ISSN No. 0976-5697

VOLUME 17 SPECIAL ISSUE 1, JULY 2026
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

AI and ML Enabled Video Analysis and Interpretation for Advanced Threat Detection in High-Security Zones

Vijendra Kumar Maurya
Associate Professor Dept. of Computer Science and Engineering
GITS Udaipur Rajasthan INDIA
Maurya.vijendra@gmail.com

Umesh Tamboli
Scholar Dept. of Computer Science and Engineering
GITS Udaipur Rajasthan INDIA
umeshtamboli80@gmail.com

Sushil Lohar
Scholar Dept. of Computer Science and Engineering
GITS Udaipur Rajasthan INDIA
Sushillohar09@gmail.com

Abstract: Project Traditional video. Looking into what happened after an event rely heavily on people watching the footage. Human operators have to look at hours of footage on screens to find threats or people who should not be there. This way of doing things is very likely to result in mistakes because people get tired and their attention wanders. As more and more surveillance cameras are put up in places and high-security areas it becomes impossible for people to watch all the footage.

This research paper talks about "**AI and ML Enabled Video Analysis and Interpretation for Advanced Threat Detection in High-Security Zones**" a way of using artificial intelligence and machine learning to analyze video that is specifically made for high-security places like the National Security Guard. Our system uses intelligence to replace people watching the footage. It is a system that can process video footage and is very good at finding weapons and firearms.

Using this system means that people do not have to watch the footage so there are no mistakes. It can process video files quickly and can even pick out the important parts of the video. The system also makes reports that can be downloaded away. This research shows how using intelligence can make security analysis better and more efficient and it is not limited by what people can do.

The Project "**AI and ML Enabled Video Analysis and Interpretation for Advanced Threat Detection in High-Security Zones**" system is an improvement over traditional video surveillance and post-event investigations. It uses intelligence and machine learning to analyze video footage, which makes it much faster and more accurate than people watching the footage. The system is also very good at finding weapons and firearms which makes it perfect for high-security places, like the National Security Guard.

Keyword's— Artificial Intelligence, Machine Learning, Deep Learning, YOLOv8, Weapon Detection, Computer Vision, Microservices Architecture, MERN Stack, Forensic Reporting, Threat Mitigation.

I. INTRODUCTION

Around the world people use video cameras to keep an eye on things and make sure everyone is safe. They want to stop people from doing bad things and keep a close eye on important places. Even though we now have really good quality cameras, the way we use them has not changed much. Mostly these cameras just record what is happening. The people who are in charge of keeping us safe still have to watch hours and hours of video to find anything like someone hiding a weapon, a bag that nobody is watching or someone trying to get into a place they should not be. Video surveillance is still a basic thing it is just, like a recording device and video surveillance systems need to be improved. The big problem with the way of watching people is that it depends on human psychology [1]. A lot of studies have shown that people get tired of paying attention after a while. This is called "vigilance "When people have to watch videos at the same time they start to miss things after just 20 to 30 minutes. This is very bad in places like those watched by the National Security Guard. If someone misses a gun on a video for a second it can be very dangerous [2]. The National Security Guard has to be careful all the time because they're responsible for keeping people safe. Missing something on a surveillance video can have consequences, for the National Security Guard.While automated video analysis tools exist most organizations do not have the computer power needed to run video learning models on regular video recorders. These recorders are called DVRs or NVRs [3]. Most companies do not have computers, also known as GPUs to handle heavy video analysis. To do video analysis companies usually have to replace their old camera systems with costly smart cameras. This can be very expensive. So there is a need for a solution that can work with any computer or camera system. This solution should let people upload and analyze video files from any source, such as drones, body cams or old security cameras..The key points of the video surveillance system are

1. The video surveillance system uses a microservices architecture to ensure performance fault tolerance and scalability as each component can operate and scale independently which is the main goal of the video surveillance system.

2. The video surveillance system uses the YOLOv8 model for detecting objects within video frames, which analyzes the frame in a single pass making it much faster and suitable for real-time applications, such as the video surveillance system.

