Multidisciplinary Journal of Engineering and Technology
Print ISSN: 3007-6897
Online ISSN: 3007-6900
DOI: https://doi.org/10.61784/mjet3038

IMPLEMENTATION OF A VIOLATION BEHAVIOR IDENTIFICATION AND EARLY WARNING SYSTEM FOR INDUSTRIAL WORKSHOPS

ZiRui Wu, XinYin Lan, FanHao Ye, ZhenNan Zhang, MuCong Chi*
School of Artificial Intelligence, Wenzhou Polytechnic, Wenzhou 325035, Zhejiang, China.
**Corresponding Author: MuCong Chi*

Abstract: Aiming at the pain points in industrial workshops, such as the low efficiency of manual monitoring, poor computing power adaptability of existing vision-based systems, and the lack of a closed-loop alarm mechanism, this paper designs and implements a violation behavior identification and early warning system. First, based on the YOLOv8n model developed by the Ultralytics team, the system incorporates depthwise separable convolutions and attention mechanisms for lightweight optimization. Combined with TensorRT quantization and compression, this ensures low-latency deployment on edge devices. Second, a four-layer architecture is constructed, comprising the Perception Layer, Edge Computing Layer, Intelligent Decision Layer, and Application Layer. The introduction of an intelligent decision mechanism enables an end-to-end automated closed-loop of identification-alarm-archiving. The system is integrated with DingTalk/WeCom and MES (Manufacturing Execution System) to facilitate real-time alarm notifications and data logging. Finally, the system's performance was validated through experiments in real-world industrial workshop scenarios. The results indicate that the average inference latency at the edge is 39 ms, the identification accuracy for violations reaches 92.3%, and the false alarm rate is reduced to 3.2%. All metrics meet the real-time monitoring requirements of industrial sites.
Keywords: Industrial workshop; Violation identification; Edge computing; YOLO; TensorRT quantization

1 INTRODUCTION

As a core scenario in manufacturing, the safety management of industrial workshops is directly linked to personnel safety and enterprise production efficiency. Currently, most industrial workshops still rely on manual monitoring models, which suffer from inherent drawbacks such as low monitoring efficiency, delayed real-time response, and high labor costs. These limitations make it increasingly difficult to meet the safety control demands of large-scale and fast-paced production environments. Although existing vision-based AI violation identification systems have achieved partial automated detection, they generally face challenges including poor adaptability to limited computing power, high false alarm rates, and a lack of closed-loop alarm mechanisms [1].

To address these pain points, this paper designs and implements a violation behavior identification and early warning system for industrial workshops based on Edge Computing and an Intelligent Decision Layer. Utilizing the YOLOv8n model as a foundation, the study achieves low-latency deployment on the edge through lightweight improvements and TensorRT quantization and compression. Furthermore, by introducing an intelligent decision mechanism, an automated closed-loop processing flow for violation events is constructed, enabling precise alarm push notifications and integration with MES system archiving. Experimental validation demonstrates that the system features low inference latency and a controllable false alarm rate, effectively enhancing the automation level and response speed of workshop safety management while reducing industrial deployment costs. This research offers significant industrial application value and provides a new technical trajectory for intelligent industrial safety monitoring.

2 OVERVIEW OF RELATED TECHNOLOGIES

The implementation of the system proposed in this paper relies on core technologies such as edge computing, lightweight object detection, and an intelligent decision framework. The core characteristics and application logic of each technology are outlined below, providing the foundational support for system design and implementation.

As a lightweight model in the YOLO series, YOLOv8n is characterized by its small parameter size [2], high inference speed, and low deployment cost. By utilizing the C2f module and an Anchor-Free detection mechanism [3], it significantly reduces computational overhead while maintaining a high degree of recognition accuracy, making it well-suited for the computing power constraints of edge devices in industrial workshops. TensorRT [4], a high-performance inference engine developed by NVIDIA, optimizes deep learning models through techniques such as quantization and pruning. By converting floating-point models into integer or half-precision formats, it effectively minimizes inference time and enhances the real-time performance of edge-side deployment.

