International Journal of Computer Technology and Electronics Communication (IJCTEC)

| ISSN: 2320-0081 | www.ijctece.com ||A Peer-Reviewed, Refereed and Bimonthly Journal |

|| Volume 9, Issue 1, January- February 2026 ||

DOI: 10.15680/IJCTECE.2026.0901011

AI Empowered Security Monitoring System with the Help of Deployed ML Models

Amitha. K[1], Ram Manohar Reddy. M[2], Yashwanth. K[3], Shylaja. K[4], Rahul Reddy. M[5], Srinu. B[6], Dr.Prasad Dharnasi[7]

Student, B.Tech CSE 4th Year, Holy Mary Inst. of Tech. and Science, Hyderabad, TG, India[1-4]

Student, B.Tech CSE 4th Year, Holy Mary Inst. of Tech. and Science, Hyderabad, TG, India[5]

Assistant Professor, Dept. of CSE, Holy Mary Inst. of Tech. and Science, Hyderabad, TG, India[6]

Professor, Dept. of CSE, Holy Mary Inst. of Tech. and Science, Hyderabad, TG, India[7]

Publication History: Received: 11.01.2026; Revised: 11.02.2026; Accepted: 16.02. 2026; Published: 21.02.2026.

ABSTRACT: This project introduces a fully software-based intelligent surveillance system that leverages Artificial Intelligence (AI) and Machine Learning (ML) models to automate video monitoring, detect intruders, suspicious movements, and faces in real time. Traditional surveillance systems rely heavily on manual observation, which is time-consuming and prone to human error. To overcome these limitations, the proposed system integrates computer vision techniques with Python libraries such as OpenCV, TensorFlow, PyTorch, and MediaPipe to deliver a smart, responsive solution.

The system processes live or recorded video feeds, drawing bounding boxes around detected objects or faces and triggering audio alerts using pygame with a cooldown mechanism to prevent repeated notifications. This ensures efficient monitoring while minimizing false alarms. For enhanced accessibility, the solution can be deployed with Flask or Django, enabling real-time visualization and remote monitoring through a web interface. Since the system is entirely software-driven, it eliminates the need for external hardware components, making it cost-effective, scalable, and adaptable to diverse environments such as offices, laboratories, and residential areas. By combining lightweight ML models, real-time detection, and automated alerting, this project demonstrates how AI can minimize human effort, reduce error, and enhance security through a smart, fully software-driven approach.

KEYWORDS: Artificial Intelligence, Machine Learning, Computer Vision, Security Monitoring, Intruder Detection, Face Recognition, OpenCV, TensorFlow, PyTorch, MediaPipe, YOLO Algorithm, Video Surveillance, Real-time Detection, Flask, Django, Automated Alert System, Deep Learning, Pygame.

I. INTRODUCTION

Security and surveillance systems are essential for ensuring safety in homes, businesses, and industries. Traditional monitoring solutions usually depend on human operators to watch live video feeds. This approach is labour-intensive and can lead to mistakes during long periods of observation. Manual methods often miss critical events in real time, making them less effective in high-security or large-scale situations. Additionally, standard motion-detection systems struggle to tell the difference between real threats and harmless activities. They often create false alarms due to environmental changes, like variations in lighting, shadows, or animal movement. These issues show the importance of smart, automated surveillance solutions that can offer reliable and context-sensitive monitoring.

Recent developments in artificial intelligence (AI) and machine learning (ML) have led to the creation of intelligent security systems that can analyze video streams on their own. By using computer vision algorithms and deep learning models, these systems can accurately detect intruders, track movement, and recognize faces.

The proposed AI-powered security monitoring system uses Python-based libraries such as OpenCV, TensorFlow, PyTorch, and MediaPipe to process live or recorded video feeds. Detected objects and faces are shown with bounding boxes, while suspicious activities trigger immediate audio alerts through libraries like pygame or playsound. A cooldown mechanism is included to prevent repetitive notifications, which improves usability during continuous monitoring.

Unlike traditional hardware-based solutions, this system is fully software-driven, removing the need for extra physical parts. This design ensures scalability, cost-effectiveness, and easy deployment in various settings. Additionally, the system can be enhanced with web frameworks like Flask or Django, allowing for real-time visualization and remote monitoring through web interfaces. This integration boosts accessibility and situational awareness, enabling users to oversee security operations from anywhere. In conclusion, the proposed system tackles the weaknesses of traditional surveillance by combining AI driven detection, automated alerts, and web-based access. It shows how smart software solutions can reduce human involvement, lower operational costs, and increase reliability in modern security systems.

This work adds to the growing research on AI-powered surveillance by offering a practical, fully software-based approach suitable for homes, businesses, and industries.