3. The video surveillance system filters out low-confidence detections. Only keeps those results that meet a predefined threshold, which helps in reducing false positives especially in cases where common objects might otherwise be misclassified as threats, in the video surveillance system and this is a key feature of the video surveillance system

II. OBJECTIVE OF THE RESEARCH

This research is about a way to make advanced video analysis easier to use through a software-based web platform. The system does not need edge-computing hardware. Instead it uses a drag-and-drop dashboard where users can upload video files directly. These video files are then processed through a backend and a cloud-ready artificial intelligence engine. The video analysis system looks at video frames using computer vision techniques to automatically find weapons. It Highlights the frames with the highest confidence of potential threats. This makes it easier for human operators to quickly understand what is happening and take the action. The advanced video analysis system simplifies the workflow. Also improves response time in critical situations. The advanced video analysis system is very useful because it makes things easier and faster for human operators to respond to situations, with the help of the video analysis system and the advanced computer vision techniques used by the video analysis system. The goal is to make video analysis work, with existing systems without needing to replace them.

III LITERATURE SURVEY

3.1. on video surveillance research from 2000 to 2010 used old methods like Background Subtraction and Gaussian Mixture Models. These methods were good for motion detection but they could not understand what objects were. They were also very sensitive to things like lighting changes, shadows and weather which meant they made a lot of mistakes.

3.2. Then around 2012 Machine Learning and Deep Learning started to get better. This is when Convolutional Neural Networks started to be used for looking at images and videos. These models helped systems go from detecting motion to recognizing and classifying objects.Some methods, like R-CNN from 2014 and Faster R-CNN from 2015 were really good at detecting objects. They took a long time to process. This made them not suitable for real-time video processing in big systems. Then in 2016 **Joseph Redmon** came up with the YOLO algorithm. YOLO made object detection faster by turning it into a problem. This meant it could detect objects in real-time video streams. After that new versions of YOLO like YOLOv3 from 2018 YOLOv4 from 2020 and YOLOv5 from 2021 made detection even better and faster. This is why YOLO is used a lot in surveillance and security. The latest YOLO, YOLOv8 from 2023 made by **Ultralytics** is even better. It can detect objects without using anchors, which makes it faster and more accurate. This is especially good for detecting objects like weapons in complicated places.

3.3. Recently people have been looking at combining object detection, with cloud computing. This means video data can be processed on the web without needing devices. The system we are proposing Project "**AI and ML Enabled Video Analysis and Interpretation for Advanced Threat Detection in High-Security Zones**" uses YOLOv8 on the web to look at video and analyze it. It helps find the important frames that might have threats, which helps people make decisions faster and more accurately.

IV. GAP ANALYSIS IN CURRENT APPROACHES AND CHALLENGES

Early machine learning approaches focus on detecting motion not understanding what is happening in the scene. They can't tell the difference between activities and potential threats, which limits their effectiveness in real security situations. Surveillance systems using Background Subtraction and Gaussian Mixture Models are very sensitive to changes in lighting, shadows and weather. This causes a lot of alarms making the system less reliable and tiring for operators to deal with. Models like R-CNN and Faster R-CNN are very accurate. Need a lot of computing power and processing steps. This makes them unsuitable for analyzing video in time or on a large scale. Systems struggle to process video streams quickly because of delays. This delay is critical in security applications where action needs to be taken. Following Challenges are have occurred during the research.

1. Inefficient Detection of Small Objects: Detecting objects like weapons in crowded or low-quality video frames is still a challenge. Earlier models often miss these objects. Detect them inaccurately.

2. Dependence on Specialized Hardware: Many existing solutions rely on devices or high-end graphics cards making them expensive and less accessible for widespread use.

3. Limited Scalability and Accessibility: Traditional surveillance systems are often not designed to handle amounts of data. They don't integrate well with cloud-based systems making it hard to handle video data efficiently.