Edge computing technology breaks through the limitations of traditional centralized cloud processing by offloading core tasks, such as data acquisition and model inference, to edge devices [5,6]. This reduces data transmission latency and network bandwidth pressure while avoiding the response lags caused by centralized cloud computing, perfectly

aligning with the real-time monitoring needs of industrial workshops. Furthermore, drawing structural inspiration from the recent advancements in Agentic AI and autonomous systemic workflows [7,8], the Intelligent Decision Layer is designed as a highly coordinated, reactive control hub. Instead of running as a disjointed script, it adopts a multi-stage "Perception-Decision-Execution" pipeline where modules collaborate via standardized asynchronous data communication. By mapping temporal edge-side inference data onto contextual rule validations, this framework independently manages the entire operational lifecycle—from localized anomaly verification to cross-platform enterprise system archiving—providing the essential technical foundation for a fully automated safety closed-loop.

3 SYSTEM DESIGN AND IMPLEMENTATION

3.1 Overall System Architecture

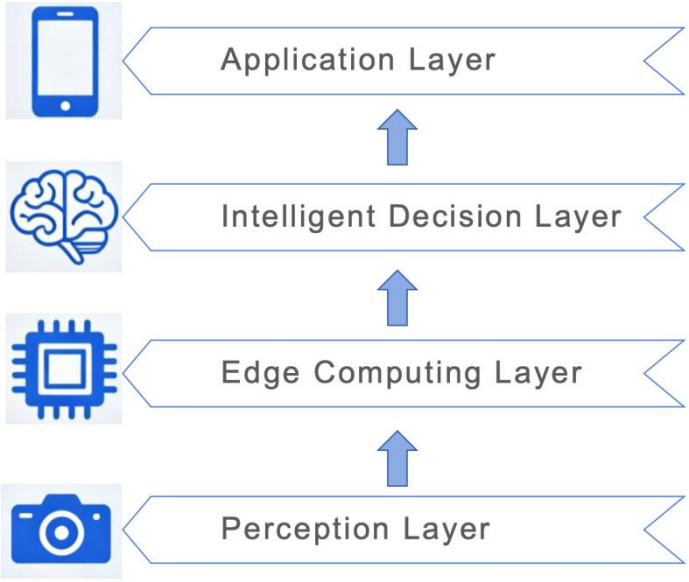

Figure 1 System Architecture

To achieve real-time recognition and closed-loop early warning of violation behaviors in industrial workshops, a four-layer system architecture has been designed, as illustrated in Figure 1. This architecture systematically consists of the Perception Layer, Edge Computing Layer, Intelligent Decision Layer, and Application Layer.

3.1.1 Perception layer design

As the source of data collection, the primary goal of the Perception Layer is to achieve comprehensive coverage and high-quality image acquisition in critical workshop areas. The system utilizes 2-megapixel industrial HD network cameras with infrared night vision, enabling clear capture of real-time data regarding personnel operations, equipment status, and area conditions under complex environments such as uneven lighting or night shifts. A partitioned layout strategy is adopted, with 1–2 cameras deployed in key areas like production zones, hazardous restricted zones, and equipment operation areas to ensure zero blind spots. Cameras are set to 25 fps with a resolution of 1920×1080, balancing image quality with data transmission volume to prevent bottlenecks or blurriness. Additionally, the Perception Layer features a built-in basic pre-processing module for denoising and size normalization, filtering out invalid images (e.g., lens obstruction or motion blur) to reduce the workload on the Edge Computing Layer [9].

3.1.2 Edge computing layer design

Serving as the real-time processing core, the Edge Computing Layer focuses on receiving data from the Perception Layer and performing inference for violation recognition. Its design emphasizes being lightweight, low-latency, and highly adaptive. This layer deploys the improved YOLOv8n model optimized with TensorRT and adopts a modular design consisting of a pre-processing sub-module and an inference sub-module. The former performs image enhancement (brightness and contrast optimization), normalization, and color space conversion to meet model input requirements. The latter executes the optimized model to identify violations, outputting structured data including violation types (e.g., missing safety helmets, unauthorized entry), spatial coordinates, and confidence scores. This layer employs a distributed deployment model, facilitating localized data processing to minimize cross-node latency and supporting node redundancy to ensure system reliability.

3.1.3 Intelligent decision layer design

This layer acts as the central decision-making hub, responsible for false alarm filtering, decision scheduling, and process control. It utilizes a modular architecture comprising a Perception Reception Module, Logic Determination Module, and Decision Scheduling Module. The reception module receives structured results via standardized JSON interfaces, ensuring data consistency. The determination module performs multidimensional logic analysis based on preset rules and historical data to filter out false positives caused by light interference or obstructions. Finally, the

scheduling module autonomously generates decision commands, sending alarm notifications and archiving instructions to the Application Layer while maintaining decision logs for future traceability and system optimization [10].

