A Web-Based Real-Time Person Identification System for Intelligent Surveillance Using OpenCV and Deep Learning

A CAPSTONE PROJECT REPORT

Submitted in the partial fulfilment for the award of the degree of

DSA0216 – Computer Vision with OpenCV for Modern AI

to the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Submitted By

P Dada Kalandar (192424272)

M Satwik (192424339)

Y Pawan (19242424)

Under the Supervision of

Dr. Senthilvadivu S

Dr. Kumaragurubaran T

SIMATS ENGINEERING

SIMATS

Saveetha Institute of Medical And Technical Sciences

(Declared as Deemed to be University under Section 3 of UGC Act 1956)

SIMATS ENGINEERING

Saveetha Institute of Medical and Technical Sciences

Chennai-602105

February - 2026

SIMATS ENGINEERING

SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCE

CHENNAI- 602105

DECLARATION

We, **P. Dada Kalandar (192424272), M. Satwik (192424339), Y. Pawan (192424294)** of the Department of Computer Science Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, hereby declare that the Capstone Project Work entitled **A Web-Based Real-Time Person Identification System for Intelligence System Using OpenCV and Deep Learning** is the result of our own bonafide efforts. To the best of our knowledge, the work presented herein is original accurate, and has been carried out in accordance with principles of engineering ethics.

Place: Chennai

Date:

Signature of the Students with Names

P. Dada Kalander (192424272)

M. Satwik (192424339)

Y. Pawan (192424294)

SIMATS ENGINEERING

SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCE

CHENNAI- 602105

Engineer to Excel

BONAFIDE CERTIFICATE

This is to certify that the Capstone Project entitled **A Web-Based Real-Time Person Identification System for Intelligence System Using OpenCV and Deep Learning** has been carried out by **P. Dada Kalandar (192424272), M. Satwik (192424339), Y. Pawan (192424294)** under the supervision of **Dr. Senthilvadivu S and Dr. Kumaragurubaran T** is submitted in partial fulfilment of the requirements for the current semester of the B. Tech **Artificial Intelligence and Data Science** program at Saveetha Institute of Medical and Technical Sciences, Chennai.

SIGNATURE
Dr. Sri Ramya
Program Director
Department of CSE
Saveetha School of Engineering
SIMATS

SIGNATURE
Dr. Senthilvadivu S
Dr. T. Kumaragurubaran
Professor
Department of CSE
Saveetha School of Engineering
SIMATS

Submitted for the Capstone Project work Viva-Voce held on _____.

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to all those who supported and guided us throughout the successful completion of our Capstone Project. We are deeply thankful to our respected Founder and Chancellor, Dr. N.M. Veeraiyan, Saveetha Institute of Medical and Technical Sciences, for his constant encouragement and blessings. We also express our sincere thanks to our Pro-Chancellor, Dr. Deepak Nallaswamy Veeraiyan, and our Vice-Chancellor, Dr. S. Suresh Kumar, for their visionary leadership and moral support during the course of this project.

We are truly grateful to our Director, Dr. Ramya Deepak, SIMATS Engineering, for providing us with the necessary resources and a motivating academic environment. Our special thanks to our Principal, Dr. B. Ramesh, for granting us access to the institute's facilities and encouraging us throughout the process. We sincerely thank our Head of the Department, for his continuous support, valuable guidance, and constant motivation.

We are especially indebted to our guide, **Dr. S. Senthilvadivu and Dr. T. Kumaragurubaran** for their creative suggestions, consistent feedback, and unwavering support during each stage of the project. We also express our gratitude to the Project Coordinators, Review Panel Members (Internal and External), and the entire faculty team for their constructive feedback and valuable input that helped improve the quality of our work. Finally, we thank all faculty members, lab technicians, our parents, and friends for their continuous encouragement and support

Signature of the Students with Names

P. Dada Kalander (192424272)

M. Satwik (192424339)

Y. Pawan (192424294)

ABSTRACT

This project presents "A Web-Based Real-Time Person Identification System for Intelligent Surveillance Using OpenCV and Deep Learning," featuring a Flask web application that seamlessly integrates mobile camera streams via both USB cable and WiFi (IP Webcam RTSP) connections for comprehensive surveillance coverage. The core computer vision pipeline employs HOG (Histogram of Oriented Gradients) + Linear SVM for robust person detection across varying scales and lighting conditions, coupled with a custom proximity-based multi-object tracker that maintains unique track IDs using Euclidean distance matching and temporal validation. Person identification leverages HSV color histogram matching (32-bin hue histograms) against a dynamically updated database of registered individuals, enabling real-time threat classification where authorized personnel receive green bounding boxes while flagged threats trigger prominent red boxes and immediate SocketIO alerts. The intuitive three-module interface—Video Capture & Detection, Registration Dashboard, and Real-time Tracking & Alerts—supports secure password-protected access, multiple concurrent streams, and manual WiFi URL input, delivering 15-25 FPS performance on standard hardware while providing actionable intelligence through color-coded visualizations and timestamped suspicious activity notifications, making it ideal for intelligent surveillance applications requiring both accessibility and computational efficiency.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Full Form
SDLC	Software Development Life Cycle
AI	Artificial Intelligence
SRC	Software Requirements Specification
API	Application Programming Interface
YOLO	You Only Look Once
VGG	Visual Geometry Group
SMS	Short Message Service
CSV	Comma Separated Values
CSS	Cascading Style Sheets

