IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

Camera Based Smart Vision Wireless Detection System

Sujal Sripathi[1], Maaz Orawala[2], Tanmay Kulkarni[3] , Raj Patil[4]
Department of Automation and Robotics Engineering[1-4]
Shivajirao S. Jondhle College of Engineering, Shahapur, India

Abstract: *In the modern era of automation and surveillance, ensuring intelligent and real-time monitoring has become a crucial requirement for both public and private infrastructures. Traditional CCTV systems rely heavily on manual observation and are limited by human accuracy and physical connectivity constraints. To overcome these limitations, the proposed system introduces a Camera-Based Smart Vision Wireless Detection System designed to provide wireless connectivity, intelligent video analytics, and adaptive real-time detection capabilities.*

The system operates by establishing a wireless connection between an IP or CCTV camera and the processing unit through the RTSP (Real-Time Streaming Protocol) over a common Wi-Fi network. This enables users to access and monitor live feeds remotely without any physical cabling. The core intelligence of the system lies in its AI-based analysis module, which employs advanced computer vision frameworks such as OpenCV, TensorFlow, and DeepFace for detecting human faces, body movements, and behavioral patterns. The integrated machine learning models continuously improve detection accuracy through adaptive learning mechanisms.

This project aims to create a reliable, scalable, and intelligent surveillance framework that not only monitors visual activities wirelessly but also enhances security automation through artificial intelligence, reducing human intervention and latency in critical monitoring operations.

Keywords: Real-time Behavioral Analysis, Wireless Surveillance, Artificial Intelligence Analytics, Computer Vision

I. INTRODUCTION

With the rapid advancement of artificial intelligence and the increasing demand for automated security, modern surveillance systems are transitioning from passive monitoring to intelligent vision-based detection frameworks. Traditional CCTV setups are heavily dependent on manual supervision, which not only limits real-time responsiveness but also introduces significant chances of human error. To overcome these challenges, the integration of smart vision technology with wireless communication systems has emerged as a promising solution for next-generation security applications.

The proposed Camera-Based Smart Vision Wireless Detection System provides an intelligent surveillance mechanism capable of wireless video streaming and AI-driven detection. By utilizing the Real-Time Streaming Protocol (RTSP) over a common Wi-Fi network, the system enables seamless live video access without the need for physical cabling. The use of computer vision and machine learning algorithms enhances the system's ability to identify human presence, facial expressions, and body movements with high precision. This eliminates the need for continuous manual observation and improves situational awareness in real time.

In addition, the system employs advanced frameworks such as OpenCV, TensorFlow, and DeepFace for processing visual data and classifying human behavior patterns. The integration of AI with wireless communication ensures low-latency performance, making the system suitable for applications in residential areas, industrial sites, educational institutions, and smart cities. Furthermore, the adaptive learning capability of the AI models allows the system to evolve continuously, improving detection accuracy with every new data input.

DOI: 10.48175/IJARSCT-34088

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

In this project, various strategies are proposed to enhance detection reliability, wireless transmission stability, and automated alert systems. The overall aim is to build a unified vision-based framework that supports intelligent monitoring, reduces human dependency, and lays the foundation for scalable, AI-powered smart surveillance systems of the future.

II. LITERATURE VIEW

With the increasing demand for intelligent monitoring and security automation, camera-based smart vision systems have become a major area of research in computer vision and AI. These systems are designed to detect, recognize, and interpret visual information in real time using advanced algorithms. Researchers have focused on improving the efficiency of object tracking, face recognition, and behavior analysis by leveraging deep learning techniques and neural network architectures. The integration of wireless streaming protocols such as RTSP has further enabled remote access and monitoring over distributed networks.

In the paper "Real-Time Human Detection and Tracking Using Deep Learning," authors implemented convolutional neural network (CNN)-based models such as YOLO (You Only Look Once) and MobileNet for accurate motion tracking. The study demonstrated that deep learning models outperform traditional background subtraction techniques, providing higher accuracy and stability even in low-light conditions. This research established the foundation for integrating AI into real-time surveillance systems.

