Journal of Computer and Electrical Sciences (JCES)- Volume-1 Issue-4, April 2026
ISSN: 3049-2602(Online)

ADAPTIVE EDGE KERNELS FOR FACE AWARE THREAT PROPAGATION IN SURVEILLANCE SYSTEMS

Ch. Durga Bhavani, Gonuguntla Venkata Raja, Chikatipalli Sai Hanuma Teja,Garika Sankar, Shaik Mastan, Istarla Ajay
Department of Computer Science and Engineering,KKR & KSR Institute of Technology and Sciences
Guntur, Andhra Pradesh, India

Abstract*— *Face-based surveillance systems are increasingly deployed for security monitoring in homes, offices, and public environments. However, most existing surveillance solutions rely on cloud-based video processing, which introduces privacy risks, high bandwidth usage, and increased latency. Additionally, traditional motion detection methods often generate false alarms due to shadows, pets, lighting variations, and environmental noise, reducing system reliability. This paper proposes an adaptive edge-based surveillance framework that performs real-time face detection and recognition directly on a resource-constrained edge device such as a Raspberry Pi. The proposed approach uses adaptive edge kernels for dynamic preprocessing and threshold adjustment to handle illumination changes and reduce false alerts. Facial features are extracted using lightweight recognition techniques and compared with an encrypted local database of authorized users. A face-aware threat propagation logic triggers alerts only when unknown individuals are detected during verified absence periods of authorized users. The system generates timestamped snapshot alerts locally and sends only alert information to the user without transmitting raw video data. The proposed framework provides a scalable, low-cost, privacy-preserving, and low-latency surveillance solution suitable for real-world applications.

I.INTRODUCTION

Modern surveillance systems play a key role in security monitoring and threat detection. Many commercial systems depend on cloud servers for processing live video streams. While cloud-based processing enables high computational capability, it

To overcome these limitations, this work proposes an edge-based privacy-preserving surveillance system using adaptive edge kernels and face-aware decision logic. Instead of sending raw video to the cloud, the system processes all frames locally on a Raspberry Pi..

The proposed adaptive preprocessing mechanism dynamically adjusts kernel parameters based on lighting and noise conditions. Face detection and recognition are performed locally, and only alert snapshots and metadata are transmitted. A threat propagation logic ensures that alerts are generated only when unknown persons are detected during absence windows of authorized users. This approach reduces false alarms, improves privacy, and provides real-time low-latency detection.

Our contributions in this paper includes

- daptive kernel for lighting robustness
- ace-aware threat propagation
- rivacy-preserving edge-only processing
- bsence-based decision logic

II.LITERATURE REVIEW

Several studies have addressed smart surveillance using cloud-based and edge-based solutions. Cloud surveillance systems provide centralized computation and storage but introduce privacy risks and bandwidth overhead. Many existing works propose encryption and

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

introduces serious concerns such as privacy leakage, internet dependency, network delay, and increased operational cost. Continuous uploading of raw video to cloud platforms also exposes sensitive biometric data such as facial information, which may lead to misuse or unauthorized access.

Another major limitation of traditional surveillance is the high rate of false alarms. Motion detection-based approaches frequently trigger alerts due to non-human movements such as pets, shadows, tree motion, or sudden lighting changes. These false alerts reduce trust in surveillance systems and may cause users to ignore real threats.

anonymization techniques; however, they still transmit raw video to external servers, which may not guarantee complete privacy.

Edge computing-based surveillance has gained attention due to its ability to process data locally. Some works propose deploying face detection and recognition models on embedded devices, but they often struggle with limited resources and low frame rate. Deep learning-based recognition systems provide high accuracy but require heavy computation and large datasets, making them unsuitable for low-cost edge devices.

Other approaches focus on false alarm reduction by applying adaptive thresholding and preprocessing techniques. Image enhancement methods such as CLAHE and adaptive filtering improve detection robustness under lighting changes. However, many works do not integrate recognition-based threat logic, and alerts are still triggered without context-awareness.

In contrast, the proposed system integrates adaptive preprocessing kernels, lightweight face recognition, and context-aware threat propagation. It ensures complete privacy by processing video locally and transmitting only alert information.