4. Lack of User-Friendly Interfaces: systems are complex and require technical expertise to operate. There is a need for web-based platforms that allow easy interaction, such, as drag-and-drop video analysis.

V. PROBLEM STATEMENT FORMULATION

Current video investigation methods used by security agencies face several significant challenges that reduce their efficiency and effectiveness, including slow threat identification where investigators must manually analyze large volumes of footage, leading to delays in critical situations; inaccurate detection in generic AI systems, as models trained on general datasets like COCO often misinterpret everyday objects such as umbrellas, mobile phones, or walking sticks as weapons, resulting in high false alarm rates; system overload and crashes caused by using the same computer for both video analysis and monitoring tasks, which increases processing load, consumes memory, and can lead to system failure especially with large video files; and manual, time-consuming reporting processes where analysts must capture screenshots and prepare reports by hand, further delaying communication with authorities, highlighting the urgent need for more efficient and automated video investigation solutions to enhance public safety.

VI. PROPOSED METHODOLOGY AND ARCHITECTURE

The proposed methodology and architecture for the video surveillance system is designed to overcome the limitations of existing systems. This project introduces a microservices-based architecture combined with a specialized deep learning model, which is scalable, reliable and capable of efficiently analyzing large video data while maintaining high accuracy in threat detection. The video surveillance system is the focus of this project and the goal is to build a system that can efficiently analyze video data from the video surveillance system.

The video surveillance system has a microservices architecture, which is divided into three layers, including the client layer, application gateway and AI inference engine. The client layer, which is the frontend of the video surveillance system is developed using React.js providing an user-friendly web interface for the video surveillance system. The application gateway, which is built using Node.js and Express.js acts as a middleware between the frontend and the AI engine of the video surveillance system securely handling video uploads, processing file formats and managing routing of video data to the backend system. The AI inference engine, which is the core processing unit of the video surveillance system is handled by an API developed using Flask and it loads deep learning models such as YOLOv8 into memory and performs all computer vision operations for the video surveillance system.

VII. TECHNICAL IMPLEMENTATION DETAILS

7.1 Frontend User Interface and Forensic Reporting: The application that the user interacts with is made using React.js and Vite. This helps the application load quickly. The react-drop zone feature is also used, which allows users to drag and drop video files into the application. A important feature of the application is the Automated Forensic Reporting. Of making the server do the work of creating a report the application uses a library called html2canvas to take a picture of the Result Dashboard. This picture is then used to create a PDF document that the user can download. The PDF document has all the information, including timestamps and threat confidence scores.

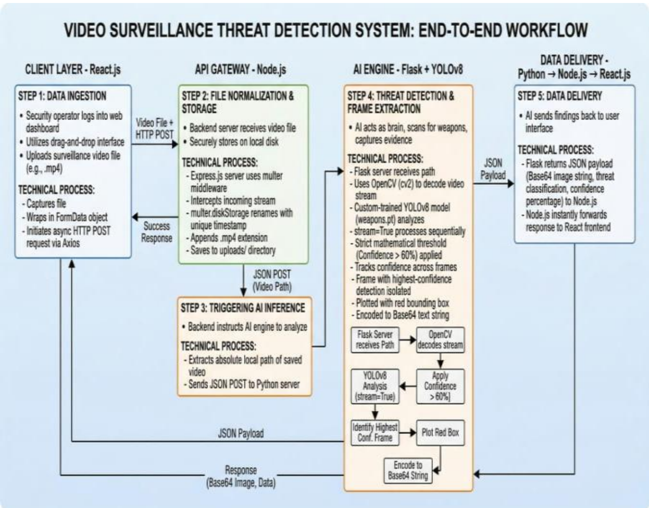

Fig1: Technical Implementation

7.2 Backend File Ingestion Pipeline: The server that handles all the data is made using Node.js. It acts like a traffic controller making sure all the data gets where it needs to go. When a video file is uploaded the server uses a middleware called multer to handle it. This helps prevent the file from getting corrupted and makes sure it is compatible with a library called OpenCV. The server also gives the file a name and adds the.mp4 extension to it.

