

An API-Driven System for Real-Time Safety Monitoring on Construction Sites: Integrating Machine Learning and Industry Foundation Classes

Mukta Nisal¹  and Xiangyu Wang¹

¹Institut für Bauinformatik, TU Dresden University, Dresden, Germany

E-mail(s): mukta_rajan.nisal@tu-dresden.de, xiangyu.wang@tu.dresden.de

Abstract: This research focuses on improving construction site safety by developing an innovative real-time hazard detection system using machine learning models. At its core is a camera-based backend API that deploys machine learning models for object detection in an SQL database. When potential safety issues arise, the system triggers alerts and logs key data points such as timestamps and alert types. Alerts are grouped into short-term (24-hour) and long-term (1-month) categories, which offer actionable insights for immediate and ongoing safety improvements. By integrating advanced object detection methods like Yolov8 with efficient database management and API-driven functionality, the system offers an effective solution to monitor construction site safety in real time. Its ability to dynamically manage machine learning models and alert categorization functionality make it an essential tool for fostering safer workplaces. Future developments include integrating location data using compatible sensors to further enhance risk monitoring. Additionally, the paper discusses future integration with the Industry Foundation Classes, which could expand the system's capabilities by linking hazard detection to specific building elements and processes.

Keywords: construction site safety, real-time object detection, machine learning models, alert generation, SQL database



DOI: 10.18154/RWTH-CONV-254924. Published in the conference proceedings of the 36. Forum Bauinformatik 2025, Aachen, Germany,
© 2025 The copyright for this article lies with the authors. This publication, except for quotations and otherwise indicated parts, is licensed under a
Creative Commons Attribution 4.0 International (CC BY 4.0) license.

1 Introduction

The construction industry remains one of the most hazardous sectors globally, with frequent safety lapses attributed to delayed hazard recognition, non-compliance, and the ever-changing site environment [1], [2]. Traditional safety approaches which are largely reactive and manual often do not provide the real-time awareness required to prevent accidents [1]. Recent advances in artificial intelligence (AI) and computer vision have opened new avenues for proactive safety monitoring. AI-driven object detection models, such as those in the YOLO (You Only Look Once) family, enables high-speed, accurate recognition of safety-critical elements like PPE compliance, unguarded edges and equipment proximity

[3], [4], [5]. This study proposes a deployable, real-time construction safety detection framework using YOLOv8n integrated with a FastAPI backend. The system processes both helmet-mounted and static camera inputs, offering a robust detection architecture. It supports modular updates, adapts to site changes and provides actionable insights through dashboard integration [6]. The goal is to validate this system under varied site conditions and demonstrate its effectiveness in enhancing safety through immediate risk detection and response.

2 Literature Review

2.1 AI for Construction Safety Monitoring

Recent advances in artificial intelligence (AI) and computer vision have significantly reshaped safety monitoring in the high-risk construction environment. Alateeq et al. [6] utilized deep learning to identify site hazards with high precision, demonstrating scalable detection of personal protective equipment (PPE) violations. Similarly, Zhou et al. [2] conducted a comprehensive review of safety management strategies, supporting the integration of AI for proactive safety enforcement. Arabshahi et al. [7] emphasized the increasing adoption of sensing technologies in the construction industry, with sensors and AI forming the backbone of modern safety infrastructures. Boje et al. [8] advocated for semantic digital twins to facilitate real-time risk analysis and intervention. Karlapudi et al. [9] demonstrated how Internet of Things (IoT) devices, digital twins, and data fusion can be leveraged to improve safety on construction sites. This approach strongly supports the integration of AI-based hazard detection with Building Information Modeling (BIM), highlighting the need for interoperable systems that combine real-time monitoring with digital representations of construction processes.

2.2 Detection Models: CNNs and YOLO-based Approaches

Detection models such as convolutional neural networks (CNNs) and the YOLO (You Only Look Once) series are crucial in automating risk identification. Redmon et al. [3] pioneered YOLO, which delivers real-time object detection with minimal latency. Jocher [4] introduced YOLOv8, offering Open Neural Network Exchange (ONNX) compatibility and a modular design suitable for deployment on edge devices. Sohan et al. [10] reviewed the improvements of YOLOv8, particularly for complex construction environments. Talib et al. [11] proposed YOLOv8-CAB, achieving enhanced detection precision in dynamic scenes. Zhang et al. [5] demonstrated high-accuracy PPE detection using sensor-integrated CNNs. Parkash et al. [12] supported RFID-based systems for location-aware risk evaluation. Arabshahi et al. [7] also discussed wearable sensor systems that enable real-time proximity detection of hazards using Ultra-Wideband (UWB) and other localization technologies.