3.1.4 Application layer design

The Application Layer focuses on practical deployment and user interaction, providing alarm notifications, data archiving, and simplified visualization. The Alarm Notification Sub-module utilizes DingTalk/WeCom open APIs to push precise alerts (including violation type, location, real-time screenshots, and timestamps) to management. It supports hierarchical notification, where minor violations are sent to workshop supervisors while critical incidents are escalated to safety heads. The Data Archiving Sub-module integrates with the MES (Manufacturing Execution System) API to automatically log incident details, linking safety data with production management for performance audits. The Visualization Sub-module provides an intuitive interface for real-time monitoring and historical data review.

3.2 Implementation of Core Modules

3.2.1 Lightweight YOLOv8n improvement and TensorRT quantization

To address the insufficient accuracy of the standard YOLOv8n model in complex industrial scenes (e.g., uneven lighting, equipment occlusion) and its high computational demand for edge devices, specific improvements were implemented. First, the C2f module of YOLOv8n was optimized by replacing traditional convolutions with depthwise separable convolutions. This reduces redundant channels and lowers the total parameter count by over 20% without significantly degrading feature extraction capabilities, following the theoretical foundations of MobileNets [11]. Second, a Convolutional Block Attention Module (CBAM) was integrated to enhance feature focusing on violation targets [12], such as personnel without safety helmets or those entering restricted zones, thereby improving recognition precision in complex backgrounds.

Following model optimization, TensorRT was introduced to perform hybrid INT8 quantization. Utilizing a localized workshop calibration dataset, the floating-point network was strategically mapped into an integer-based representation. To prevent the significant accuracy drop common in YOLOv8 edge deployment, a mixed-precision paradigm was adopted, keeping the precision-sensitive regression layers in FP16 while quantizing the remaining layers to INT8. This process drastically reduces computational density and memory footprint during runtime inference. As a result, the optimized model successfully satisfies the stringent hardware constraints of the edge device while maximizing real-time processing throughput in practical industrial environments.

3.2.2 Edge-side low-latency deployment optimization

An industrial-grade edge computing gateway equipped with an NVIDIA Jetson Nano chip was selected as the deployment carrier due to its compact size, low power consumption, and high adaptability to workshop environments. The software environment was built on Ubuntu, integrating TensorRT and OpenCV libraries.

To minimize latency, the inference engine configuration was optimized using batch inference and image pre-processing acceleration strategies, which reduce time spent on scaling and normalization. Furthermore, a data transmission optimization mechanism was implemented: image data captured at the edge is compressed before being fed into the model. These strategies, aligned with current trends in edge computing, ensure the system meets real-time detection requirements [13].

3.2.3 Intelligent decision-driven closed-loop management

The intelligent decision layer adopts a modular design consisting of three sub-modules: Perception, Decision, and Execution, which collaborate via structured data interfaces.

- Perception Module: Receives real-time recognition results (violation types, coordinates, confidence scores) from the edge layer and performs data standardization.
- Decision Module: The Decision Module functions as the core validation engine of the intelligent decision layer, engineered to eliminate transient false alarms caused by lighting changes, dust, or brief occlusions. It tracks violations across a temporal sliding window W_t. An alarm transitions from "Pending" to "Verified" only if the cumulative score satisfies the validation formula:

$$\sum_{i \in W_t} \left(\alpha \cdot \ \mathrm{Conf}_i + \beta \cdot \ \mathrm{IoU}_i \right) \geq T_d \tag{1}$$

where Conf_i is the detection confidence score at frame i, IoU_i is the bounding box spatial stability metric, α and β are experimental weights, and T_d is the verification threshold.

- Execution Module: Triggers precise alarm notifications via DingTalk/WeCom open APIs to relevant personnel. Simultaneously, it interfaces with the MES API to archive incident details (including status, screenshots, and results), completing the "identification-alarm-archiving" automation loop in accordance with smart manufacturing standards.

3.2.4 System integration implementation

Data interaction between modules is facilitated by standardized HTTP interfaces using JSON for structured data transmission, ensuring high accuracy and efficiency. Integration with third-party platforms (DingTalk, WeCom, and MES) is handled through API calls to simplify the deployment process. To ensure system stability in industrial settings, an exception handling mechanism was established for all interfaces, preventing data loss during network fluctuations and ensuring the rapid and reliable landing of the system.