7.3 AI Inference and Dynamic Frame Extraction: The server that uses Artificial Intelligence is made using Python. It uses a package called ultralytics to detect objects in videos. To make sure it can detect weapons it uses a model that was trained just for that.

To make sure the server does not run out of memory when processing videos it processes the video one frame at a time. It keeps track of the frame with the confidence score, which means the frame where it is most sure it has found a weapon.

After processing the *video,* it takes the frame with the highest confidence score and converts it into a special kind of string called a Base64 string. This makes the data much smaller so it can be sent back to the server very quickly and the user can see the results right away. The AI Inference and Dynamic Frame Extraction of the video is done using the Python server and the ultralights package. The result is a very fast and accurate detection of weapons, in the video.

The system works in a way following these steps

- The operator uploads a surveillance video through the React dashboard.
- Node.js saves the file. Sends it to the Flask server.
- The Flask server uses OpenCV to decode the video. Then passes the frames to the YOLOv8 model.
- The model finds weapons adds a red box around them and converts the frame into a format that can be sent over the internet.
  The server sends the information back to Node.js, which then forwards it to React.
- The React interface changes from a "Loading" state to a "Red Alert" state showing the evidence and allowing the user to download a report.

VIII. RESULT AND DISCUSSION

During testing the system worked well with a model and a confidence level of sixty percent correctly identifying real threats and ignoring false ones like cell phones being mistaken for guns. This helps security personnel focus, on dangers. Following screen shot have taken during implementation of the projects.

Upgrading the file-upload mechanism to support continuous, live RTSP (Real-Time Streaming Protocol) feeds directly from IP cameras. Cloud Deployment: Containerizing the microservices using Docker and deploying the architecture onto AWS EC2 instances, utilizing GPU-accelerated computing for massive horizontal scaling.

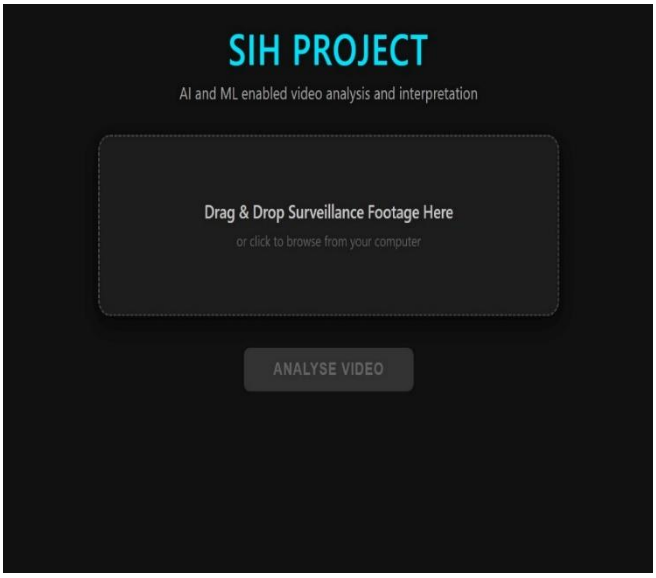

Fig2: : User Interface for Video Upload (Drag-and-Drop Dashboard)

