iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IX **Month of publication:** September 2026

DOI: https://doi.org/10.22214/ijraset.2026.84966

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IX Sep 2026- Available at www.ijraset.com

Missing Person Identification Using AI

Vishnu[1], Shruthi Rampure[2]
Master of Computer Application, Visvesvaraya Technological University Regional Campus Kalaburagi

Abstract: ***This paper presents MissingPersonAI Pro, a web-based artificial intelligence system designed to support the identification of missing individuals through automated facial image comparison. Conventional missing-person searches often involve manual photograph examination, eyewitness information, and records maintained across separate sources, which can make the identification process slow and difficult to manage at scale. The proposed system addresses this challenge by combining facial recognition, image processing, similarity matching, and structured database management within a single platform. Administrators can register missing individuals by entering relevant personal information and uploading multiple facial photographs. These images are processed to obtain facial feature representations that can be stored for subsequent comparison. When a user submits an image, the system detects the face, extracts its features, and compares them with the stored representations to determine a potential matching record. The resulting match is displayed together with a similarity score and relevant information. The system also maintains identification history and provides PDF report generation to support documentation and further investigation. Python and Flask are used for application development, while SQLite manages user records, missing-person information, facial-image references, and matching history. Role-based access is incorporated to control administrative and general-user operations. The functional tests documented in the project report successfully covered authentication, person registration, image uploading, face detection, feature extraction, matching, and report generation. The proposed framework can reduce repetitive manual comparison and provide a structured technological aid for missing-person investigations. Future improvements may include real-time CCTV analysis, age-progression support, multi-face detection, mobile integration, and connection with authorized large-scale databases.***

Keywords: Missing Person Identification, Artificial Intelligence, Facial Recognition, Deep Learning, Computer Vision, Facial Features, Similarity Matching, Flask, Image Processing.

I. INTRODUCTION

Missing person identification is an important social and humanitarian concern, as locating individuals often requires the examination of photographs, personal information, eyewitness descriptions, and records collected from different sources. Traditionally, the identification process has depended largely on manual comparison of photographs and separate database searches, which can be time-consuming, difficult to manage for large image collections, and affected by variations in image quality, lighting, facial expressions, and physical appearance. With the advancement of Artificial Intelligence (AI) and Computer Vision technologies, there is an increasing opportunity to develop automated and reliable systems that can assist in identifying missing individuals through facial image analysis. Facial recognition provides a non-contact approach in which distinctive facial characteristics can be extracted from an image and compared with previously registered facial information. This study focuses on the design and implementation of MissingPersonAI Pro, an AI-based web application that integrates face detection, facial feature extraction, similarity matching, database management, and automated report generation. The system allows authorized administrators to register missing persons with multiple facial images, while users can upload an image to search for a potential match. The proposed system processes the uploaded image, extracts facial representations, compares them with stored records, and presents the potential matching result together with a similarity or confidence score. The system also maintains identification history and generates PDF reports containing relevant match information for documentation and further investigation. By combining artificial intelligence, structured data management, authentication, and a user-friendly web interface, MissingPersonAI Pro provides a systematic technological approach for reducing repetitive manual effort and supporting missing person identification.

II. PROBLEM STATEMENT

Conventional missing person identification practices rely heavily on manual examination of photographs, eyewitness descriptions, public reports, and searches across separately maintained records. These approaches can require considerable time and human effort, particularly when investigators need to examine a large number of photographs.

©IJRASET: All Rights are Reserved

Differences in lighting conditions, facial expressions, image quality, viewing angles, and changes in physical appearance can further complicate manual comparison and may affect the identification process. In addition, fragmented record management can make it difficult to maintain identification attempts and retrieve previous matching information efficiently.

These limitations highlight the need for an automated, structured, and user-friendly system capable of assisting with facial image comparison. With advancements in Artificial Intelligence and Computer Vision, facial recognition can be used to detect faces, extract distinctive facial representations, and compare them against registered records. However, challenges such as variations in facial appearance, image quality, reliable similarity matching, secure access, and proper documentation must be considered when developing a practical identification platform. The MissingPersonAI Pro project addresses these requirements by developing a web-based system that integrates facial feature extraction, similarity-based matching, structured database management, authentication, match-history storage, and PDF report generation to support efficient missing person identification.

III. OBJECTIVES OF THE STUDY

The primary objective of this study is to design and implement MissingPersonAI Pro, an AI-powered web-based system that assists in identifying missing individuals through facial recognition and similarity matching. The system aims to minimize manual effort by allowing authorized users to upload facial images, automatically detect and extract facial features, and compare the generated facial embeddings with registered missing-person records. It is designed to provide suitable matching results along with confidence scores while maintaining secure and structured records.

Another key objective is to improve the management and documentation of identification activities by providing role-based access, maintaining match history, and generating downloadable PDF reports containing relevant identification details and visual evidence. The system also aims to provide a user-friendly platform that can support authorized organizations in performing image-based searches efficiently and maintaining organized missing-person information.

IV. METHODOLOGY

The methodology adopted for this project follows a structured, step-by-step approach to support efficient and reliable missing person identification. The system integrates Artificial Intelligence, Computer Vision, facial feature extraction, similarity matching, database management, and a Flask-based web interface to provide functionality for both administrators and general users. The process begins with the registration of missing individuals, where administrators enter relevant personal details and upload multiple facial images. These images are processed to detect faces and generate numerical facial embeddings that represent distinctive facial characteristics.

The next stage involves storing the processed information and facial image references in a structured database. During an identification request, a user uploads an image through the web interface, and the system first validates whether a detectable face is present. The detected face is then converted into a facial embedding using the recognition model. This embedding is compared with the stored embeddings of registered missing persons using similarity-based matching, and the system identifies the closest available match along with a confidence score.

