A. Raja, ‘An ensemble of deep learning architectures for accurate plant disease classification’, Ecol. Inform., vol. 81, p. 102618, Jul. 2024, doi: 10.1016/j.ecoinf.2024.102618.