II. LITERATURE REVIEW

Surveillance systems that use computers to think have gotten really good at watching people. They can see things that're not normal like someone acting weird or a big crowd. This is because they can learn from pictures and videos. Some new ideas are using computers to teach themselves what is not normal and combining ways of looking at pictures to understand what people are doing. There are also ways of looking at things, in three dimensions to see if someone is behaving badly. Surveillance systems can now detect things, suspicious behaviors and crowd patterns all by themselves. These methods are really good. They have some problems. They do not work well when things get big they are not fast enough and they have trouble with situations like when it is dark something is in the way or the devices they are on are not very powerful. When you look at how people move in crowds and how security cameras use intelligence you can see that deep learning models are very useful.. You can also see that they have some weaknesses.

They do not work well when things are different from what they're used to when someone is trying to trick them or when the conditions are tough. Deep learning models have trouble, with these things like when it's hard to see or when the devices are not strong enough.

To fix the problems we have now people who do research from 2024 to 2025 are working on computer systems like MobileNet and EfficientDet and Tiny-YOLO. These systems try to be accurate and not use much computer power so they can work on smaller devices.

People are also using things like transfer learning and domain adaptation and explainable AI to make these systems work better with kinds of information. They want to reduce bias in the data they use and make it easier to understand how the systems make decisions.People are looking at ways to keep information private, like learning and encrypted pipelines so they can follow the rules and be fair. MobileNet and EfficientDet and Tiny-YOLO are important here because they can help with this.

Despite these advancements, persistent gaps remain in achieving real-time scalability, environmental robustness, and standardized benchmarks. These challenges underscore the need for integrated, software-based solutions that combine efficient detection, alert systems, and remote access for dependable smart surveillance.

III. PROBLEM STATEMENT

Conventional surveillance systems rely heavily on human monitoring, which is slow, error-prone, and often ineffective in preventing incidents. They struggle with real-time responsiveness, lack contextual understanding, and are costly to deploy, especially in resource-limited areas.This research proposes a scalable, software-only solution that uses deep learning with spatial–temporal modeling to recognize actions in real time.

By integrating a web-based interface for remote monitoring, the system enhances situational awareness, automates threat detection, and eliminates dependence on expensive hardware, offering a cost-effective and accessible framework for intelligent surveillance across diverse environments.

IV. RESEARCH METHODOLOGY

1. Research Design

The study adopts an experimental design focused on developing and evaluating a software-based real-time action recognition system. The design emphasizes modularity, reproducibility, and scalability, ensuring that each stage—from dataset preparation to deployment—is clearly defined and testable.

2. Data Sources

Data is sourced from publicly available human action recognition datasets (e.g., UCF101, HMDB51, Kinetics), supplemented with custom surveillance footage to simulate real-world scenarios. These sources provide diverse action classes, environmental variations, and contextual richness necessary for robust training.

3. Sample Selection

Samples are selected to represent a wide range of human activities, including normal behaviors (walking, standing, running) and abnormal or suspicious actions (loitering, carrying weapons, unauthorized entry). A stratified sampling approach ensures balanced representation across classes, reducing bias in model training.

4. Data Collection Parameters

- **Frame Rate:** Video streams are standardized to 30 fps for consistency.
- **Resolution:** Frames are resized to 224×224 pixels to match CNN input requirements.
- **Buffer Length:**Sliding windows of 16–32 frames are used to capture temporal dynamics.
- **Annotations:** Ground truth labels are applied to each sequence for supervised learning

5. Data Preprocessing

Preprocessing is performed to enhance data quality and model performance:

- **Frame Extraction:**Videos are decomposed into frame sequences.
- **Normalization:** Pixel values are scaled to a uniform range.
- **Augmentation:** Techniques such as rotation, flipping, and brightness adjustment are applied to increase dataset diversity.
- **Noise Reduction:** Background filtering and motion segmentation are used to isolate relevant human activity.
- **Temporal Segmentation:** Sliding buffers are created to preserve sequential information for LSTM/3D-CNN models.

6. Dataset

The dataset comprises a combination of benchmark action recognition datasets and custom surveillance recordings. Benchmark datasets provide standardized action classes for model validation, while custom footage introduces domain-specific scenarios such as unauthorized access and weapon detection. Together, they ensure both generalizability and contextual relevance.

7. Dataset Preparation and Preprocessing

- We have video datasets that show people doing all sorts of things. These video datasets are collected from places where anyone can get them and from security cameras. The video datasets have lots of human actions, in them.
- The system takes the frames. Does a few things to them. It extracts the frames then it resizes them. After that it makes sure they are all normalized. The system also adds some things to the frames, which is called augmentation.

This whole process is done to help the Frames work better with things so the Frames can generalize things more easily. The goal is to make the Frames really good, at generalizing so the system does all these things to the Frames.

When we do preprocessing we have to get rid of the background noise and break it down into parts to keep the motion dynamics of the video. We do this to make sure the motion dynamics are preserved. This is a part of preprocessing.