Finally, the system records the identification attempt in the match-history database and generates a downloadable PDF report containing relevant match information, confidence details, reporter information, and visual evidence. Authentication and role-based authorization are incorporated to restrict administrative operations and protect stored records. The developed system is tested through different functional operations, including authentication, person registration, image uploading, face detection, feature extraction, matching, and report generation. This methodology provides a structured approach for automating missing person identification while reducing dependence on manual image comparison.

V. LITERATURE SURVEY

Facial recognition and artificial intelligence have gained increasing attention in recent years for assisting with missing person identification and reducing the dependence on manual photograph comparison. Earlier approaches commonly used conventional face detection and tracking techniques, while recent studies have increasingly adopted deep learning models, facial embeddings, similarity measures, and computer vision frameworks for automated identification. The literature included in the project report shows the use of techniques such as YOLOv8, CNN, FaceNet, VGG-Face, VGG16, SVM, Haar Cascade, DeepFace, PCA, LBP, and cosine or Euclidean similarity matching for missing person detection and recognition. Several reported systems also explore real-time CCTV monitoring, mobile applications, geolocation information, and automated alerts to extend identification beyond static image searches.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Recent studies demonstrate different approaches to improving recognition performance under practical conditions. For example, research using YOLOv8 and CNN-based facial processing has explored real-time detection, while other systems have combined FaceNet embeddings with Flask and OpenCV for web-based recognition. CNN-based approaches using VGG-Face and SVM have also been investigated for missing person identification. Other works have considered age progression, multi-source data, CCTV integration, mobile deployment, and improved performance under lighting and appearance variations.

The reviewed literature indicates that AI-based facial recognition can provide an effective foundation for automating missing person searches; however, practical challenges remain, including variations in image quality, lighting conditions, facial appearance, occlusion, aging, scalability, and the need for secure data management. The MissingPersonAI Pro project addresses these requirements through a web-based approach that combines face detection, facial embedding extraction, similarity-based matching, structured database storage, authentication, match-history management, confidence scoring, and PDF report generation. The report also identifies future possibilities such as CCTV integration, age-progression models, multi-face detection, large-scale databases, mobile applications, and automated alerts.

VI. EXISTING SYSTEM AND PROPOSED SYSTEM

A. *Existing System*

Traditional missing person identification methods depend largely on manual examination of photographs, eyewitness descriptions, printed information, and basic database searches. These approaches require considerable human effort and become difficult to manage when a large number of photographs or records must be examined. Manual face comparison can also be affected by differences in lighting, facial expressions, image quality, viewing angles, and changes in appearance over time. In addition, records may be maintained separately, making it difficult to organize identification attempts and retrieve previous matching information. Existing approaches may also lack automated facial matching, systematic match-history management, and proper reporting facilities.

B. *Proposed System*

The proposed MissingPersonAI Pro introduces an AI-powered web application for assisting in missing person identification through facial recognition and similarity matching. Administrators can register missing individuals by entering personal details and uploading multiple facial images. The system processes these images through face detection and facial feature extraction to generate facial embeddings. When a user uploads an image for identification, the system validates the presence of a face, extracts its facial representation, and compares it with stored embeddings using similarity measures. The closest matching record is displayed with a confidence score. The system also provides role-based authentication, structured database management, match-history storage, and downloadable PDF reports containing identification details and visual evidence.

Unlike conventional approaches that depend heavily on manual comparison, the proposed system provides an integrated platform for image-based searching, automated matching, secure record management, and documentation. The architecture uses a web-based interface with Python, Flask, computer vision and facial recognition technologies, SQLite, HTML, CSS, JavaScript, and FPDF to support the major functions of the application.

VII. FEASIBILITY STUDY

A. *Economic Feasibility*

The proposed MissingPersonAI Pro system is economically feasible because it is developed using commonly available and open-source technologies such as Python, Flask, SQLite, and computer vision libraries. The application does not require specialized hardware for its basic operation, which helps reduce implementation and maintenance costs. Its web-based design can support organizations in reducing the time and manual effort involved in missing person identification.

B. *Operational Feasibility*

The system is operationally feasible because it provides a user-friendly web interface for both administrators and general users. Administrators can register missing persons, upload facial images, manage records, and review identification history, while general users can upload images and view matching results and reports. Automated face detection, matching, and report generation reduce manual intervention and make the identification workflow easier to operate.

C. *Technical Feasibility*

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The project is technically feasible as it uses established technologies including Python, Flask, SQLite, computer vision techniques, and deep learning-based facial recognition models. The system follows a client-server architecture in which image processing, facial feature extraction, similarity matching, authentication, and database operations are handled by the application server. The report indicates that standard CPU-based hardware and storage resources are sufficient for basic system operation.

D. Legal and Ethical Feasibility

The report emphasizes secure authentication and role-based authorization to restrict access to administrative operations and protect stored information. Since the application processes facial images and personal information, appropriate access control and responsible handling of identification data are important considerations. The current project focuses on controlled use by authorized organizations, while future development should continue to address privacy, security, and responsible deployment requirements.

VIII. TOOLS AND TECHNOLOGIES USED

A. Programming Language

Python is used as the primary programming language for developing MissingPersonAI Pro. It supports the backend implementation, image processing, database operations, and integration of facial recognition and machine learning components. Its extensive library ecosystem makes it suitable for developing the application's AI-based functionality.

