ECO-ATTENDANCE THROUGH FACIAL RECOGNITION

A DESIGN PROJECT REPORT

Submitted by

ARULNITHI K

BALAJI K

HARIPRASATH S

In partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

NOVEMBER, 2024

ECO-ATTENDANCE THROUGH FACIAL RECOGNITION

A DESIGN PROJECT REPORT

Submitted by

ARULNITHI K (811722104015)

BALAJI K (811722104020)

HARIPRASATH S (811722104046)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

NOVEMBER, 2024

K.RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report titled **"ECO-ATTENDANCE THROUGH FACIAL RECOGNIATION"** is the Bonafide work of **ARULNITHI K (811722104015), BALAJI K (811722104020) , HARIPRASATH S (811722104046)** who carried out the project under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Dr.A.DELPHIN CAROLINA RANI, M.E.,Ph.D.,
HEAD OF THE DEPARTMENT
PROFESSOR
Department of CSE
K Ramakrishnan College of Technology
(Autonomous)
Samayapuram – 621 112

SIGNATURE
Ms. S.UMA MAGESHWARI, M.E.,
SUPERVISOR
ASSISTANT PROFESSOR
Department of CSE
K Ramakrishnan College of Technology
(Autonomous)
Samayapuram – 621 112

Submitted for the viva-voice examination held on ………………

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION

We jointly declare that the project report on **"ECO-ATTENDANCE THROUGH FACIAL RECOGNIATION"** is the result of original work done by us and best of our knowledge, similar work has not been submitted to **"ANNA UNIVERSITY CHENNAI"** for the requirement of Degree of **BACHELOR OF ENGINEERING**. This project report is submitted on the partial fulfilment of the requirement of the award of Degree of **BACHELOR OF ENGINEERING**.

Signature

ARULNITHI K

BALAJI K

HARIPRASATH S

Place:Samayapuram

Date:

ACKNOWLEDGEMENT

It is with great pride that we express our gratitude and in-debt to our institution **"K RAMAKRISHNAN COLLEGE OF TECHNOLOGY (AUTONOMOUS)"**, for providing us with the opportunity to do this project.

We are glad to credit honorable chairman **Dr. K RAMAKRISHNAN, B.E.,** for having provided for the facilities during the course of our study in college.

We would like to express our sincere thanks to our beloved Executive Director **Dr. S KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering adequate duration to complete it.

We would like to thank **Dr. N VASUDEVAN, M.Tech., Ph.D.,** Principal, who gave opportunity to frame the project with full satisfaction.

We whole heartily thank **Dr.A DELPHIN CAROLINA RANI,M.E., Ph.D.,** Head of the Department, **COMPUTER SCIENCE AND ENGINEERING** for providing her support to pursue this project.

We express our deep and sincere gratitude and thanks to our project guide **Ms.S.UMA MAGESHWARI , M.E.,** Department of **COMPUTER SCIENCE AND ENGINEERING,** for his incalculable suggestions, creativity, assistance and patience which motivated us to carry our this project.

We render our sincere thanks to Course Coordinator and other staff members for providing valuable information during the course. We wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

ABSTRACT

The integration of facial recognition technology into attendance systems offers a highly efficient and automated solution for tracking student presence. A system that uses facial recognition to determine whether a student is present or absent and, if present, automatically records their attendance in an Excel sheet. The system operates by capturing real-time video frames through a camera, detecting and recognizing student's faces, and matching them against a pre-stored database of student images. When a match is found, the student's attendance is marked as "present" and logged with a timestamp in an Excel sheet, otherwise, the student is considered "absent." This automated solution eliminates the need for manual attendance marking, reduces administrative workload, and minimizes errors associated with traditional methods. the proposed facial recognition attendance system provides a reliable and scalable solution for educational institutions, enhancing both security and convenience. the student's name and the date of the infraction, enhancing accountability and creating a transparent record of policies, foster a culture of discipline, the system incorporates a weekly dress code enforcement mechanism.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATION		EXPANSION
ROM	-	Range of Motion
UML	-	Unified Modeling Language
CRT	-	Cathode Ray Tube
LCD	-	Liquid Crystal Displays
CCD	-	Charge Coupled Device
RGB	-	Red Green and Blue
NMR	-	Nuclear Magnetic Resonance
CT	-	Computed Tomography
OCR	-	Optical Character Recognition
HP	-	Hewlett-Packard
WISN	-	Wireless Inertial Sensor Networks
EMG	-	Electromyography
IOT	-	Internet of Things

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO IMAGE PROCESSING

Digital image processing involves using computer algorithms to manipulate digital images. These images are typically represented as matrices of binary digits, with each element representing a pixel at the intersection of rows and columns in a 2D grid. This approach allows for processing various types of images, such as photographs, X-rays, or transparencies, which are initially digitized for computer storage. Once digitized, the image can undergo various processing techniques to enhance or modify its characteristics. These techniques can include filtering, edge detection, color enhancement, and more. The processed image is then stored in a rapid-access buffer memory for quick retrieval and display on high-resolution monitors, ensuring a smooth visual experience for viewers.

Face recognition has gained significant importance due to its wide range of applications, including computer access control, entry management in restricted areas, and database searches for identification purposes. This technology is utilized by organizations like the Roads and Maritime Services in Australia and the Australian Department of Immigration, which maintain extensive databases of facial images for identification and verification.

The core concept of face recognition involves extracting specific data from a human facial image's region of interest and comparing it with stored data for identification purposes. In simple terms, face identification determines if an input image corresponds to a person in a database, while face verification checks if a person's image exists in the database.

1.2 IMAGE PROCESSING SYSTEM

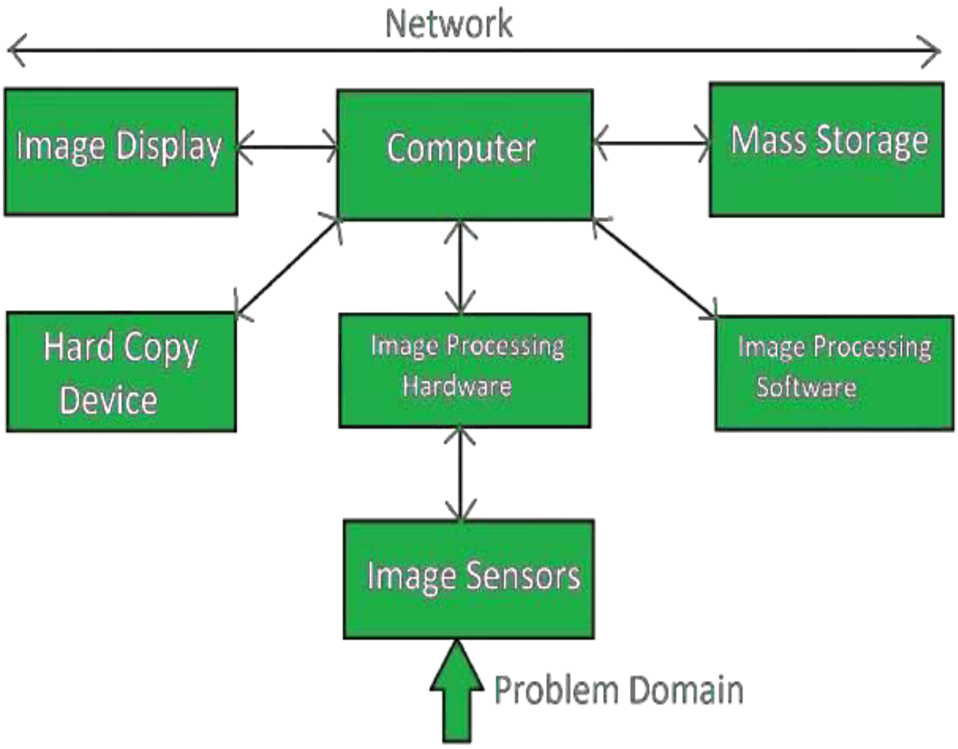


Fig 1.1 Image Processing System

Digitization involves representing an object, image, sound, document, or signal using a discrete set of points or samples, typically in the form of binary digits (bits), which are either 0 or 1. This process converts analog information into digital format, enabling storage and manipulation using computers or digital storage media like hard disks or CD-ROMs.

To digitize an image, various methods can be employed, such as using a scanner or a video camera connected to a computer with a frame grabber board. Once the image is digitized, it becomes accessible for a range of image processing operations.

Common tools and technologies used for digitization include microdensitometers, flying spot scanners, image dissectors, videocon cameras, and photosensitive solid-state arrays. These tools capture the analog data and convert it into digital form, facilitating further analysis, enhancement, or storage of the digitized content.

A computer is an electronic device that takes in raw data and follows a set of instructions to produce a desired outcome. When dealing with digitized images, the computer performs various mathematical processes like convolution, averaging, addition, and subtraction to manipulate the image.

Digital computers belong to a category of devices that work by processing information in discrete form. They handle data represented in binary code, which includes numerical values, characters, and symbols. This binary representation allows computers to execute operations and solve problems efficiently.

Mass storage devices in desktop and server computers are typically organized using a file system. These devices include floppy disks, CD- ROMs, and other similar mediums. Relational databases can also operate directly on mass storage devices without requiring an intermediate file system or storage manager.In embedded computers, such as those found in devices like smartphones or IoT devices, it's common to memory map the contents of a mass storage device like ROM or flash memory. mapping allows the device's contents to be accessed and manipulated as if they were in- memory data structures, or they can be executed directly by programs. This approach is beneficial for efficient data access and program execution in resource-constrained embedded systems.

