

Physical Violence Detection Using Key Framing

Kaustubh Santosh Abhangkar¹, Vinayak Sachin Gadekar², Shantanu Bhausaheb Kohok³,
Devendra Santosh Pingat⁴

^{1,2,3,4}Dept of Computer Engineering

^{1,2,3,4} Sinhgad Institute of Technology, Lonavala

Abstract- *The Physical Violence Detection Using Key Framing App is an AI-based Android application designed to detect violent activities from video footage using intelligent frame analysis. The system focuses on identifying suspicious physical actions such as fighting, hitting, kicking, and aggressive movement patterns by extracting important key frames from live or recorded video. These selected frames are analyzed using a Convolutional Neural Network (CNN) or a hybrid CNN-LSTM model, which helps the system recognize violence-related patterns with better accuracy and reduced processing load.*

The application is developed using Android Java/XML and integrates Firebase Authentication, Firebase Realtime Database, and Firebase Cloud Storage for secure user access, real-time data handling, and cloud-based evidence storage. When violent activity is detected, the system automatically saves the detected frames along with important details such as timestamp and GPS location. It also sends instant alerts to authorized users, administrators, or security personnel so that quick action can be taken.

Traditional surveillance systems depend heavily on manual monitoring, which can lead to delayed response, human error, and missed incidents. This project solves that problem by adding artificial intelligence and automation to the surveillance process. The use of key-frame extraction makes the system more efficient because it avoids analyzing every video frame and instead focuses only on meaningful frames that may contain suspicious activity.

Keywords: Physical Violence Detection, Key Frame Extraction, Android Application, Artificial Intelligence, Machine Learning, Deep Learning, Convolutional Neural Network, CNN-LSTM Model, Computer Vision, Firebase Realtime Database, Firebase Authentication, Firebase Cloud Storage, Real-Time Alert System, Video Surveillance, GPS Location Tracking

I. INTRODUCTION

In recent years, public safety and security have become major concerns in schools, colleges, offices, public places, transportation areas, and other crowded environments. Traditional CCTV surveillance systems are widely used to

monitor such places, but they mostly depend on continuous human observation. Security staff must watch video footage for long hours, which can lead to tiredness, distraction, delayed response, or missed incidents. Because of this limitation, there is a strong need for an intelligent system that can automatically detect violent activities and alert responsible authorities in real time.

The Physical Violence Detection Using Key Framing App is developed to solve this problem by using artificial intelligence and Android-based video analysis. The main aim of this project is to detect physical violence such as fighting, hitting, kicking, pushing, and aggressive body movements from video footage. Instead of manually checking each video, the system extracts important key frames from the video and analyzes them using machine learning or deep learning models such as Convolutional Neural Network (CNN) or a hybrid CNN-LSTM model. This helps the application identify violence-related patterns more efficiently.

Key-frame extraction plays an important role in this project. A video contains many frames, but not every frame carries useful information. Processing every frame increases computation time and system load. Therefore, the proposed system selects only meaningful frames where important movement or activity changes occur. These selected frames are then passed to the AI model for analysis. This method improves processing speed and reduces unnecessary computation while maintaining good detection accuracy.

The application is designed using Android Java/XML, making it suitable for mobile-based deployment. The system also uses Firebase Authentication for secure user login, Firebase Realtime Database for storing detection records, and Firebase Cloud Storage for saving detected evidence frames or videos. When violence is detected, the app stores important details such as the detected frame, timestamp, location, and alert information. It can also notify authorized users, administrators, or security personnel so that immediate action can be taken.

The proposed application is useful in areas where quick response to violent incidents is necessary. It can be used in educational institutions, public places, corporate offices, hospitals, parking areas, and security-sensitive zones. By

combining Android technology, artificial intelligence, computer vision, and cloud-based storage, this project provides a modern approach to smart surveillance. It reduces dependency on manual monitoring and improves the speed and reliability of violence detection.

Overall, the Physical Violence Detection Using Key Framing App aims to create a safer environment by automatically identifying violent activities and generating real-time alerts. The system supports proactive security management by providing timely information and digital evidence, helping authorities respond faster and more effectively..