2.3 Alert Systems and Smart Integration

Arabshahi et al. [7] and Zhang et al. [5] demonstrated the integration of sensor data with AI-based real-time alert systems to reduce incident response times in construction settings. Talib et al. [11] further improved this concept with their YOLOv8 framework, which combines high-speed detection with enhanced alerting mechanisms suitable for dynamic site conditions. Despite progress, challenges remain: sensor noise, model robustness, and audit transparency. As Alateeq et al. [6] suggest, improved data fidelity and edge-device compatibility are essential for next-generation deployments.

2.4 Categorization of Hazards

Hossen [13] analyzed and categorized hazard situations and locations at construction sites in his master's thesis at TU Dresden. His work emphasizes that hazards can be systematically classified into categories such as physical, environmental, and procedural risks. This categorization approach highlights the importance of not only identifying hazards but also contextualizing them within their specific risk domains, thereby improving targeted mitigation strategies.

3 Methodology

This section outlines the technical foundation of our real-time hazard detection system tailored for high-risk construction sites. The pipeline integrates AI-driven object detection and a FastAPI microservice for scalable deployment.

3.1 FastAPI-Based Object Detection Service

The Hazard Detection API is a FastAPI-based microservice that provides real-time computer vision capabilities. It includes intuitive endpoints like `POST /detect` (submit images), `GET /yolo/models` (model metadata), and `POST /update_model` (YOLOv8 weight hot-swapping). The service fuses UWB tag coordinates with 360-degree helmet-mounted and static camera feeds, returning annotated bounding boxes, confidence scores, and 3D hazard positions. Analytical endpoints such as `/detection/alerts` and `/detection/analyze` allow retrieval of historical detection logs and safety summaries.

3.2 Object Detection Workflow

Figure 2 illustrates the end-to-end pipeline for object detection. It begins with data collection using 360-degree and standard cameras. The data collected are enriched through Open Experience tools like BauMangel and BauPhotos. Data was collected using two major tools:

1. **360-Degree Camera:** The foundation of object detection is the data collection process. This process was done with the help of two major tools: the 360-degree Camera and a standard camera. The 360-degree camera system from Open Experience GmbH is a panoramic device seamlessly integrated into a construction helmet. Equipped with two 12MP cameras and high-quality wide-angle lenses, this system ensures exceptional detail and clarity in every 360-degree image.
2. **Standard Camera:** In addition to the 360° camera, a standard camera provides object-focused views with varying fields of view, improving recognition under diverse conditions. Combined with panoramic data, this creates a rich dataset covering both contextual and detailed perspectives.



Figure 1: Helmet-mounted 360-degree camera for data acquisition

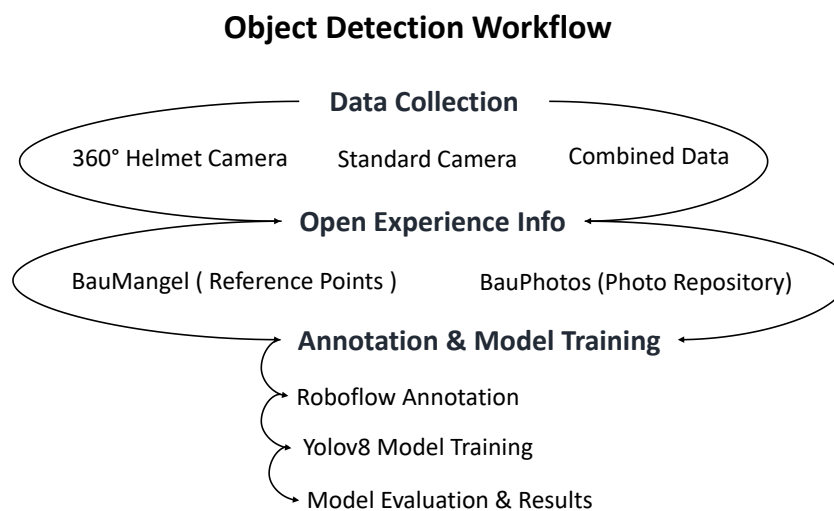


Figure 2: Object detection workflow pipeline from data collection to model evaluation.

3.3 Annotation and Model Training

1. **Roboflow Annotation:** (Figure 3) All collected images were uploaded to the RoboFlow Platform for manual annotation using web interface and bounding boxes were drawn around the critical safety classes such as `hazardous_object`, `big_obstacle`, `open_wires`. The final RoboFlow-annotated dataset was exported into a YOLOv8-compatible format.