The Operator Console is a versatile tool utilized across various industries for call management, support, and security purposes. It offers a graphical interface that allows designated operators or administrators to oversee and control active conferences system-wide.

This console comprises equipment and setups designed for verifying intermediate results and making software alterations as needed. Operators can also check for any errors that may arise and input necessary data as required. In essence, the Operator Console serves as a central hub for managing communications, ensuring operational efficiency, and addressing security concerns within an organization.

Spots may be binary (example: monochrome LCD), achromatic (example: so-called black-and-white, actually grayscale for intensity), pseudo color or false colors (example for intensity or hyper spectral data), or true color (color data displayed as such)

1.3 IMAGE PROCESSING FUNDAMENTALS

Digital image processing involves the manipulation of images in a digital format. While modern cameras can capture images directly in digital form, most images start in optical form and are then converted to digital format through the use of video cameras and digitalization processes, which include sampling and quantization. These digital images undergo processing through at least one of the five fundamental processes, which may not necessarily include all of them. Enhancement techniques like histogram equalization improve contrast, while filtering operations such as blurring and sharpening manipulate image details by modifying pixel values through linear and non-linear filters. Edge detection algorithms like Sobel, Canny, and Laplacian of Gaussian help highlight boundaries and significant transitions in an image.

Thresholding is a crucial technique used to convert an image into binary form, either globally or adaptively. They are the operations such as erosion and dilation alter the shape of objects in binary images, helping to remove noise or smooth object boundaries. Segmentation techniques divide an image into regions based on specific characteristics, enabling more efficient analysis. Lastly, feature extraction techniques identify key features like edges and textures, which are essential for tasks like object recognition and tracking.

1.4 FUNDAMENTAL STEPS IN IMAGE PROCESSING

First need to produce a digital image from a paper envelope. They can be done using either a CCD camera, or a scanner.

Preceding the main image processing tasks is a crucial step focused on executing basic operations to optimize the resulting image for subsequent tasks. This preliminary phase involves actions such as enhancing contrast, eliminating noise, and identifying areas potentially containing the postcode, all aimed at making the image more suitable for further processing.

Segmentation refers to the partitioning of a digital image into multiple segments, which are sets of pixels also referred to as super pixels. The objective of segmentation is to simplify and alter the image representation into a format that is more meaningful and simpler to analyse.

This technique is commonly employed to identify objects and delineate boundaries within images. Specifically, image segmentation involves assigning a label to each pixel based on shared visual attributes, ensuring that pixels with the same label exhibit similar visual characteristics.

Image process is the process of convert the input data to a form suitable for computer processing

Image description is the process of extract features that result in some quantitative information of interest or features that are basic for differentiating one class of objects from another.

Image recognition is the process of assign a label to an object based on the information provided by its descriptors. Image interpretation is the process of assign meaning to an ensemble of recognized objects.

1.5 IMAGE TYPES

Each pixel is just black or white. Since there are only two possible values for each pixel(0, 1), it only need one bitper pixel.

Each pixel is a shade of gray, normally from 0 (black) to 255(white). The range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally they are a power.

An indexed image consists of an array and a color map matrix. The pixel values in the array are direct indices into a color map. By convention, uses the variable name X to refer to the array and map to refer to the color map.

Each pixel has a particular color; that color is described by the amount of red, green and blue in it. If each of these components has a range 0–255, the gives a total of 2563different possible colors. Such an image is a “stack” of three matrices; representing the red, green and blue values for each pixel. The means that for every pixel there correspond 3 values.

Hyperspectral images are similar to multispectral images but capture data across a much wider range of wavelengths, often consisting of hundreds of narrow bands. These images provide very detailed spectral information, enabling precise material identification and analysis. Hyperspectral imaging is used in advanced applications such as mineral exploration, environmental monitoring, and medical diagnostics, where a detailed spectral signature can distinguish between different substances or conditions.

Multispectral images capture data at multiple wavelengths of light, beyond just visible colors. These images are often used in remote sensing, where different spectral bands (such as infrared, ultraviolet, and visible light) help to analyze environmental conditions, land cover, vegetation health,

1.6 IMAGE PROCESSING GOALS

In virtually all image processing applications, however, the goal is to extract information from the image data. Obtaining the information desired may require filtering, transforming, coloring, interactive analysis, or any number of other methods. To be somewhat more specific, one can generalize most image processing tasks to be characterized by one of the following categories:

The simply means improvement of the image being viewed to the (machine or human) interpreter's visual system. Image enhancement types of operations include contrast adjustment, noise suppression filtering, application of pseudo color, edge enhancement, and many others.

Image restoration aims to rectify or reverse imperfections that diminish the quality of an image. These imperfections can manifest in various forms such as motion blur, noise, and camera misfocus. For instance, in scenarios involving motion blur, it is feasible to generate a highly accurate estimate of the blurring function and reverse the blur to restore the image to its original state. However, when an image is marred by noise, the most achievable outcome may be to mitigate the degradation caused by the noise rather than completely eliminating it.

Image segmentation divides an image into meaningful regions to isolate specific objects or areas for detailed analysis, while object detection and recognition are crucial for locating and identifying objects within images, often used in security and automated systems. Pattern recognition identifies regularities and structures, which are essential for applications like facial recognition or quality control. Image compression reduces file sizes without significant quality loss, making storage and transmission more efficient. Morphological operations manipulate the structure of objects in binary images, improving tasks such as medical imaging or defect detection. Motion analysis and tracking are used to detect and follow moving objects over time, vital in surveillance and autonomous driving. Additionally, 3D image processing is used to analyze and visualize complex three-dimensional data in fields like medical imaging and engineering design.

Image analysis involves extracting meaningful data from images, producing numerical or graphical information based on the image's characteristics. These operations involve breaking down objects within the image and categorizing them, relying on image statistics for accuracy. Typical tasks include extracting and describing scene and image features, automated measurements, and object classification. Image analysis finds extensive use in machine vision applications, where it plays a crucial role in interpreting visual data for various purposes.

Image registration is a fundamental process that involves aligning two or more images of the same scene, captured under varying conditions such as different times, viewpoints, or sensors. This alignment, which geometrically aligns a reference image with sensed images, is crucial in image analysis tasks where data from multiple sources must be combined, as seen in applications like image fusion, change detection, and multichannel image restoration.

In various fields, image registration plays a vital role. In remote sensing, it facilitates tasks such as multispectral classification, environmental monitoring, change detection, image mosaicing, and weather forecasting. In medicine, image registration is essential for combining data from different modalities like CT and NMR to enhance patient information, monitor tumor growth, verify treatments, and compare patient data with anatomical atlases. Cartography benefits from image registration for map updating, while computer vision uses it for tasks like target localization and automatic quality control. These examples highlight the versatility and significance of image registration across diverse domains.

Image compression aims to minimize the redundancy and irrelevant data in images for efficient storage or transmission. This compression can be either lossy or lossless. Lossless compression, favored for archival and precise imaging like in medical fields or technical drawings, preserves all image information without quality loss. On the other hand, lossy compression, while introducing compression artifacts at low bit rates, is ideal for natural images like photographs where a minor loss of fidelity is acceptable for a significant reduction in bit rate. In cases where the loss is imperceptible, it's termed as visually lossless compression.

Feature extraction is a critical process that aims to streamline the resources needed for accurately describing a vast dataset. The challenge in analyzing complex data often arises from dealing with a multitude of variables, which can demand significant memory, computational power, or lead to overfitting issues with classification algorithms, resulting in poor generalization to new samples. To address these challenges, feature extraction encompasses various methods for creating combinations of variables that circumvent these issues while maintaining sufficient accuracy in describing the data.

Image synthesis operations create images from other images or non-image data. Image synthesis operations generally create images that are either physically impossible or impractical to acquire.

image processing are diverse and depend on the specific application, but generally focus on enhancing, analyzing, and extracting valuable information from images. Image enhancement aims to improve visual quality by adjusting contrast, reducing noise, or sharpening details, making important features more noticeable. Image restoration, on the other hand, seeks to recover degraded images by removing distortions or noise, effectively restoring them to their original form. Feature extraction identifies key elements such as edges, textures, or shapes within an image, aiding in tasks like object recognition and tracking

Image visualization focuses on presenting processed images in a manner that is easy to interpret and analyze by humans or machines. This goal is essential for fields like medical diagnostics (where doctors need clear, interpretable images of body scans).

CHAPTER 2

LITERATURE SURVEY

Literature survey is a text written by someone to consider the critical points of current knowledge including substantive findings, as well as theoretical sections.

2.1 TEXTURE-GUIDED TRANSFER LEARNING FORLOW-QUALITY FACE RECOGNITION

AUTHORS: Meng Zhang , Rujie Liu, Daisuke Deguchi , Member, IEEE

Although many advanced works have achieved significant progress for face recognition with deep learning and large-scale face datasets, low-quality face recognition remains a challenging problem in real-word applications, especially for unconstrained surveillance scenes. We propose a texture-guided transfer learning approach under the knowledge distillation scheme to improve low-quality face recognition performance. Unlike existing methods in which distillation loss is built on forward propagation; e.g., the output logits and intermediate features, in this study, the backward propagation gradient texture is used. More specifically, the gradient texture of low-quality images is forced to be aligned to that of its high-quality counterpart to reduce the feature discrepancy between the high- and low-quality images. Moreover, attention is introduced to derive a soft-attention version of transfer learning, termed as SA-TG, to focus on informative regions. Experiments on the benchmark low-quality face DB's TinyFace and QMUL-SurFace confirmed the superiority of the proposed method, especially more than6.6% Rank1 accuracy improvement is achieved on TinyFace. Index Term Low-quality face recognition, transfer learning, backward propagation gradient texture, knowledge distillation.