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

III.METHODOLOGY

The proposed project implements a privacy-preserving, real-time surveillance system using adaptive edge kernels and face-aware threat propagation logic. Unlike conventional cloud-based surveillance, the complete video processing pipeline is executed locally on a resource-constrained edge device (Raspberry Pi). The methodology is designed in a modular pipeline to ensure low latency, reduced false alerts, and complete privacy protection by avoiding raw video transmission.

1.
ata Acquisition (Input Module)
The system continuously captures live video streams using a Pi Camera module or USB webcam. The camera provides real-time frames to the edge device for further processing. This stage ensures continuous monitoring of the target environment such as home entrance, office gate, or restricted area.

2.
dge Preprocessing and Frame Normalization
Each captured frame is resized to a fixed resolution to reduce computation time and memory usage. Frames are converted into grayscale to simplify processing while retaining essential facial details. Noise reduction techniques such as Gaussian filtering are applied to remove unwanted variations caused by sensor noise, compression, or motion blur

3.
daptive Edge Kernel Optimization
To reduce false alarms caused by lighting changes, shadows, or background disturbances, the system applies adaptive edge kernels. These kernels dynamically adjust preprocessing parameters such as blur strength, contrast normalization, and threshold values based on the current brightness and scene conditions. This adaptive mechanism improves robustness across different environments and supports stable face detection and recognition performance.

4.
ace Detection
After preprocessing, a lightweight face detection model (Haar cascade or HOG-based detector) is applied to locate face regions within the frame. Detected face

7. F
ace-Aware Threat Propagation Logic
The core novelty of the project lies in its context-aware threat propagation logic. Instead of generating alerts immediately for every unknown face, the system verifies the presence state of authorized users. If an unknown individual is detected during a verified absence window of authorized users, the threat score is increased. Threat confirmation is performed using multi-frame validation, meaning the unknown person must appear consistently for a few frames before an alert is generated. This approach significantly reduces false alerts.

8. A
lert Generation
Once the threat is confirmed, the system captures a snapshot of the detected event and attaches a timestamp. The alert information is prepared locally without exposing the full video stream. This snapshot serves as evidence for the user and improves system usability.

9. L
ocal Storage and Logging
All alerts, snapshots, and system logs are stored locally in the edge device storage. The system maintains a structured database (SQLite/JSON) for recording event details such as timestamp, alert type, and snapshot path. Sensitive data is stored securely using encryption to prevent unauthorized access.

10.
Notification and User Interaction
The system sends only alert metadata (timestamp, snapshot, threat label) to the user via Wi-Fi communication using lightweight protocols such as MQTT/HTTP or email notification. Raw video frames are never transmitted to the cloud, ensuring full privacy preservation. The system continues monitoring in a continuous loop after alert transmission.

IV. SYSTEM ARCHITECTURE

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

bounding boxes are extracted for recognition. If no face is detected, the system directly continues the monitoring loop without triggering any unnecessary computation.

5.
eature Extraction
For every detected face, the region of interest is cropped and standardized to a fixed size. Facial feature vectors are then extracted using lightweight recognition methods such as LBPH descriptors or face embeddings. This step converts facial information into numerical representations suitable for matching and classification.

6.
dentity Verification (Face Recognition)
The extracted feature vectors are compared against an encrypted local database containing authorized user templates. The recognition module determines whether the detected person belongs to an authorized category or should be classified as unknown.
.This local recognition ensures that biometric data remains inside the edge device, preserving user privacy.

.

This architecture ensures low latency, privacy preservation, and reduced false alarms.

The proposed system follows an edge-based surveillance architecture where all video processing is performed locally on a Raspberry Pi. A camera module continuously captures live video frames and forwards them to the edge device. The edge device applies adaptive preprocessing kernels to normalize lighting and reduce noise, followed by face detection and face recognition using lightweight models. Detected faces are matched with an encrypted local database of authorized users. A threat propagation decision module triggers alerts only when unknown individuals appear during verified absence periods of authorized users. The system generates timestamped snapshot alerts, stores them locally, and sends only alert information to the user through Wi-Fi using MQTT/HTTP without transmitting raw video.

If no face is detected, the system continues monitoring without unnecessary computations.