B. Web Framework

Flask is used to develop the web application and manage HTTP requests, routing, and web-page rendering. It provides the connection between the user interface and backend processing components, including authentication, image uploading, facial matching, and report generation

C. Computer Vision and Face Recognition

Computer vision and facial recognition technologies are used to detect faces and extract distinctive facial representations from uploaded images. Deep learning-based facial embedding extraction converts detected faces into numerical representations that can be compared using similarity measures for identification.

D. Database/Storage

SQLite is used for structured storage of user information, missing person details, facial image references, and identification history. The database contains logically related records that support secure authentication, person management, image association, and storage of matching results.

E. Frontend Technologies

HTML, CSS, and JavaScript are used to create the web-based user interface. HTML provides the page structure, CSS supports the visual presentation, and JavaScript adds interactive functionality to the application. Together, these technologies provide an accessible interface for administrators and general users.

F. PDF Report Generation

FPDF is used to generate downloadable PDF reports for identification results. The generated reports can contain match information, confidence scores, reporter details, and supporting visual evidence, providing structured documentation for identification attempts.

IX. SYSTEM DESIGN

A. System Perspective

The proposed MissingPersonAI Pro is designed as a centralized web-based application that integrates users, administrators, facial recognition components, and a structured database into a unified system. The architecture follows a client-server approach in which users interact with the application through a web browser, while authentication, image processing, facial feature extraction, similarity matching, database operations, and report generation are handled on the server side. This modular organization allows the major components to operate together while keeping their individual responsibilities clearly separated. The system supports two primary user roles: Administrator and General User. Administrators can authenticate themselves, register missing individuals, provide personal information, upload multiple facial images, manage records, and review identification history.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

General users can upload an image for identification, view the resulting match, and access the corresponding report. Role-based authorization restricts administrative functions to permitted users and helps protect the stored informationThe main processing flow begins when a facial image is submitted to the application. The Face Detection module verifies the presence of a valid face, after which the Feature Extraction module generates a numerical facial embedding. The generated representation is compared with stored facial embeddings using the Matching module to determine the closest available record. The resulting match and confidence information are then presented to the user, while the Report Generation module prepares a structured PDF report. Identification information and match history are maintained in the database for subsequent retrieval and documentation.The database forms an important part of the system design by maintaining user credentials, missing person information, facial image references, and matching records. The overall architecture is designed with emphasis on modularity, security, scalability, maintainability, and organized data flow, allowing the system to support its current identification workflow and future extensions.

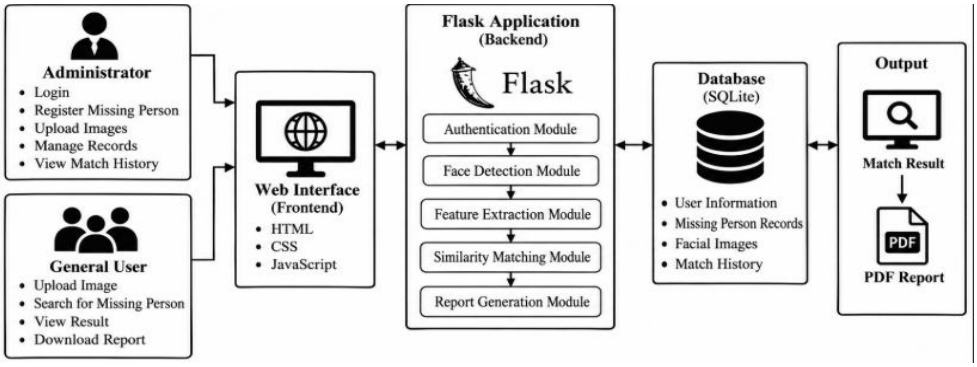


Figure 1: System Architecture of MissingPersonAI Pro

B. Context Diagram

The context diagram presents MissingPersonAI Pro as a single high-level system and shows how it exchanges information with the entities that interact with it. The main external entities are the Administrator, General User, and Database. The Administrator provides missing-person information and facial images and receives identification results and generated reports. The General User submits an image for identification and receives the corresponding matching result and PDF report. The Database stores and retrieves user information, missing-person records, facial images or embeddings, and match history. This representation focuses on the system boundary and major data exchanges rather than the internal processing performed by individual modules

Main Data Flows

- Administrator → System: Login details, missing-person details, facial images, record-management requests.
- System → Administrator: Match results, confidence information, match history, and reports.
- General User → System: Login information and image/search requests.
- System → General User: Identification results, confidence information, and PDF reports.
- System ↔ Database: User records, missing-person information, facial data/embeddings, and match history.

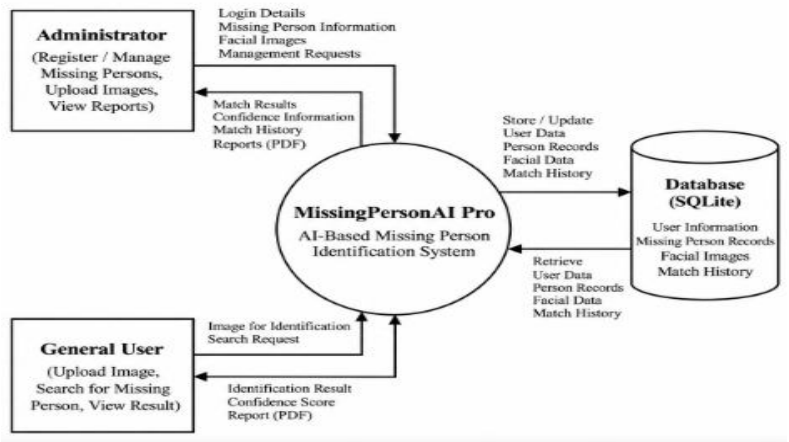


Fig. 2: Context Diagram of MissingPersonAI Pro

1) Use Case Diagram

The Use Case Diagram of MissingPersonAI Pro represents the functional interaction between the system and its two primary actors: Administrator and General User. The Administrator has privileged access to manage missing-person information, upload multiple facial images, review identification records, and monitor generated reports. The General User can access the identification functionality by uploading an image, searching for a missing person, viewing the matching result, and downloading the corresponding report. Both actors interact with the authentication functionality, while administrative operations are protected through role-based access control.