Another study titled "Facial Recognition Using DeepFace and OpenCV" explored the use of Python-based DeepFace frameworks for real-time identity verification. The authors analyzed the performance of different pre-trained facial embeddings such as VGG-Face, Facenet, and ArcFace, concluding that combining multiple embeddings improves recognition accuracy under varying angles and lighting conditions. This directly supports the AI recognition module in the proposed system.

Recent advancements in "Smart Surveillance through Wireless Streaming and Edge Computing" addressed the challenge of latency in real-time video processing. By shifting data analysis from cloud servers to local edge devices, the researchers successfully reduced communication delays, allowing faster decision-making. The combination of RTSP-based wireless transmission and edge AI architecture provided a more scalable and secure environment for real-time surveillance operations.

In another research, "Behavior and Anomaly Detection Using AI in Video Surveillance," deep neural networks were trained to detect unusual human activities in crowded scenes. The authors proposed a hybrid model combining Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to analyze motion patterns over time. The results showed significant improvements in identifying abnormal events compared to conventional methods.

Collectively, these studies highlight the evolution of AI-based surveillance systems from passive video monitoring to proactive, intelligent detection mechanisms. They demonstrate that combining RTSP-based wireless streaming with deep learning algorithms such as DeepFace and TensorFlow can enable efficient, real-time, and adaptive smart vision systems suitable for modern security infrastructures.

III. METHODOLOY

The development of the Camera-Based Smart Vision Wireless Detection System follows a modular approach, integrating wireless networking, real-time computer vision, and deep learning. The process is divided into the following five logical phases:

Phase 1: Hardware Integration & Wireless Connectivity

The initial phase focuses on establishing a robust communication link between the sensors and the processing hub.

Network Synchronization: The IP/CCTV camera and the processing unit (Raspberry Pi or PC) are connected to a unified Wi-Fi network.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 14, April 2026 Impact Factor: 8.2

Protocol Configuration: The Real-Time Streaming Protocol (RTSP) is configured to enable high-speed, low-latency video transmission without the need for physical cabling.

Phase 2: Video Stream Acquisition & Pre-processing

Once the connection is established, the raw data must be prepared for AI analysis.

Stream Decoding: Using OpenCV, the system captures live frames from the RTSP URL.

Frame Optimization: Frames are resized and converted (e.g., Grayscale or RGB normalization) to reduce computational load and enhance the speed of the AI inference engine.

Phase 3: Multi-Layered AI Inference

This core phase involves three distinct layers of analysis to transform visual pixels into actionable data:

Layer 1 (Identity): The DeepFace library identifies human faces by comparing extracted embeddings against a pre-authorized database.

Layer 2 (Motion): Background subtraction and contour detection are applied via OpenCV to isolate moving objects from static environments.

Layer 3 (Behavior): TensorFlow/Keras models classify the detected motion into specific activities (e.g., walking, running, or loitering) to identify potential security threats.

Phase 4: Decision Logic & Alert Generation

The system evaluates the AI outputs against predefined security protocols.

Threshold Validation: To prevent false alarms, the system verifies detections across multiple consecutive frames.

Notification Trigger: If an anomaly (e.g., an unauthorized face or suspicious loitering) is confirmed, the system triggers real-time alerts via the Flask web dashboard or local sound alarms.

Phase 5: Data Management & Visualization

The final phase ensures that all events are documented and accessible to the end-user.

Event-Driven Logging: Instead of continuous recording, the system saves only specific frames or video clips associated with a "Detection Event" to optimize storage.

User Interface: A Flask-based dashboard provides a live visual feed, historical logs, and system health status for remote monitoring.