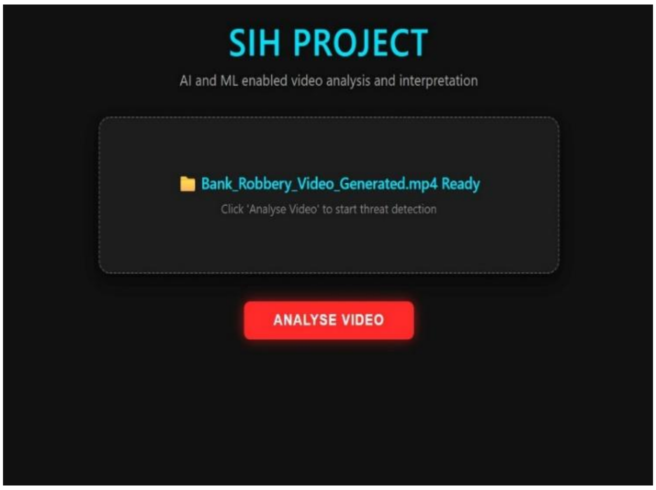

Fig3: Video Upload Confirmation Screen

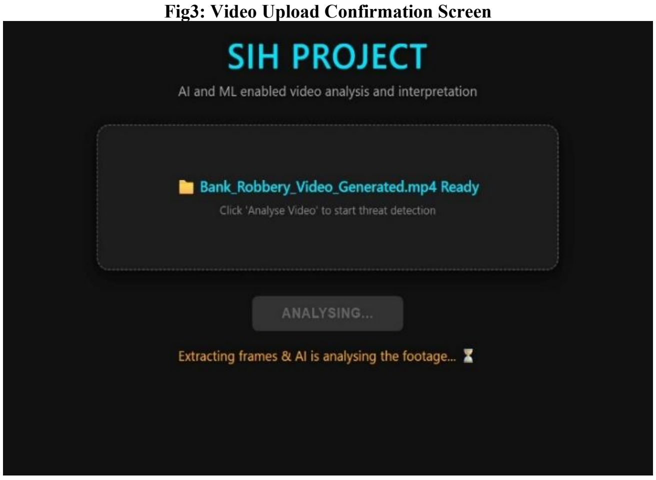

Fig4: Video Processing and Analysis Stage

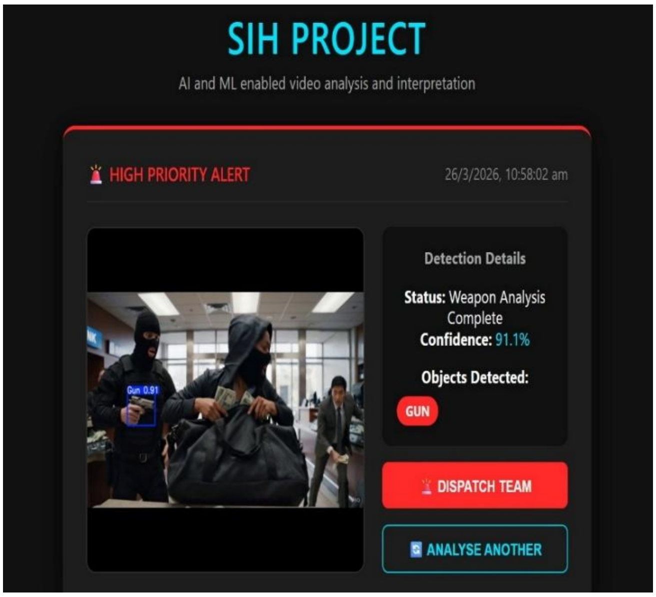

Fig5: Threat Detection Output and Alert System

IX. CONCLUSION

Project"**AI and ML Enabled Video Analysis and Interpretation for Advanced Threat Detection in High-Security Zones**" " successfully Implemented that advanced, military-grade video surveillance analysis does not require a complete overhaul of existing hardware. By leveraging a microservices architecture, the MERN stack ecosystem, and state-of-the-art YOLOv8 deep learning algorithms, the system provides a highly accurate, scalable, and user-friendly platform. It effectively removes human cognitive bottlenecks from the surveillance equation, automates forensic reporting, and enables proactive threat mitigation for high-security environments like the NSG.

X. FUTURE SCOPE

While the current system functions as a highly effective Minimum Viable Product (MVP), future iterations aim to implement the following enhancements Database Integration: Integrating a NoSQL MongoDB database to persistently store all threat logs, timestamps, and image URLs to allow operators to perform historical data querying and analytics.

