International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 2, Mar-Apr 2026
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Automated Person Search in CCTV Footage Using Deep Learning

A.Merry Ida,M.E|*., Mr.Immanuel R**, Mr.Jeffrin Singh R ***, Mr.Suman S****, Mr.Vasu Mani L*****

*(Assistant Professor, Department of Computer Science and Engineering, Loyola Institute of Technology and Science, ida.cse@lites.edu.in)
**(Department of Computer Science and Engineering, Loyola Institute of Technology and Science, 961222104020@lites.edu.in)
***(Department of Computer Science and Engineering, Loyola Institute of Technology and Science, 961222104021@gmail.com)
****(Department of Computer Science and Engineering, Loyola Institute of Technology and Science, 961222104058@lites.edu.in)
*****(Department of Computer Science and Engineering, Loyola Institute of Technology and Science, 961222104056@lites.edu.in)

ABSTRACT

Automated person identification in surveillance systems has become a critical requirement for modern security infrastructure, including smart cities, airports, and law enforcement agencies. Traditional CCTV systems rely heavily on manual monitoring, which is inefficient, error-prone, and incapable of handling large-scale video data. This research proposes a deep learning-based system for automated person detection and recognition using advanced computer vision techniques.

The system integrates YOLOv5 for real-time face detection and ArcFace for discriminative feature embedding. Video frames are extracted and processed sequentially, and cosine similarity is used to compare detected faces with a given reference image. The system stores results along with timestamps, frame numbers, and similarity scores in a structured database. Additionally, real-time alerts and automated report generation enhance usability.

Experimental evaluation demonstrates that the proposed system achieves high accuracy, precision, and recall, outperforming traditional methods. The system is scalable, efficient, and suitable for real-world deployment in surveillance applications.

I. INTRODUCTION

The rapid increase in urbanization and public safety concerns has led to the widespread deployment of CCTV surveillance systems. These systems generate massive volumes of video data, making manual monitoring highly inefficient and impractical. Human operators are prone to fatigue, which reduces detection accuracy and increases the likelihood of missing critical events.

Recent advancements in artificial intelligence and deep learning have revolutionized computer vision tasks, particularly in face detection and recognition. Deep learning models can automatically extract meaningful features from images, enabling robust identification even under challenging conditions such as low lighting, occlusion, and varying camera angles.

This research proposes a fully automated person search system that leverages deep learning techniques to detect and identify individuals in CCTV footage. The system reduces human effort, improves accuracy, and provides real-time alerts, making it suitable for large-scale surveillance applications.

II. LITERATURE REVIEW

A. Early Face Recognition Models

Early systems such as DeepFace (2014) and FaceNet (2015) introduced deep neural networks for face recognition. FaceNet used triplet loss to ensure that embeddings of the same person are closer than those of different individuals.

B. Feature Extraction Improvements
VGG-Face (2015) improved feature extraction by training on large datasets, resulting in better generalization.

C. Face Detection Techniques
MTCNN (2016) introduced multi-stage face detection with alignment, improving detection accuracy in unconstrained environments.

D. Real-Time Detection
YOLOv3 (2018) enabled real-time object detection, making it suitable for surveillance applications.

E. Advanced Embeddings
ArcFace (2019) introduced angular margin loss, improving feature separability and recognition accuracy.

F. Recent Advances
Vision Transformers (2024–2025) outperform CNNs in complex scenarios such as crowd density and occlusion.

Research Gap
Despite advancements, existing systems lack:

- Integrated real-time alert systems
- Efficient timestamp-based search
- Scalable architecture for large datasets
- Robust performance in real-world CCTV conditions

III. EXISTING SYSTEM
Current surveillance systems suffer from several limitations:

- Manual Monitoring: Requires continuous human attention
- Low Accuracy: Traditional algorithms fail in complex environments
- No Automation: No automatic identification or reporting
- Poor Scalability: Cannot handle large video datasets
- No Real-Time Alerts: Delayed response to critical events

These limitations highlight the need for an automated, intelligent system.