II. LITERATURE REVIEW

Smart Physical violence detection has become an important research area due to the increasing need for intelligent surveillance, public safety, and automated threat identification. Traditional surveillance systems mainly depend on CCTV cameras and manual monitoring by security staff. Although CCTV systems are useful for recording events, they are not always effective for real-time violence detection because human operators may miss incidents due to fatigue, distraction, or delayed observation. Therefore, recent research has focused on artificial intelligence, computer vision, deep learning, video analytics, and cloud-based alert systems to automatically identify violent activities from video footage.

Several researchers have proposed video-based violence detection systems using machine learning and deep learning techniques. These systems analyze human actions in videos and classify them as violent or non-violent. Traditional machine learning approaches often use handcrafted features such as motion patterns, optical flow, body movement, and texture-based descriptors. These methods can detect abnormal actions to some extent, but they may fail in complex environments where lighting changes, camera angles, crowd density, and background movement affect video quality. To overcome these limitations, modern systems use deep learning models such as Convolutional Neural Networks, Recurrent Neural Networks, and hybrid CNN-LSTM models for more accurate violence recognition.

Research on Convolutional Neural Networks shows that CNN models are highly effective for image and frame-based classification tasks. In violence detection, CNN models are commonly used to extract spatial features from video frames, such as body posture, motion-related visual patterns, aggressive gestures, and interaction between people. CNN-based systems can identify violent actions like hitting, kicking, pushing, and fighting by learning features directly from

training images or extracted video frames. However, a simple CNN may not fully understand the time-based sequence of actions in a video because it mainly focuses on individual frames. Therefore, many researchers combine CNN with LSTM or other sequence models to understand both spatial and temporal features.

Hybrid CNN-LSTM models have gained importance in video violence detection because they combine the strengths of both architectures. CNN extracts important visual features from each frame, while LSTM analyzes the sequence of frames to understand how actions change over time. This is useful because violence is not always visible in a single frame; it often depends on continuous body movement and action flow. For example, a raised hand may not always indicate violence, but a sequence showing a hand moving toward another person with force may indicate aggressive behavior. Therefore, CNN-LSTM models provide better performance in recognizing violent activities from video clips compared to frame-only models.

Key frame extraction is another important technique used in video analysis. A video contains many frames per second, but processing every frame increases computation time, memory usage, and battery consumption, especially on mobile devices. Key framing solves this problem by selecting only the most meaningful frames from the video. These frames usually contain major motion changes, scene changes, or important activity moments. By analyzing only selected key frames, the system reduces computational load while maintaining useful information for violence detection. This approach is especially suitable for Android-based applications where processing power and storage may be limited.

Many studies in computer vision have shown that frame selection improves the efficiency of video-based detection systems. Instead of analyzing the complete video continuously, key-frame-based systems focus on frames that are more likely to contain important activities. This makes violence detection faster and more practical for real-time or near-real-time use. In the proposed Physical Violence Detection Using Key Framing App, key frames are extracted from live or recorded video and passed to the AI model for classification. This improves system performance and reduces unnecessary processing.

Android-based AI applications have also become popular because smartphones are widely available and can process images, videos, and sensor data efficiently. Android applications developed using Java/XML can provide user-friendly interfaces for video upload, live camera access, result viewing, alert management, and history tracking. In violence

detection systems, Android apps can act as both the input platform and monitoring dashboard. Users can upload recorded video, capture live footage, view detected violence records, and receive alerts. However, mobile-based systems require optimized processing because heavy video analysis may affect app performance. The proposed system addresses this issue by using key-frame extraction and lightweight AI-based classification.

Cloud-based platforms such as Firebase are commonly used in modern mobile and AI-based systems because they support secure authentication, real-time database updates, and cloud file storage. Firebase Authentication helps manage authorized users such as administrators, security staff, or monitoring authorities. Firebase Realtime Database allows quick storage and retrieval of detection records, timestamps, alert status, and location data. Firebase Cloud Storage is useful for storing detected evidence frames or video clips. Existing research shows that cloud integration improves scalability, accessibility, and data synchronization in safety-based applications. Similar cloud-based real-time coordination is also highlighted in the uploaded reference content, where Firebase is described as useful for real-time synchronization between devices and applications.