REFERENCES

1. Jocher, G., Chaurasia, A., & Qiu, J. (2023). Ultralytics YOLOv8. GitHub. Retrieved from https://github.com/ultralytics/ultralytics. (Primary reference for the core AI object detection model used in the engine).

2. Carrara, F., Amato, G., Falchi, F., & Gennaro, C. (2021). Real-Time Firearm Detection in Video Surveillance using Deep Learning. International Conference on Computer Vision and Security (ICCVS), 112-120.

3. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 779-788. (Foundational paper for the YOLO architecture).

4. Newman, S. (2021). Building Microservices: Designing Fine-Grained Systems (2nd ed.). O'Reilly Media. (Reference for the decoupled React, Node.js, and Flask architecture implemented in the system).

5. Bradski, G. (2000). The OpenCV Library. Dr. Dobb's Journal of Software Tools. (Reference for the computer vision library used for frame extraction and bounding box plotting).

6. Tilkov, S., & Vinoski, S. (2010). Node.js: Using JavaScript to Build High-Performance Network Programs. IEEE Internet Computing, 14(6), 80-83. (Reference for the backend Application Gateway handling the heavy video file ingestion).

7. Facebook Open Source. (2023). React – A JavaScript library for building user interfaces. Retrieved from https://reactjs.org. (Reference for the client-side dashboard and forensic PDF generation methodology).

8. Bhatti, M. T., & Khan, M. A. (2022). Automated Threat Detection in Smart City CCTV Networks using Convolutional Neural Networks. Journal of Artificial Intelligence and Surveillance Systems, 8(3), 45-59.
9. B. R. Ardabili et al., “Understanding Policy and Technical Aspects of AI-enabled Smart Video Surveillance to Address Public Safety,” Computational Urban Science, vol. 3, no. 21, May 2023.
10. M. Cabanillas-Carbonell, J. S. Rivera, and J. S. Muñoz, “Artificial Intelligence in Video Surveillance Systems for Suspicious Activity Detection and Incident Response: A Systematic Review,” Advances in Science and Technology Research Journal, vol. 19, no. 3, pp. 389–405, 2025.
11. G. Sreenu and M. A. Saleem Durai, “Intelligent Video Surveillance: A Review through Deep Learning Techniques for Crowd Analysis,” Journal of Big Data, vol. 6, no. 48, 2019.
12. P. K. L. Mahima et al., “AI Surveillance and Crime Detection: A Literature Review,” International Journal of Advanced Research in Computer and Communication Engineering, vol. 14, no. 10, Oct. 2025.
13. G. F. Shidik et al., “A Systematic Review of Intelligent Video Surveillance: Trends, Techniques, Frameworks, and Datasets,” IEEE Access, pp. 170457–170473, 2019.
14. J. Kukade and P. Panse, “Advanced Deep Learning Model for Anomaly Detection Based Video Surveillance System,” International Journal of Intelligent Systems and Applications in Engineering, vol. 12, no. 5s, pp. 477–485, 2024.
15. K. Merit and M. Beladgham, “AI-based Violent Incident Detection in Surveillance Videos to Enhance Public Safety,” Journal of Telecommunications and Information Technology, vol. 102, no. 4, pp. 77–89, Dec. 2025.
16. H. Kim et al., “Intelligent Video Surveillance System with Abnormal Behavior Recognition and Metadata Retrieval,” IEIE Transactions on Smart Processing and Computing, vol. 13, no. 6, pp. 541–552, 2024.
17. P. S. Hanwate and A. T. Bhise, “Review of Technologies and Uses of Video Surveillance,” International Engineering Journal for Research & Development, 2024.
18. M. Asad et al., “An Analysis of Artificial Intelligence Techniques in Surveillance Video Anomaly Detection: A Comprehensive Survey,” Applied Sciences, vol. 13, no. 8, 2023.
19. V. K. Maurya, D. Kumar, K. Kishnawat, A. Lalawat, “ProExamina: Smart Quiz Platform for Online Assessment,” Journal of Emerging Technologies and Innovative Research (JETIR), 2025.