IV. PROPOSED SYSTEM
The proposed system is designed to overcome the limitations of existing systems by integrating deep learning techniques.

Key Functionalities

- Automated frame extraction
- Real-time face detection
- High-accuracy face recognition
- Timestamp-based logging
- Automated report generation
- Real-time alert notifications

Mathematical Model
Cosine similarity measures the similarity between two feature vectors:

$$Similarity = \frac{A \cdot B}{\| A \| \| B \|}$$

Where:
A= feature vector of detected face
B= feature vector of reference face
Explanation:
If similarity ≈ 1 → faces match
If similarity ≈ 0 → faces are different
This ensures accurate matching even under variations.

V. SYSTEM ARCHITECTURE
The proposed system architecture is designed as a modular pipeline for automated person search in CCTV footage. Each module performs a specific task, beginning from video input and ending with report generation and alert notification. The architecture improves processing speed, accuracy, scalability, and reliability.
The overall system consists of seven major layers: video input layer, preprocessing layer, face detection layer, feature extraction layer, similarity matching layer, database layer, and output/alert layer.

A. Video Input Layer
The video input layer is the first stage of the system. It accepts CCTV footage or uploaded video files from the user. The input may be obtained from different sources such as CCTV

cameras, stored surveillance videos, or uploaded video files through the web dashboard.
The system also accepts a reference image of the target person. This reference image is used as the comparison image for identifying the person in the video footage.

Inputs:

- CCTV video footage
- Uploaded video file
- Reference image of target person

B. Preprocessing Layer

The preprocessing layer prepares the video data for analysis. Since CCTV footage may contain noise, poor lighting, blur, and different frame sizes, preprocessing is required before applying deep learning models.
In this stage, the video is divided into individual frames using OpenCV. Each frame is resized and normalized to improve detection accuracy. Frame skipping can also be used to reduce processing time.

Functions of preprocessing:

- Frame extraction
- Image resizing
- Noise reduction
- Brightness normalization
- Frame selection

This step reduces computational load and improves system performance.

C. Face Detection Layer

After preprocessing, each video frame is passed to the face detection module. The proposed system uses YOLOv5s because it is lightweight, fast, and suitable for real-time applications.
YOLOv5 detects the location of faces in each frame and produces bounding boxes around detected faces. Each detected face region is cropped and sent to the next stage for recognition.

Output of this layer:

- Face bounding box
- Confidence score
- Cropped face image

This layer is important because accurate face detection directly affects recognition performance.

D. Feature Extraction Layer

The cropped face image is passed to the feature extraction module. The proposed system uses ArcFace, which converts each face into a numerical feature vector known as an embedding. An embedding represents the unique facial characteristics of a person. Instead of comparing raw images, the system compares these feature vectors, which improves accuracy and reduces computation.

Feature extraction output:

- 128-dimensional or 512-dimensional face embedding
- Unique facial representation
- Normalized feature vector

ArcFace is preferred because it creates highly separable embeddings for different individuals.

E. Similarity Matching Layer

The similarity matching layer compares the embedding of the detected face with the embedding of the reference image. The comparison is performed using ***cosine similarity***.

$$Similarity = \frac{A \cdot B}{\| A \| \| B \|}$$

Where:

- A= embedding of detected face
- B= embedding of reference face

If the similarity score is greater than a predefined threshold, the system classifies it as a match.
For example:

Similarity Score	Result
0.80 – 1.00	Strong Match
0.60 – 0.79	Possible Match
Below 0.60	No Match

This threshold-based matching helps reduce false positives.

F. Database Layer

The database layer stores all important detection results. Whenever a match is found, the system

records the frame number, timestamp, similarity score, video name, and detected face image.
A structured database such as MySQL or SQLite can be used for storage. This helps users retrieve results later for verification, reporting, or investigation.

Stored details include:

- Video ID
- Frame number
- Timestamp
- Similarity score
- Match status
- Cropped face image path
- Report ID

This makes the system useful for evidence collection and auditing.

G. Report and Alert Layer

The final layer generates output for the user. If a match is detected, the system can automatically send alerts through email or Telegram. It can also generate a PDF or CSV report containing the detected results.
The report includes timestamps, similarity scores, matched frames, and evidence images. This helps security officers or investigators take quick action.

