ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2026.0400

AI-ASSISTED REAL-TIME EMBEDDED FACIAL IDENTIFICATION AND ALERT GENERATION FOR SURVEILLANCE-BASED INVESTIGATIONS

A. Nandhini, M. Amilesh, P. Archana, A. Nisath and R. Siva
Department of Computer Applications, Nehru College of Management, India

Abstract

Public-space surveillance produces large volumes of video that can be difficult for personnel to review promptly. This paper presents Spot Crime, a prototype surveillance-support system that combines face detection, CNN-based embedded facial feature representation, database-based similarity matching, and alert generation in a web application. The proposed pipeline accepts CCTV frames or uploaded images, preprocesses the input, detects and aligns facial regions, generates facial embeddings, and compares them with authorized reference records. When a similarity score exceeds a threshold validated for the selected model and operating conditions, the system creates a preliminary alert containing the timestamp, camera identifier, candidate score, and reference-record identifier for review by authorized personnel. OpenCV and dlib support image processing and face localization, while Python/PHP and MySQL provide application and record-management functions. Unlike a system that labels a person as a criminal, the proposed design treats a match only as a candidate identity association; it does not establish guilt, and human verification remains essential. The paper details the architecture, workflow, database structure, operational safeguards, and an evaluation framework. Since quantitative benchmark results are not provided in the current manuscript, claims about accuracy, latency, and field effectiveness are treated as objectives for future validation rather than established outcomes. The principal contribution is an integrated design for traceable candidate matching and alert review, with explicit attention to false matches, demographic performance, privacy, and responsible operational use.

Keywords:

Public-Space Surveillance, Video, Spot Crime, CNN-based Embedded Facial Features

1. INTRODUCTION

The criminal record contains personal data about a specific person together with a photo. In order to identify any criminal, an eyewitness identification of that person is needed. Identification can be done using fingerprint, eyes, DNA, etc. One such application is facial identification. The face is the main focus of attention in social interaction and plays a major role in conveying identity and emotions. The facial recognition system uses a database of stored images and compares the captured image to find a match, if any. For each face image, identification can be done using RGB values for eye colour, face width, and height.

This system is aimed at identifying offenders for any investigative department. In this system, the criminal's image is stored in the database. Eyewitnesses select the slices that appear on the screen, and with this a face image is retrieved from the database. Thus, this system provides a very friendly environment for the operator as well as the eyewitness to easily identify the criminal if the crime record exists in the database.

The developed system is also the first milestone for image detection and recognition based on surveillance video. The number of recorded criminal incidents is increasing, including robbery, vandalism, assault, murder, and kidnapping. However, the conventional methods of criminal investigation and prevention are often labour-intensive and inefficient. To address this challenge, a novel deep-learning-based system named "Spot Crime" has been proposed.

Spot Crime is a web application designed to enhance public safety and support law enforcement efforts by automating the monitoring of live CCTV footage and alerting police officials to suspicious activities. This innovative system employs custom convolutional neural networks (CNNs) and advanced models for behaviour classification, allowing it to analyze human activity in real-time video frames. The selection of the best-performing model is based on validation accuracy, and its purpose is to aid in the early detection of criminal incidents. In light of increasing crime rates in India, especially during lockdowns, the need for such a system has become paramount. Spot Crime offers a solution to the challenges posed by manual surveillance, providing a more effective means of crime prevention and promoting public safety by promptly notifying authorities of potential criminal activity and its location.

1.1 MODULE DESCRIPTION

- *Image preprocessing module*: Processing features to be extracted to improve face recognition speed. The face image is cropped and resized to a smaller pixel value.
- *Image segmentation module*: Various facial features are extracted. The grayscale images from this step are used to identify the criminal.
- *Criminal identification module*: The extracted images are identified for faces using the dlib library and a deep learning algorithm, so the person's face alone is captured without interruption from the background.
- *Database and dataset module*: A tool for collecting and organizing information. Many pictures of criminals along with their identities are stored in the database and dataset. Image matching module: The resulting images are compared with existing images in the dataset. If a match is found, data related to that image is returned; otherwise, the recognized person is not a criminal.
- *Warning module*: If a criminal is detected, an alert message is sent along with the captured images and details of the criminal.
- *Alert and notification generation*: If a significant match is found, the system generates an alert that is sent to the relevant authorities or security personnel in real time, including the appearance, location and relevant details of the potential criminal.