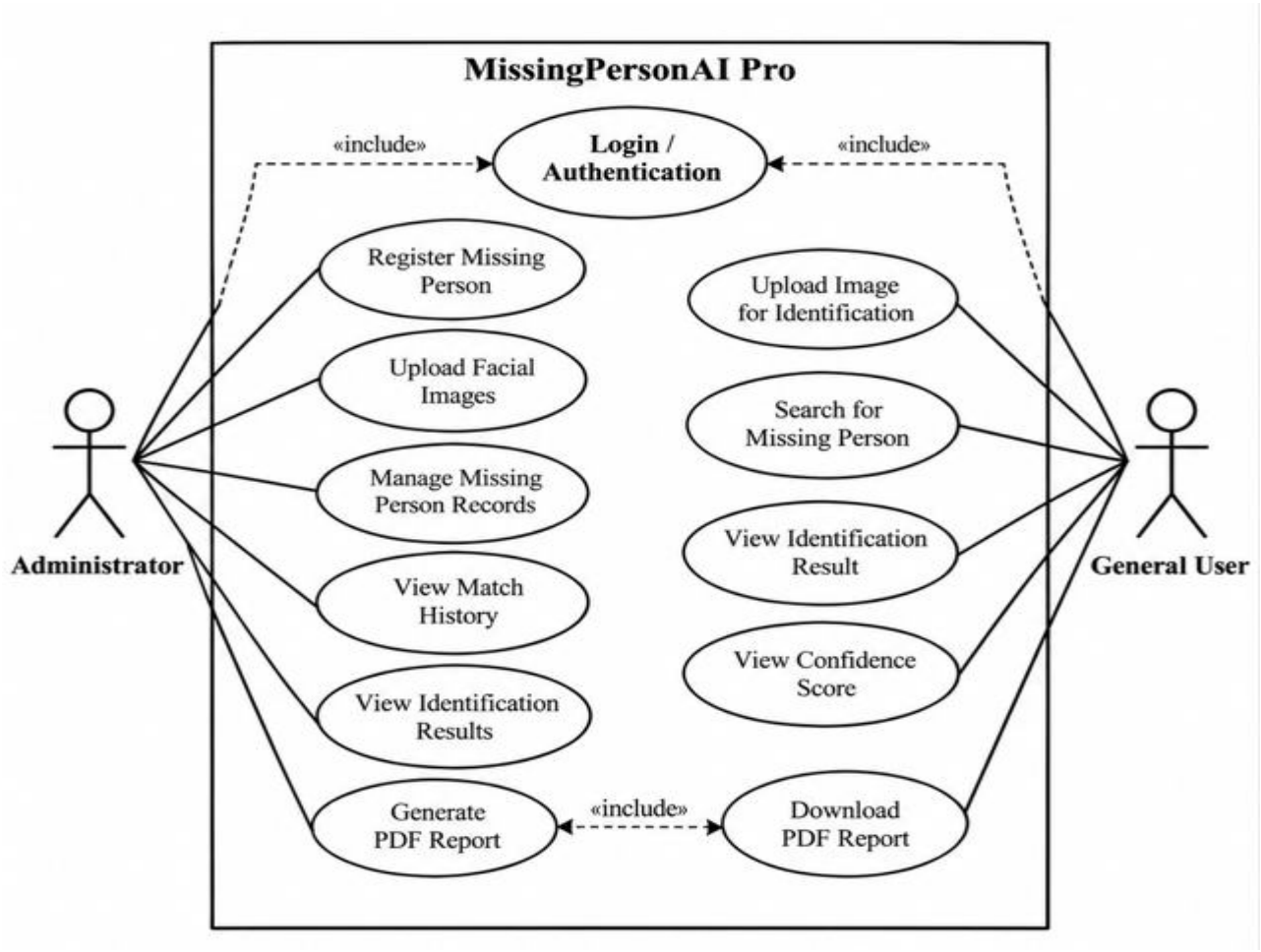


Fig. 3: Use Case Diagram of MissingPersonAI Pro

The diagram focuses on what each user can do within the system, rather than describing the internal implementation of face detection, feature extraction, or similarity matching. These internal processes are handled by the system after the user initiates the relevant operation.

2) Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) describes how information moves through the MissingPersonAI Pro system. It represents the movement of user requests, facial images, missing-person records, facial features, matching information, and generated reports between external users, system processes, and the database. The system begins with authentication and image input, followed by face validation, feature extraction, similarity matching, result generation, and storage of the identification history. This structure provides a clear representation of the processing sequence without depending on the implementation details of individual programming modules.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

a) DFD Level 0 — Context Level

At Level 0, MissingPersonAI Pro is represented as a single process. The Administrator provides missing-person details and facial images, while the General User provides an image for identification. The system returns identification results, confidence information, and PDF reports. The database stores and retrieves user information, missing-person records, facial data, and match history.

b) DFD Level 1

The Level 1 DFD expands the main system into its principal processing activities:

- User Authentication – validates user credentials and access roles.
- Missing Person Registration – accepts personal details and facial images from administrators.
- Image Validation and Face Detection – checks the uploaded image and identifies a valid face.
- Feature Extraction – converts the detected facial information into a facial embedding.
- Similarity Matching – compares the extracted embedding with stored facial information.
- Result Processing – prepares the identification result and confidence score.
- Report Generation – produces a downloadable PDF report.
- Match History Storage – records the identification attempt for later reference.