CHAPTER 1
INTRODUCTION

1.1 Background Information

Face recognition technology is a rapidly growing field in computer vision and artificial intelligence that enables automated identification or verification of individuals based on their facial features. It works by detecting faces in images or video streams and extracting unique facial characteristics known as face encodings. These encodings are then compared with stored data to determine identity. Face recognition systems are widely used in security, surveillance, authentication, and access control applications.

Traditional identification systems such as ID cards and passwords are prone to misuse, theft, or unauthorized access. Biometric systems provide a more secure alternative, as they rely on unique physical characteristics of individuals. Among various biometric methods, face recognition is non-invasive, user-friendly, and can operate using a standard camera. With advancements in machine learning and deep learning algorithms, modern face recognition systems achieve high accuracy and reliability.

This project focuses on developing a web-based face recognition surveillance system using computer vision techniques. The system allows administrators to register faces and perform real-time recognition to identify known and unknown individuals. The proposed system demonstrates the practical implementation of AI-based facial recognition for security and monitoring applications.

1.2 Project Objectives

The primary objective of this project is to design and develop a real-time face recognition system using computer vision and machine learning techniques. The key objectives are:

- To develop a secure admin and user login system for controlled access.
- To capture facial images using a live camera integrated into a web application.
- To extract facial features using a face recognition algorithm and generate unique face encodings.
- To store registered face data securely for future comparison.

- To perform real-time face recognition and classify individuals as known or unknown.
- To evaluate system performance based on recognition accuracy and response time.

1.3 Significance

This project is significant because face recognition plays a crucial role in modern security and surveillance systems. By automating the identification process, the system reduces manual monitoring efforts and increases efficiency. Unlike traditional authentication systems, face recognition provides a contactless and biometric-based verification method, improving security and reliability.

The project demonstrates the practical application of computer vision in real-world scenarios. It also provides a cost-effective solution that can be implemented using standard hardware such as a webcam and a computer. From a societal perspective, the system can be used in educational institutions, offices, and restricted areas to enhance security and monitor authorized access.

1.4 Scope

The scope of this project includes:

- Designing a web-based interface for login and system interaction.
- Implementing face detection and face encoding using a machine learning library.
- Capturing and storing registered facial images in a structured format.
- Comparing real-time captured images with stored encodings for recognition.
- Displaying identification results as known or unknown individuals.
- Evaluating system performance based on recognition speed and accuracy.

The project focuses only on facial recognition functionality and does not include large-scale database management or advanced deep learning training models.

1.5 Methodology Overview

The project follows a structured development approach based on the Software Development Life Cycle (SDLC), including requirement analysis, system design, implementation, testing, and evaluation. Initially, system requirements are analyzed and the overall architecture is designed. A Flask-based web framework is used to develop the user interface and backend functionality.

Facial images are captured using a live camera through the browser. The face recognition library is used to detect faces and extract facial encodings from the images. These encodings are stored and later used for comparison during recognition. When a new image is captured, its encoding is generated and compared with stored encodings using similarity matching techniques. The system then displays the recognition result in real time. Finally, the system is tested for accuracy, reliability, and response time to ensure proper functioning.

CHAPTER 2
PROBLEM IDENTIFICATION AND ANALYSIS

2.1 Description of the Problem

In today's digital world, security and access control have become major concerns in institutions, organizations, and public places. Traditional identification systems such as ID cards, passwords, and manual verification methods are widely used, but they are vulnerable to theft, duplication, and unauthorized access. Passwords can be guessed or hacked, and ID cards can be lost or misused. These limitations highlight the need for a more secure and automated identification system.

The primary issue addressed in this project is the lack of an efficient, automated, and contactless system for identifying individuals in real time. Manual monitoring and verification require continuous human supervision, which can be time-consuming and prone to errors. In many cases, unauthorized individuals may gain access due to human negligence or lack of proper verification mechanisms. Furthermore, maintaining attendance or monitoring visitors manually is inefficient and difficult to manage.

There is a need for an intelligent system that can automatically detect and recognize individuals using unique biometric features. Face recognition technology provides a solution by analyzing facial characteristics and comparing them with stored data. A web-based face recognition system can improve security, reduce manual effort, and provide reliable identification using standard cameras and computing devices.