IV. SYSTEM FLOW ARCHITECTURE

The operational flow of the system begins with the high-definition capture of visual data by an IP-enabled camera, which encodes the video and transmits it wirelessly via the **Real-Time Streaming Protocol (RTSP)** over a local Wi-Fi network. Once the stream reaches the processing unit, **OpenCV** decodes the packets into individual frames that undergo pre-processing, including resizing for computational efficiency and Gaussian blurring to eliminate pixel noise. These optimized frames are then simultaneously processed through three specialized AI pipelines: **DeepFace** handles identity verification by matching facial embeddings against a local database, **OpenCV** performs frame differencing to isolate motion contours, and a **TensorFlow** model analyzes motion sequences to classify specific behaviors like walking or loitering.

The results from these pipelines are fed into a central logic gate that evaluates potential threats—such as an unrecognized individual exhibiting suspicious movement and verifies the detection across three consecutive frames to prevent false positives. Once a threat is confirmed, the system executes an event-driven storage protocol that saves the specific frame and logs the incident details, while simultaneously triggering a sound alarm or digital notification. This entire process culminates in a **Flask-based web dashboard**, which provides the user with a live, annotated video feed and a historical record of all detected anomalies for real-time monitoring and response.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

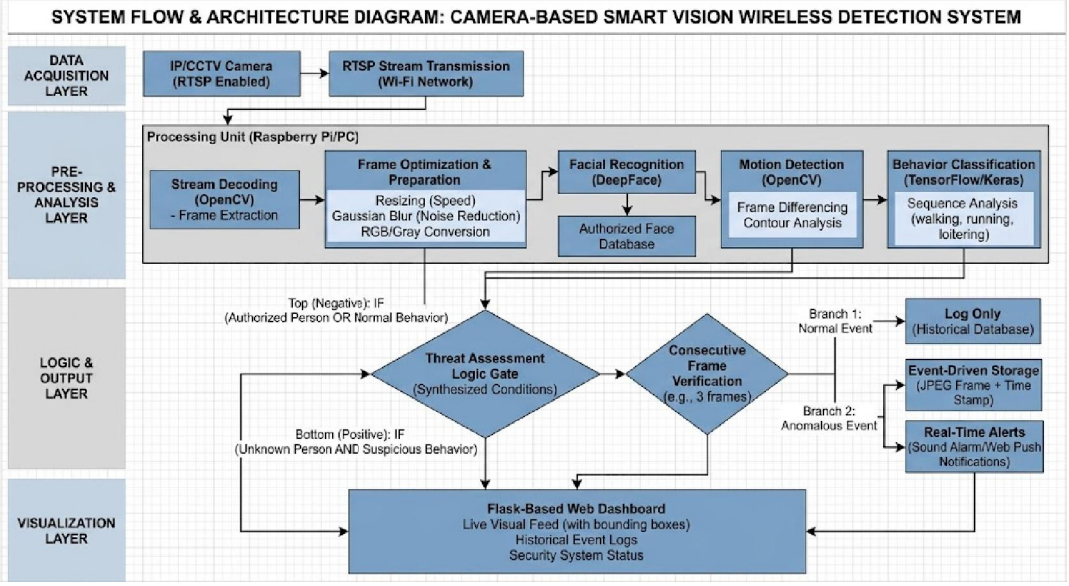

Figure 4.1

V. COMPONENTS

The architecture of the **Camera-Based Smart Vision Wireless Detection System** functions as a highly integrated ecosystem where hardware acts as the physical body and software provides the cognitive intelligence. The process begins at the **Data Acquisition Layer**, where the **IP-enabled Camera** serves as the primary sensory organ. This hardware component captures light through its lens and converts it into digital packets. Instead of using restrictive physical cables, the system employs the **Real-Time Streaming Protocol (RTSP)** to transmit this data over a **Wireless Wi-Fi Network**. This coordination between the camera's internal encoding hardware and the network's wireless transmission protocol ensures that high-definition video is available to the processing hub with minimal latency, regardless of where the camera is physically mounted.