Figure 3: Roboflow annotation interface showing bounding boxes

2. **YOLOv8 Model Training:** The annotated dataset was exported directly in YOLOv8 format, eliminating manual conversion steps. We selected the YOLOv8n (“nano”) variant for its balance of real-time inference speed and detection accuracy, leveraging its anchor-free heads and CSP-based feature fusion. Training was carried out via the Ultralytics YOLOv8 Python library, with hyperparameters tuned (learning rate, batch size, augmentation mix) over 300 epochs. Key performance metrics mAP@0.5 (0.75), precision (0.82), recall (0.78), and average inference time (50 ms) were monitored to guide further augmentation and learning-rate adjustments.

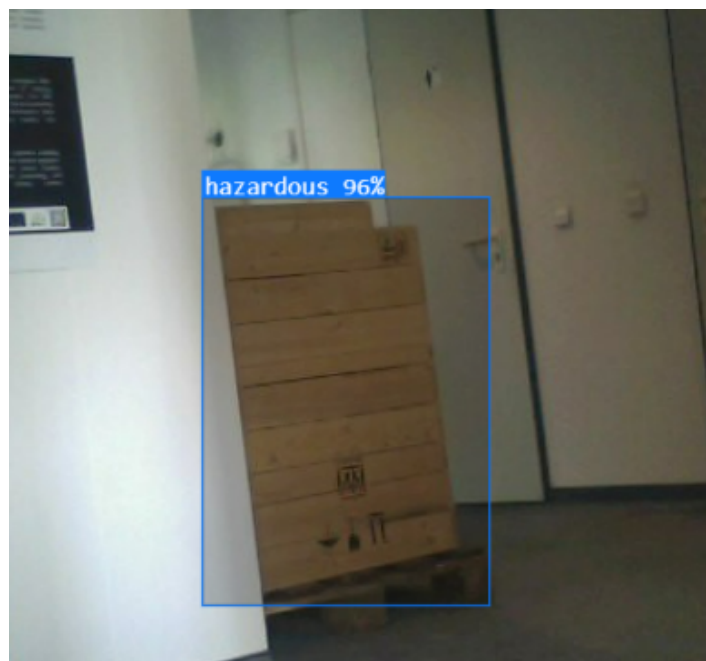


Figure 4: YOLOv8 model output on construction dataset

3.4 API Architecture Overview

The architecture, shown in Figure 5, illustrates the proposed hazard detection system comprising multiple interacting components, each assigned specific responsibilities to facilitate real-time safety monitoring and system responsiveness.