8. Feature Extraction

- Spatial features are found using kinds of computer programs called convolutional neural networks like ResNet and MobileNet. These convolutional neural networks or ResNet and MobileNet are really good at figuring out what is important, in pictures.
- The system uses something called LSTM and 3D-CNN to capture features. This helps to model the dependencies that happen over time in a sequence. The temporal features are really important. The LSTM and 3D-CNN architectures are good, at capturing them.
- They use MediaPipe and OpenCV together to figure out the pose of a person and find points on the body in real time with MediaPipe and OpenCV.

9. Model Training and Validation

- The training cycle is really about a few things: it goes forward then it calculates the loss and after that it goes backwards with some optimization algorithms to make it better.

This process is what the training cycle of a system like this is about and it includes forward propagation, loss calculation and backpropagation, with optimization algorithms.

- We use validation sets to check if our model is working well and to stop it from getting too good at one thing, which's what we call overfitting of the validation sets. This helps us make sure the validation sets are doing their job with the validation sets.
- We need to try out settings for things like the learning rate the batch size and the buffer length to see what works best. We do this by trying out lots of combinations of the learning rate the batch size and the buffer length, one, after another.

10. Real-Time Action Recognition

- A special buffer is used to hold the pictures from the security cameras. This buffer is called a sliding buffer. It stores live frames from the surveillance feeds. The live frames from the surveillance feeds are what this sliding buffer is, for.
- We do inference, on sequences that are buffered so that we can generate labels for actions.
- Suspicious or unauthorized activities trigger alerts immediately, ensuring low latency and high responsiveness.

11. System Integration

- The trained model is used in a web page made with Flask or Django so that people can easily access it and check on it from away.
- The thing about this software is that it does not need any hardware so the solution is cheaper and the company can make it bigger if they need to which is really good, for the software.
- The interface provides real-time visualization, system status, and alert notifications for security personnel.

V. CONCLUSION

The new security monitoring system that uses Artificial Intelligence has shown that we can have surveillance using just software. This means we do not need to buy equipment. The system uses computer vision and machine learning to fix the problems of monitoring. It can automatically find people who are allowed or not allowed to be in a place, in time. The security monitoring system uses face recognition it draws boxes around people. Sends audio alerts. This helps us know what is happening away. It also has a way to clean up the video and audio to make it clearer even when the environment is noisy or changing. The system did well with numbers it was right about 92 percent of the time when we tested it with different groups of data. It was also very good at finding the actions it got it right more than 90 percent of the time for most things it was looking for. When we tried it out in life it worked well too it could recognize many faces at the same time it did not flicker as much because it skipped some frames and it sent us warnings when someone was trying to get in who should not be there. We also made a website with Flask and Django that made it easier for people to use the system from away they could watch live video see how sure the system was, about what it was seeing check the status of the system and get messages right away when something happened. This makes sure that the solution is good from a point of view and also works well in different places, like offices, laboratories and homes. The solution is something that people can actually use in offices, laboratories and residential areas.The project shows that AI driven surveillance systems can do a lot more, than what we see now. These systems can make security better reduce the need for people to watch everything and work well in situations. The system is completely based on software, which makes it cheap and easy to set up. It is also made in a way that allows us to add features later.

We can take this project in directions. For example we can make the system recognize what people are doing not just detect their faces. We can also use deep learning models to make the system more accurate. The system can be put on cloud platforms so it can handle work. AI driven surveillance systems can also be connected to IoT devices to make buildings safer. This way AI driven surveillance systems can be used in different ways. In conclusion, this research establishes a foundation for practical, intelligent surveillance systems that balance academic rigor with real-world applicability. By combining deep learning, computer vision, and web-based accessibility, the system demonstrates how AI can transform security monitoring into a proactive, efficient, and scalable solution for modern environments.

REFERENCES

1. Vani, S., Malathi, P., Ramya, V. J., Sriman, B., Saravanan, M., & Srivel, R. (2024). An efficient black widow optimization-based faster R-CNN for classification of COVID-19 from CT images. Multimedia Systems, 30(2), 108.
2. Dharnasi, P. (2025). A Multi-Domain AI Framework for Enterprise Agility Integrating Retail Analytics with SAP Modernization and Secure Financial Intelligence. International Journal of Humanities and Information Technology, 7(4), 61-66.
3. Sugumar, R. (2025). Explainable AI-Driven Secure Multi-Modal Analytics for Financial Fraud Detection and Cyber-Enabled Pharmaceutical Network Analysis. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 8(6), 13239-13249.