These processes correspond to the functional operations documented for MissingPersonAI Pro, including authentication, facial processing, similarity matching, result presentation, history management, and PDF generation.

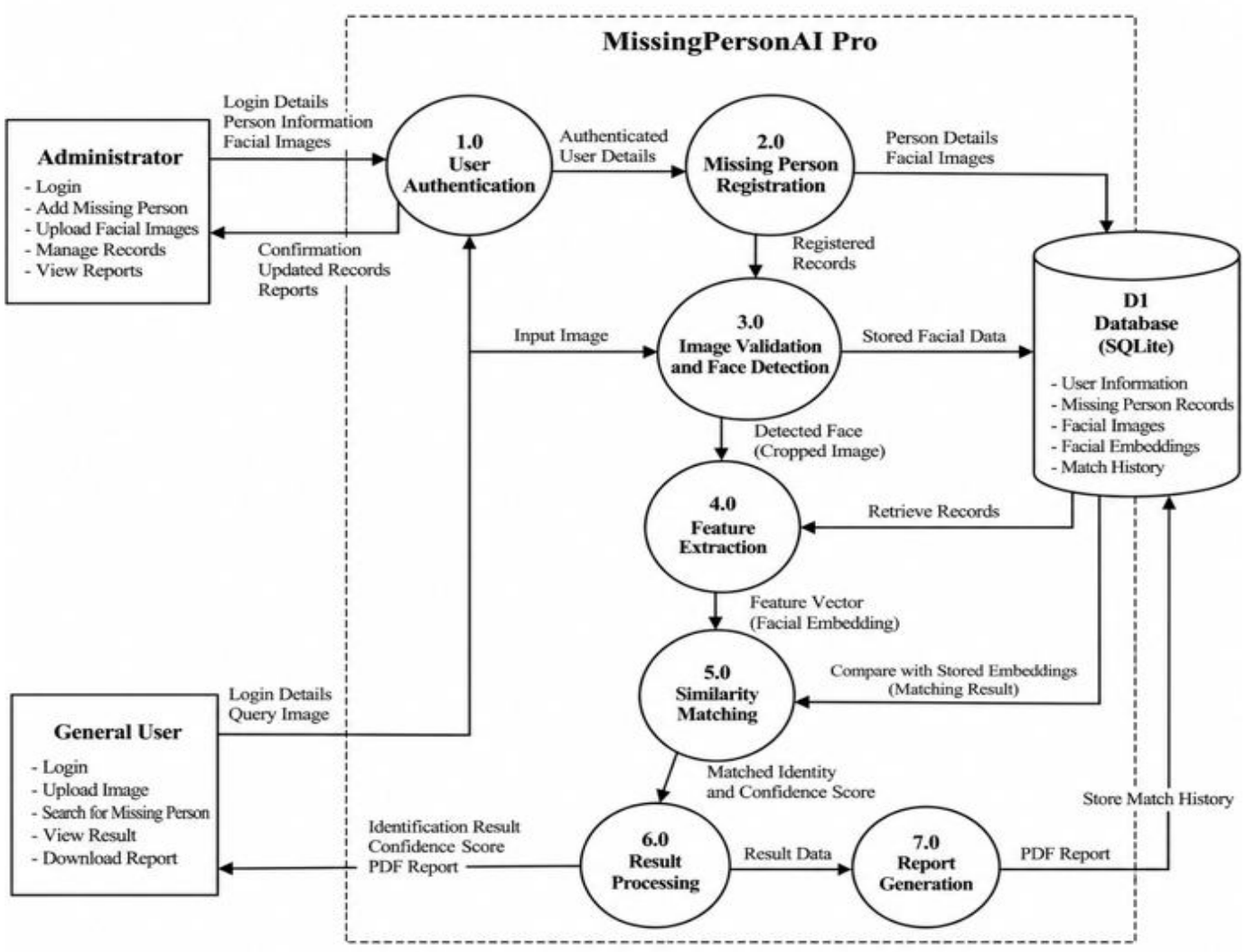


Fig. 4: Data Flow Diagram (DFD) of MissingPersonAI Pro

3) Class Diagram

The Class Diagram of MissingPersonAI Pro represents the structural organization of the system and the relationships among its major classes. The User class manages account information, authentication details, and user roles, allowing the system to provide role-based access to administrators and general users. The Person class represents registered missing individuals and maintains information such as name, age, gender, description, and last-seen details. Each Person can be associated with multiple PersonPhoto objects, where facial image paths and corresponding facial information are maintained for identification. The Match class records identification attempts, including the uploaded image reference, matched person, confidence score, reporter details, and timestamp. The system also contains supporting functional classes such as Authentication, FaceRecognition, and ReportGenerator. Authentication verifies user credentials and controls access, while FaceRecognition performs face detection, feature extraction, embedding generation, and facial comparison. The ReportGenerator prepares identification information and produces downloadable PDF reports. These classes work together to support user management, missing-person registration, facial-image processing, similarity matching, match-history storage, and report generation. The database structure documented in the project contains the Users, Persons, Person_Photos, and Matches entities, with relationships maintained using identifiers to support organized and consistent data management.