2. LITERATURE REVIEW AND RESEARCH GAP

2.1 TRADITIONAL AND FEATURE-BASED FACE RECOGNITION

Earlier face-recognition systems commonly used geometric measurements, template matching, Eigenfaces, local descriptors, and cascade-based face detectors. These methods can be computationally economical and useful in constrained environments, but their performance may be affected by changes in illumination, pose, image resolution, occlusion, and camera distance. Haar cascade classifiers are suitable for locating candidate facial regions, but face detection and identity recognition are separate tasks: a detector identifies where a face may be present, whereas a recognition model estimates whether it corresponds to a reference identity. Consequently, a complete surveillance workflow must evaluate both stages rather than treating detection as proof of identity.

2.2 DEEP LEARNING AND FACIAL EMBEDDINGS

Deep-learning methods learn facial representations from image data and compare these representations rather than relying only on hand-crafted geometry. Schroff, Kalenichenko, and Philbin introduced FaceNet, which maps face images into a compact embedding space where distances represent facial similarity and supports verification, recognition, and clustering [11]. Embedding-based matching is relevant to the proposed system because reference images and incoming CCTV faces can be represented in a common feature space. However, published benchmark performance does not automatically transfer to low-resolution surveillance video, different camera hardware, or a locally collected reference database. The selected model, preprocessing pipeline, and matching threshold therefore require application-specific validation.

2.3 RELIABILITY, DEMOGRAPHIC VARIATION, AND OPERATIONAL RISK

Recognition performance depends on image quality, the composition of the reference population, and the algorithm used. The US National Institute of Standards and Technology (NIST) evaluated demographic differentials across face-recognition algorithms and reported that false-positive and false-negative rates can vary across demographic groups and capture conditions [12]. NIST further notes that poor exposure and other image-quality problems can increase failure-to-match rates, while false-match differentials may remain even for good-quality images [13]. These findings make it important to report error rates by relevant demographic and image conditions, rather than relying on a single overall accuracy value. In a policing-support context, false matches can lead to serious consequences; alerts should therefore be treated as leads requiring independent human verification.

2.4 RESEARCH GAP AND LIMITATIONS OF EXISTING APPROACHES

The reviewed literature and the existing approaches described in this manuscript reveal several practical gaps. First, many face-recognition demonstrations emphasize matching accuracy but provide limited detail about the end-to-end connection between camera input, identity records, alert delivery, investigator review, and audit logging. Second, performance reported on curated datasets may not represent real CCTV conditions, where faces may be small, blurred, partially occluded, or captured under uneven lighting. Third, a single similarity threshold can produce different error rates across datasets and demographic groups unless it is calibrated and tested carefully. Fourth, system descriptions often omit operational measures such as notification failures, duplicate alerts, access control, data retention, and the handling of uncertain matches. Finally, the references in the original draft primarily concern attendance systems, so they do not provide a sufficiently focused evidence base for surveillance-based identification.

2.5 POSITIONING OF THE PROPOSED SYSTEM

To address these design-level gaps, Spot Crime combines face localization, CNN embedding-based matching, authorized reference-record retrieval, preliminary alert generation, and an investigator-facing review workflow. It also proposes timestamped audit records, role-based access, image-quality checks, and a manual-review path for uncertain cases. The contribution is an integrated system design rather than a claim of a newly invented recognition algorithm or proven superiority over existing models. Its novelty should therefore be evaluated through implementation evidence and controlled comparisons with baseline methods.

2.6 RESEARCH QUESTIONS AND EVALUATION OBJECTIVES

The evaluation should address the following questions: (i) how accurately does the selected model match identities under representative surveillance conditions; (ii) how do false-match and missed-match rates change with image quality and database size; (iii) what end-to-end latency is introduced by frame sampling, inference, database search, and notification; and (iv) can authorized users review and audit candidate alerts reliably? The corresponding objectives are to measure precision, recall, false-match rate, false-non-match rate, processing latency, and notification-delivery success, and to report results across lighting, pose, resolution, occlusion, and relevant demographic subgroups. These are proposed evaluation objectives; results must be populated only after experiments are conducted.

3. PROPOSED METHODOLOGY

3.1 OVERVIEW OF THE PROPOSED REAL-TIME CRIMINAL IDENTIFICATION SYSTEM

The proposed Real-Time Criminal Identification System, termed Spot Crime, integrates computer vision, convolutional neural networks (CNNs), facial feature extraction, database-based face matching, and real-time alert generation to support surveillance-assisted criminal investigation. The system processes images acquired from CCTV cameras, web cameras, or uploaded image files and compares the detected faces with authorized facial records maintained in a centralized database. Its primary objective is to reduce the time required to identify a

potential match during an investigation while providing law enforcement personnel with a structured interface for reviewing detection results.