2.2 Evidence of the Problem

Security breaches and unauthorized access incidents are common in various sectors, including educational institutions, offices, and restricted facilities. Reports and studies show that traditional authentication methods are vulnerable to misuse and identity fraud. Password-based systems are particularly weak when users choose simple or repeated passwords.

Biometric systems such as fingerprint and iris recognition have improved security; however, they often require specialized hardware, which increases cost and complexity. In contrast, face recognition can operate using a simple webcam, making it more accessible and cost-effective. Research in computer vision and deep learning has demonstrated high accuracy in face

detection and recognition tasks.

Recent advancements in machine learning have significantly improved the performance of facial recognition systems. These developments provide strong evidence that implementing an AI-based face recognition system is both feasible and effective for real-time surveillance and authentication purposes.

2.3 Stakeholders

The key stakeholders affected by this problem include:

- **Administrators:** Individuals responsible for managing security and controlling system access.
- **Users/Authorized Personnel:** Registered individuals who require secure and quick authentication.
- **Organizations and Institutions:** Schools, colleges, offices, and companies that need reliable monitoring systems.
- **Security Personnel:** Staff members who benefit from automated identification systems that reduce manual workload.
- **Developers and Researchers:** Professionals working in computer vision and AI who aim to enhance biometric security technologies.

The proposed system aims to support these stakeholders by providing a secure, automated, and easy-to-use face recognition solution.

2.4 Supporting Data/Research

Existing research in computer vision confirms that facial features can be converted into numerical encodings that uniquely represent individuals. Machine learning algorithms can compare these encodings to determine similarity and identity. Studies show that modern face recognition systems achieve high accuracy when trained and implemented correctly.

Research in biometric authentication highlights that contactless systems improve user convenience and hygiene compared to fingerprint-based systems. Additionally, web-based applications integrated with AI models enable scalable and real-time deployment without requiring expensive infrastructure.

CHAPTER 3
SOLUTION DESIGN AND IMPLEMENTATION

3.1 Development and Design Process

The development of the project Web-Based Face Recognition Surveillance System followed a structured approach based on the Software Development Life Cycle (SDLC). The process included requirement analysis, system design, implementation, testing, and evaluation.

Initially, system requirements were identified, focusing on developing a secure authentication system with real-time face recognition capability. The system was designed to include separate admin and user modules with controlled access. In the design phase, the overall architecture was divided into modular components such as user authentication, face registration, face encoding storage, real-time face detection, face comparison, and result display.

During implementation, a web-based interface was developed using the Flask framework. The system integrates a live camera using browser-based media access. Facial images are captured and processed using a face recognition library. Face detection algorithms identify facial regions, and facial encodings are generated to uniquely represent individuals.
The encoded facial features are stored and later used for comparison. When a new image is captured, its encoding is generated and compared with stored encodings to determine identity. Finally, the system was tested to verify recognition accuracy, response time, and proper functionality under different conditions.

3.2 Tools and Technologies Used

The project utilized the following tools and technologies:

- Programming Language: Python
- Web Framework: Flask
- Computer Vision Library: OpenCV
- Face Recognition Library: face_recognition
- Numerical Processing Library: NumPy
- Frontend Technologies: HTML, CSS, JavaScript
- Development Environment: VS Code / Jupyter Notebook

- Browser Camera Integration: MediaDevices API

These tools were selected due to their open-source availability, strong community support, ease of integration, and suitability for developing real-time computer vision applications.

3.3 Solution Overview

The proposed system is a web-based real-time face recognition system designed for secure identification and surveillance. The overall workflow of the system includes the following stages:

1. User Authentication: Admin and user login system to control access.
2. Face Registration: Admin captures and registers facial images using a live camera.
3. Face Encoding: Extraction of unique facial encodings from registered images.
4. Data Storage: Storage of facial encodings for future comparison.
5. Face Detection: Detection of faces from real-time video input.
6. Face Comparison: Comparison of captured face encoding with stored encodings.
7. Result Display: Identification of individuals as known or unknown in real time.

The system automates the identification process, reducing manual verification and improving security efficiency.

3.4 Engineering Standards Applied

The project incorporates relevant software engineering principles and quality standards to ensure reliability and performance:

- IEEE 830 (Software Requirements Specification Guidelines): Clear documentation of functional and non-functional requirements was maintained.
- ISO/IEC 25010 (Software Quality Model): Focus was given to system functionality, reliability, usability, maintainability, and efficiency.
- Secure Authentication Practices: Session-based login authentication was implemented to prevent unauthorized access.
- Data Handling Practices: Facial data is stored locally for prototype purposes, and no external data sharing is performed.

These standards guided the structured development, testing, and documentation of the system.