X. SEQUENCE DIAGRAM

The Sequence Diagram of MissingPersonAI Pro describes the chronological interaction between the user, web application, authentication service, facial-processing components, database, matching module, and report-generation module. The interaction begins when an administrator or general user accesses the application and submits authentication information. After successful verification, an administrator can register a missing person and upload facial images, while a general user can upload an image for identification. The submitted image is validated for the presence of a face, after which facial features are extracted and represented as an embedding. The generated embedding is compared with stored facial embeddings retrieved from the database to determine the closest matching record. The system then prepares the identification result and confidence score, stores the identification activity in the match history, and generates a PDF report containing the relevant result information. The final result and report are returned to the requesting user through the web interface. This sequence reflects the authentication, image processing, feature extraction, similarity matching, database interaction, and reporting operations documented for MissingPersonAI Pro.

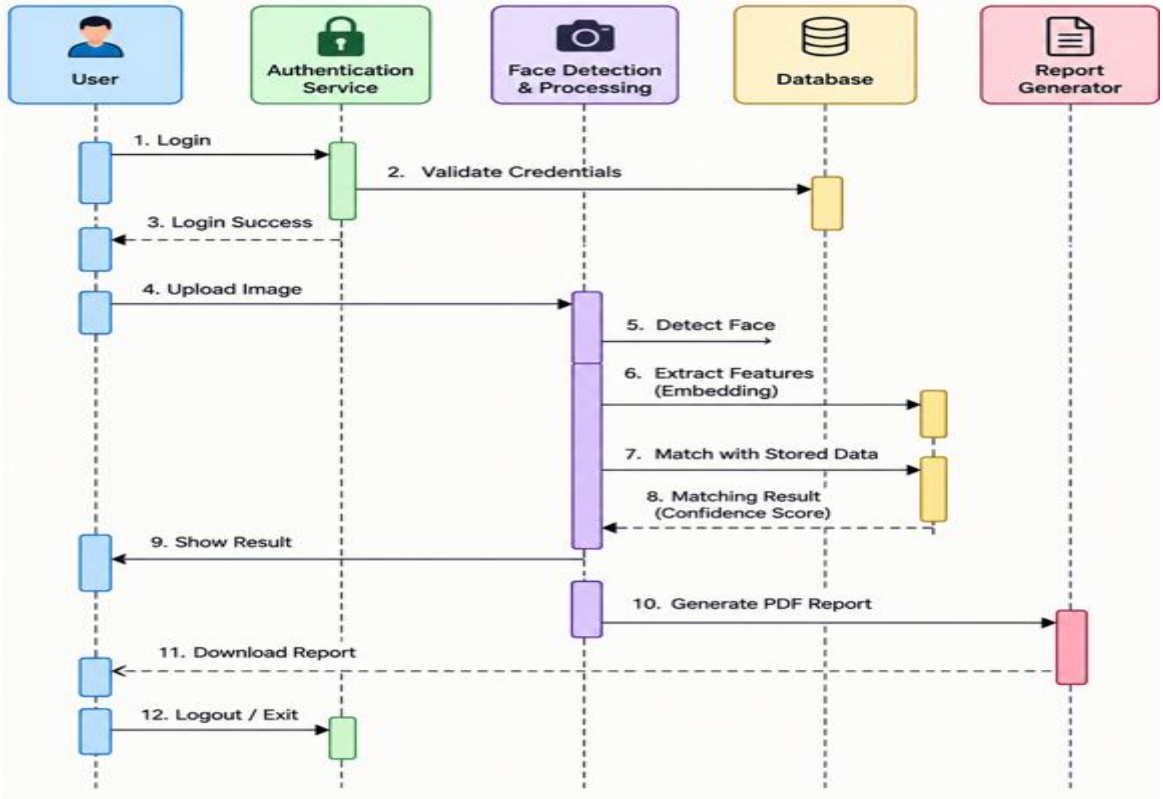


Fig.5 Sequence Diagram

XI. COLLABORATION DIAGRAM

The Collaboration Diagram of MissingPersonAI Pro illustrates how the major system components communicate with one another to complete the missing-person identification process. The interaction begins when a user, either an administrator or a general user, communicates with the Authentication Module, which verifies the login credentials and determines the permitted access level. After successful authentication, the administrator can register missing-person details and upload facial images, while a general user can submit an image for identification. The uploaded image is forwarded to the Face Detection Module for validation, and the detected facial information is then passed to the Feature Extraction Module to generate a facial embedding. The generated embedding is stored in or retrieved from the Database, where registered facial information is maintained. The Matching Module compares the new embedding with stored embeddings and produces the closest matching result together with a confidence score. The resulting information is then sent to the Report Generation Module, which prepares a PDF report, while the identification result and report are returned to the user through the web application. The interaction also allows match information to be stored as historical records for later reference. This collaboration structure reflects the coordinated operation of authentication, facial processing, database management, matching, and reporting components described in the project report.