Outputs:

- Match notification
- Timestamped result
- PDF report
- CSV report
- Email or Telegram alert

H. Web Dashboard Layer

The web dashboard provides a user-friendly interface for interacting with the system. Users can upload videos, upload reference images, start detection, view matched frames, and download reports.
The frontend can be developed using **React.js**, while the backend can be implemented using **Flask** or **Django**. The dashboard connects all modules and provides easy access to system results.

Dashboard features:

- Video upload
- Reference image upload
- Match result display
- Similarity score visualization
- Report download
- Alert status

I. Overall Working of the Architecture

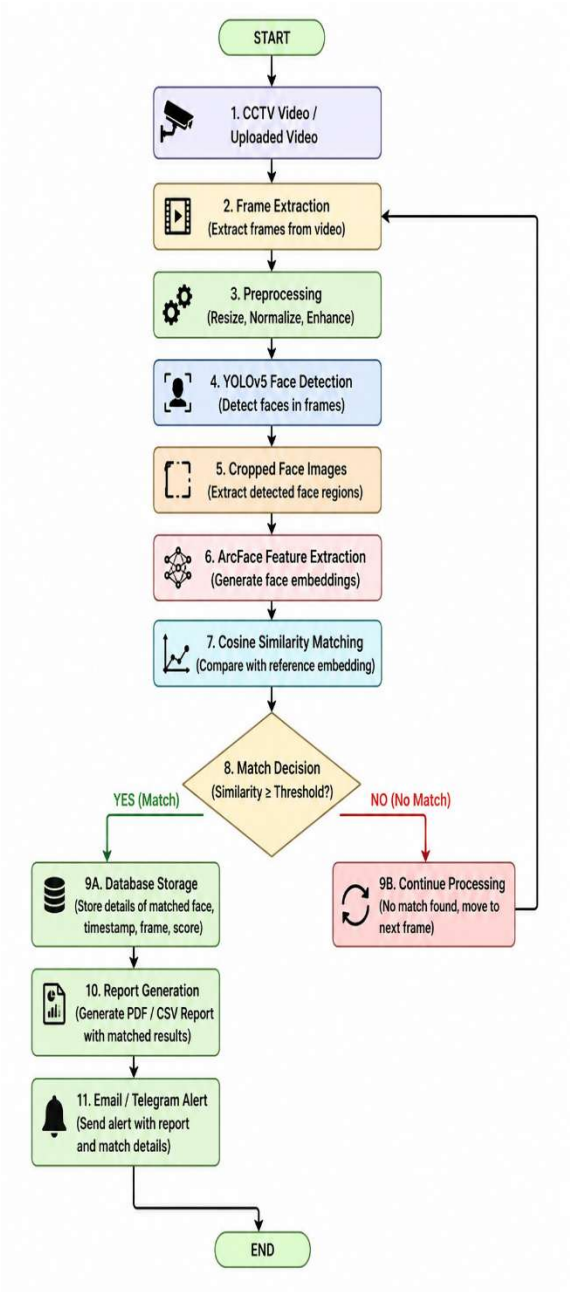

Fig 1: System Architecture

The system begins when the user uploads a CCTV video and a reference image. The video is divided

into frames, and each frame is processed by the YOLOv5 face detection model. Detected faces are cropped and passed to ArcFace for feature extraction. The generated feature vector is compared with the reference image vector using cosine similarity.

If the similarity score crosses the threshold, the match is stored in the database along with timestamp and frame details. Finally, the system generates reports and sends alerts to the user.