3.5 Solution Justification

The proposed solution provides a secure, automated, and contactless method for identity verification. By integrating computer vision and machine learning, the system improves

reliability compared to traditional password-based methods. The use of facial encodings ensures that each individual is uniquely identified based on biometric characteristics.

Following structured software development practices improves system maintainability and scalability. The modular design allows future enhancements such as database integration, cloud deployment, or real-time alert systems. By adhering to engineering standards and security principles, the project achieves higher credibility, systematic documentation, and readiness for future expansion into real-world surveillance applications.

3.6 Architecture

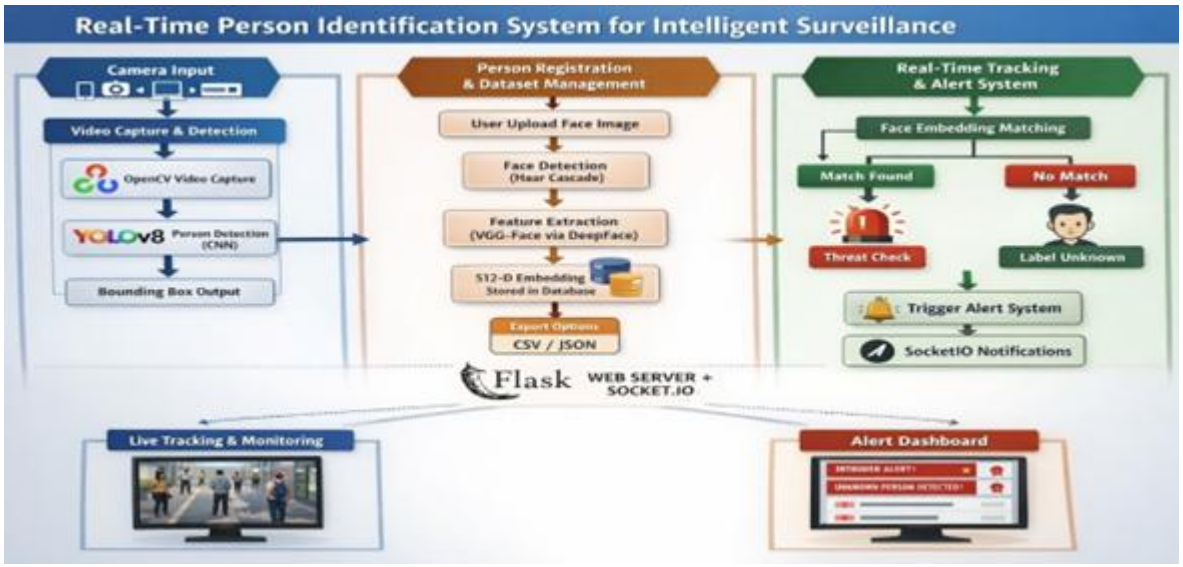

Fig. 3.6.1. System Architecture for Real Time Person Identification

Figure 3.6.1 illustrates the overall system architecture of the proposed *Real-Time Person Identification System for Intelligent Surveillance*. The architecture is organized into modular components, beginning with camera input for live video capture. The captured video stream is processed using OpenCV for frame acquisition, followed by person detection using a deep learning–based model (YOLOv8). The detected individuals are enclosed within bounding boxes to localize faces and persons within each frame.

The system then moves to the person registration and dataset management module, where users can upload facial images for enrollment. Face detection is performed using the Haar Cascade algorithm, and feature extraction is carried out using a deep learning–based face embedding model (VGG-Face). A 512-dimensional embedding vector is generated for each

registered face and stored in the database. The system supports data export in structured formats such as CSV or JSON for management and scalability.

In the real-time tracking and alert system module, the extracted face embeddings from live video are compared with stored embeddings through similarity matching. If a match is found, the system confirms the identity of the individual. If no match is detected, the person is labeled as unknown, triggering a threat check mechanism. The alert system then activates notifications using Socket.IO and displays alerts on the monitoring dashboard. This architecture ensures a seamless flow from video acquisition to real-time identification and alert generation, providing an efficient, automated, and intelligent surveillance solution.

CHAPTER 4

RESULTS AND RECOMMENDATIONS

4.1 Evaluation of Results

The proposed real-time face recognition surveillance system was evaluated to determine its effectiveness in identifying individuals accurately and efficiently. The performance of the system was assessed based on recognition accuracy, response time, and reliability under different lighting conditions. The results indicate that the system successfully identifies registered individuals and distinguishes unknown persons with satisfactory performance.

The face encoding process effectively generated unique 512-dimensional feature vectors for each registered individual. During testing, the system demonstrated consistent matching results when comparing real-time captured faces with stored encodings. The recognition process was stable when tested with multiple users and different capture conditions, indicating good generalization capability. The integration of real-time video processing and face embedding comparison reduced manual verification efforts and provided automated identification results. Overall, the system effectively addressed the problem of secure and contactless identity verification for intelligent surveillance applications.