4 EXPERIMENTAL TESTING AND ANALYSIS

To verify the feasibility, real-time performance, and accuracy of the system in industrial scenarios, and in accordance with the concise requirements of journal publications, experimental tests were conducted focusing on core performance indicators. These tests clarify the environment, scheme, and results to validate whether the system meets the practical requirements for violation identification and early warning in industrial workshops.

4.1 Test Environment Setup

The test environment is divided into hardware and software components, ensuring consistency with actual industrial deployment to guarantee the authenticity and reference value of the results.

- Hardware Environment: The edge computing node utilizes an industrial-grade edge gateway equipped with an NVIDIA Jetson Nano chip (1.43GHz CPU, 4GB RAM), which is well-suited for lightweight deployment. The perception layer employs 2-megapixel industrial HD network cameras deployed according to the actual workshop layout, capturing images at a resolution of 1920×1080 and a frame rate of 25 fps. A standard office computer serves as the testing terminal for data statistics and analysis.
- Software Environment: The edge node operates on Ubuntu 18.04 LTS (JetPack 4.6), with core dependencies including TensorRT 8.2, OpenCV 4.5, and PyTorch 1.10. The intelligent decision layer is developed in Python, interfacing seamlessly with third-party messaging platforms (DingTalk/WeCom) and manufacturing enterprise platforms (MES) via standard APIs.
- Dataset: The dataset comprises 5,000 images from real industrial workshop scenarios, covering three categories: missing safety helmets, unauthorized entry, and safety guard anomalies (such as open machine shield doors, removed protective grilles, or displaced interlock covers). The data is split into a training set (4,000 images) and a test set (1,000 images) in a 4:1 ratio, encompassing complex variables such as uneven lighting, equipment occlusion, and night operations to ensure comprehensiveness.

4.2 Test Scheme Design

The testing focuses on core system performance, utilizing both performance and functional test schemes while omitting redundant items to ensure efficiency.

- Performance Testing: This prioritizes three key metrics: inference latency (time taken for a single image inference on the edge node), identification accuracy (the ratio of correctly identified violations to total violations), and false alarm rate (the ratio of incorrectly identified violations to total images). During testing, images from the test set were continuously processed. Average values were recorded as final results.
- Functional Testing: This phase emphasizes the verification of the system's closed-loop management pipeline. It assesses the accuracy of the edge recognition output, the effectiveness of the intelligent decision engine’s false alarm filtering, the timeliness of alert notifications, and the completeness of MES data archiving. Real-world violation scenarios were simulated to ensure that the end-to-end "identification-alarm-archiving" workflow operates smoothly and reliably.

4.3 Test Results and Analysis

The results indicate that all performance metrics meet industrial requirements, and the functional closed-loop is stable and reliable.

- Performance Metrics: The optimized system achieved an average inference latency of 39 ms on the edge node, well below the 50ms target. As illustrated in Table 1, the average identification accuracy reached 92.3%.

Table 1 Performance Verification Across Different Violation Categories

Violation Category	Test Samples	Accuracy
Safety Guard Anomalies	450	91.8%
Missing Safety Helmets	300	93.7%
Unauthorized Entry	250	91.6%
Total / Average	**1000**	**92.3%**

- Decision Engine Optimization: After logical determination by the intelligent decision engine, the false alarm rate dropped to 3.2%, representing a 21% relative reduction compared to the unoptimized baseline. This significant filtering effect highlights the structural advantages of the intelligent decision layer in industrial automation.
- Functional Reliability: Functional testing confirmed that the edge computing layer accurately outputs recognition results, the intelligent decision engine effectively filters false positives, and alarm notifications are pushed to management terminals within 3 s. Furthermore, violation details are archived in the MES system automatically and completely.

Comprehensive analysis shows that by utilizing lightweight model optimization and edge deployment, the system effectively resolves the issues of limited computing power and latency in industrial settings. The introduction of the intelligent decision layer enables robust false alarm filtering and process closure. The system demonstrates excellent

performance, comprehensive functionality, and high practical value for industrial safety management, offering clear advantages over existing monitoring systems.

5 CONCLUSION AND FUTURE WORK

5.1 Conclusion

This paper addresses the core pain points in industrial workshop safety management, such as the low efficiency of manual monitoring, the lack of a closed-loop alarm mechanism in existing vision-based AI systems, and poor adaptability to limited computing power. A violation behavior identification and early warning system based on Edge Computing and an Intelligent Decision Layer was designed and implemented.