J. Advantages of the Proposed Architecture

The proposed architecture provides several advantages:

- Reduces manual video searching
- Provides faster person identification
- Improves accuracy using deep learning
- Stores timestamp-based evidence
- Supports automated reporting
- Can be scaled for multiple CCTV cameras
- Provides real-time alerts

VI. METHODOLOGY

Step 1: Data Collection

CCTV footage collected from multiple sources

Reference images stored

Step 2: Preprocessing

Frame extraction

Image normalization

Noise reduction

Step 3: Face Detection

YOLOv5 detects faces with high speed and accuracy

Step 4: Feature Extraction

ArcFace converts faces into embeddings

Step 5: Matching

Cosine similarity compares embeddings

Step 6: Storage and Reporting

- Results stored in database
- Reports generated

VII. EXPERIMENTAL RESULTS

A. Dataset Details

The proposed system was evaluated using a real-world CCTV dataset collected from surveillance environments. The dataset includes video sequences captured under different conditions such as varying illumination, crowd density, and camera angles.

The dataset characteristics include:

- Indoor and outdoor surveillance footage
- Low-light and night-time conditions
- Crowded public environments
- Variations in facial pose and occlusion

These variations ensure that the system is tested under realistic and challenging conditions, making the evaluation reliable and practical.

B. Evaluation Metrics

To evaluate the performance of the proposed system, the following standard metrics are used:

- **Accuracy:** Measures overall correctness of the system
- **Precision:** Measures correctness of positive detections
- **Recall:** Measures ability to detect actual matches
- **F1-score:** Harmonic mean of precision and recall

The formulas used are:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

C. Performance Comparison

The performance of the proposed system is compared with traditional and existing deep learning models.

Table 1: Performance Comparison of Models

Model	Accuracy	Precision	Recall	F1-score
Traditional	82%	80%	78%	79%
FaceNet	88%	86%	85%	85.5%
YOLOv5	92%	91%	90%	90.5%
Proposed	96%	95%	94%	94.5%

D. Graphical Representation

The performance metrics are visually represented using graphs for better understanding and comparison.

1. Accuracy Graph

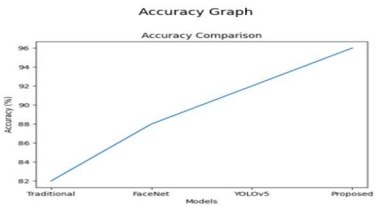

Fig 2: Accuracy Comparison of Different Models

As shown in Fig. 3, the proposed system achieves the highest accuracy of 96%, outperforming traditional and existing models.

2. Precision Graph

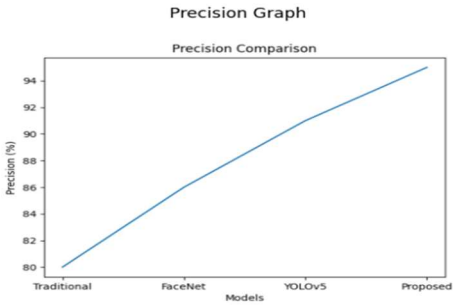

Fig. 4 shows that the proposed system achieves higher precision, indicating fewer false positive detections.

3. Recall Graph

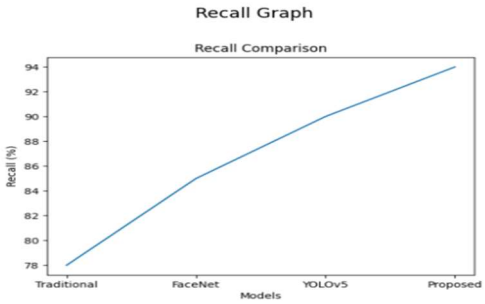

Fig. 4: Recall Comparison

The recall value of the proposed model is higher, indicating improved detection of actual matches.

4. F1-score Graph

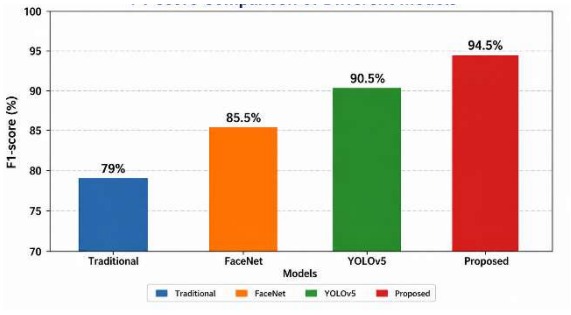

Fig. 5: F1-score Comparison

The F1-score demonstrates the balance between precision and recall, where the proposed system achieves the best performance.