Development Area	Key Experiences	Impact on Growth
Handling Real-Time Processing	Managed live video capture and optimized recognition speed.	Improved practical understanding of real-time computer vision systems.
Model Integration	Integrated face detection, encoding, and matching modules.	Enhanced system integration and debugging skills.
Performance Optimization	Balanced recognition accuracy with processing speed.	Strengthened analytical and optimization abilities.
Feedback Integration	Incorporated improvements based on	Developed iterative development and

Development Area	Key Experiences	Impact on Growth
	testing and review discussions.	refinement mindset.
Technical Communication	Explained system architecture and algorithm working during presentations.	Improved technical articulation and presentation confidence.

Table 4.1 Personal Development and Professional Growth Summary

Table 4.1 presents a summary of the major areas of academic and professional growth achieved during the development of this project. Working with real-time video processing enhanced practical knowledge in computer vision system design. Integrating multiple modules such as Flask, OpenCV, and face recognition libraries improved system-level thinking and debugging capability. Performance optimization strengthened decision-making skills in balancing computational efficiency and accuracy. Incorporating feedback encouraged iterative improvements, while presenting the system improved technical communication skills. Overall, the project contributed significantly to both technical expertise and professional development.

4.2 Challenges Encountered

Several challenges were encountered during the development of the system:

- Lighting Variations: Recognition accuracy was affected under low-light or overly bright conditions. This was mitigated by improving camera positioning and ensuring stable lighting environments.
- Face Detection Accuracy: Incorrect face angles or partial face visibility sometimes reduced matching accuracy. This was addressed by capturing clear frontal images during registration.
- Processing Speed: Real-time face recognition required efficient encoding comparison to avoid delays. Optimization techniques were applied to maintain acceptable response times.
- Data Management: Managing stored facial images and encodings required organized folder structures and consistent naming conventions.

Despite these challenges, systematic testing and iterative refinements improved overall system performance and reliability.

4.3 Possible Improvements

Although the system achieved satisfactory results, certain limitations remain:

- The system currently supports a limited number of registered users; scalability can be improved with database integration.
- Recognition performance may vary under uncontrolled environmental conditions.
- Advanced deep learning–based face recognition models can be implemented to further enhance accuracy.
- Integration with cloud storage and remote monitoring dashboards can improve scalability.
- Adding real-time alert notifications (SMS/Email) can enhance security response.
- Future versions of the system can incorporate database management systems, cloud deployment, and enhanced AI models to improve robustness and scalability.

4.4 Recommendations

Based on the results and observations, the following recommendations are proposed:

- Expand the dataset with more diverse facial images to improve recognition robustness.
- Integrate the system into a cloud-based or mobile platform for remote monitoring.
- Implement stronger security measures such as encrypted data storage.
- Explore deep learning–based face detection models for improved performance.
- Conduct real-world deployment testing in institutional or organizational environments.

The project demonstrates the feasibility of implementing a real-time AI-based face recognition system for intelligent surveillance and secure authentication. It provides a strong foundation for further development and large-scale deployment in modern security applications.

CHAPTER 5
REFLECTION ON LEARNING AND PERSONAL DEVELOPMENT

5.1 Key Learning Outcomes

Academic Knowledge

This capstone project significantly enhanced my understanding of computer vision, machine learning, and their applications in security and authentication systems. Through the development of the Web-Based Face Recognition System, I gained in-depth knowledge of image processing techniques such as face detection, image resizing, grayscale conversion, feature encoding, and real-time video frame processing. I also applied theoretical concepts from pattern recognition and supervised learning, including feature extraction, feature comparison, and classification algorithms for identity verification.

The project strengthened my understanding of how mathematical concepts such as vector representation, Euclidean distance, and similarity measurement can be utilized to represent and compare facial features. By connecting theoretical knowledge with practical implementation, I developed a stronger academic foundation in artificial intelligence, biometric authentication, and web-based system integration.

Technical Skills

Throughout the project, I developed strong technical proficiency in Python programming and practical implementation of computer vision algorithms using OpenCV and face recognition libraries. I gained hands-on experience in capturing live video streams, detecting faces in real-time, encoding facial features, and matching them with stored datasets. Additionally, I worked with backend frameworks such as Flask to develop a web-based authentication system and integrated frontend technologies like HTML and CSS to create user-friendly interfaces. I improved my skills in handling image data using NumPy and managing structured user information efficiently.

The project also strengthened my understanding of model evaluation techniques such as recognition accuracy, false acceptance rate, and false rejection rate. Overall, this experience enhanced my coding ability, debugging skills, and confidence in building AI-based biometric security systems accessible across multiple devices, including mobile phones.