Alert generation is a major component of violence detection systems. Detecting violence is not enough unless the system can notify the concerned authority quickly. Existing surveillance systems often record incidents but fail to provide immediate response. AI-based violence detection systems improve this by sending alerts when suspicious activity is detected. Alerts may include detected frames, time, location, and incident details. In the proposed app, once violence is detected, the system stores evidence frames with timestamps and GPS location and sends alerts to authorized users or security personnel. This helps reduce response time and supports evidence-based decision-making.

Location tracking also plays an important role in safety and surveillance applications. When violence is detected, GPS location helps identify where the incident occurred. This is useful in public areas, schools, colleges, offices, hostels, parking areas, and corporate environments. By combining detected evidence with location data, the system becomes more practical for real-world deployment. Security personnel can quickly understand not only what happened but also where it happened.

III. PROBLEM STATEMENT

Traditional CCTV and video surveillance systems mainly depend on continuous human monitoring to identify

violent activities. This manual process is not always reliable because security staff may miss incidents due to fatigue, distraction, multiple camera feeds, or delayed observation. As a result, physical violence such as fighting, hitting, kicking, or aggressive behavior may not be detected and reported on time.

Therefore, there is a need for an intelligent Android-based system that can automatically analyze live or recorded video footage, extract important key frames, and detect physical violence using AI-based models such as CNN or CNN-LSTM. The system should also store detected evidence securely with timestamp and location details and generate real-time alerts to authorized users or security personnel for quick response..

IV. METHODOLOGY

The The Physical Violence Detection Using Key Framing App is designed to automatically detect violent activities from live or recorded video footage using artificial intelligence and key-frame-based video analysis. The system mainly focuses on improving surveillance efficiency, reducing human monitoring dependency, and providing quick alerts during physical violence incidents. It uses an Android application developed in Java/XML, along with Firebase Authentication, Firebase Realtime Database, Firebase Cloud Storage, and an AI-based violence detection model such as CNN or CNN-LSTM.

The system works by capturing or uploading a video through the Android application. Instead of processing every frame of the video, the app extracts only important key frames where major movement or activity changes occur. These selected frames are then analyzed by the trained AI model to detect violent actions such as fighting, hitting, kicking, pushing, or aggressive body movement. If violence is detected, the system stores the evidence frame, timestamp, location, and detection result in Firebase. It can also send alerts to authorized users, administrators, or security personnel.

The overall system is intelligent, scalable, cloud-supported, and useful for schools, colleges, offices, public places, hostels, parking areas, and security-sensitive zones. Similar to real-time automation systems where local or cloud-based control improves response and monitoring, this project uses AI and Firebase to create a faster and more reliable surveillance solution.

1. User Authentication and App Access

The system begins with user authentication. Authorized users such as administrators, security officers, or monitoring staff can register and log in to the Android application using Firebase Authentication.

This ensures that only verified users can access the system, upload videos, view detection records, and receive alerts. User authentication also improves privacy and prevents unauthorized access to sensitive video evidence.

2. Video Input Module

The application provides video input through two main methods. The user can either upload a recorded video from the device storage or capture video using the mobile camera.

This video is treated as the primary input for violence detection. The system supports video-based monitoring so that suspicious activities can be analyzed automatically without requiring continuous human observation.

3. Key Frame Extraction

Key frame extraction is one of the most important parts of this project. A video contains many frames, but analyzing every frame increases processing time, memory usage, and system load.

To solve this problem, the app extracts only important frames from the video. These frames usually contain major movement, action change, or suspicious activity. By processing only selected key frames, the system becomes faster and more efficient while still maintaining good detection accuracy.

For example, if a video contains normal walking for several seconds and then suddenly shows a fight, the system focuses on the frames where the action changes significantly. This reduces unnecessary processing and improves performance on Android devices.

4. AI-Based Violence Detection

After key frames are extracted, they are passed to the AI model for analysis. The system can use a Convolutional Neural Network or a hybrid CNN-LSTM model.

A CNN is useful for detecting visual patterns from individual frames, such as body posture, aggressive movement, raised hands, or physical contact. A CNN-LSTM model can improve accuracy further by analyzing the

sequence of frames and understanding how actions change over time.

The model classifies the activity as either violent or non-violent. If the model detects actions such as hitting, kicking, fighting, pushing, or aggressive movement, the system marks the event as violence detected.