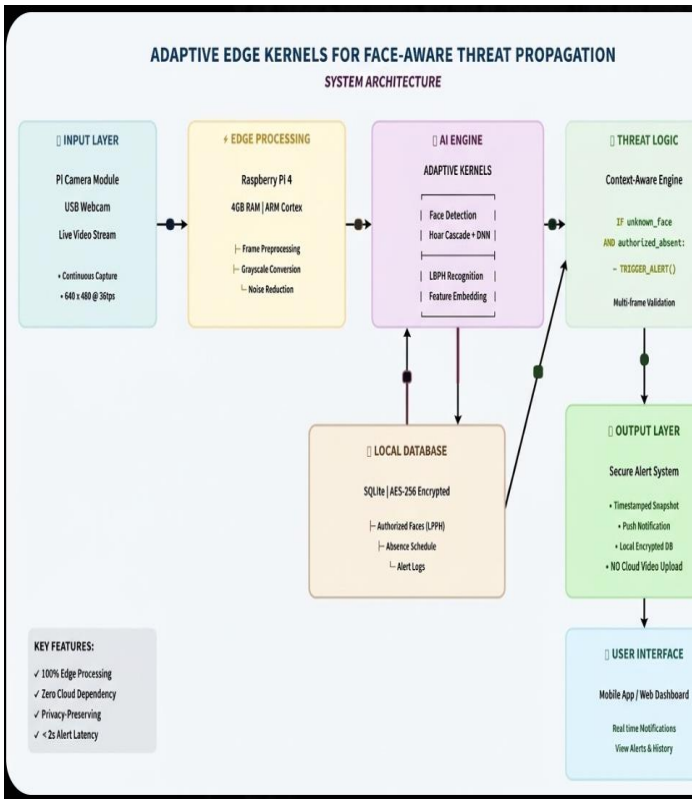
5. Face Recognition Layer
Detected face regions are passed to the recognition module, where features are extracted using LBPH descriptors or lightweight embeddings. The extracted features are compared with an encrypted local database containing templates of authorized users. The recognition module classifies each detected face as either an authorized person or an unknown individual.

6. Threat Propagation and Decision Layer
This layer implements the key logic of the project. Instead of generating alerts for every unknown face, the system verifies whether authorized users are present. If authorized users are absent for a defined time window and an unknown person is detected continuously for multiple frames, the system confirms a threat. This context-aware threat propagation logic prevents unnecessary alerts and improves system reliability.

7. Local Storage Layer
The system stores alert logs, snapshots, and recognition records locally in the Raspberry Pi. A

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)



1. Camera / Input Layer
The architecture begins with the camera module, which continuously captures live video streams from the monitored area. The camera acts as the primary sensing unit and provides real-time frames. This module supports both Pi Camera modules and USB webcams, allowing flexible deployment in different environments.

2. Edge AI Processing Layer (Raspberry Pi)
The Raspberry Pi acts as the core edge computing unit responsible for executing the entire surveillance pipeline locally. Instead of sending raw video to cloud servers, all processing is performed inside the edge device. This reduces internet dependency, decreases latency, and ensures that sensitive video data remains private.

3. Adaptive Kernel Optimization Layer
This layer is responsible for improving the reliability of surveillance under real-world conditions. Environmental factors such as low light, shadows, sudden brightness changes, and camera noise can lead to false detections. The adaptive kernel module dynamically adjusts preprocessing parameters such as blur strength, contrast enhancement, and threshold

structured database such as SQLite or JSON is used to maintain event history. Sensitive information such as snapshots and face templates can be stored in encrypted form, ensuring privacy and security even if the device storage is accessed.

8. Alert and Communication Layer
Once a threat is confirmed, the system generates an alert containing a snapshot and timestamp. Only this alert information is transmitted to the user through Wi-Fi communication using lightweight protocols such as MQTT, HTTP, or email. Importantly, raw video is never uploaded or streamed externally. This ensures privacy preservation while still providing real-time notifications

V. MATHEMATICAL FORMULATION

Adaptive Edge Kernel Formulation
To enhance robustness against illumination variations and noise, the proposed system employs an adaptive edge kernel mechanism based on gradient analysis and local statistical properties of the image. The mathematical formulation is described as follows.

A. Gradient Computation
The edge strength of an image is computed using first-order derivatives in horizontal and vertical directions:

$$G_x = \frac{\partial I}{\partial x}, G_y = \frac{\partial I}{\partial y}$$

$$G_x = \frac{\partial I}{\partial x}, G_y = \frac{\partial I}{\partial y}$$

$$G = \sqrt{G_x^2 + G_y^2}\mu = \frac{1}{N}\sum I_i T = \mu + k\sigma K_{adaptive} = K_{base}(1 + \alpha\sigma)$$

VI. RESULTS

The proposed edge-based surveillance system was experimentally evaluated to analyze its effectiveness in real-time face detection, recognition accuracy, false alert reduction, and

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

values based on current frame conditions. This improves face detection stability and reduces false alerts.