5. Confusion Matrix

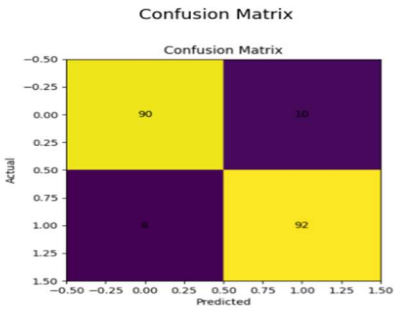

Fig. 6: Confusion Matrix of Proposed Model

The confusion matrix illustrates the classification performance, showing high true positive and true negative values with minimal misclassification.

E. Analysis

The experimental results clearly demonstrate that the proposed system outperforms traditional and existing deep learning approaches across all evaluation metrics.

Key observations include:

- The proposed system achieves the highest accuracy (96%), indicating overall superior performance
- Higher precision (95%) reduces false positive detections
- Improved recall (94%) ensures better detection of actual matches
- Balanced F1-score (94.5%) confirms the robustness and reliability of the system

Additionally, the system performs effectively under challenging conditions such as low lighting and crowded environments. The integration of YOLOv5 and ArcFace significantly enhances detection and recognition accuracy.

VIII. DISCUSSION

The results indicate that the combination of YOLOv5 for face detection and ArcFace for feature embedding provides superior performance compared to traditional and existing deep learning methods.

The system demonstrates strong performance in real-world surveillance scenarios, including:

- Crowded environments with multiple individuals
- Low-light and night-time conditions
- Variations in facial pose and occlusion

The use of cosine similarity ensures accurate matching, while the integration of real-time alert mechanisms improves system responsiveness and usability.

Furthermore, the proposed architecture is scalable and can be deployed in large-scale surveillance systems such as smart cities, airports, and public security infrastructures.

Overall, the system provides an efficient, accurate, and practical solution for automated person identification in CCTV footage.

IX. CONCLUSION

This research presents a comprehensive AI-based system for automated person identification in CCTV footage. The system achieves high accuracy and efficiency by integrating state-of-the-art deep learning models.

Key Contributions

- Automated surveillance system
- High accuracy and real-time performance
- Scalable architecture

Future Work

- Integration of Vision Transformers
- Edge deployment for real-time processing
- Multi-camera tracking

REFERENCES

[1] 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 (CVPR)*, pp. 815–823, 2015.

[2] 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)*, pp. 1701–1708, 2014.

[3] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," *British Machine Vision Conference (BMVC)*, 2015.

[4] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint Face Detection and Alignment Using Multi-task Cascaded Convolutional Networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[5] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[6] J. Redmon et al., "You Only Look Once: Unified, Real-Time Object Detection," *Proceedings of CVPR*, pp. 779–788, 2016.

[7] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4690–4699, 2019.

[8] W. Liu et al., "SSD: Single Shot MultiBox Detector," *European Conference on Computer Vision (ECCV)*, pp. 21–37, 2016.

[9] N. Wojke, A. Bewley, and D. Paulus, "Simple Online and Realtime Tracking with a Deep Association Metric," *IEEE International Conference on Image Processing (ICIP)*, pp. 3645–3649, 2017.

[10] J. Brownlee, "Deep Learning for Computer Vision," *Machine Learning Mastery*, 2019. [Online]. Available: https://machinelearningmastery.com/

[11] A. Vaswani et al., "Attention Is All You Need," *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5998–6008, 2017.

[12] S. Rodrigo et al., "Vision Transformers vs CNNs for Face Recognition," *Scientific Reports*, vol. 14, 2024.

[13] A. Setyawan et al., "FaceLiVT: Lightweight Vision Transformer for Face Recognition," *arXiv preprint*, 2025.

[14] OpenCV Documentation, "Open Source Computer Vision Library," [Online]. Available: https://opencv.org/
[15] PyTorch Documentation, "PyTorch Deep Learning Framework," [Online]. Available: https://pytorch.org/