The paper introduces an innovative solution for low-quality face recognition by focusing on texture-guided transfer learning. This approach leverages the power of pre-trained deep learning models and enhances texture features in degraded face images to achieve better recognition performance. The technique has significant potential for applications in security, surveillance, and identity verification, where face recognition is often required under challenging conditions.

2.2 AN OVERVIEW OF THE TESSERACT OCR ENGINE" BY RAY SMITH, PUBLISHED IN 2007

AUTHOR: Smith R

The work "An Overview of the Tesseract OCR Engine" by Ray Smith provides a comprehensive overview of the Tesseract Optical Character Recognition (OCR) engine, detailing its development, architecture, and capabilities. Tesseract is one of the most accurate open- source OCR engines available and has been widely used for various text recognition tasks, including the extraction of text from ID cards and other documents.

Tesseract was originally developed by Hewlett-Packard (HP) in the 1980s and 1990s. The development ceased in 1995, but the project was revived by HP and the University of Nevada, Las Vegas in 2005. In 2005, Tesseract was released as open-source software, which significantly increased its development pace and community contributions.

Tesseract is known for its high accuracy in recognizing printed text. The engine has been extensively benchmarked against other OCR engines and has demonstrated superior performance, especially with clean, printed text. Tesseract supports a wide range of languages and can be trained to recognize new scripts.The work highlights Tesseract's application in recognizing text from ID cards, where its ability to handle various fonts and layouts is particularly beneficial. The work by Ray Smith provides an in-depth look at the Tesseract OCR engine, highlighting its development history, modular architecture, and high accuracy in text recognition. Originally developed by HP and later made leading OCR engines.

It excels at recognizing printed text and supports many languages, though it has limitations with handwritten text and complex layouts. The work underscores Tesseract's application in ID card recognition and suggests areas for future enhancement, ensuring its continued relevance and improvement in the field of OCR technology.

2.3 FACIAL HAIR RECOGNITION USING FREQUENCY DOMAIN FEATURE

AUTHOR: Fang, H., & Costen, N

Fang and Costen (2009) explore the use of frequency domain features for the recognition of facial hair patterns in images. The study focuses on developing methods to accurately identify and distinguish various types of facial hair, which can significantly affect the performance of facial recognition systems. The primary objective is to enhance facial recognition systems by incorporating the ability to recognize and differentiate facial hair patterns. This is motivated by the observation that facial hair can drastically alter a person's appearance and thus impact the effectiveness of facial recognition technologies. The authors propose a technique that leverages frequency domain features for facial hair recognition. The method involves transforming facial images into the frequency domain using techniques like the Fourier Transform. Frequency domain features are extracted to capture the unique patterns and textures associated with different types of facial hair.

These features are advantageous as they can represent periodic patterns and textures more effectively than spatial domain features. By focusing on specific frequency bands, the method can isolate the characteristics of facial hair from other facial features. The authors describe the process of applying their method to a dataset of facial images. The steps include pre- processing the images, applying the frequency domain transformation, extracting on relevant features, and then classifying the facial hair types. The results demonstrate that frequency domain features can significantly improve the accuracy of facial hair recognition.

The approach is shown to be robust against variations in lighting, pose, and facial expressions. Enhancing the performance of facial recognition systems in security and surveillance. Improving identification accuracy in systems where facial hair is a significant variable.

2.4 ARCFACE: ADDITIVE ANGULAR MARGIN LOSS(AAML) FOR DEEP FACE RECOGNITION

AUTHOR: Deng, J., Guo, J., Xue, N., & Zafeiriou,N

Deep face recognition has undergone a remarkable transformation, driven by the advancements in deep learning. Early methods, such as Eigenfaces and Fisherfaces, relied heavily on handcrafted features, which limited their performance, especially in real-world scenarios with variations in pose, lighting, and occlusion. The advent of deep learning introduced more robust and discriminative feature extraction methods, exemplified by DeepFace (2014) and FaceNet (2015), which utilized convolutional neural networks (CNNs) to learn embeddings optimized for face verification and recognition tasks. Among the state-of-the-art approaches, ArcFace, proposed by Deng et al. (2019), introduced an additive angular margin loss that significantly improved the discriminative power of learned face embeddings. This innovation enhanced both inter-class separability and intra-class compactness, addressing key challenges in face recognition.

ArcFace's effectiveness was demonstrated through its superior performance on prominent benchmarks such as LFW, MegaFace, and IJB-C, where it outperformed prior methods like SphereFace and CosFace, which employed different margin-based loss functions. ArcFace's contributions extend beyond academic benchmarks, finding applications in security, biometric authentication, and entertainment, including real-time identity verification, AR/VR systems, and tagging in social media.

Despite its success, challenges remain, including addressing biases in training datasets, improving robustness against adversarial attacks, and ensuring user privacy. These challenges motivate future research directions, such as lightweight models for resource-constrained devices, self- supervised learning for unlabeled data, and domain adaptation for diverse environments. As face recognition continues to evolve, methods like ArcFace serve as a foundation for further advancements in this critical field.

2.5 FACENET AND UNIFIED FACE RECOGNITION IN DEEP CONVOLUTIONAL NEURAL NETWORKS

AUTHOR: Schroff, F., Kalenichenko, D., & Philbin, J.

FaceNet, introduced by Schroff, Kalenichenko, and Philbin in 2015, marked a significant breakthrough in face recognition and clustering by proposing a unified deep learning approach to learn facial embeddings directly optimized for the task. Traditional methods relied on feature extraction followed by classification, often requiring separate processes for feature engineering and recognition. FaceNet, in contrast, employed a single deep convolutional neural network (CNN) trained with a triplet loss function, ensuring that embeddings for the same identity were closer in the embedding space while those for different identities were pushed apart. This end-to-end training pipeline eliminated the need for intermediate stages, resulting in highly discriminative embeddings that could be used for various tasks such as face verification, identification, and clustering.

The triplet loss function was the core innovation of FaceNet, requiring careful selection of triplets (anchor, positive, and negative samples) to maximize learning efficiency. By utilizing large-scale datasets and hard sample mining, FaceNet achieved state-of-the-art performance on benchmarks like LFW (achieving 99.63% accuracy) and YouTube Faces. Furthermore, FaceNet demonstrated scalability, handling millions of identities while maintaining accuracy. Its embeddings facilitated face clustering in large datasets without requiring explicit labels, making it versatile for applications in surveillance, biometrics, and multimedia retrieval.

Despite its successes, FaceNet also highlighted challenges such as the computational complexity of triplet selection and the dependency on large-scale labeled datasets. These limitations paved the way for further innovations, including margin-based losses (e.g., ArcFace) and self-supervised learning approaches to reduce reliance on labeled data. FaceNet's pioneering design laid the groundwork for modern face recognition systems, establishing a foundation for continued advancements in accuracy, efficiency, and real-world applicability.

2.6 DEEP FACE RECOGNITION BY PARKHI ET AL IN TRANSFER LEARNING AND PRE-TRAINING

AUTHOR: Parkhi, O. M., Vedaldi, A., & Zisserman, A

Parkhi, Vedaldi, and Zisserman (2015) proposed a deep learning approach to face recognition, emphasizing the use of large-scale datasets and convolutional neural networks (CNNs) to improve performance. The study introduced the VGG-Face model, which leveraged a deep CNN architecture pre-trained on a large dataset of 2.6 million images from 2,622 individuals. This extensive training allowed the model to learn robust facial features, achieving state-of-the-art accuracy on various benchmarks.

The VGG-Face architecture, based on the VGGNet framework, consisted of multiple convolutional and pooling layers followed by fully connected layers. This deep architecture enabled the extraction of high-level features that were invariant to variations in pose, lighting, and occlusion. The embeddings learned by VGG-Face could be used for tasks such as face verification and identification, providing a universal representation of facial features.

Parkhi et al. also emphasized the importance of high-quality training datasets. Their use of a proprietary dataset with diverse poses, expressions, and lighting conditions contributed to the robustness of their model. On the LFW benchmark, the VGG-Face model achieved a high accuracy of 98.95%, showcasing its effectiveness.

Despite its successes, the study highlighted challenges in training deep networks, such as the computational cost of handling large datasets and the need for extensive labeled data. These challenges spurred further research into efficient architectures and unsupervised or semi-supervised training methods. The VGG-Face model remains a significant milestone in face recognition, influencing subsequent innovations like FaceNet and ArcFace, which built upon its foundational principles to achieve even greater accuracy and scalability.

CHAPTER 3

SYSTEM ANALYSIS

The process of analyzing the current attendance system and the proposed improvements using facial recognition technology is detailed in this system analysis.

3.1 EXISTING SYSTEM

In the existing system, Wireless Inertial Sensor Networks (WISN) are used, composed of three inertial modules placed on the trunk, upper arm, and forearm of an individual to assess shoulder range of motion (ROM) in real time. Each inertial module contains an ARM-based 32-bit microcontroller, a triaxial accelerometer, a triaxial gyroscope, a triaxial magnetometer, and a Controller Area Network (CAN) transceiver. These sensors work together to measure the accelerations, angular velocities, and magnetic signals generated by human shoulder movements. The collected data is transmitted to a personal computer via a Bluetooth wireless transmission module. The shoulder ROM estimation algorithm employed in this system includes procedures for data collection, signal preprocessing, quaternion-based orientation estimation, and shoulder joint compensation. To accurately evaluate shoulder ROM, the system integrates accelerations, angular velocities, and magnetic signals into a quaternion-based complementary nonlinear filter, which minimizes cumulative errors caused by sensor drift. However, this existing system faces several limitations. There is often disagreement around definitions, diagnoses, and study methodologies, which makes it difficult to compare research findings and draw consistent conclusions. Variations in definitions and diagnostic criteria can lead to inconsistencies and confusion, while differences in study designs and methodologies create heterogeneity that limits the ability to synthesize and compare results. Additionally, direct comparisons of the effectiveness of exercise-based rehabilitation to other treatments can be problematic due to varying nature and severity of injuries, differences in exercise programs, and confounding factors such as co-morbidities, medication use, and patient compliance. Differences in study design, patient selection, and outcome measures further complicate comparisons. The classification of musculoskeletal shoulder disorders based on pathoanatomic diagnosis helps guide treatment decisions and prognosis, but it can also add to the complexity of research and treatment comparison.