The proposed architecture consists of seven principal modules: image acquisition, preprocessing, face detection and segmentation, facial feature extraction, identity matching, criminal-record retrieval, and alert generation. Each module performs a specific function and transfers its output to the subsequent stage. The system uses OpenCV for image processing, a CNN-based model for facial representation, dlib for facial landmark localization where required, and MySQL for storing identity records and associated metadata. A web-based interface developed using Python and PHP supports authorized user registration, authentication, record management, and alert review.

Unlike a simple face-classification system, which attempts to classify a face directly as criminal or non-criminal, the proposed method treats facial recognition as an identity-matching problem. The system searches for a sufficiently similar facial representation in the authorized reference database. A potential match indicates that the detected person may correspond to an existing record; it does not establish that the person has committed a crime. Accordingly, the final identification decision remains subject to verification by an authorized investigator.

3.2 SYSTEM ARCHITECTURE

The system architecture follows a sequential image-processing and identity-verification pipeline. Initially, the camera captures video frames at regular intervals. The preprocessing module improves image consistency by resizing frames, correcting image quality where feasible, and converting suitable representations to grayscale for conventional detection operations. A face detector then identifies facial regions and extracts individual face images from the original frame. The CNN extracts discriminative facial embeddings, which are compared with reference embeddings generated from authorized database images.

The matching module computes similarity scores between the detected face and stored facial representations. When a candidate exceeds the configured matching threshold, the application retrieves the associated record and generates a preliminary alert containing the capture timestamp, camera identifier, location, similarity score, and reference-record identifier. The alert is transmitted to authorized personnel through the web application. When no candidate meets the threshold, the system records the outcome as an unmatched observation rather than automatically classifying the person as non-criminal.

3.3 IMAGE ACQUISITION AND PREPROCESSING

The image acquisition module obtains visual data from live surveillance cameras or user-uploaded images. For video input, the system extracts frames at a predefined sampling interval to control computational overhead while retaining sufficient temporal coverage for detecting individuals. Each frame is assigned a timestamp and a camera identifier, allowing the application to associate a detected face with its observation location and time: $V = \{I_t \mid t = 1,2,\ldots,T\}$, where I_t denotes the frame acquired at time t, and T denotes the number of processed frames. The preprocessing module transforms each frame into a standardized representation:

$$\tilde{I}_t = P(I_t) = N\left(R\left(C(I_t)\right)\right)$$

where $C(\cdot)$ denotes optional color-space conversion, $R(\cdot)$ represents resizing, and $N(\cdot)$ denotes image normalization. Grayscale conversion may be applied to detection operations that require intensity information, whereas the CNN receives the input format expected by its training configuration.

The preprocessing stage also evaluates image quality using indicators such as blur, illumination, and visible facial area. Images with inadequate quality are flagged for reprocessing or manual review rather than being forced through an unreliable identification decision. This step reduces unnecessary comparisons and helps maintain consistent input conditions.

3.4 FACE DETECTION AND FACIAL REGION EXTRACTION

Face detection identifies the position and dimensions of each face within the preprocessed frame. The proposed system supports OpenCV Haar cascade classifiers as a lightweight detection option and a CNN-based face detector when greater robustness is required. Dlib facial landmarks may subsequently be used to align the detected facial region. These components perform distinct functions: Haar cascades locate candidate faces, landmark detectors estimate facial geometry, and the recognition network generates identity-related features.

For a frame $\tilde{I}_t$, the detector produces a set of facial bounding boxes: $B_t = \{b_{t,1}, b_{t,2}, \ldots, b_{t,n_t}\}$, where n_t is the number of detected faces in frame t, and each bounding box is represented by $b_{t,j} = (x_{t,j}, y_{t,j}, w_{t,j}, h_{t,j})$, where, $x_{t,j}$ and $y_{t,j}$ indicate the upper-left coordinates, while $w_{t,j}$ and $h_{t,j}$ denote the width and height of the j-th detected face. The system extracts each facial region from the original frame and applies alignment when reliable landmark estimates are available. This operation reduces the influence of background objects, camera framing, and moderate facial misalignment. Multiple faces in the same frame are processed independently, allowing the system to compare several individuals against the reference database.