Problem-Solving and Critical Thinking

This project required continuous analytical thinking and systematic problem-solving. One major challenge was accurately detecting and recognizing faces under varying lighting conditions, camera angles, and background noise. To address this issue, I experimented with different preprocessing techniques and optimized recognition parameters to improve system accuracy.

Selecting appropriate face encoding methods and reducing false matches required detailed experimentation and validation. Through debugging, testing, and iterative refinement, I learned to approach complex technical challenges logically and methodically. Breaking the system into smaller modules—such as face detection, encoding, database management, and web authentication—made the development process more structured and manageable.

The project strengthened my critical thinking skills and highlighted the importance of validation, testing, and performance optimization in developing reliable AI-powered authentication systems.

Aspect	**Description**	**Skills Developed**
Face Detection Optimization	Faced difficulty detecting faces under different lighting and background conditions; applied preprocessing techniques and parameter tuning to improve detection accuracy.	Image preprocessing, parameter tuning
Feature Encoding and Matching	Tested multiple face encoding and similarity comparison methods to improve recognition accuracy and reduce false matches.	Analytical thinking, feature optimization
System Performance Improvement	Optimized recognition speed and adjusted similarity thresholds to balance accuracy and false detection rates.	Model evaluation, performance analysis

Aspect	Description	Skills Developed
Debugging and Testing	Systematically tested login modules, camera integration, and recognition workflow to identify and fix errors.	Debugging, systematic testing
Modular System Design	Divided the system into modules such as face detection, encoding, database management, and authentication for structured implementation.	Structured development, system design

Table 5.1: Problem-Solving and Critical Thinking – Summary

5.2 Challenges Encountered and Overcome

Personal and Professional Growth

During the development process, I encountered challenges such as inconsistent face recognition accuracy, limited training images, and ensuring smooth integration between backend and frontend components. In some cases, the system failed to recognize faces correctly, requiring reassessment of encoding methods and parameter adjustments.

These challenges taught me patience, persistence, and adaptability. I learned that AI system development requires continuous testing and improvement rather than immediate perfection. Overcoming these difficulties strengthened my resilience and increased my confidence in handling real-world technical problems effectively.

Collaboration and Communication

Regular discussions with my supervisor and peers helped refine the system architecture and improve implementation strategies. Presenting technical concepts, explaining recognition workflows, and demonstrating system functionality enhanced my communication and presentation skills.

5.3 Application of Engineering Standards

The project followed a structured Software Development Life Cycle (SDLC) approach to ensure systematic development and quality assurance. Proper documentation, modular coding practices, and organized testing procedures were maintained throughout the implementation process.

Adhering to software engineering principles such as reliability, scalability, maintainability, and performance efficiency improved the robustness of the system. Ethical considerations, including secure storage of facial data and user privacy protection, were also taken into account to ensure responsible AI deployment. These structured engineering practices enhanced the credibility and professionalism of the overall project.

5.4 Insights into the Industry

This project provided valuable insight into the role of artificial intelligence and biometric technologies in modern security systems. It demonstrated how face recognition systems are widely used in attendance management, surveillance, access control, banking authentication, and smart device security.

The experience highlighted the importance of data quality, algorithm accuracy, and privacy protection in real-world AI applications. It also showed how integrating computer vision with web technologies can create scalable and user-friendly authentication platforms. This exposure has motivated me to further explore opportunities in AI-driven security systems, cybersecurity, and intelligent authentication technologies.

5.5 Conclusion of Personal Development

Overall, this capstone project significantly contributed to my academic, technical, and personal development. It strengthened my foundation in computer vision, machine learning, and web application development while improving my practical implementation skills.

The project enhanced my confidence in solving real-world security problems, conducting structured system development, and building AI-powered web applications. It has also clarified my career goals, encouraging me to pursue advanced studies and professional opportunities in artificial intelligence and cybersecurity.

CHAPTER 6
CONCLUSION

The Web-Based Face Recognition System successfully demonstrates the practical implementation of computer vision and artificial intelligence techniques in real-time security and authentication applications. The primary objective of the project was to design and develop a secure, automated, and user-friendly system capable of detecting and recognizing individuals using facial features. Through the integration of face detection, face encoding, and similarity matching algorithms within a web-based framework, the system achieved reliable identification performance under controlled conditions.

The project effectively combined backend development using Python and Flask with frontend technologies such as HTML, CSS, and JavaScript to create a complete, functional application. The implementation of separate admin and user login modules further enhanced system security and structured access control.

Throughout the development process, various challenges such as lighting variations, pose differences, and recognition accuracy were addressed through preprocessing techniques and parameter tuning. The modular system design improved maintainability and scalability, allowing future enhancements such as database expansion, cloud deployment, and advanced deep learning model integration.