The existing system, utilizing Wireless Inertial Sensor Networks (WISN) for shoulder range of motion (ROM) assessment, is accompanied by several notable limitations. One significant challenge lies in the lack of consensus and standardization across definitions, diagnoses, and research methodologies related to musculoskeletal shoulder disorders. This lack of agreement makes it inherently difficult to compare research findings and draw consistent conclusions across studies. The variations in definitions and diagnostic criteria not only lead to inconsistencies and confusion but also hinder the synthesis and comparison of results. Moreover, the diverse nature and severity of injuries or conditions being treated pose another layer of complexity. The effectiveness of exercise- based rehabilitation, a key aspect of the existing system, faces challenges in direct comparison to alternative treatments due to these varying factors. Additionally, the specific details of exercise programs, including intensity, frequency, duration, and mode of delivery, vary widely, further comparative analysis.

Confounding factors such as co- morbidities, medication usage, and patient compliance introduce additional variability that can skew outcomes and hinder meaningful comparisons. Furthermore, differences in study design, patient selection criteria, and outcome measures contribute to the complexity of assessing and comparing the effectiveness of rehabilitation treatments. These limitations collectively underscore the need for greater standardization, consensus-building, and methodological clarity in research and clinical practice related to shoulder ROM assessment and rehabilitation strategies.

3.2 PROPOSED SYSTEM

The proposed system aims to enhance the current method of assessing limb mobility by incorporating advanced AI techniques, motion and rotational movement analysis, and electromyography (EMG) sensing. The limb mobility computation method includes data collection, signal analysis, and identification and modification of upper-limb movements. Physical therapists traditionally use a goniometer to measure the range of motion (ROM), which measures joint inclination. Various forms and sizes of goniometers are used to accommodate different joints in the human body.

In this proposed system, the focus is on the complex anatomical components involved in limb movement, specifically the ribs, upper arm, and wrist, at the beginning of the therapy plan. During data collection, motion impulses, rotational movement markers, and electromuscular signals from the WISN are transmitted to a smartphone app or online platform via an IoT component. Students are instructed to perform five arm movements, including extending their ROM, while standing in the center of the room and then returning to their natural postures. An EMG monitor is placed on the infraspinatus muscle, and electrodes are used to detect activity in the supraspinatus and teres minor muscles. The infraspinatus muscle, one of the four muscles of the rotator cuff, primarily functions to laterally rotate the upper arm and support the joint. The proposed system enhances limb movement by considering compensatory alignments captured by a quasi-static device, which evaluates limb treatment effectiveness for post-operative rotator cuff replacement. Research indicates good inter-ethnic agreement in the WISN for various directions of limb movements, suggesting that the WISN could effectively power limb motion tracking devices.

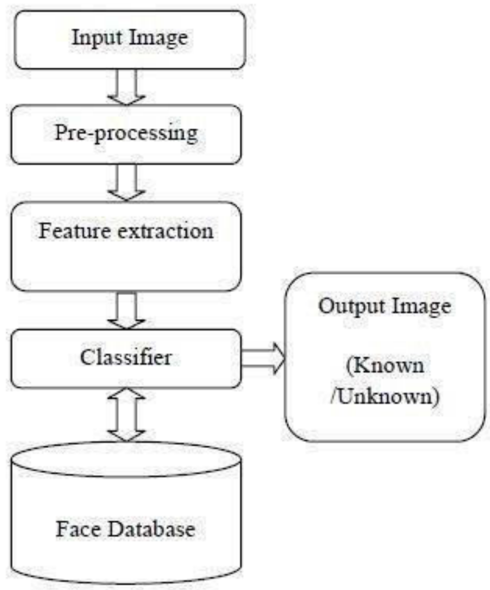


Fig 3.1: Proposed System diagram

The proposed system offers a range of significant advantages over traditional methods of shoulder range of motion (ROM) assessment and rehabilitation. Firstly, it provides a more accurate and objective measurement of ROM by integrating advanced AI techniques, motionanalysis, and electromyography (EMG) sensing.

This comprehensive approach ensures precise data collection and analysis, leading to more effective treatment planning and monitoring. Secondly, the system enhances the patient experience by offering personalized and tailored rehabilitation programs based on individualized assessments of limb mobility and muscle activity. This personalized approach not only improves treatment outcomes but also enhances patient satisfaction and engagement in the rehabilitation process. Additionally, the system promotes early intervention and proactive management of musculoskeletal shoulder disorders, reducing the risk of complications and long-term disability. By restoring normal range of motion and strength to affected joints, the system improves overall function and mobility, enhancing the patient's quality of life. Furthermore, the cost- effectiveness of the proposed system makes it a viable alternative to more invasive treatments like surgery, reducing healthcare costs and resource utilization. Finally, the system empowers patients by providing them with the knowledge, tools, and support needed to manage their condition independently, fostering self-confidence and long-term well-being. Overall, the proposed system represents a significant advancement in shoulder ROM assessment and rehabilitation, offering comprehensive, personalized, and cost-effective solutions for patients and healthcare providers alike.

CHAPTER 4

SYSTEM REQUIREMENTS

The requirements specification is a technical specification of requirements for the hardware products. It is the first step in the requirements analysis process it lists the requirements of a particular hardware system including functional, performance and safety requirements. The requirements also provide usage scenarios from a user and an operational perspective. The purpose of hardware requirements specification is to provide a detailed overview of the hardware project, its parameters and goals. This describes the project target and its user interface, hardware and software requirements.

4.1 HARDWARE REQUIREMENTS

Ram: 4.00 GB

Processor: Intel(R) Pentium(R) 2.11 GHz

Hard Disk: 520 GB

Camera: 1080p HD Night vision

4.2 SOFTWARE REQUIREMENTS

Operating System: Windows Software:

Python IDLE

Language: Python

4.3 TECHNOLOGIES USED

PYTHON

Python emerges as a powerful and versatile programming language ideally suited for developing sophisticated facial recognition systems. Unlike Arduino Sketch, which is primarily focused on microcontroller- based development, Python offers a broader ecosystem and toolset for building complex algorithms and applications. Python's simplicity, readability, and extensive libraries, such as OpenCV and dlib, provide developers with robust tools for image processing, computer vision, and facial recognition tasks. These libraries offer pre-trained models and algorithms for facial detection, landmark identification.

Moreover, Python's flexibility extends to its integration capabilities with various hardware platforms and IoT frameworks. It allows seamless connectivity with cameras, sensors, and IoT devices, facilitating real-time data acquisition and processing for facial recognition tasks. Additionally, Python's compatibility with cloud services enables developers to leverage cloud-based computing resources for intensive facial recognition computations, enhancing scalability and performance.

Furthermore, Python's active community and rich documentation provide developers with access to a vast knowledge base and support network. Overall, Python stands out as a preferred choice for implementing facial recognition systems, offering a blend of versatility, scalability, and community support essential for developing cutting-edge solutions in attendance tracking and security

HISTORY OF PYTHON

Python, a versatile and widely-used programming language, has a rich history that traces back to the late 1980s and early 1990s. Its development was initiated by Guido van Rossum, a Dutch programmer, who aimed to create a language that prioritized code readability and simplicity while offering powerful features for software development. The name "Python" was inspired by the British comedy series "Monty Python's Flying Circus," reflecting van Rossum's fondness for the show's humor.

The origins of Python can be traced to the ABC programming language, which van Rossum had worked on during his time at the Centrum Wiskunde & Information (CWI) in the Netherlands. ABC was designed to be an easy-to-understand language for teaching programming concepts. Van Rossum envisioned Python as a successor to ABC, incorporating its simplicity and ease of use while adding features for general-purpose programming.

Python's first version, Python 0.9.0, was released in February 1991. It included fundamental features such as exception handling, functions, and modules, laying the groundwork for its subsequent evolution. Python's design philosophy, known as "The Zen of Python," emphasizes readability, simplicity, and explicitness, guiding developers to write clear and understandable code.

Over the years, Python gained popularity and underwent significant enhancements and updates. The release of Python 2.0 in 2000 introduced list comprehensions, garbage collection, and a unified object model, further solidifying Python's capabilities as a versatile and efficient language. Python 2.x versions continued to evolve, with Python 2.7 being the last major release in the Python 2 series, maintaining compatibility while paving the way for the transition to Python 3.

Python 3, introduced in December 2008 with Python 3.0, marked a significant milestone in Python's development. It addressed various design flaws and inconsistencies present in Python 2, introducing features syntax, and enhanced standard library modules. While the transition from Python 2 to Python 3 initially faced challenges due to compatibility issues, the Python community actively supported and encouraged migration to Python 3, leading to widespread adoption of the newer version.