By performing lightweight improvements on the YOLOv8n model and applying TensorRT quantization, the research achieves real-time, low-latency recognition of violations. The model optimization is theoretically supported by depthwise separable convolutions and attention mechanisms, and technically grounded in the YOLOv8 algorithm and TensorRT optimization, yielding significant results. Furthermore, the introduction of the intelligent decision mechanism enables a fully automated "identification-alarm-archiving" closed-loop, facilitating precise alarm notifications and seamless data linkage with the MES system. Experimental results demonstrate that the system's edge-side inference latency is as low as 39 ms, with a recognition accuracy of 92.3% and a 21% relative reduction in the false alarm rate compared to the unoptimized baseline.

These metrics meet industrial site requirements and outperform existing edge-AI safety monitoring benchmarks. The system effectively lowers the hardware threshold for industrial deployment, significantly enhances the automation and responsiveness of workshop safety management, and provides a novel technical roadmap for intelligent industrial safety monitoring.

5.2 Limitations and Future Work

While this research successfully implements the core functions of the system, certain limitations remain. First, the identification scenarios are still focused on common violation types; the recognition accuracy for specialized behaviors under complex conditions (e.g., improper operation of precision equipment) needs further improvement. Second, the decision logic of the intelligent decision layer is primarily based on heuristic rules, lacking sufficient autonomous learning and adaptive capabilities, which represents a gap compared to the evolving trends of adaptive control systems in industrial automation.

Future research will focus on two primary directions:

- Expanding Violation Categories: Optimizing the model's feature extraction capabilities to enhance recognition precision in complex and specialized scenarios.
- Introducing Reinforcement Learning: Enhancing the autonomous decision-making and adaptive capabilities of the intelligent decision engine [14]. This will enable the system to dynamically adjust determination rules according to changes in the workshop environment, further improving the intelligence and industrial adaptability of the system.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Jha S. Computer Vision for Surveillance and Monitoring. 2025 5th International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), MANDYA, India, 2025: 1-5. DOI: 10.1109/ICERECT65215.2025.11376098.

[2] Jocher G, Chaurasia A, Qiu J. Ultralytics YOLO (Version 8.0) [Software], 2023, Accessed: May 2026. https://github.com/ultralytics/ultralytics.

[3] Zhai H, Du J, Ai Y, et al. Edge deployment of deep networks for visual detection: a review. IEEE Sensors Journal, 2024, 25(11): 18662-18683.

[4] NVIDIA Corporation. NVIDIA TensorRT: Programmable Deep Learning Accelerator. [Technical Documentation], 2023, Accessed: May 2026. https://developer.nvidia.com/tensorrt.

[5] Bayar A, Şener U, Kayabay K, et al. Edge computing applications in industrial IoT: A literature review. International Conference on the Economics of Grids, Clouds, Systems, and Services, GECON 2022. Lecture Notes in Computer Science, 2023: 124-131. DOI: 10.1007/978-3-031-29315-3_11.

[6] Savaglio C, Mazzei P, Fortino G. Edge intelligence for industrial IoT: Opportunities and limitations. Procedia Computer Science, 2024, 232, 397-405.

[7] Abou Ali M, Dornaika F, Charafeddine J. Agentic AI: a comprehensive survey of architectures, applications, and future directions. Artificial Intelligence Review, 2025, 59(1): 11.

[8] Jaggavarapu MKR. The evolution of agentic AI: architecture and workflows for autonomous systems. Journal Of Multidisciplinary, 2025, 5(7): 418-427.

[9] Zhang Y, Liao X. Asymmetric Training and Symmetric Fusion for Image Denoising in Edge Computing. Symmetry, 2025, 17(3): 424.
[10] Wang J, Yang F, Chen T, et al. An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems. IEEE Transactions on Automation Science and Engineering, 2015, 13(2): 1045-1061.
[11] Howard AG, Zhu M, Chen B, et al. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint, 2017. DOI: 10.48550/arXiv.1704.04861.
[12] Woo S, Park J, Lee JY, et al. CBAM: Convolutional block attention module. Proceedings of the European Conference on Computer Vision (ECCV), 2018, 11211, 3-19. DOI: 10.1007/978-3-030-01234-2_1.
[13] Wu Y, Guo H, Chakraborty C, et al. Edge computing driven low-light image dynamic enhancement for object detection. IEEE Transactions on Network Science and Engineering, 2022, 10(5): 3086-3098.
[14] Alginahi YM, Sabri O, Said W. Reinforcement Learning for Industrial Automation: A Comprehensive Review of Adaptive Control and Decision-Making in Smart Factories. Machines, 2025, 13(12): 1140.