Beyond technical implementation, the project strengthened understanding of biometric security systems and their growing importance in modern industries. Face recognition technology plays a significant role in surveillance, access control, attendance management, and digital authentication systems. This project highlights how artificial intelligence can provide efficient, contactless, and secure identification solutions while reducing manual intervention.

In conclusion, the Web-Based Face Recognition System meets its intended objectives and provides a strong foundation for further research and development in AI-driven security applications. The knowledge and skills gained during this project contribute significantly to academic growth, technical expertise, and professional readiness in the fields of computer vision, artificial intelligence, and cybersecurity.

REFERENCES

- Ultralytics, *YOLOv8 Documentation and Release Notes*, Ultralytics Inc., 2024. [Online].
- OpenCV Organization, *OpenCV Documentation (Version 4.x)*, 2023–2024. [Online].
- S. Serengil and A. Ozpinar, "DeepFace: A Lightweight Face Recognition and Facial Attribute Analysis Framework," *GitHub Repository*, 2024.
- J. Deng et al., "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," *IEEE CVPR*, with ongoing InsightFace implementations updated 2024.
- InsightFace Developers, *InsightFace: 2D and 3D Face Analysis Project*, GitHub Repository, 2024.
- X. Li et al., "Transformer-Based Person Re-Identification: A Survey," *IEEE Access*, 2024.
- Y. Zhang et al., "Lightweight Multi-Object Tracking for Real-Time Surveillance Systems," *arXiv preprint*, 2024.
- S. Liu et al., "Open-Vocabulary Object Detection via Vision-Language Models," *arXiv preprint*, 2024.
- Kumar et al., "Edge Computing for Real-Time Video Surveillance: A Survey," *IEEE Internet of Things Journal*, 2024.
- National Institute of Standards and Technology (NIST), *Face Recognition Vendor Test (FRVT) Report*, 2024.
- European Union, *Artificial Intelligence Act (AI Act)*, Official Journal of the European Union, 2024.

APPENDICES

Appendix I

```
# Sample Code: Real-Time Person Identification System
from flask import Flask, Response, request, jsonify
import cv2
import numpy as np
import base64
import time

app = Flask(_name_)

# In-memory database
KNOWN_FACES = {}
KNOWN_DETAILS = {}

# Load face detector
face_cascade = cv2.CascadeClassifier(
   cv2.data.haarcascades + 'haarcascade_frontalface_default.xml'
)

def generate_frames():
   cap = cv2.VideoCapture(0)
   while True:
      success, frame = cap.read()
      if not success:
         break
      gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
      faces = face_cascade.detectMultiScale(gray, 1.3, 5)

      for (x, y, w, h) in faces:
         cv2.rectangle(frame, (x, y), (x+w, y+h), (0,255,0), 2)
         cv2.putText(frame, "Person Detected",
                  (x, y-10),
                  cv2.FONT_HERSHEY_SIMPLEX,
                  0.6, (0,255,0), 2)

      ret, buffer = cv2.imencode('.jpg', frame)
      frame_bytes = buffer.tobytes()

      yield (b'--frame\r\n'
           b'Content-Type: image/jpeg\r\n\r\n' +
           frame_bytes + b'\r\n')
@app.route('/')
def home():
   return "Real-Time Person Identification System Running"

@app.route('/video')
```

```
def video():
    return Response(generate_frames(),
                    mimetype='multipart/x-mixed-replace; boundary=frame')

@app.route('/register', methods=['POST'])
def register():
    data = request.json
    name = data.get('name')
    details = data.get('details')
    KNOWN_DETAILS[name] = {
        "details": details,
        "time": time.strftime("%Y-%m-%d %H:%M:%S")
    }
    return jsonify({"status": "Registered", "name": name})

if _name_ == "_main_":
    app.run(host='127.0.0.1', port=5000)
```

Appendix II
Sample Output

Fig A.1. illustrates the first module of the proposed web-based real-time person identification system. This module provides an interactive interface where users can enter a camera source such as RTSP, HTTP stream, mobile camera feed, or laptop webcam. The interface includes control buttons for starting and stopping the stream, loading the YOLO-based detection model, and selecting either a phone or laptop camera. Once the stream is initiated, the live video feed is displayed on the right panel, enabling continuous monitoring. The system uses OpenCV for video capture and frame handling, while the backend processes frames in real time to prepare them for detection and analysis.

It also demonstrates the integration of deep learning-based person detection within the live streaming pipeline. When the detection model is loaded, the system applies YOLO (You Only Look Once) object detection to identify persons in each frame. Detected individuals are highlighted with bounding boxes, where color coding can differentiate between identified threats (red) and non-threat or unknown individuals (green). This module forms the foundation of the intelligent surveillance system by combining live video acquisition, real-time processing, and deep learning-based person detection in a user-friendly web environment.