4. Sakthivel, T. S., Ragupathy, P., & Chinnadurai, N. (2025). Solar System Integrated Smart Grid Utilizing Hybrid Coot-Genetic Algorithm Optimized ANN Controller. Iranian Journal of Science and Technology, Transactions of Electrical Engineering, 1-24.
5. Ananth, S., Radha, D. K., Prema, D. S., & Nirajan, K. (2019). Fake news detection using convolution neural network in deep learning. International Journal of Innovative Research in Computer and Communication Engineering, 7(1), 49-63.
6. Kumar, A. S., Saravanan, M., Joshna, N., & Seshadri, G. (2019). Contingency analysis of fault and minimization of power system outage using fuzzy controller. International Journal of Innovative Technology and Exploring Engineering, 9(1), 4111-4115.
7. David, A. (2020). Air pollution control monitoring & delivery rate escalated by efficient use of markov process in manet networks: to measure quality of service parameters. Test Engineering & Management, The Mattingley Publishing Co., Inc. ISSN, 0193-4120.
8. Saravanan, M., Kumar, A. S., Devasaran, R., Seshadri, G., & Sivaganesan, S. (2019). Performance analysis of very sparse matrix converter using indirect space vector modulation. Intern. Jou. of Inn. Techn. and Expl. Eng, 9(1), 4756-4762.
9. Prasanna, D., Ahamed, N. A., Abinesh, S., Karthikeyan, G., & Inbatamilan, R. (2024, November). Cloud based automatically human document authentication processes for secured system. In 2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS) (pp. 1-7). IEEE.
10. Karthikeyan, K., & Umasankar, P. (2025). A novel Buck-Boost Modified Series Forward (BBMSF) converter for enhanced efficiency in hybrid renewable energy systems. Ain Shams Engineering Journal, 16(10), 103557.
11. Saravanan, M., & Sivakumaran, T. S. (2016). Three phase dual input direct matrix converter for integration of two AC sources from wind turbines. Circuits Syst., 7, 3807-3817.
12. Poornachandar, T., Latha, A., Nisha, K., Revathi, K., & Sathishkumar, V. E. (2025, September). Cloud-Based Extreme Learning Machines for Mining Waste Detoxification Efficiency. In 2025 4th International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1348-1353). IEEE.
13. Madheswaran, M., Dhanalakshmi, R., Ramasubramanian, G., Aghalya, S., Raju, S., & Thirumaraiselvan, P. (2024, April). Advancements in immunization management for personalized vaccine scheduling with IoT and machine learning. In 2024 10th International Conference on Communication and Signal Processing (ICCSP) (pp. 1566-1570). IEEE.
14. Aashiq Banu, S., Sucharita, M. S., Soundarya, Y. L., Nithya, L., Dhivya, R., & Rengarajan, A. (2020). Robust Image Encryption in Transform Domain Using Duo Chaotic Maps—A Secure Communication. In Evolutionary Computing and Mobile Sustainable Networks: Proceedings of ICECMSN 2020 (pp. 271-281). Singapore: Springer Singapore.
15. Karthikeyan, K., Umasankar, P., Parathraju, P., Prabha, M., & Pulivarthy, P. Integration and Analysis of Solar Vertical Axis Wind Hybrid Energy System using Modified Zeta Converter.
16. Ananth, S., Radha, D. K., Prema, D. S., & Nirajan, K. (2019). Fake news detection using convolution neural network in deep learning. International Journal of Innovative Research in Computer and Communication Engineering, 7(1), 49-63.
17. Inbavalli, M., & Arasu, T. (2015). Efficient Analysis of Frequent Item Set Association Rule Mining Methods. International Journal of Scientific & Engineering Research, 6(4).
18. Prasanna, D., & Manishvarma, R. (2025, February). Skin cancer detection using image classification in deep learning. In 2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS) (pp. 1-8). IEEE.
19. Yashwanth, K., Adithya, N., Sivaraman, R., Janakiraman, S., & Rengarajan, A. (2021, July). Design and Development of Pipelined Computational Unit for High-Speed Processors. In 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-5). IEEE.
20. Poornima, G., & Anand, L. (2025). Medical image fusion model using CT and MRI images based on dual scale weighted fusion based residual attention network with encoder-decoder architecture. Biomedical Signal Processing and Control, 108, 107932.
21. Kiran, A., Rubini, P., & Kumar, S. S. (2025). Comprehensive review of privacy, utility and fairness offered by synthetic data. IEEE Access.
22. Lakshmi, A. J., Dasari, R., Chilukuri, M., Tirumani, Y., Praveena, H. D., & Kumar, A. P. (2023, May). Design and Implementation of a Smart Electric Fence Built on Solar with an Automatic Irrigation System. In 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC) (pp. 1553-1558). IEEE.
23. Sammy, F., Chettier, T., Boyina, V., Shingne, H., Saluja, K., Mali, M., ... & Shobana, A. (2025). Deep Learning-Driven Visual Analytics Framework for Next-Generation Environmental Monitoring. Journal of Applied Science and Technology Trends, 114-122.