3.5 CNN-BASED EMBEDDED FACIAL FEATURE EXTRACTION

The facial feature extraction module converts each aligned face into a numerical representation that captures discriminative facial characteristics. A CNN is used because its successive convolutional layers can learn local patterns, intermediate facial structures, and higher-level representations from training images. The recognition network should be trained or pretrained for facial representation learning rather than treating a generic object-classification output as an identity embedding. Let the aligned facial image be denoted by $F_{t,j}$. The CNN transforms this input into an embedding vector:

$$z_{t,j} = f_\theta(F_{t,j}); z_{t,j} \in R^d$$

where $f_\theta(\cdot)$ denotes the CNN with learned parameters θ, and d represents the embedding dimension. The embedding is normalized to reduce the influence of its magnitude:

$$\hat{z}_{t,j} = \frac{z_{t,j}}{\| z_{t,j} \|_2 + \varepsilon}$$

where ε is a small positive constant that prevents division by zero.

The resulting representation allows the system to compare faces without relying exclusively on raw RGB values, face width, or face height. These geometric measurements can vary substantially with pose, camera distance, and illumination. Learned embeddings instead encode more useful patterns for facial similarity, although their reliability still depends on image quality, training data, demographic coverage, and camera conditions.

If a training stage is required, the CNN can be optimized using a suitable identity-classification or metric-learning objective. For example, a contrastive objective can encourage embeddings of the same identity to remain close while separating embeddings of different identities:

$$L_c = \frac{1}{2} y D^2 + \frac{1}{2}(1-y)\left[(0, m-D)\right]^2$$

where D denotes the distance between two embeddings, $y = 1$ indicates a same-identity pair, $y = 0$ indicates a different-identity pair, and m is a margin parameter. This objective is one possible training strategy; the final implementation must specify the selected CNN architecture and its actual training procedure.

3.6 CRIMINAL REFERENCE DATABASE CONSTRUCTION

The database module maintains facial reference images, identity metadata, and associated case records. Only records lawfully authorized for the specific identification purpose should be included. Each enrolled identity is associated with one or more suitable reference images, and the system computes an embedding for each image using the same preprocessing and recognition model employed during live inference.

The database can be organized into four principal tables:

- *Administrator*: Stores administrator credentials, role assignments, and account status.
- *Police Station*: Stores station identifiers, jurisdictional details, and authorized personnel information.
- *Criminal Record*: Stores the reference-record identifier, legally relevant case metadata, and authorized facial image references.
- *Detection and Alert Log*: Stores observation timestamps, camera identifiers, candidate-match scores, verification status, and alert history.

The original reference images and their derived embeddings should be stored with appropriate access controls. Passwords must be protected using a suitable password-hashing scheme rather than stored as plain text. Sensitive identifiers should be encrypted or otherwise protected, and access to facial records should be logged. For each reference identity k, the system stores a set of facial embeddings: $D = \{(r_k, E_k, m_k)\}_{k=1}^{N}$, where N denotes the number of enrolled reference identities, r_k represents the reference-record identifier, E_k contains the associated facial embeddings, and m_k represents the authorized metadata associated with the record. This structure separates identity information from the computational representation used for matching. It also supports updates when new reference images become available or when a record requires correction or removal.

3.7 FACIAL SIMILARITY COMPUTATION AND CANDIDATE IDENTIFICATION

The matching module compares the embedding extracted from a surveillance image against the reference embeddings stored in the database. For normalized embeddings, cosine similarity provides a convenient measure of feature alignment:

$$S_{j,k} = \frac{\hat{z}_{t,j}^T \hat{z}_k}{\| \hat{z}_{t,j} \|_2 \| \hat{z}_k \|_2}$$

where $\hat{z}_{t,j}$ represents the detected-face embedding and $\hat{z}_k$ denotes a stored reference embedding. For unit-normalized vectors, the similarity lies between -1 and 1, with higher values indicating greater feature similarity. The actual interpretation depends on the selected model and must be validated empirically.

For each detected face, the system identifies the highest-scoring reference candidate:

$$k^* = argarg\,max_{k \in \{1,..,N\}} S_{j,k}$$

The candidate is marked for further review only when the corresponding score satisfies a validated threshold:

$$\delta_j = \{1, S_{j,k^*} \geq \tau, 0, S_{j,k^*} < \tau$$

where τ denotes the selected decision threshold and δ_j indicates whether a candidate match has been generated. The threshold should be established using a representative validation dataset, taking false-match rates, missed matches, image conditions, and the consequences of incorrect identification into account.

A threshold crossing does not establish that the detected individual is the person named in the record. The system therefore presents the candidate image, reference image, similarity score, and associated record information to an authorized investigator for verification. Where image quality is poor or the score is insufficiently decisive, the observation is marked for manual review instead of producing a definitive identity claim.