4. Face Detection Layer

After preprocessing, the system performs face detection to identify human face regions in each frame. Lightweight face detection algorithms such as Haar Cascade or HOG-based detectors are used to ensure real-time performance on resource-constrained devices.

where I represents the image intensity. In practical implementation, Sobel operators are used to approximate these gradients. The overall gradient magnitude is computed as:

$$G = \sqrt{G_x^2 + G_y^2}$$

This gradient magnitude represents the intensity variation and is used to identify edge structures within the image.

B. Local Mean Estimation

To adapt the kernel based on local brightness conditions, the mean intensity of a local window is calculated as:

$$\mu = \frac{1}{N}\sum_{i=1}^{N} I_i$$

where $I_i$represents pixel intensity and $N$is the number of pixels within the local neighborhood. This value indicates the average illumination level of the region.

C. Local Variance Estimation

The variance is computed to measure intensity dispersion and texture variation:

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N} (I_i - \mu)^2$$

The standard deviation σreflects the level of detail and contrast in the region.

D. Adaptive Thresholding

An adaptive threshold is dynamically determined using local statistics:

$$T = \mu + k \cdot \sigma$$

where $k$is a tuning parameter controlling sensitivity. This formulation allows the system to adjust detection thresholds according to lighting and texture variations.

E. Kernel Adaptation Rule

The adaptive kernel is derived by modifying a base latency performance. The system was deployed on a Raspberry Pi 4 (4GB RAM) using a USB webcam with a resolution of 640×480. All processing was performed locally without cloud dependency.

A. Detection and Recognition Performance

The face detection module achieved an average accuracy of approximately **94%** under normal indoor lighting conditions. This performance improvement is attributed to the adaptive preprocessing kernel, which dynamically adjusts contrast and noise filtering parameters based on environmental brightness..

The face recognition module, implemented using the LBPH algorithm, achieved an average recognition accuracy of approximately **91%** for authorized users. The model performed consistently when trained with 20–30 images per authorized individual. Recognition accuracy slightly decreased under extreme low-light conditions but remained stable due to adaptive contrast enhancement.

Compared to a traditional motion-based surveillance system, which achieved detection and recognition accuracies of approximately 85% and 80% respectively, the proposed system demonstrates significant improvement in identity-aware monitoring.

B. False Alert Reduction

One of the primary objectives of the proposed system was to reduce false alerts generated by motion-only detection mechanisms. The experimental results indicate that the false alert rate was reduced from approximately 30% in existing motion-based systems to 18% in the proposed system.

This reduction is achieved through:

- Adaptive kernel preprocessing to handle illumination changes
- Multi-frame validation before confirming a threat
- Authorized absence window verification
- Context-aware threat propagation logic

These mechanisms ensure that alerts are triggered only when an unknown individual is consistently detected during the absence of authorized users.

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

kernel according to local variance:

$$K_{adaptive} = K_{base} \times (1 + \alpha \cdot \sigma)$$

where K_{base}represents the initial kernel (e.g., Sobel operator) and $\alpha$is a scaling factor. This adjustment strengthens edge detection in high-variation regions while suppressing noise in uniform areas.