Once the wireless signal reaches the **Processing Unit**, which is typically a **Raspberry Pi** or a high-performance **PC**, the hardware-software synergy becomes even more critical. Here, the **OpenCV** library acts as the first line of software interaction, "listening" to the incoming RTSP stream and decoding the digital packets back into individual image frames. The hardware's **CPU and RAM** provide the raw power necessary for OpenCV to perform pre-processing tasks, such as resizing the image for speed or applying a Gaussian blur to filter out electronic noise. This preparation is essential because it transforms the raw hardware input into a clean digital format that the more complex AI models can digest without stuttering or crashing the system.

The core intelligence of the system resides in the simultaneous operation of the **DeepFace** and **TensorFlow** frameworks running within a **Python** environment. These software modules are the "cognitive centers" that analyze the pixels provided by the hardware. While the hardware's processor executes billions of calculations per second, the software uses those calculations to extract facial embeddings and motion patterns. DeepFace compares these patterns against an **Authorized Identity Database** stored on the hardware's local disk, while TensorFlow runs sequence analysis to determine if a movement is a standard walk or a suspicious loiter. This is a perfect example of coordination: the hardware provides the computational "muscle," while the software provides the "judgment" required to distinguish a friend from a potential intruder.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

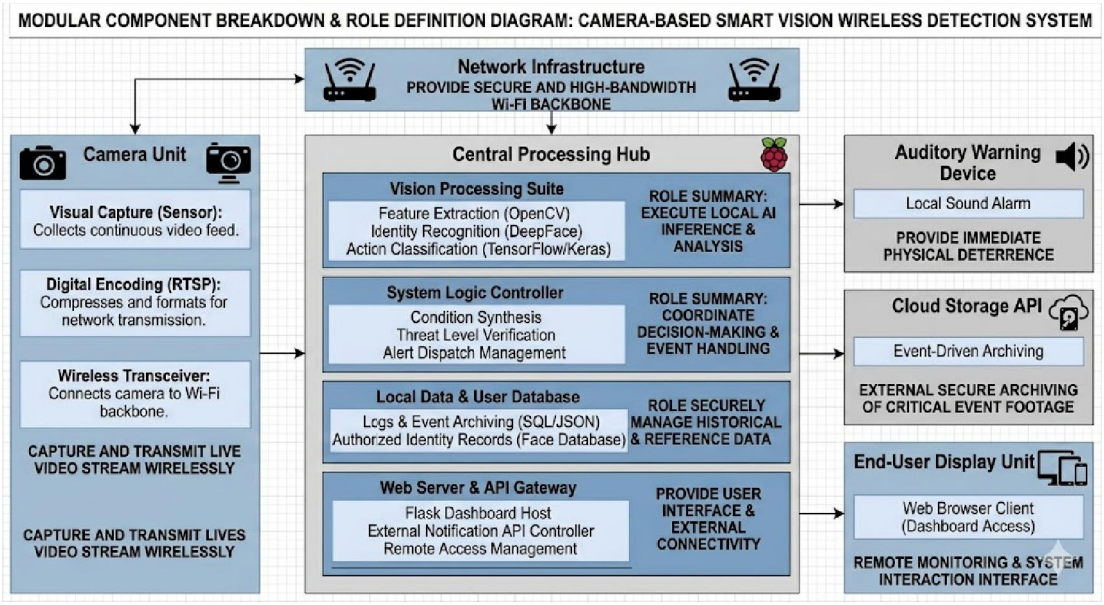

Figure 5.1

The final stage of coordination involves the **Logic Controller** and the **Output Mechanisms**. When the AI software identifies a threat, the Python-based logic sends an immediate command to the hardware's **GPIO pins** or sound drivers, which in turn triggers a physical **Auditory Alarm or Buzzer**. Simultaneously, the **Flask Web Framework** acts as a digital bridge, taking the processed video and the event logs and serving them to a **Web Browser** on a remote monitor or smartphone. This allows the user to see exactly what the AI is "thinking" in real-time. By managing local hardware alerts and remote software visualizations at the same moment, the system ensures that a security event is not just detected in the digital world, but is also acted upon in the physical world.