3.8 REAL-TIME ALERT GENERATION AND NOTIFICATION

The alert generation module communicates potential matches to authorized personnel through the Spot Crime application. When the matching module identifies a candidate above the configured threshold, the system retrieves the corresponding reference record and creates a preliminary alert. Each alert includes the observation timestamp, camera identifier, location, detected face image, reference image, similarity score, and record identifier, subject to access permissions. The alert record can be represented as

$$A_{t,j} = \left(r_{k^*}, S_{j,k^*}, t, c_t, l_t, v_{t,j} \right)$$

where r_{k^*} is the candidate-record identifier, S_{j,k^*} is the matching score, t is the capture timestamp, c_t is the camera identifier, l_t is the recorded camera location, and $v_{t,j}$ represents the verification status.

The application sends notifications to the designated police station or authorized security personnel through the configured notification service. The dashboard displays pending alerts and permits authorized personnel to inspect the evidence, confirm or reject the candidate match, and record the outcome. Every action is stored in an audit log to support traceability and subsequent investigation.

To prevent repeated alerts for the same observation, the system can apply a deduplication mechanism based on the camera, timestamp window, detected-face track, and candidate record. Notification delivery failures should be logged and retried according to a defined policy. The system must also enforce retention rules and restrict access to surveillance images and identity information.

3.9 WEB APPLICATION AND USER AUTHENTICATION

The web application provides a controlled interface for managing users, reference records, surveillance observations, and alerts. Authorized administrators register police stations and assign user roles. After authentication, users can upload approved reference images, retrieve relevant records, inspect potential matches, and view previous alerts.

The application separates the presentation layer from the recognition and database layers. Python manages image processing and CNN inference, while PHP may provide the web-facing interface and application integration. MySQL stores structured records, access permissions, and detection logs. Communication between the application components should use authenticated interfaces and validated input data.

The principal application functions include:

- *User authentication*: Verifies user credentials and applies role-based permissions.
- *Reference management*: Adds, updates, and removes authorized facial records.
- *Live monitoring*: Receives camera frames and displays processing status.
- *Candidate review*: Presents detected faces alongside possible reference matches.
- *Alert management*: Records notification status and investigator decisions.
- *Audit and reporting*: Maintains timestamped records of access, alerts, and verification outcomes.

This modular design allows the recognition model to be updated without requiring a complete redesign of the database or user interface.

3.10 SYSTEM ANALYSIS

System analysis is concerned with analyzing, designing, implementing and evaluating the information system in the organization. It is carried out to make the system more effective either by modification or by substantial redesign. In system analysis, the problem is identified, alternative solutions are studied, and the most suitable solution meeting technical, economic and social demands is selected

3.11 EXISTING SYSTEM

It is very difficult to track a criminal person at the local, state or district level in India. There is no live criminal-tracking system, so criminals can easily move anywhere in the country with the help of a passport. Existing methods generally address the problem of face recognition (FR) in two steps: feature extraction and recognition, calculating a similarity score between two compared faces using the feature representation of each face. Many approaches have been implemented earlier for FR, such as neural networks, geometrical features, Eigen faces, template matching, and graph matching.

3.12 LIMITATIONS OF THE EXISTING SYSTEM

Crime increases rapidly and existing solutions struggle to keep pace.

- Large, pre-existing face databases are required.
- Accuracy of face recognition needs to remain very high.
- Multiple images must be detected at a time from video or CCTV footage.
- Face detection is only the first step of a fully automated face recognition and localization pipeline, and recognition remains a challenge.
- Knowledge-based methods rely on what constitutes a typical face, such as the relationship between facial features.
- Feature-invariant approaches must find structural features of a face that persist even when viewpoint, lighting or pose vary.

4. PROPOSED SYSTEM

This project is focused on the development of an application called the Real-Time Criminal Identification System, based on facial recognition. It is able to detect and recognize criminals from both images and a video stream obtained in real time. An authorized person logs in through the welcome page to upload images for viewing, which are stored in a database. A web camera installed at the entry point records people passing through, and image processing and segmentation is performed on the captured images.

Using dlib's point-detection algorithm, only faces are identified, and the identified images are compared with the images stored in the database. If a match is found, the captured image is sent to the authorized person along with the date and time via the application. This saves a lot of time and is a highly secure process, and criminals can be easily detected. Advantages of the Proposed System

- Provides information about a specific criminal that is being searched for.
- Police personnel can use this application anytime, anywhere, to find a criminal.
- Accessible over the internet from anywhere at any time.
- Criminals can also be identified from live CCTV camera feeds.
- Fast, robust, reasonably simple and accurate, with a relatively simple and easy-to-understand GUI.