Today, Python stands as one of the most popular and widely-used programming languages globally, renowned for its simplicity, readability, versatility, and extensive ecosystem of libraries and frameworks. Its applications span diverse domains, including web development, data science, artificial intelligence, automation, scientific computing, and more. Python's vibrant community, robust documentation, and continuous development make it a preferred choice for developers and organizations seeking efficient and scalable solutions.

4.4 HARDWARE

CAMERA

Cameras play a pivotal role in the realm of face recognition technology, serving as the primary input source for capturing facial images and enabling the identification and authentication of individuals. The importance of cameras in face recognition systems lies in their ability to capture high-quality images and videos of faces, which are then processed and analyzed using sophisticated algorithms to extract unique facial features.

Modern cameras used in face recognition systems are equipped with advanced technologies such as high-resolution sensors, autofocus capabilities, low-light sensitivity, and fast shutter speeds. These features ensure that facial images captured by the camera are clear, detailed, and suitable for accurate analysis and recognition. Additionally, cameras may incorporate features like facial detection autofocus and real-time image processing, enhancing their effectiveness in capturing facial data in various environments and lighting conditions.

In face recognition systems, cameras serve multiple crucial functions. Firstly, they acquire facial images or video streams from individuals seeking access or identification. These images are then processed to detect and extract facial landmarks, such as eyes, nose, and mouth, forming a unique facial signature or template for each individual. Cameras also play a role in capturing facial expressions, gestures, and movements, which can be used for additional behavioral analysis and authentication purposes.

The importance of cameras extends beyond mere image capture; they contribute to the overall accuracy and reliability of face recognition systems. A well-calibrated and strategically positioned camera ensures optimal image quality and consistency, reducing false positives and negatives in facial recognition tasks. Moreover, cameras with advanced features like depth sensing and 3D imaging enhance the system's ability to detect and differentiate between real faces and spoofing attempts using masks or photographs.

Furthermore, cameras integrated into surveillance and security systems leverage face recognition technology to enhance situational awareness, monitor access control, and improve forensic analysis. They enable real-time identification of individuals, tracking of suspicious activities, and alerting security personnel to potential threats or unauthorized access attempts.

CHAPTER 5

SYSTEM DESIGN

System architecture is the process of defining the architecture, modules, interfaces and data for a system to satisfy specified requirements.

5.1 BLOCK DIAGRAM

A block diagram is a fundamental tool used in various fields of engineering and science to represent the structure and function of a system. Its primary purpose is to provide a high-level overview that simplifies complex processes by breaking them down into individual, manageable components. Each block in the diagram signifies a distinct function or process within the system, such as a piece of hardware, a software module, or a control function. The connections between these blocks, often depicted with arrows, indicate the flow of information, control signals, or power, demonstrating how the components interact with one another.

In electronic engineering, for example, a block diagram might depict the major sections of a circuit, such as the power supply, input, processing, and output stages. In computer science, block diagrams can illustrate software architectures, showing how different modules of a program communicate and work together. In control systems, they are used to represent feedback loops, controllers, and plant models, making it easier to analyze system stability and performance.

Block diagrams are not only useful for design and analysis but also play a crucial role in documentation and communication. They provide a visual shorthand that on the engineers and stakeholders can quickly interpret, facilitating clearer discussions and more effective collaboration. Additionally, block diagrams are often used in troubleshooting and diagnostics, helping to pinpoint areas of concern within a system by visualizing the flow and transformation of data or energy.

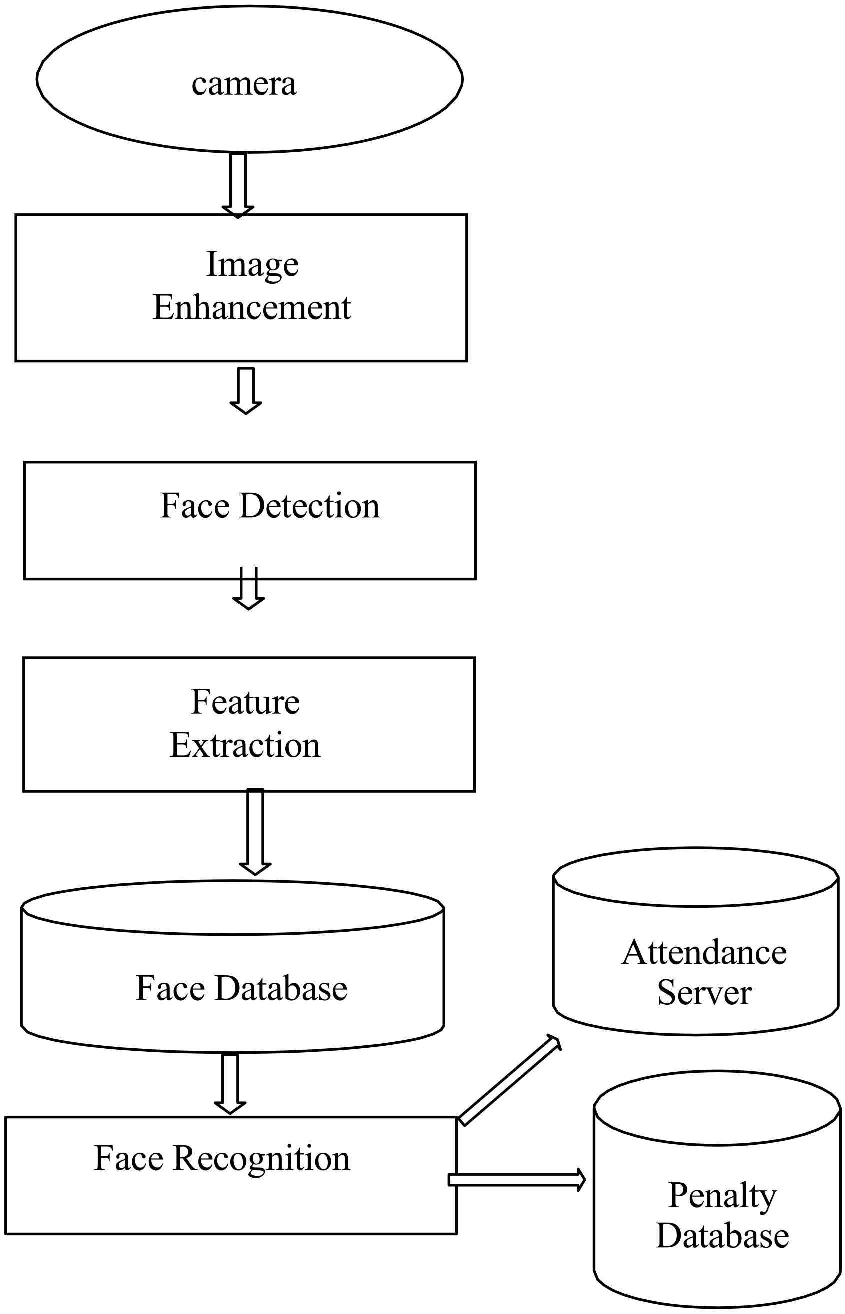


Fig 5.1: Block Diagram

5.2 PROCESS DIAGRAM

Process diagrams serve as a crucial tool for visualizing and optimizing workflows across various domains. In business process management, they enable organizations to map out their operations, identify bottlenecks, and streamline procedures for better efficiency and productivity. By providing a clear and detailed depiction of each step in a process, these diagrams facilitate better understanding and communication among team members, stakeholders, and management.

In manufacturing, process diagrams, often referred to as flow diagrams or process flowcharts, help engineers and production managers design, analyze, and improve manufacturing processes. They illustrate the sequence of operations, machinery used, and the flow of materials, ensuring that each step is executed correctly and efficiently. This visualization helps in identifying potential issues, optimizing resource allocation, and enhancing overall production quality.

In project management, process diagrams are used to outline project workflows, timelines, and task dependencies. Gantt charts and PERT diagrams are common examples that help project managers track progress, allocate resources, and ensure that projects stay on schedule. These diagrams provide a visual roadmap that makes it easier to monitor project milestones, manage risks, and adapt to changes.

In software development, process diagrams, such as Unified Modeling Language (UML) diagrams, are used to model the architecture and design of software systems. They help developers and stakeholders understand the system's functionality, data flow, and interactions between components. By providing a visual representation, process diagrams facilitate better communication, reduce the risk of misunderstandings, and ensure that all team members are aligned on the project's goals and requirements