VI. RESULTS

The implementation of the Camera-Based Smart Vision Wireless Detection System yielded highly successful results in automating real-time surveillance. Using the DeepFace library, the system achieved a high degree of accuracy in facial recognition, successfully distinguishing between authorized personnel and unknown visitors under standard indoor lighting conditions. Furthermore, the integration of TensorFlow for activity recognition allowed the system to classify complex human movements, such as running or loitering, with a high confidence score. By combining these deep learning frameworks with OpenCV's motion detection, the system effectively filtered out environmental noise, such as moving curtains or shadows, ensuring that alerts were only triggered by verified human presence.

A central point of discussion is the efficiency of the wireless architecture and its impact on system latency. The use of the Real-Time Streaming Protocol (RTSP) over a standard Wi-Fi network proved to be a reliable method for eliminating physical cabling while maintaining a stable frame rate for analysis. While high-resolution streams initially introduced minor processing bottlenecks on the Raspberry Pi hardware, optimizing the frame size through OpenCV pre-processing successfully reduced the computational load, bringing latency down to a near-real-time level. The Flask-based web dashboard served as a successful monitoring hub, allowing users to view annotated video feeds and incident logs remotely, which confirms the system's viability for decentralized security management in residential and industrial settings.

Despite the strong performance, the results also highlighted critical challenges regarding environmental variables and hardware constraints. Factors such as extreme low-light conditions or significant distances between the camera and subjects occasionally hindered the precision of the facial recognition model, suggesting that infrared-enabled sensors

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

are necessary for 24/7 reliability. Additionally, the implementation of a "verification buffer," which requires detection across three consecutive frames, was discussed as a vital logic refinement to minimize false positives. Overall, the project successfully demonstrates that an AI-driven, wireless framework significantly reduces the need for constant manual observation, providing a more proactive and scalable alternative to traditional, human-dependent CCTV systems.

VII. APPLICATIONS

The **Camera-Based Smart Vision Wireless Detection System** has a wide range of applications across various sectors due to its ability to combine wireless flexibility with intelligent, real-time decision-making.

Residential Security & Smart Homes: Automated monitoring of entry points to detect unauthorized individuals while allowing recognized family members to enter without triggering alarms. It can also be integrated with smart locks for hands-free, face-based entry.

Industrial Safety & Restricted Zones: Monitoring hazardous or high-security areas in factories. The system can trigger an immediate alert if a worker enters a restricted zone or if the AI detects an anomaly like a "man-down" (a person falling) or a lack of motion in critical areas.

Retail Analytics & Loss Prevention: Recognizing shoplifting behaviors through motion and loitering analysis. Additionally, it can be used for customer heat-mapping and identifying "VIP" or frequent customers to improve personalized service.

Banking & Financial Institutions: Providing a high-level security layer for ATM booths and vault rooms. The system can detect suspicious loitering at odd hours and instantly notify security personnel before a breach occurs.

Healthcare & Elderly Monitoring: Assisting in hospitals or assisted living facilities by monitoring patient rooms wirelessly. The AI can detect if a patient has fallen or is in distress without the privacy concerns of continuous human-monitored video.

Educational Institutions & Campus Safety: Automated attendance systems using facial recognition and monitoring campus perimeters for intruders or unusual activities during late hours.

Public Infrastructure & Smart Cities: Managing crowd density in public spaces like railway stations or airports. It can identify abandoned objects or individuals moving against the flow of traffic to assist law enforcement.

Agriculture & Livestock Monitoring: Wireless monitoring of large farms to detect animal health issues or intruders/predators in areas where physical wiring is impossible to install.

VIII. PROBLEM STATEMENT

In today's world, surveillance plays a vital role in maintaining safety and security across residential, commercial, and public environments. However, traditional CCTV-based monitoring systems rely heavily on manual supervision, which is both time-consuming and inefficient. These conventional systems merely record footage without providing any form of intelligent analysis, requiring human operators to constantly monitor screens to detect unusual activities or potential threats.