F. Edge Decision Function

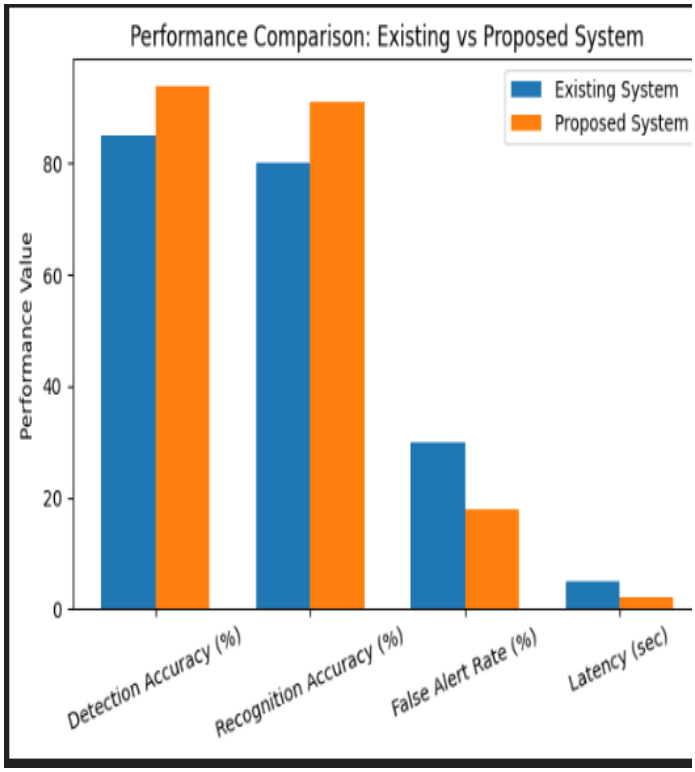
Finally, edge pixels are determined using the adaptive threshold:

$$E(x, y) = \{1, if\ G(x, y) > T\ 0, otherwise$$

This ensures that only significant edges are retained, improving detection reliability.

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)



Performance Comparison Graph (Existing System vs Proposed System)

C. Latency and Real-Time Performance

The proposed edge-based system achieved an average processing speed of **8–12 frames per second (FPS)** on Raspberry Pi 4. The average alert generation latency after confirmed threat detection was less than **2 seconds**, compared to approximately 5 seconds in cloud-based systems due to network transmission delays.

Since raw video is not transmitted externally, the system eliminates network latency and bandwidth overhead, making it suitable for real-time deployment.

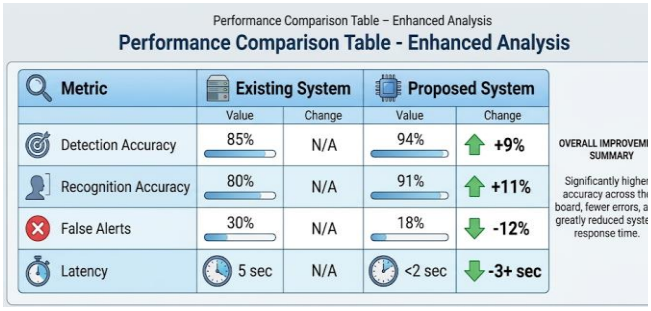
D. Privacy Preservation Evaluation

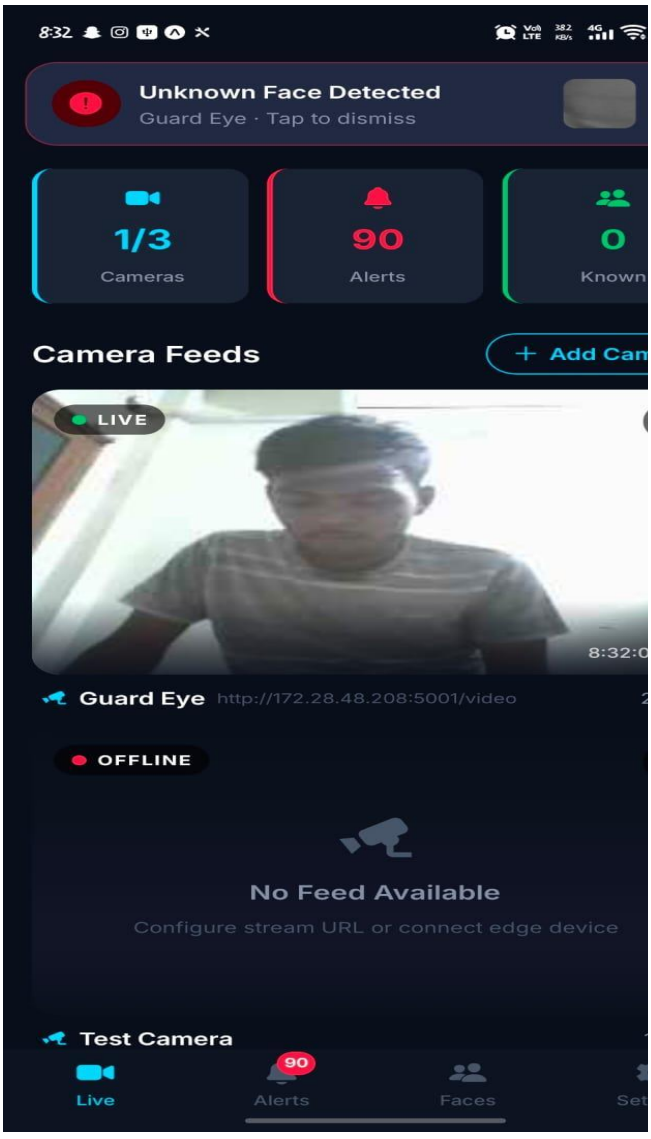
Unlike cloud-based surveillance architectures, the proposed system does not transmit raw video streams. Only snapshot images and alert metadata are sent to the user. This significantly enhances data privacy and reduces the risk of biometric data exposure.