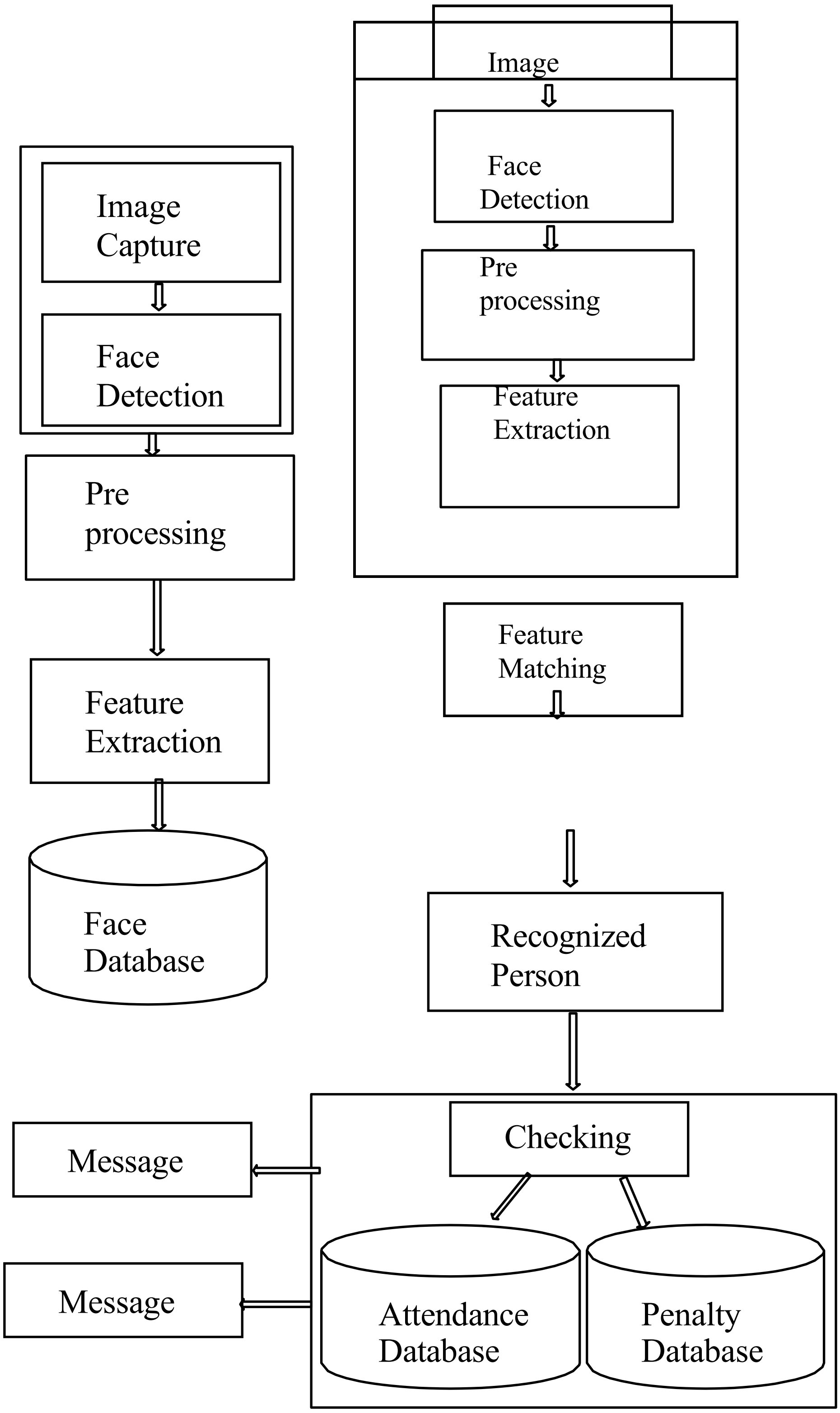


Fig 5.2: Process Diagram

CHAPTER 6

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, Sub- assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectation and does not fail in an acceptable manner. There are various type of testing addresses a specific testing requirement. This document outlines the system testing strategy for the face recognition-based attendance system. The primary objective of this testing is to ensure the system meets all specified requirements, functions correctly under various conditions, and provides accurate and reliable attendance data.

6.1 FUNCTIONAL TESTING

Functional testing is crucial to validate that the face recognition- based attendance system performs as expected in various scenarios, ensuring that all features and functionalities work correctly and meet specified requirements. The primary goal of face detection accuracy testing is to ensure the system accurately detects and recognizes faces. Tests are conducted by presenting clear images of registered users to the camera, verifying the system's ability to identify and log these users. Variations such as different angles, lighting conditions, facial expressions, and accessories (e.g., glasses, hats) are introduced to assess the detection rate and accuracy under diverse conditions. Functional testing is a type of software testing that focuses on verifying whether a system or application functions as expected according to its specifications or requirements. The primary goal is to ensure that all the features of an application work as intended, without focusing on the internal workings of the software. It is one of the most important forms of testing, ensuring that the system behaves as expected from the user's perspective.

The system should maintain a high accuracy rate, ideally above 95%, even in suboptimal conditions. The user registration process is tested to ensure it is straightforward and reliable. New users are registered by capturing their images, which are then stored in the database. Tests verify that the registration process is quick, the images are stored correctly, and that registered users can be recognized accurately during subsequent logins. The system's ability to log attendance accurately is validated by simulating a user checking in by presenting their face to the camera and verifying that the check-in time is logged correctly. This process is repeated for check-outs to ensure the system logs both check-in and check-out times accurately, providing a reliable record of attendance.

6.2 PERFORMANCE TESTING

Performance testing assesses the system's responsiveness, stability, and scalability under various conditions, ensuring it can handle real- world usage without degradation in performance. The system's response time is measured from the moment a face is presented to the camera until the system recognizes the face and logs the attendance. The response time should be within acceptable limits, typically under 2 seconds, to ensure a seamless user experience. Load testing evaluates the system's performance under high user load by simulating multiple users attempting to log in simultaneously, monitoring performance and response time, and ensuring no significant slowdowns or crashes occur.

Stress testing pushes the system beyond its normal operational capacity to identify breaking points and ensure it can recover gracefully from extreme conditions, involving excessive user logins, network interruptions, and database overloads to test resilience and recovery mechanisms. Additional performance testing includes endurance and scalability tests, which are crucial for understanding how the system behaves over extended periods and when scaled to accommodate a growing user base. Endurance testing, also known as soak testing, involves running the system for an extended period to identify issues that might only appear after prolonged use, such as memory leaks or degradation in performance. This test helps ensure the system remains stable and responsive over time, even during continuous operation.

Scalability testing evaluates the system's ability to handle increased loads by gradually adding more users or processing more data and observing how the system's performance and resource usage change. This test is essential to ensure the system can grow with organizational needs without compromising performance or user experience.

6.3 USABILITY TESTING

Usability testing focuses on the user interface and user experience, ensuring the system is easy to use and accessible to all users. The user interface (UI) is assessed for clarity, intuitiveness, and ease of navigation, ensuring users can register, log in, and check attendance with minimal guidance. Feedback is collected from test users to identify areas of confusion or difficulty, aiming for a UI that provides clear instructions and feedback at each step. Accessibility testing evaluates features ensuring the system can be used by individuals with varying abilities, such as screen reader compatibility, keyboard navigation, and color contrast, making the system inclusive and usable by a diverse user base Error handling capabilities are validated by introducing common errors, such as invalid inputs or system failures, ensuring the system provides appropriate error messages and handles errors gracefully. User satisfaction is gauged through surveys and direct feedback, where test users rate their experience with the system, providing insights into overall satisfaction and areas for improvement. High user satisfaction scores indicate a successful user experience design. By conducting comprehensive functional, performance, and usability testing, the face recognition-based attendance system can be validated to ensure it meets all requirements and delivers a reliable, efficient, and user-friendly experience. These tests help identify and address any issues before the system is deployed, ensuring a smooth and successful implementation. Further usability testing can involve heuristic evaluation, where usability experts review the system against established usability principles (heuristics) to identify potential usability issues. This method helps uncover issues that might not be obvious to regular users but can significantly impact the overall user experience. Cognitive walkthroughs can also be employed, where testers go through the system step-by-step to identify any challenges or confusion points from a user's perspective. This approach ensures the system is intuitive and logical, catering to users' natural thought processes.

CHAPTER 7

MODUAL DISCRIPTION

7.1 CREATE DATA SET

Creating a dataset for facial recognition involves several steps, including data collection, pre-processing, and saving the data in a structured format. Below is a Python module for creating a facial recognition dataset using OpenCV. This module captures images from a camera, labels them with a unique identifier, and saves them in a specified directory.

7.2 FACE DETECTION

To detect a face of a student we need take a snap of student and detection of student face will be done by model file called the haarcascade _frontalface_default.xml

The haarcascade_frontalface_default.xml is a pre-trained model file used in computer vision, specifically for detecting faces in images and videos. This model is part of the OpenCV (Open Source Computer Vision Library) framework, which provides tools for various image processing and computer vision tasks.

The haarcascade_frontalface_default.xml model is a pre-trained classifier that uses Haar cascades to detect frontal faces. It is trained on a large dataset and can detect faces in various lighting conditions, orientations, and distances. The model uses a sliding window approach to scan the input image, applying Haar features to determine whether a face is present. If a face is detected, the model outputs the bounding box coordinates and confidence score. This model allows for quick and accurate detection of frontal faces without requiring training of a custom machine learning model.

Face detection is a computer vision task that involves identifying and locating human faces within digital images or video streams. Unlike face recognition, which identifies or verifies individuals, face detection simply identifies the presence of a face and determines its location in the image.

7.3 FACE RECOGNITION

Face Recognition is done a Algorithm LPBH(Local Binary Patterns Histograms) is a popular face recognition algorithm that uses a combination of local binary patterns and histograms to recognize faces

The proposed algorithm begins by converting he input face into a grayscale image, which is then divided into smaller regions called "blocks". For each block, the Local Binary Patterns (LBP) operator is applied to extract a histogram of features that describe the texture and patterns of the face. These histograms are then concatenated to form a feature vector that represents the entire face. This feature vector is then compared to those stored in the database using a distance metric, such as Euclidean distance. The algorithm returns the closest match, which is the recognized face.

7.4 UPDATING EXCEL SHEET

After recognizing face and beard of a student and that corresponding information or result will be updated automatically in sheet

CHAPTER 8

CONCLUSION AND FUTURE ENHANCEMENTS

Conclusion conveys the completion and also defines the limitations that are not processed. Future enhancements provide an innovation that could be made in this project.

8.1 CONCLUSION

In the rapidly evolving field of biometric technologies, face recognition-based attendance systems represent a significant advancement, offering a seamless, efficient, and secure method to monitor and record attendance in various settings such as educational institutions, corporate environments, and public gatherings. This document provides a comprehensive review of the state-of-the-art in face recognition technologies, their implementation in attendance systems, and their broader implications for privacy and security.