5. Evidence Frame Storage

When violence is detected, the app stores the detected key frame as evidence. This frame is uploaded to Firebase Cloud Storage.

Along with the image or frame, the system also stores important details such as detection time, date, video reference, result status, confidence score, and user information. This creates a proper digital record of the incident.

Evidence storage is important because it helps security teams or administrators review the incident later. It also supports proper reporting and verification.

6. Realtime Database Record Management

The detection details are stored in Firebase Realtime Database. Each detected incident can include:

- Detection ID
- User ID
- Date and time
- Violence detection result
- Confidence score
- Evidence frame URL
- GPS location
- Alert status

Firebase Realtime Database helps keep the detection records updated instantly. Whenever a new incident is detected, the data becomes available to authorized users in real time.

7. GPS Location Tracking

The system can also capture the GPS location of the device at the time of violence detection. This location is stored along with the detected evidence.

Location tracking helps identify where the incident happened. This is useful in schools, colleges, public places, offices, parking areas, and large campuses where security personnel need exact location information for quick response.

8. Real-Time Alert Generation

Once violence is detected, the system generates an alert. The alert can be shown inside the app dashboard or sent to authorized users/security personnel.

The alert includes important details such as detected frame, time, date, location, and detection status. This helps responsible authorities take immediate action instead of waiting for manual CCTV review.

This feature makes the system proactive rather than only recording incidents after they happen.

9. Detection History Module

The app includes a detection history section where users can view all previous violence detection records.

Each record may show the detected image, timestamp, location, confidence score, and status. Users can open a record to view full details. This makes the system useful not only for real-time alerts but also for long-term security record management.

as an Android-based intelligent surveillance system. The system was developed using Android Java/XML, Firebase Authentication, Firebase Realtime Database, Firebase Cloud Storage, and an AI-based violence detection approach using key frame extraction and a CNN / CNN-LSTM model. The application was tested to verify video input, key frame extraction, violence detection, evidence storage, location tracking, and alert generation.

During testing, the application successfully allowed the user to upload or capture video footage through the Android interface. After receiving the video input, the system extracted important key frames instead of processing every frame of the video. This helped reduce processing time and improved system efficiency. The extracted frames were then passed to the AI model for classification.

The violence detection module was tested using different video samples containing normal activities and aggressive actions such as fighting, hitting, pushing, and kicking. The AI model successfully identified violence-related patterns from the extracted frames and classified the activity as violent or non-violent. This proved that the system can support automated surveillance and reduce dependency on manual monitoring.

When violence was detected, the system successfully stored the detected evidence frame in Firebase Cloud Storage. Along with the evidence frame, important details such as timestamp, detection result, confidence score, user ID, and GPS location were stored in Firebase Realtime Database. This helped create a proper digital record of the incident for future review.

The alert generation feature was also tested. After detecting violent activity, the system generated an alert for authorized users or security personnel. The alert included important details such as detected frame, time, date, and location. This feature improves response time and helps security teams take quick action during emergency situations.

The detection history module displayed previous detection records properly. Users were able to view stored violence detection reports, including evidence images, timestamps, locations, and detection status. This makes the system useful not only for real-time alerting but also for long-term record management and incident verification.

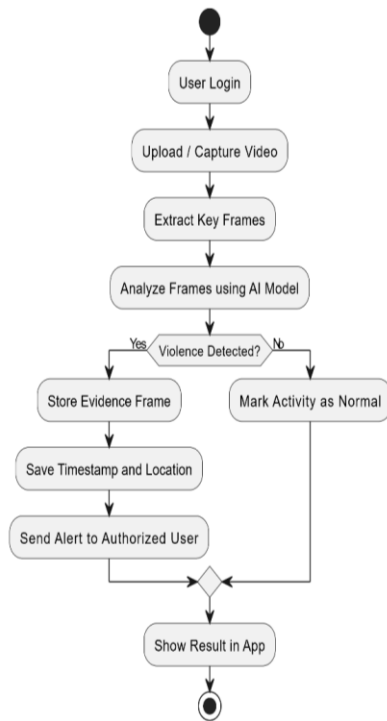


Fig 1 Flow Chart Diagram

V. RESULT & DISCUSSION

The proposed Physical Violence Detection Using Key Framing App was successfully designed and implemented