4.1 PROPOSED METHODOLOGY AND ARCHITECTURE

4.1.1 System Specification:

- ***Hardware requirements***: Processor i5 or above, RAM 4 GB or more, Hard Disk Drive 500 GB.
- ***Software requirements***: Development platform Windows, front end Python and PHP, back end a dataset with MySQL.

4.2 DATA FLOW AND SYSTEM ARCHITECTURE

Input images from a live stream are acquired and pre-processed to grayscale. Feature extraction derives face features, which are matched against a reference database through a comparison process to produce a similarity measure.

A decision process then flags the result to the output system. At the application level, a user registers with police details, logs in, and can raise or view complaints, view cases, and submit feedback, each of which is stored and retrieved through the corresponding database tables.

4.2.1 Criminal Identification Methodology:

Image acquisition, feature extraction and classification using a CNN are used to determine whether a detected face is a criminal or non-criminal match, using the dataset of stored criminal images as the reference set.

The system operates in a highly modular fashion, maintaining strict relational structures to bind user logins, station authorities, and individual criminal records. The primary processing module ensures that input surveillance frames are filtered for high contrast, skin-tone boundaries, and structural alignment before matching.

4.3 DATABASE AND SYSTEM TABLES

A database is an organized collection of data as in Table.1. The data is typically organized to model relevant aspects of reality in a way that supports processes requiring this information. A general-purpose database management system (DBMS) such as MySQL is utilized to structure administrators, police departments, and criminal records.

The relational schema in Table.2 allows authorized law enforcement entities to register their designated credentials while anchoring alerts to specific jurisdictional branches. All records preserve timestamp integrity to establish seamless legal evidence chains during live tracking events.

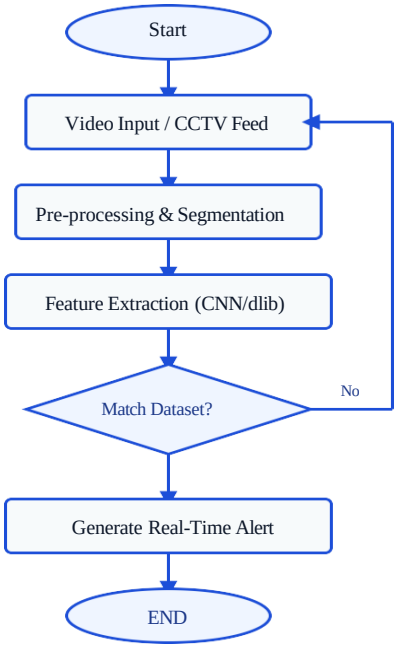


Fig.1. Criminal Tracking System Proposed Flowchart

Table.1. Table Name: Admin

Field Name	Data Type	Size	Constraints	Description
Aid	Int	11	Primary Key	Administrator id
Name	Varchar	50	Not Null	Name
Age	Int	1 0	Not Null	Age
Sex	Varchar	1 5	Not Null	Sex
Address	Text	-	Not Null	Address
Contact_no	Varchar	1 0	Not Null	Contact Number
Photo	Text	-	Not Null	Photo
Passwrd	Varchar	1 0	Not Null	Password
Date	Date	-	Not Null	Date

Table.2. Table Name: Police Station

Field Name	Data Type	Size	Constraints	Description
PL_id	Int	11	Primary key	Police station id
Name	Varchar	1 5	Not Null	Name
Sex	Varchar	1 0	Not null	Sex
Age	Int	1 0	Not Null	Age
Address	Text	-	Not Null	Address
Contact_no	Varchar	1 0	Unique	Contact Number
email	Varchar	20	Not null	Email
Passwrd	Varchar	20	Unique	Password
Photo	Text	-	Not Null	Photo
Date	Date	-	Not Null	Date

Table.3. Table Name: Criminal

Field Name	Data Type	Size	Constraints	Description
cr_id	Int	11	Primary key	Criminal Id

Name	Varchar	1 5	Not Null	Name
Parent_name	Varchar	20	Not Null	Parent Name
Sex	Varchar	1 5	Not Null	
Age	Int	1 0	Not Null	
Address	Text	-	Not Null	
Adhar_no	Int	1 0	Unique	
State	Varchar	20	Not Null	
Qualification	Varchar	20	Not Null	
Job	Varchar	20	Not Null	
Contact_no	Varchar	1 0	Unique	
Email	Varchar	20	Not Null	
Photo	Text	-	Not Null	
Date	Date	-	Not Null	

4.4 INPUT AND OUTPUT DESIGN

4.4.1 Input Design:

A major part in the design of the system is the preparation of the input. Input is necessary for the successful development and implementation of the system. The input design is the link that ties the information system into the process of converting oriented inputs to computer-based formats. The quality of the inputs determines the quality of the output. Input specifications describe the manner in which data enter the system for processing. Inaccurate input is the common cause of errors in data processing. This system is developed in a user-friendly manner. The forms are being designed in such a way that during processing the cursor is placed in the position where data must be entered. Validation routines verify each field, presenting appropriate error dialogs whenever invalid or incomplete input is detected.