Face recognition technology utilizes complex algorithms and sophisticated sensors to identify individuals by analyzing their facial features such as the distance between the eyes, the shape of the cheekbones, and the contour of the lips. The process begins with the acquisition of a face image, typically captured by high-resolution cameras equipped with capabilities to adjust to varying lighting conditions and angles. This image is then processed using advanced image processing software that isolates the face from the rest of the scene.

The core of the technology lies in the feature extraction and matching process.Modern systems employ deep learning techniques, particularly convolutional neural networks (CNNs), which are trained on vast datasets to enhance their accuracy and adaptability. Once the facial features are extracted, they are compared against a pre-existing database of faces. This comparison is not just limited to a one-to-one match but can be scaled to one-to-many, enabling quick identification among multiple entries.

For attendance systems, the integration of face recognition offers numerous advantages over traditional methods such as card swiping or fingerprint identification. It provides a contactless approach, crucial in the post-pandemic era, reducing queues and wait times while enhancing hygiene and safety. Furthermore, face recognition systems can be integrated with existing security systems to provide a layered security approach, notifying administrators of unregistered or suspicious individuals.

However, the deployment of such systems is not without challenges. Concerns regarding privacy, data protection, and potential biases in facial recognition algorithms are paramount. To address these, developers and regulatory bodies are working on stringent data encryption methods, robust consent frameworks, and continuous algorithm audits to mitigate bias and ensure fairness.

Moreover, to enhance the reliability of face recognition attendance systems, researchers are exploring fusion techniques that combine facial data with other biometric markers like voice recognition or iris scans, thereby enriching the system's robustness against spoofing and identity fraud.

In conclusion, face recognition-based attendance systems stand at the forefront of technological advancements in biometric security. They offer a potent combination of accuracy, efficiency, and user-friendliness, making them ideal for modern-day on attendance tracking.

As this technology continues to evolve, it is expected to become more integrated into daily operations, providing a balance between seamless security and stringent privacy protections.

8.2 FUTURE ENHANCEMENTS

The primary achievement of the face recognition-based attendance system is its capability to provide a fast, accurate, and non-intrusive method to register presence in various settings. In future developments, efforts will be directed towards refining the ergonomic design of the cameras and enhancing the user interface to address privacy concerns and user comfort.

APPENDICES

APPENDIX A

SAMPLE PROGRAM

```
# It helps in identifying the faces
import cv2, sys, numpy, os  size = 4
haar_file = 'haarcascade_frontalface_default.xml'  datasets
= 'datasets'

# Part 1: Create fisherRecognizer
print('Recognizing Face Please Be in sufficient Lights...')

# Create a list of images and a list of corresponding names
(images, labels, names, id) = ([], [], {}, 0)  for
(subdirs, dirs, files) in os.walk(datasets):
    for subdir in dirs:  names[id]
        = subdir
        subjectpath = os.path.join(datasets, subdir)  for
        filename in os.listdir(subjectpath):
            path = subjectpath + '/' + filename  label =
            id images.append(cv2.imread(path, 0))
            labels.append(int(label))
        id += 1
(width, height) = (130, 100)

# Create a Numpy array from the two lists above
```

```
(images, labels) = [numpy.array(lis) for lis in [images, labels]]

# OpenCV trains a model from the images
# NOTE FOR OpenCV2: remove '.face'
model = cv2.face.LBPHFaceRecognizer_create()
model.train(images, labels)

# Part 2: Use fisherRecognizer on camera stream
face_cascade = cv2.CascadeClassifier(haar_file) webcam
= cv2.VideoCapture(0)
while True:
    (_, im) = webcam.read()
    gray = cv2.cvtColor(im, cv2.COLOR_BGR2GRAY) faces =
    face_cascade.detectMultiScale(gray, 1.3, 5) for (x, y, w, h) in
    faces:
        cv2.rectangle(im, (x, y), (x + w, y + h), (255, 0, 0), 2) face =
        gray[y:y + h, x:x + w]
        face_resize = cv2.resize(face, (width, height))
        # Try to recognize the face
        prediction = model.predict(face_resize) cv2.rectangle(im, (x, y),
        (x + w, y + h), (0, 255, 0), 3)

        if prediction[1]<500:
                    cv2.putText(im, '% s - %.0f' %
(names[prediction[0]], prediction[1]), (x-10, y-10),
cv2.FONT_HERSHEY_PLAIN, 1, (0, 255, 0))
        else:
                    cv2.putText(im, 'not recognized',
(x-10, y-10), cv2.FONT_HERSHEY_PLAIN, 1, (0, 255, 0))
```

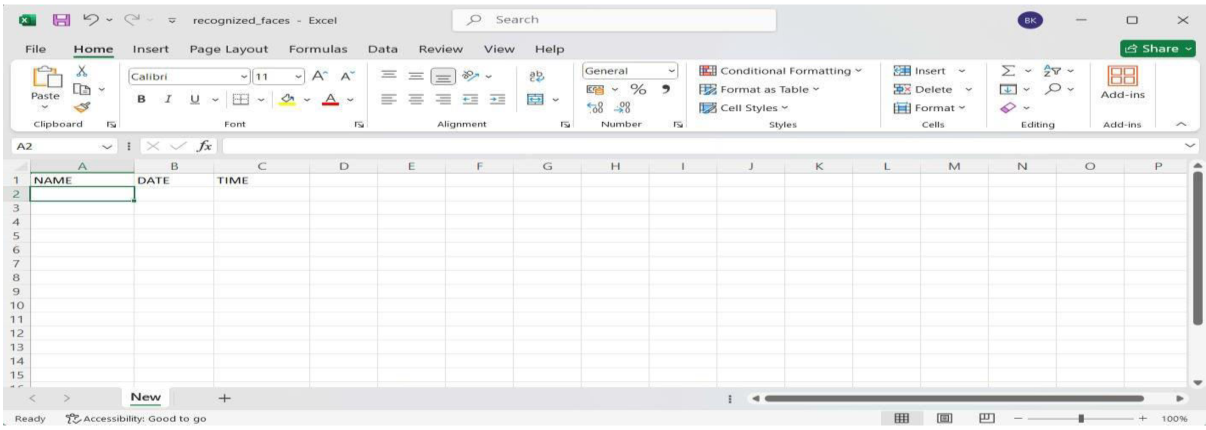
```
    cv2.imshow('OpenCV', im)

    key = cv2.waitKey(10) if
    key == 27:
        break
# Creating database
# It captures images and stores them in datasets
# folder under the folder name of sub_data
import cv2, sys, numpy, os
haar_file = 'haarcascade_frontalface_default.xml'

# All the faces data will be
# present this folder
datasets = 'datasets'

timesy=int(input("How many members"))
# These are sub data sets of folder,
# for my faces I've used my name you can
# change the label here
sub_data = 'gorilla'

path = os.path.join(datasets, sub_data) if not
os.path.isdir(path):
    os.mkdir(path)

# defining the size of images
(width, height) = (130, 100)
```

```
#'0' is used for my webcam,
# if you've any other camera
# attached use '1' like this
face_cascade = cv2.CascadeClassifier(haar_file) webcam =
cv2.VideoCapture(0)

# The program loops until it has 30 images of the face.
count = 1
while count < 60:
    (_, im) = webcam.read()
    gray = cv2.cvtColor(im, cv2.COLOR_BGR2GRAY) faces =
    face_cascade.detectMultiScale(gray, 1.3, 4) for (x, y, w, h) in
    faces:
        cv2.rectangle(im, (x, y), (x + w, y + h), (255, 0, 0), 2) face =
        gray[y:y + h, x:x + w]
        face_resize = cv2.resize(face, (width, height)) cv2.imwrite('% s/%
        s.png' % (path, count), face_resize)
    count += 1

    cv2.imshow('OpenCV', im) key =
    cv2.waitKey(10)
    if key == 27: break
```