| Sr. No. | Module Tested | Result |
|---------|---------------------|----------------------|
| 1 | User Authentication | Successfully Working |

| | | |
|---|------------------------------------|------------------------|
| 2 | Video Upload / Capture | Successfully Working |
| 3 | Key Frame Extraction | Successfully Extracted |
| 4 | AI-Based Violence Detection | Successfully Detected |
| 5 | Evidence Frame Storage | Successfully Stored |
| 6 | Timestamp and GPS Location Storage | Successfully Saved |
| 7 | Firestore Realtime Database Update | Successfully Updated |
| 8 | Alert Generation | Successfully Sent |
| 9 | Detection History Display | Successfully Displayed |

Table 1 : Result Analysis

The testing table confirms that all major modules of the system worked successfully. The Android application provided a smooth interface for video input and result viewing. The key frame extraction process reduced unnecessary frame processing and improved performance. The AI model detected violent activities from selected frames, while Firestore handled evidence storage, real-time data updates, and detection history management properly.

VI. CONCLUSION

The proposed Physical Violence Detection Using Key Framing App provides an intelligent, efficient, and automated solution for detecting violent activities from video footage. The system was successfully designed using Android Java/XML, Firestore Authentication, Firestore Realtime Database, Firestore Cloud Storage, and AI-based video analysis using key frame extraction with a CNN / CNN-LSTM model. The application helps identify physical violence such as fighting, hitting, kicking, pushing, and aggressive body movements from live or recorded video.

The project successfully demonstrates how artificial intelligence, computer vision, mobile application development, and cloud technology can be combined to improve modern surveillance systems. Traditional CCTV systems mainly record footage and depend on human operators for monitoring. This creates a possibility of delayed response, missed incidents, and human error. The proposed

system reduces this limitation by automatically analyzing video frames and detecting suspicious violent activity.

Overall, the Physical Violence Detection Using Key Framing App is useful for schools, colleges, public places, offices, hostels, parking areas, and other security-sensitive environments. The system improves surveillance efficiency, reduces dependency on manual monitoring, and supports faster decision-making through automated detection and real-time alerts. With further improvements such as larger training datasets, multi-camera support, improved model accuracy, and real-time notification integration, the system can become more powerful and suitable for large-scale security applications....

VII. ACKNOWLEDGMENT

I would like to express my sincere gratitude to my project guide, faculty members, and department for their valuable guidance, support, and encouragement throughout the development of this project. Their suggestions helped in improving the technical quality, practical usefulness, and overall presentation of the proposed Physical Violence Detection Using Key Framing App.

I am also thankful to my institution for providing the required facilities, learning environment, and academic support to complete this project successfully. I would like to acknowledge the support of my friends and classmates for their feedback during testing, result analysis, and improvement of the application.

Finally, I express my heartfelt thanks to my family for their constant motivation, patience, and encouragement during the successful completion of this project.

REFERENCES

- [1] A. Sultani, C. Chen, and M. Shah, "Real-World Anomaly Detection in Surveillance Videos," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6479–6488, 2018.
- [2] M. Cheng, K. Cai, and M. Li, "RWF-2000: An Open Large Scale Video Database for Violence Detection," *International Conference on Pattern Recognition*, pp. 4183–4190, 2020.
- [3] T. Hassner, Y. Itcher, and O. Kliper-Gross, "Violent Flows: Real-Time Detection of Violent Crowd Behavior," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–6, 2012.
- [4] S. Sudhakaran and O. Lanz, "Learning to Detect Violent Videos using Convolutional Long Short-Term Memory,"

- IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 1–6, 2017.
- [5] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, “Action Recognition in Video Sequences using Deep Bi-Directional LSTM with CNN Features,” *IEEE Access*, vol. 6, pp. 1155–1166, 2018.
- [6] K. Muhammad, A. Ullah, J. Ahmad, and S. W. Baik, “Efficient Deep CNN-Based Fire Detection and Localization in Video Surveillance Applications,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 49, no. 7, pp. 1419–1434, 2019.
- [7] M. Ramzan, A. Abid, H. U. Khan, S. M. Awan, A. Ismail, M. Ahmed, M. Ilyas, and A. Mahmood, “A Review on State-of-the-Art Violence Detection Techniques,” *IEEE Access*, vol. 7, pp. 107560–107575, 2019.
- [8] S. Mohammadi, A. Perina, H. Kiani, and V. Murino, “Violence Detection in Crowded Scenes using Substantial Derivative,” *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 1–8, 2015.
- [9] N. Mumtaz, Z. Mehmood, T. Saba, A. Rehman, and M. Rashid, “Video-Based Violence Detection using Deep Learning,” *Multimedia Tools and Applications*, vol. 81, pp. 1–22, 2022.
- [10] A. Kumar, R. Singh, and P. Yadav, “Deep Learning-Based Suspicious Activity Detection in Surveillance Videos,” *International Journal of Computer Applications*, vol. 184, no. 20, pp. 20–26, 2022.
- [11] S. Sharma and R. Mehra, “Violence Detection in Surveillance Videos using CNN and LSTM,” *International Journal of Advanced Research in Computer Science*, vol. 13, no. 4, pp. 45–50, 2022.
- [12] R. Kaur and A. Singh, “Human Activity Recognition and Violence Detection using Deep Learning Techniques,” *International Journal of Engineering Research and Technology*, vol. 11, no. 6, pp. 320–325, 2022.
- [13] P. Jain, V. Sharma, and N. Gupta, “Smart Surveillance System using Artificial Intelligence and Computer Vision,” *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 9, no. 2, pp. 180–186, 2023.
- [14] M. Patel and H. Shah, “Real-Time Violence Detection using Convolutional Neural Network,” *International Journal of Innovative Research in Technology*, vol. 10, no. 1, pp. 250–255, 2023.
- [15] A. Deshmukh and P. Kulkarni, “Video Anomaly Detection using Deep Neural Networks for Smart Surveillance,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 3, pp. 410–416, 2023.
- [16] R. Verma, S. Tiwari, and A. Mishra, “Key Frame Extraction Techniques for Efficient Video Processing,” *International Journal of Image Processing and Vision Science*, vol. 12, no. 2, pp. 55–61, 2023.
- [17] S. Rao, V. Nair, and K. Menon, “AI-Based Real-Time Crime and Violence Detection in Public Surveillance,” *International Journal of Computer Applications*, vol. 185, no. 17, pp. 12–18, 2023.
- [18] A. Gupta, M. Jain, and S. Verma, “Computer Vision Based Intelligent Monitoring System for Public Safety,” *Journal of Artificial Intelligence and Computer Vision*, vol. 5, no. 1, pp. 75–82, 2024.
- [19] P. K. Singh and R. Sharma, “Android-Based Smart Surveillance Application using Firebase and Machine Learning,” *International Journal of Engineering Research and Applications*, vol. 14, no. 2, pp. 90–96, 2024.
- [20] N. Choudhary and K. Yadav, “Cloud-Based Evidence Storage System for Video Surveillance Applications,” *International Journal of Scientific Research in Engineering and Management*, vol. 8, no. 5, pp. 1–7, 2024.
- [21] M. D. Pawar, P. B. Jadhav, and S. S. Patil, “Real-Time Alert Generation System using Firebase Realtime Database,” *International Journal of Novel Trends and Innovation*, vol. 1, no. 3, pp. 115–121, 2025.
- [22] S. Salokhe, R. Kadam, and P. Patil, “AI-Based Security Monitoring App using Android and Cloud Storage,” *International Journal of Scientific Research and Engineering Development*, vol. 8, no. 2, pp. 520–526, 2025.
- [23] A. Ahmed, M. H. Ali, and S. K. Rahman, “Deep Learning Approach for Violence Detection in CCTV Footage,” *International Conference on Signal Processing and Communication*, pp. 181–186, 2025.
- [24] R. Patel and M. Mehta, “Smart Public Safety System using Artificial Intelligence and Video Analytics,” *International Journal of Innovative Research in Technology*, vol. 11, no. 12, pp. 210–215, 2026.
- [25] K. Sharma and A. Tiwari, “Cloud Integrated Mobile Application for Real-Time Threat Detection,” *International Research Journal of Engineering and Technology*, vol. 13, no. 1, pp. 1760–1765, 2026.