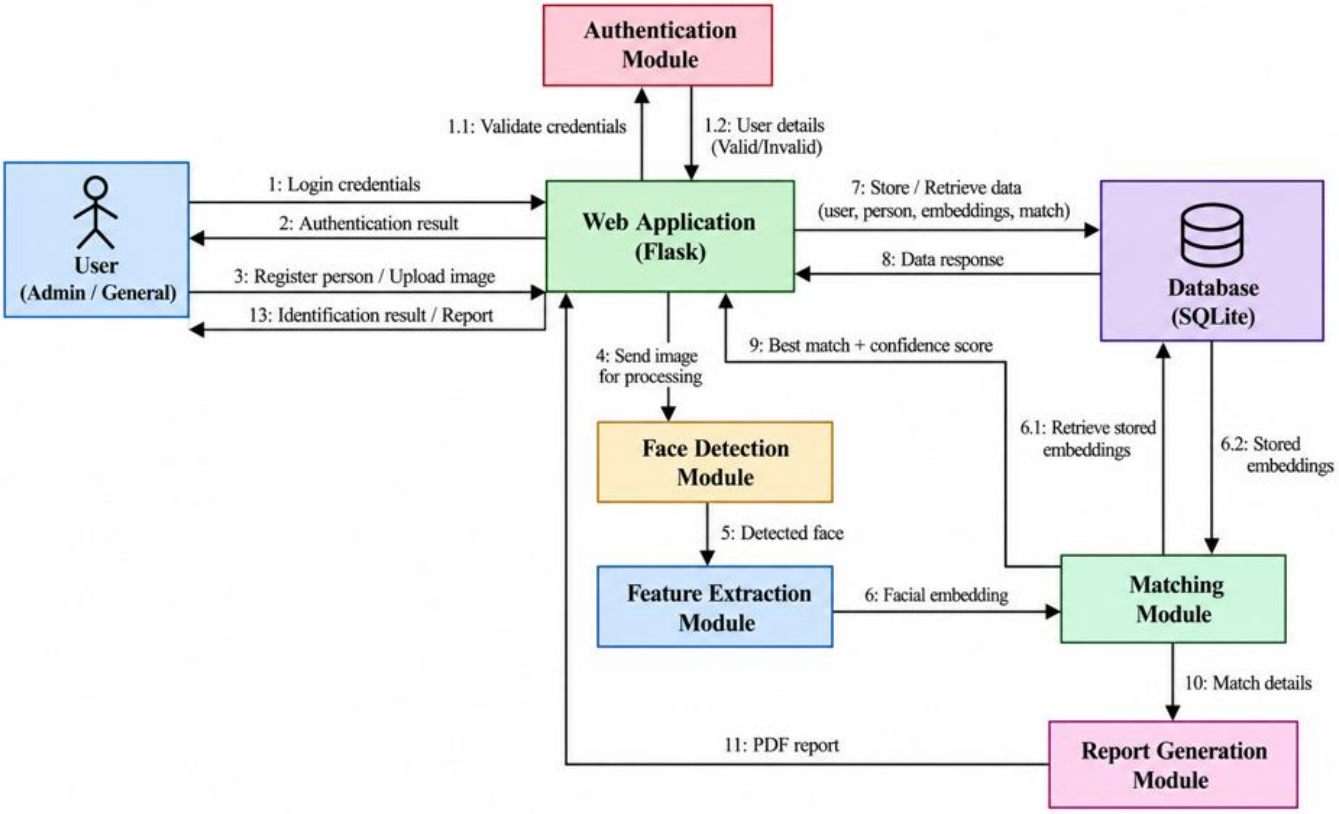


Fig. 6 Collaboration Diagram

XII. SYSTEM IMPLEMENTATION

The MissingPersonAI Pro system is implemented as a web-based application that integrates a Flask backend, facial-recognition processing, SQLite database management, and a browser-based user interface. The implementation begins with user authentication and role-based authorization, through which administrators and general users are provided access according to their assigned roles. Administrators can register missing-person information and upload multiple facial images, while general users can submit an image for identification. The uploaded images are validated for the presence of a face and then processed to extract facial features and generate embeddings. These embeddings are compared with the facial information stored in the database using similarity matching, after which the system returns the closest matching record and associated confidence information. Identification activities are recorded in the match history, and the FPDF-based reporting module generates downloadable PDF reports containing relevant match information.

The implementation uses Python and Flask for backend processing, HTML, CSS, and JavaScript for the web interface, SQLite for structured storage of users, missing-person records, facial images, and match history, and computer-vision/deep-learning components for facial detection and feature extraction. The implementation was also subjected to functional and integration testing covering authentication, missing-person registration, image uploading, face detection, feature extraction, facial matching, database operations, and PDF report generation. The documented test cases include successful and unsuccessful login, administrator authorization, missing-person registration, multiple-image upload, face detection, rejection of images without faces, embedding generation, and database-based face matching.

XIII. RESULTS AND OUTPUT IMAGES

The implementation of MissingPersonAI Pro was evaluated by executing functional test scenarios covering authentication, role-based access, missing-person registration, facial-image uploading, face detection, feature extraction, facial matching, database operations, and report generation. The documented test cases show successful execution for valid login, administrator verification, missing-person registration, multiple facial-image processing, face detection, embedding generation, and database-based face matching. The system also handles invalid credentials and images without detectable faces by displaying appropriate error messages.

The implemented application produces identification results containing the matched-person information and confidence score. Match information is maintained in the database as history, while the reporting component generates downloadable PDF documents containing identification details. The report also documents output testing for match results, confidence scores, visual displays, and PDF reports, including checks for report readability and completeness.

XIV. CONCLUSION

MissingPersonAI Pro demonstrates a web-based approach for supporting missing-person identification through artificial intelligence and facial-recognition techniques. The system enables administrators to register missing-person details and associated facial images, while authorized users can upload an image for identification. The submitted image is processed through face detection and facial feature extraction, and the resulting facial representation is compared with stored records to produce a matching result and confidence score. The system further integrates role-based access control, structured database management, match-history storage, and PDF report generation, providing a unified workflow for identification and documentation. The implemented system was evaluated through functional testing of authentication, authorization, missing-person registration, image processing, feature extraction, face matching, database operations, and report generation, with the documented test cases showing successful results for the tested scenarios. Overall, MissingPersonAI Pro provides a practical technological framework for reducing manual effort in missing-person identification while maintaining organized records and supporting documentation through generated reports.