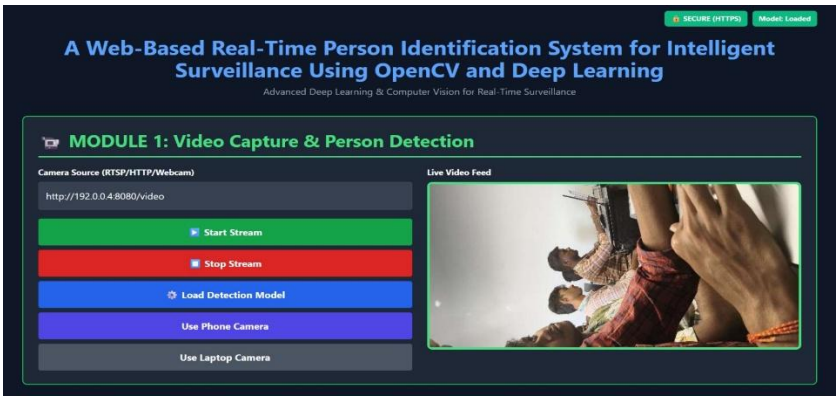

Fig A.1 Video Capture &Person Detection

Fig A.2. presents the second module of the proposed system, which is responsible for enrolling individuals into the surveillance database. This interface allows the administrator to enter the person's full name, additional details such as ID or role, and optionally upload a face image for accurate embedding generation. The "Mark as THREAT" option enables classification of individuals based on security risk. Once registered, the person's information is stored along with their facial embedding, and the system uses this data for real-time identification during live video processing. This module ensures structured data collection for reliable face recognition performance.

It also highlights the dataset control and management features integrated into the system. The Registered Database panel displays all enrolled individuals along with their status (Safe or Threat), enabling easy monitoring and verification. The Dataset Management section provides options to export data in CSV or JSON format, download the entire database, or clear the dataset when required. These functionalities support proper data handling, backup, and maintenance, ensuring that the surveillance system remains organized, scalable, and ready for deployment in real-world security environments.

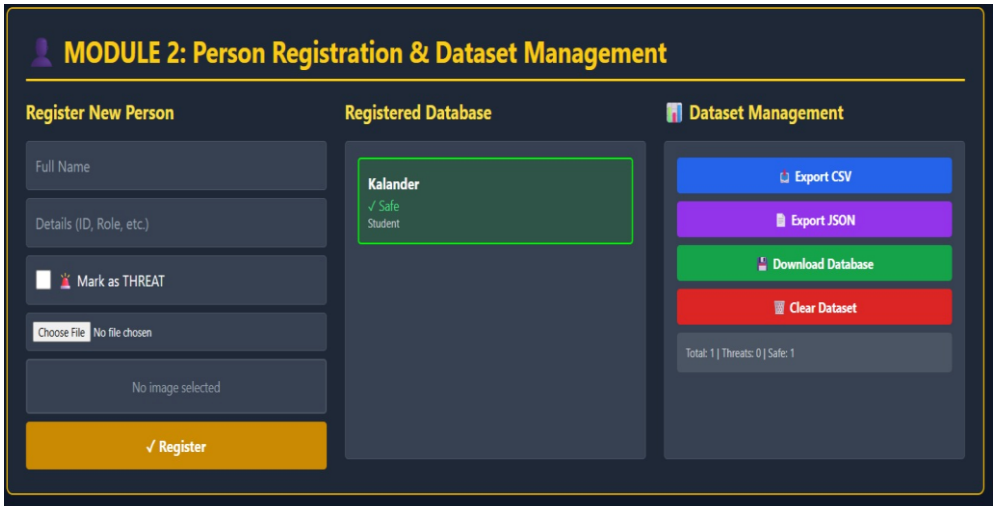

Fig A.2 Person Registration & Dataset Management

Fig A.3. illustrates the third module of the proposed surveillance system, which focuses on continuous monitoring, performance evaluation, and security alert generation. This module displays live accuracy metrics such as overall accuracy, detection rate, identification rate, and average confidence score, allowing real-time assessment of system performance. The live statistics panel shows the number of active tracks, people identified, detected threats, processing FPS, and current stream status. These metrics are dynamically updated during video processing, enabling administrators to monitor both system efficiency and identification effectiveness.

It also highlights the alert and activity management features of the system. When a registered individual marked as a threat is detected, the system immediately generates an alert notification, which appears in the alert panel and logs the event in the recent activity section. The alert mechanism ensures rapid response to potential security risks, while the tracking system maintains consistent identification of individuals across frames. This module integrates detection results, tracking logic, and real-time notifications to provide an intelligent and responsive surveillance solution.

Fig A.3 Real-Time Tracking & Alert System