4.4.2 Output Design:

The output is the most important and direct source of information to the user and should be provided in the most efficient, formatted way. Efficient and intelligible output design improves the system's relationship with the user and assists in swift decision-making. Outputs are classified into four principal categories:

- *External Outputs*: Information destined for external agencies or statutory systems.
- *Internal Outputs*: Inter-departmental records and operational summary documents.
- *Operational Outputs*: Live screen alerts, dispatch notices, and daily logs.
- *Interactive Outputs*: Direct human-computer query responses rendered on live dashboards.

5. RESULTS AND DISCUSSION

5.1 CURRENT EVIDENCE AND IMPLEMENTATION SCOPE

The manuscript describes the system architecture, processing modules, database entities, matching logic, and alert-review workflow. These details establish the proposed design and its intended operation; however, the supplied document does not report a defined test dataset, sample size, train/validation/test split, confusion matrix, benchmark comparison, measured recognition accuracy, or end-to-end latency. Accordingly, quantitative performance cannot be concluded from the current evidence. Statements in earlier drafts suggesting that the system is highly accurate or reduces crime should be interpreted as intended benefits, not experimentally demonstrated findings.

5.2 INTERPRETATION OF THE PROPOSED PIPELINE

The separation of face detection, embedding extraction, similarity matching, record retrieval, and alert delivery is a useful design choice because each stage can be tested independently. Image-quality screening can prevent low-quality frames from being treated as reliable identity evidence, while a threshold-based candidate list can make the matching decision inspectable. Nevertheless, the threshold creates a trade-off: lowering it may recover more true matches but can increase false alerts; raising it may reduce false alerts while missing genuine matches. The operating threshold should be selected on a representative validation set and then assessed on a separate test set. Reporting only accuracy would be insufficient, particularly where non-matching faces greatly outnumber matching faces.

5.3 REQUIRED QUANTITATIVE AND ROBUSTNESS EVALUATION

A reproducible evaluation should include genuine and impostor comparisons and report precision, recall, false-match rate, false-non-match rate, F1-score, and receiver operating characteristic or precision–recall analysis where appropriate. Runtime testing should measure average and percentile latency from frame acquisition to alert display, throughput under multiple faces, and notification delivery success. Robustness experiments should vary illumination, blur, pose, occlusion, face size, compression, and camera distance. Results should be compared with at least one conventional baseline and one embedding-based baseline using the same test protocol. Where sample sizes permit, confidence intervals and statistical tests should be reported. Performance should also be examined across relevant demographic groups, with suitable safeguards and lawful data governance. These experiments are necessary before making claims about operational readiness.

5.4 PRACTICAL IMPLICATIONS AND LIMITATIONS

The proposed web application can support the organization of candidate observations and reduce the need to manually search every stored image, but the practical benefit depends on the quality of the reference records, camera coverage, database maintenance, and timely alert delivery. The system has important limitations: the chosen CNN architecture and training configuration are not fully specified; no quantitative evaluation is reported; the reference dataset and its demographic coverage are not described; and deployment-scale behavior has not been established. In addition, an identity similarity score cannot establish criminal conduct. Operational use requires human verification, access restrictions, retention limits, auditability, and compliance with applicable law and institutional policy. These

limitations define the next steps for validating the design rather than negating its potential as an investigative-support prototype.

6. CONCLUSION

This paper presented Spot Crime, an AI-assisted surveillance-support design that integrates image acquisition, face detection, CNN-based embedded facial embeddings, similarity-based candidate matching, authorized record retrieval, and preliminary alert generation. The architecture connects recognition components with a web interface and relational database so that candidate observations can be reviewed and audited. A key design principle is that a similarity match is not proof of criminality: the system should provide investigative leads for authorized personnel, and identity decisions must be independently verified by a human reviewer. The manuscript also identifies important safeguards, including image-quality screening, validated thresholds, role-based access, secure credential handling, audit logs, and controlled retention of sensitive data. At present, the document establishes the proposed workflow but does not provide experimental measurements sufficient to claim a particular recognition accuracy, real-time performance level, or reduction in crime. The system should therefore be regarded as a prototype design whose effectiveness and reliability remain to be demonstrated through a transparent, independent evaluation.