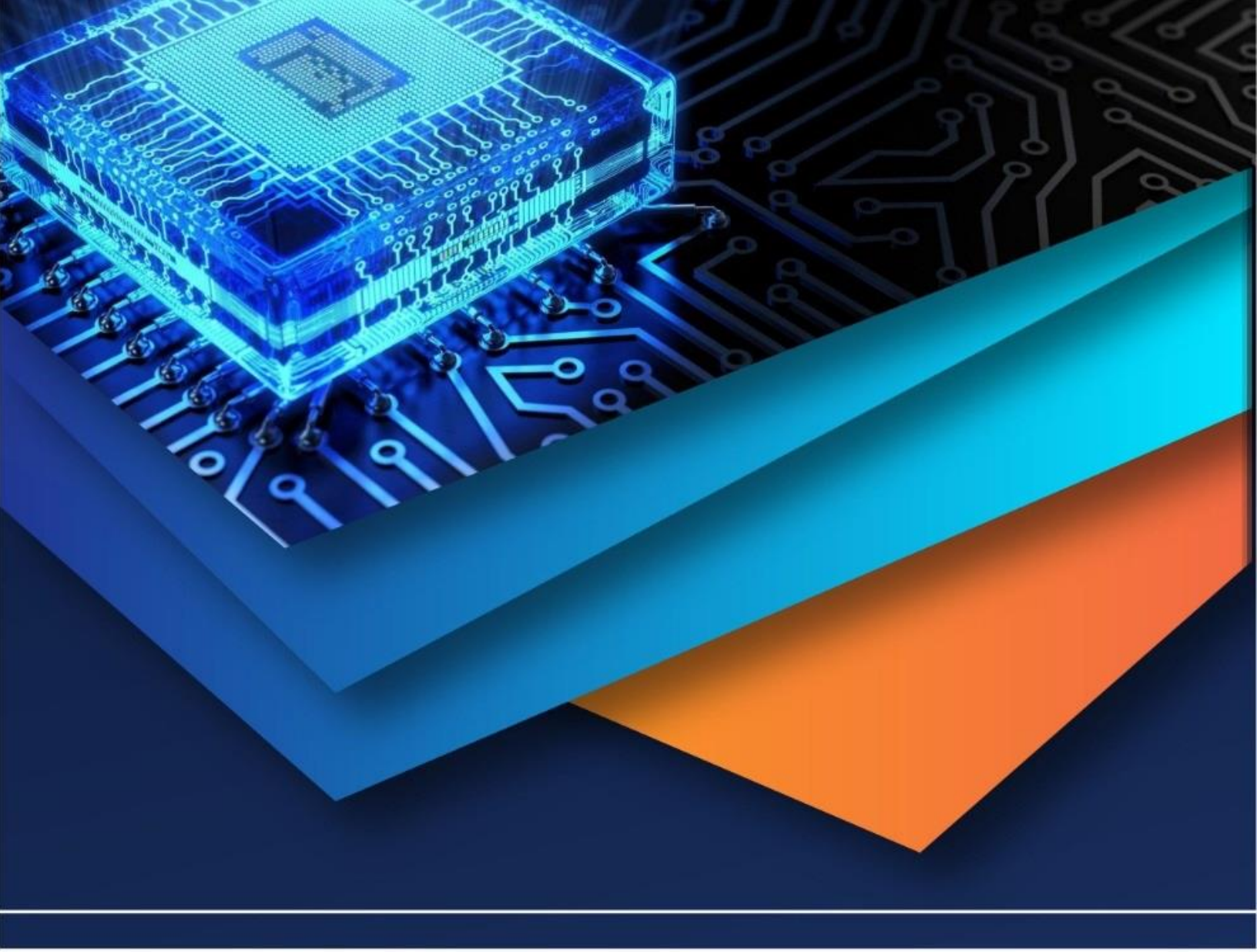
XV. FUTURE ENHANCEMENTS

The future development of MissingPersonAI Pro can focus on extending the system beyond its current image-based identification workflow. Real-time CCTV integration can be introduced to support continuous monitoring and automated identification from surveillance feeds. Age-progression techniques can be incorporated to assist in cases where a missing person has been absent for a long period and their current appearance may differ significantly from available photographs. The system can also be extended with multi-face detection so that images containing several individuals can be processed, while integration with larger regional or national missing-person databases could support broader searches. A dedicated mobile application can provide convenient image submission through smartphones, and advanced analytics and alert mechanisms can be developed to notify authorized personnel when a high-confidence match is identified. These enhancements can improve the scalability, accessibility, and practical applicability of MissingPersonAI Pro in future deployments.

REFERENCES

[1] S. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A Unified Embedding for Face Recognition and Clustering,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[2] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep Face Recognition,” British Machine Vision Conference (BMVC), 2015.
[3] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, “DeepFace: Closing the Gap to Human-Level Performance in Face Verification,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
[4] F. Schroff, D. Kalenichenko, and J. Philbin, “Face Recognition Using Deep Neural Networks and Facial Embeddings,” IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[5] P. Viola and M. Jones, “Rapid Object Detection Using a Boosted Cascade of Simple Features,” Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001.
[6] G. Bradski, “The OpenCV Library,” Dr. Dobb’s Journal of Software Tools, 2000.
[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” Advances in Neural Information Processing Systems, 2012.
[8] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.
[9] C. Szegedy et al., “Going Deeper with Convolutions,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[10] D. Patel, A. Solanki, J. Magajwala, and H. Sutariya, “AI-Based Missing Person Detection System,” 2025. VISHNU_3VY24MC077_Report(1)(1)
[11] L. Sridhara Rao, A. Thanmai, M. Hemanth, and N. Balraj, “Missing Person Identification System Using Deep Learning,” 2025. VISHNU_3VY24MC077_Report(1)(1)
[12] S. B. Patil, R. Fegade, P. Nanaware, and S. Kadu, “Finding Missing Person Using AI and ML,” 2025.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)