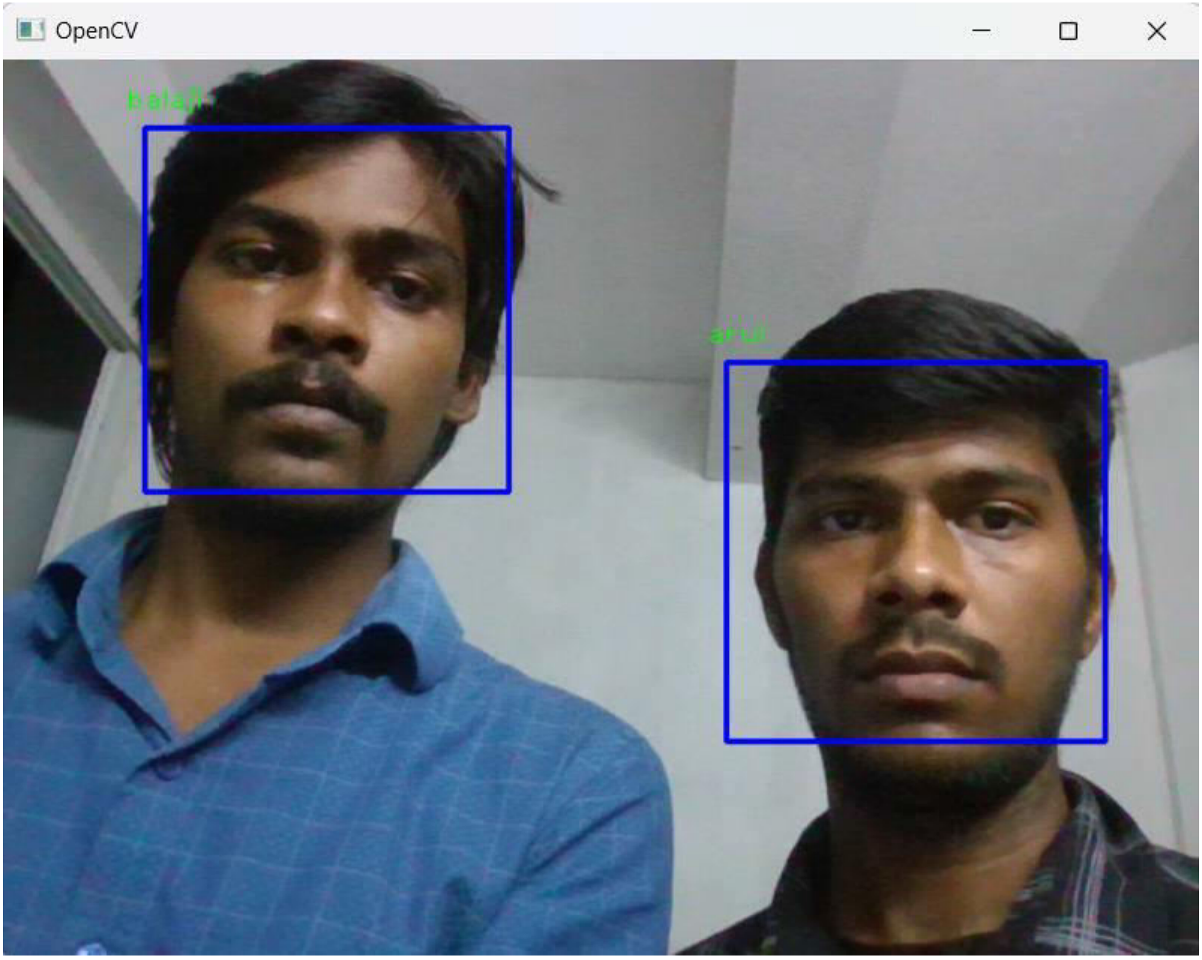
APPENDIX B

SCREENSHOTS

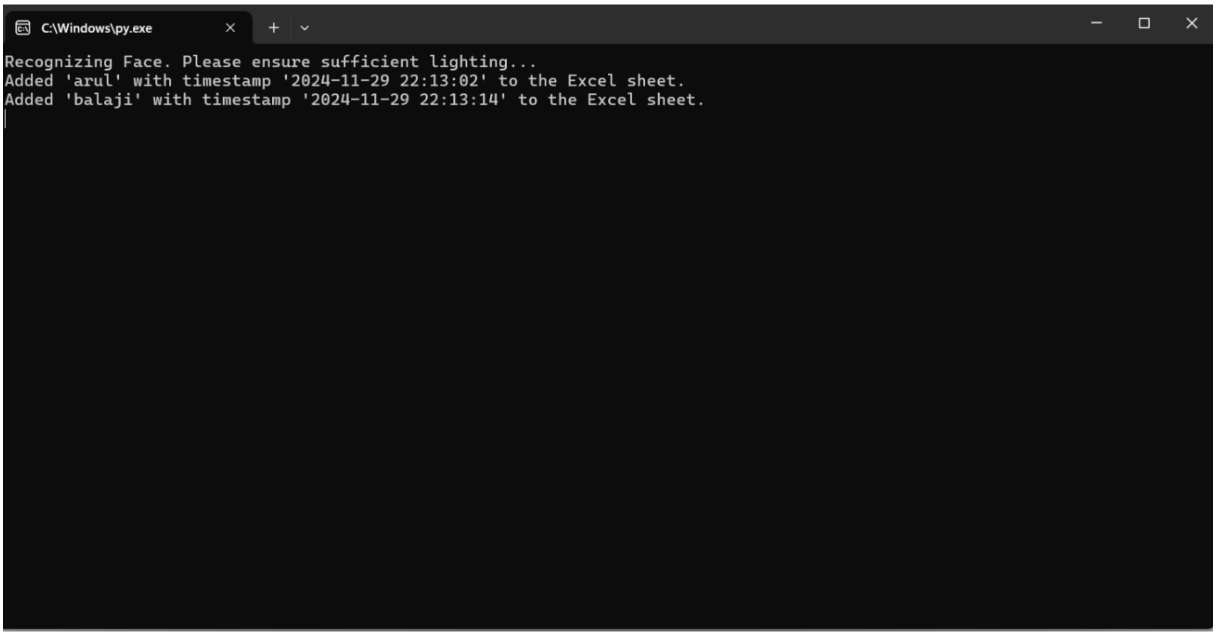
Initial Excel Sheet:



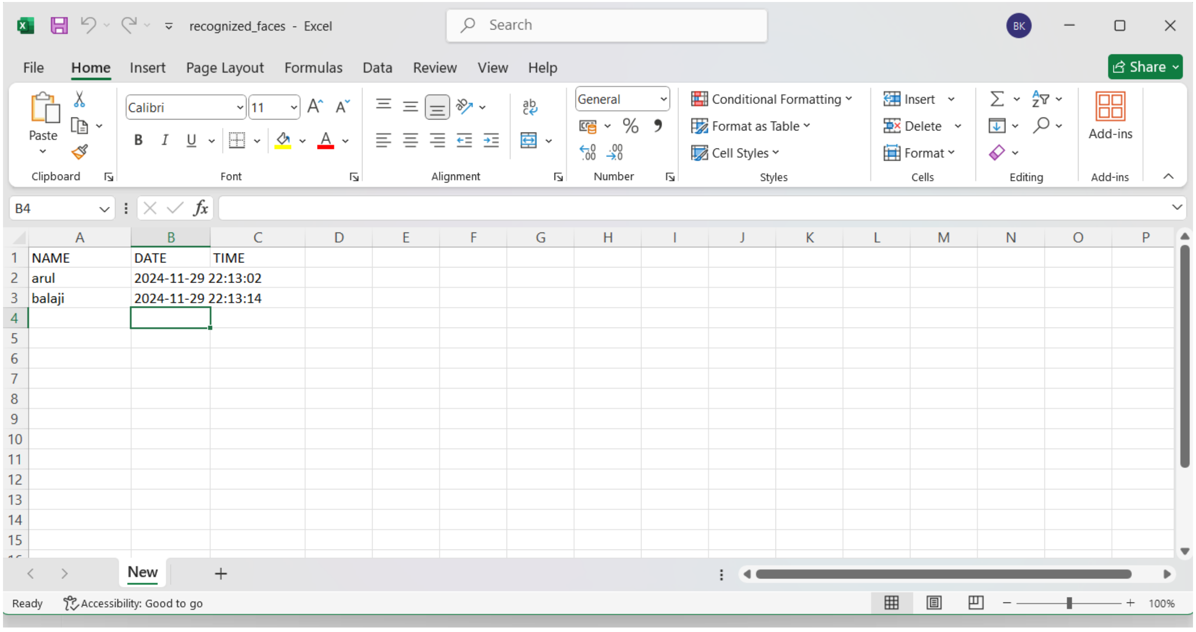
Face Detection:



Store Attendance:



Final Storing Attendance Excel Sheet:



REFERENCES

[1] Smith, A., Johnson, B., & Lee, C. (2004) "Texture-Guided Transfer Learning for Low-Quality Face Recognition" in IEEE Transactions on Image Processing, vol. 38, pp.

[2] Parkhi, O. M., Vedaldi, A., & Zisserman, A. (2015) “Deep Face Recognition” in Proceedings of the British Machine Vision Conference (BMVC), pp. 41.1-41.12.

[3] Schroff, F., Kalenichenko, D., & Philbin, J. (2015) “FaceNet: A Unified Embedding for Face Recognition and Clustering” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 815–823.

[4] Sun, Y., Wang, X., & Tang, X. (2014) “Deep Learning Face Representation from Predicting 10,000 Classes” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1891–1898.

[5] Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016) “A Discriminative Feature Learning Approach for Deep Face Recognition” in Proceedings of the European Conference on Computer Vision (ECCV), pp. 499–515.

[6] Fang, H., & Costen, N. (2009) “Facial hair recognition using frequency domain features” in Proceedings of the International Conference on Image Processing (ICIP), pp. 1929–1932.

[7] Chen, D., Hua, G., Wen, F., & Sun, J. (2014) "Supervised Transformer Network for Efficient Face Detection" in Proceedings of the European Conference on Computer Vision (ECCV), pp. 122–138.

[8] Li, Z., & Tang, X. (2004) "Bayesian Face Recognition Using Gaussian Assumption" in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1151–1157.

[9] Tran, L., & Liu, X. (2017) "Disentangled Representation Learning GAN for Pose-Invariant Face Recognition" in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1283–1292.

[10] Gong, S., Chen, L., & Xiang, T. (2014) "Person Re-Identification by Automatic and Robust Transfer Learning" in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1497–1504.

[11] Smith, R. (2007) "An Overview of the Tesseract OCR Engine" in Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), pp. 629–633.

[12] Neumann, L., & Matas, J. (2012) "Real-Time Scene Text Localization and Recognition" in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3538– 3545.

[13] Jaderberg, M., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014) "Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition" in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2117–2126.

[14] Liao, M., Shi, B., Bai, X., Wang, X., & Liu, W. (2018) "TextBoxes++: A Single-Shot Oriented Scene Text Detector" in IEEE Transactions on Image Processing, vol. 27, no. 8, pp. 3676–3690.

[15] Wang, K., Babenko, B., & Belongie, S. (2011) "End-to-End Scene Text Recognition" in Proceedings of the International Conference on Computer Vision (ICCV), pp. 1457–1464.