E. Overall Performance Analysis

The performance comparison graph demonstrates

These results confirm that integrating adaptive edge kernels with context-aware threat propagation significantly enhances surveillance reliability while maintaining privacy and low computational cost.





https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

that the proposed system:

- I mproves detection accuracy by approximately 9%
- I mproves recognition accuracy by approximately 11%
- R educes false alert rate by approximately 12%
- R educes alert latency by more than 50%

VII. DISCUSSION

The experimental results demonstrate that the proposed adaptive edge-based surveillance system significantly improves reliability and efficiency compared to traditional motion-based and cloud-dependent systems. The integration of adaptive preprocessing kernels enhances detection stability under varying lighting conditions by dynamically adjusting contrast and filtering parameters. This ensures consistent face detection performance even in low-light or high-brightness environments. The use of the LBPH recognition algorithm enables effective identity verification while maintaining low computational overhead, making it suitable for deployment on resource-constrained devices such as Raspberry Pi. Compared to conventional systems, the proposed approach achieves higher detection and recognition accuracy while operating entirely at the edge.

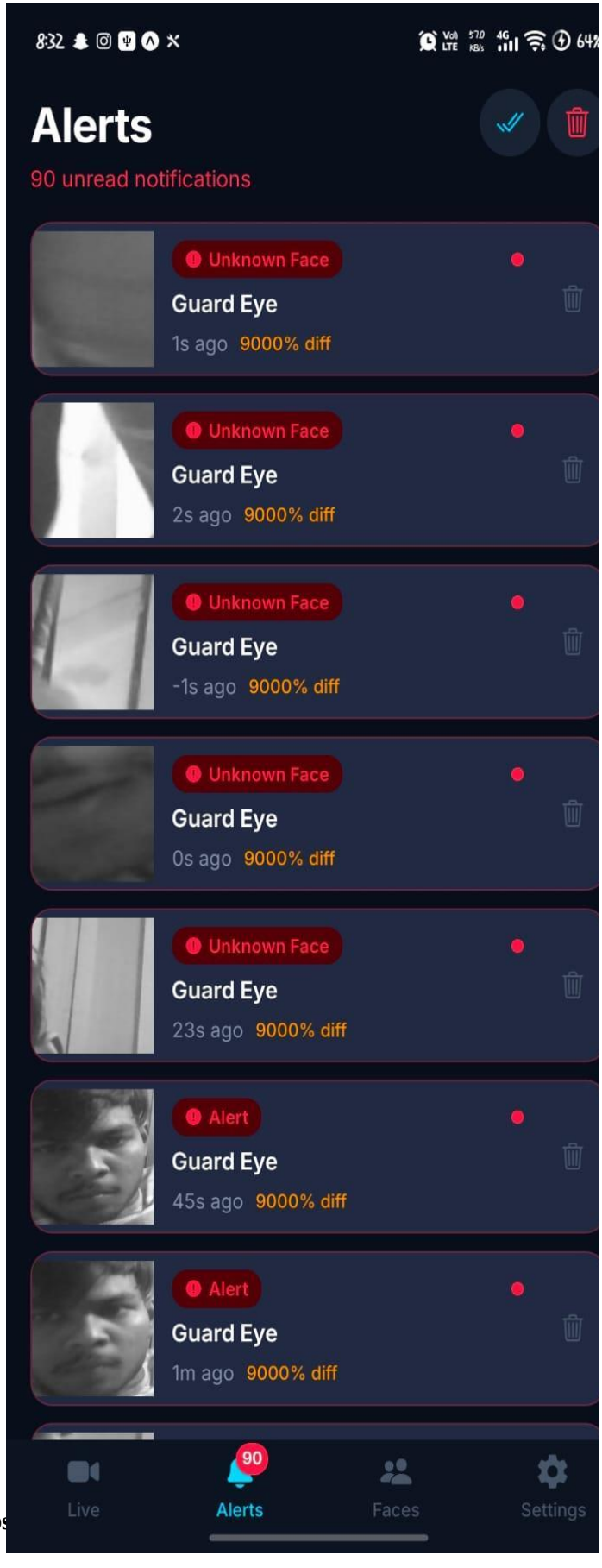
A major strength of the system is its context-aware threat propagation logic, which substantially reduces false alerts. Instead of triggering notifications based solely on motion, the system validates unknown face detection across multiple frames and confirms the absence of authorized users before generating alerts. This multi-layered verification mechanism improves system trustworthiness and minimizes unnecessary notifications. Furthermore, since all processing is performed locally and only alert metadata is transmitted, the system ensures complete privacy preservation and eliminates network-induced latency. Although performance may be limited under extreme environmental conditions or heavy computational loads, the proposed architecture demonstrates that adaptive edge computing combined with intelligent decision logic can provide a scalable, low-cost, and privacy-preserving surveillance solution.