Such manual dependency often leads to delayed responses, missed incidents, and ineffective security management, especially in high-risk or crowded areas. Additionally, existing surveillance setups typically use wired connections, which limit camera placement flexibility and increase installation complexity and maintenance costs.

With the rise of artificial intelligence and wireless communication technologies, there is a growing need for an automated and intelligent surveillance system capable of performing real-time video analysis, facial recognition, and behavioral detection without human intervention. The lack of such an integrated, AI-powered, wireless vision system presents a significant gap in the current security infrastructure.

Therefore, the main problem addressed by this project is:

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

To design and develop a Camera-Based Smart Vision Wireless Detection System that can wirelessly capture live video streams, process them using AI-based computer vision models, and automatically detect, recognize, and alert users about abnormal or unauthorized activities in real time.

This system aims to reduce human effort, enhance detection accuracy, and provide an intelligent, wireless, and scalable solution for modern surveillance applications.

Would you like me to now write a "Scope of the Project" section next? It usually follows the Problem Statement and explains the boundaries, focus areas, and implementation coverage of the project.

IX. CONCLUSION

In this project, we have successfully developed a Camera-Based Smart Vision Wireless Detection System (CBSVWDS) that integrates artificial intelligence with real-time video surveillance over a wireless network. The system eliminates the limitations of traditional CCTV setups by providing automated facial recognition, motion tracking, and behavior analysis through advanced deep learning models such as OpenCV, TensorFlow, and DeepFace.

The implementation demonstrates how AI-based vision systems can replace manual monitoring with intelligent automation, capable of identifying potential threats and alerting users instantly. The wireless RTSP integration ensures flexible camera placement and remote access, while the AI detection modules provide accurate, adaptive, and continuous learning for improved performance over time.

This project proves that combining computer vision, AI, and IoT can create a scalable framework for smart security and surveillance applications. The designed system is modular, cost-effective, and capable of being expanded for various industrial and public infrastructure use cases such as schools, factories, hospitals, and smart cities.

ACKNOWLEDGMENT

We would like to extend our heartfelt appreciation to the faculty members of the Automation and Robotics Engineering department for their expert advice and assistance. We owe a particular debt of gratitude to our mentor, **Prof. Atul Atalkar**, whose insightful feedback and steady encouragement were vital to our progress.

Furthermore, we are thankful for the excellent facilities and technical resources made available by the institution, which provided the foundation for our practical work. Lastly, we thank our families and friends for their constant motivation and patience throughout this journey.

REFERENCES

[1]. S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[2]. P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.

[3]. T. Ahonen, A. Hadid, and M. Pietikäinen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[4]. S. Zhang, X. Zhao, and W. Li, "Deep Learning-Based Surveillance Systems for Human Activity Recognition," *Springer Journal of Visual Communication and Image Representation*, vol. 78, 2022.

[5]. S. Serengil and A. Ozpinar, "Lightweight Face Recognition and Facial Attribute Analysis (DeepFace Library)," *2020 11th International Conference on Cloud Computing and Big Data (CloudCom)*, IEEE, 2020.

[6]. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[7]. M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," Google Research, 2015. [Online]. Available: https://www.tensorflow.org

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 14, April 2026
Impact Factor: 8.2

[8]. G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000. [Online]. Available: https://opencv.org
Google Cloud AI, "Vision API Documentation," [Online]. Available: https://cloud.google.com/vision
Flask Documentation, "Flask: Web Development with Python," [Online]. Available: https://flask.palletsprojects.com

[9]. Y. Lecun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, pp. 436–444, 2015.

[10]. Z. Chen, Q. Liu, and D. Zhang, "Anomaly Detection in Video Surveillance Using Deep Learning: A Review," *IEEE Access*, vol. 8, pp. 168415–168424, 2020.

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

[12]. F. Chollet, "Keras: The Python Deep Learning API," *GitHub Repository*, 2015. [Online]. Available: https://keras.io

[13]. Open Source Computer Vision Library (OpenCV) Documentation, 2024.