7. FUTURE ENHANCEMENTS

Future work should begin with implementation and controlled benchmarking on a legally authorized, representative dataset containing clear documentation of its source, consent or legal basis, demographic coverage, and train/validation/test partitions. The selected CNN architecture, pretrained weights, embedding dimension, preprocessing settings, and similarity threshold should be specified so that the system can be reproduced. Evaluation should report precision, recall, F1-score, false-match and false-non-match rates, confidence intervals, and end-to-end alert latency, with comparisons against conventional and contemporary face-recognition baselines. Robustness tests should cover low illumination, motion blur, pose changes, occlusion, low resolution, camera variation, and increasing database size. The system should also be tested for demographic performance differences and notification failures, including duplicate-alert suppression and retry behavior. Additional engineering work includes encrypted storage, stronger password hashing, role-based permissions, access auditing, retention and deletion controls, and security testing of application interfaces. Any integration with external identity databases must be considered only where specifically lawful, necessary, proportionate, and governed by formal authorization; it should not be assumed as a default design choice. Finally, a pilot study with trained users should evaluate usability, false-alert handling, review time, and the quality of audit records before any operational deployment.

REFERENCES

[1] Mayur Surve, Priya Joshi, Sujata Jamadar and Minakshi Vharkate, "Automatic Attendance System using Face Recognition Technique", *International Journal of Recent Technology and Engineering*, Vol. 9, No. 1, pp. 1-14, 2020.

[2] Alankar Patil, K.P. Priya, Prajwal More and Aniruddh Joshi, "Attendance Monitoring using Face Recognition and Machine Learning", *International Journal of Future Generation Communication and Networking*, Vol. 13, No. 3, pp. 94-102, 2020.

[3] Munmun Bhagat, Chaitanya Kirkase, Yash Hawaldar, Aishwarya Paigude and Shivani Nimbalkar, "A Study on Automated Attendance System using Facial Recognition", *International Research Journal of Engineering and Technology*, Vol. 6, No. 1, pp. 1-12, 2019.

[4] G.R. Pooja, M. Poornima and S. Palakshi, "Automated Attendance System using Image Processing", *International Journal of Advanced Networking and Applications*, Vol. 15, No. 3, pp. 23-35, 2022.

[5] Naman Gupta, Purushottam Sharma, Vikas Deep and Vinod Kumar Shukla, "Automated Attendance System using OpenCV", *Proceedings of International Conference on Reliability, Infocom Technologies and Optimization*, pp. 1-5, 2020.

[6] V. Suresh, Srinivasa Chakravarthi Dumpa, Chiranjeevi Deepak Vankayala, Haneesha Aduri and Jayasree Rapa, "Facial Recognition Attendance System using Python and OpenCv", *Quest Journals Journal of Software Engineering and Simulation*, Vol. 5, No. 2, pp. 18-29, 2019.

[7] Adrian Rosebrock, "Face recognition with OpenCV, Python, and Deep Learning", Available at: pyimagesearch.com, Accessed in 2018.

[8] S. Poornima, N. Sripriya, B. Vijayalakshmi and P. Vishnupriya, "Attendance Monitoring System using Facial Recognition with Audio Output and Gender Classification", *Proceedings of International Conference on Computer, Communication and Signal Processing*, pp. 1-6, 2017.

[9] Venkata Kalyan, Muralidhar Reddy, Dheeraj Ravi and Mahitha Sai, "Attendance System based on Face Recognition", *International Research Journal of Engineering and Technology*, Vol. 5, No. 4, pp. 111-118, 2018.

[10] Manop Phankokkruad and Phichaya Jaturawat, "Influence of Facial Expression and Viewpoint Variations on Face Recognition Accuracy by Different Face Recognition Algorithms", *Proceedings of International Conference on Computer, Communication and Signal Processing*, pp. 61-66, 2017.

[11] F. 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*, pp. 815-823, 2015.

[12] P. Grother, M. Ngan and K. Hanaoka, "Face Recognition Vendor Test (FRVT) Part 3: Demographic Effects", Available at: https://nvlpubs.nist.gov/nistpubs/ir/2019/nist.ir.8280.pdf, 2019.

[13] National Institute of Standards and Technology, Available at: https://pages.nist.gov/frvt/reports/demographics/nistir_8429.pdf, Accessed in 2022.