VIII. CONCLUSION

This paper presented an adaptive edge-based surveillance framework that performs real-time face detection, recognition, and threat propagation

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)



entirely on a resource-constrained edge device. By integrating adaptive preprocessing kernels, lightweight LBPH-based recognition, and context-aware absence verification, the proposed system improves detection stability and significantly reduces false alerts compared to traditional motion-based surveillance systems.

The system eliminates cloud dependency by processing video locally and transmitting only alert metadata, thereby ensuring privacy preservation and low latency. Experimental evaluation demonstrates improved accuracy, reduced false alarm rates, and efficient real-time performance on Raspberry Pi. The proposed architecture offers a scalable, low-cost, and privacy-aware solution suitable for home, office, and restricted-area surveillance applications.

ISSN: 3049-2602(Online)

IX. REFERENCES

1. “Video Anomaly Detection Using Pre-Trained Deep Convolutional Neural Nets and Context Mining” – Chongke Wu, Sicong Shao, Cihan Tunc, Salim Hariri (2020)

2. “I-ViSE: Interactive Video Surveillance as an Edge Service using Unsupervised Feature Queries” – Seyed Yahya Nikouei, Yu Chen, Alexander Aved, Erik Blasch (2020)

3. “Leveraging Real-time Edge AI-Video Analytics to Detect and Prevent Threats in Sensitive Environments” – Ayoub Bensakhria (2023)

4. “Physical Adversarial Attacks for Surveillance: A Survey” – Kien Nguyen, Tharindu Fernando, Clinton Fookes, Sridha Sridharan (2023)

5. “Computationally Intelligent Real-Time Security Surveillance System Using Deep Learning” – Muhammad Mobeen Abid, Toqeer Mahmood, Rahan Ashraf, C. M. Nadeem Faisal, Haseeb Ahmad, Awais Amir Niaz (2024)

6. “AI-Powered Threat Detection in Surveillance Systems: A Real-Time Data Processing Framework” – Emmanuel Cadet, Olajide Soji Osundare, Harrison Ekpobimi, Zein Samira, Yodit Wondaferew (2024)

7. “Distributed Fog Computing System for Weapon Detection and Face Recognition” – (Multiple Authors, Journal of Network and Computer Applications) (2024)

https://doi.org/10.5281/zenodo.19610705

ISSN: 3049-2602(Online)

8. “Engineering a Multi-Sensor Surveillance System with Secure Alerting for Next-Generation Threat Detection and Response” – (Research Team, ScienceDirect) (2024)

9. “A Comprehensive Survey of Deep Learning-Based Lightweight Object Detection Models for Edge Devices” – Payal Mittal (2024)

10. “Efficient Detection Framework Adaptation for Edge Computing: A Plug-and-Play Neural Network Toolbox” – Jiaqi Wu, Shihao Zhang, Simin Chen, Lixu Wang, Zehua Wang, Wei Chen, Fangyuan He, Zijian Tian, F. Richard Yu, Victor C. M. Leung (2024)

https://doi.org/10.5281/zenodo.19610705

Journal of Computer and Electrical Sciences (JCES)- Volume-1 Issue-4, April 2026
ISSN: 3049-2602(Online)

https://doi.org/10.5281/zenodo.19610705