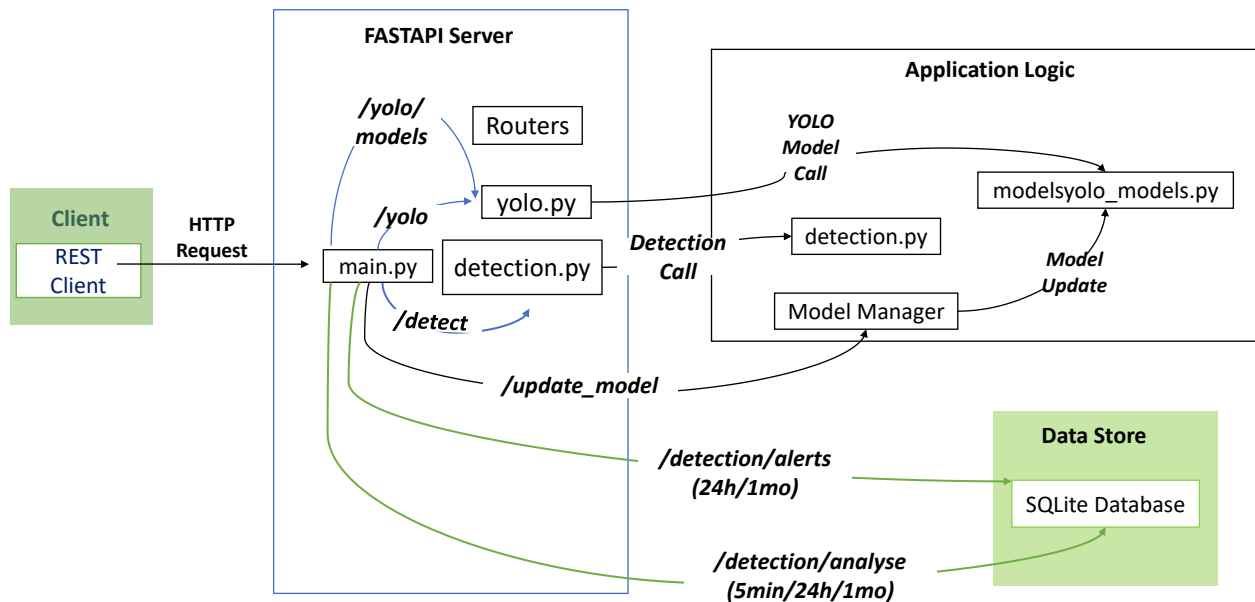


Figure 5: FastAPI Microservices Architecture for Real-time Hazard Detection

1. The Client represents the entry point into the system. It could be a web interface, mobile application, or REST client tools such as Postman or curl. These clients interact with the backend server by sending HTTP requests for tasks like submitting images for detection, updating models, or retrieving analytics.
2. The Server Layer operates as the command center of the system, built using FastAPI. The core routing begins with `main.py`, which directs requests to relevant handlers. Three primary routers include `yolo.py` for managing YOLO models and their metadata, `detection.py` for processing detection requests, and `/update_model` for managing updates to model weights. These routers expose endpoints such as `/yolo`, `/yolo/models`, `/detect`, `/detection/alerts`, and `/detection/analyse`, supporting metadata access, detection submissions, and real-time analytics.
3. The Application Logic encompasses the detection engines and control modules. `models/yolo_models.py` defines the YOLO model architecture, while `detector.py` handles the logic for processing detections.

4. Data Store is implemented using SQLite, archives detection logs, alerts, and analytics outputs. It supports the backend endpoints `/detection/alerts` and `/detection/analyze`, which allow users to extract insights over predefined periods.
5. The Alerting Mechanism is responsible for communicating safety risks in real time. When the system identifies a hazard, it instantly generates alerts both visually and digitally to notify workers or supervisors (see Fig. 5). This rapid alert delivery is vital for minimizing incident response time on active construction sites.

Table 1: API Component Overview and Interaction Flow

Component	Role	Interactions
REST Client	Sends HTTP requests	Communicates with FastAPI Server
main.py	Entry point, routes requests	Calls routers and application logic
yolo.py	Handles YOLO-related endpoints	Interacts with YOLO model logic
detection.py	Handles detection endpoints	Calls detector logic
detector.py	Performs detection using YOLO models	Uses models/yolo_models.py
models/yolo_models.py	Implements YOLO models	Used by detector and model manager
Model Manager	Updates YOLO models	Interacts with models/yolo_models.py
SQLite Database	Stores alerts and analysis results	Accessed via detection endpoints

4 Conclusion

This study introduces a modular and scalable real-time hazard detection framework tailored to the unique safety demands of construction environments. By integrating the YOLOv8n model with a FastAPI-based backend, the system provides fast, reliable, and high-fidelity detection suitable for deployment on-site. Challenges such as inconsistent lighting, environmental variability, and real-time alerting were systematically addressed through panoramic 360-degree data acquisition, edge-optimized model design, and a well-structured API interface. The pipeline, validated on annotated datasets from both standard and helmet-mounted 360-degree cameras, achieved precision and recall scores above 0.75. While performance was promising, thin or occluded hazards such as open wires remain particularly challenging; addressing false negatives in such safety-critical scenarios will therefore be a key focus of future work. Overall, the framework demonstrates strong potential as a critical layer of risk mitigation in digitally enabled construction sites. Future developments may extend alerting to mobile devices for supervisors, enabling on-site decision support. In addition, the modular API-based structure supports integration with dashboards, edge systems, and Industry Foundation Classes (IFC), linking hazard detection to BIM entities for automated and long-term safety management.

References

- [1] A. Pinto, I. L. Nunes, and R. A. Ribeiro, "Occupational risk assessment in construction industry – overview and reflection", *Safety Science*, vol. 49, no. 5, pp. 616–624, 2011. DOI: 10.1016/j.ssci.2011.01.003
- [2] Z. Zhou, Y. M. Goh, and Q. Li, "Overview and analysis of safety management studies in the construction industry", *Safety Science*, vol. 72, pp. 337–350, 2015. DOI: 10.1016/j.ssci.2014.10.006
- [3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [4] G. Jocher, *Yolov8: Next generation real-time object detector*, <https://github.com/ultralytics/ultralytics>, 2023.
- [5] M. Zhang, T. Cao, and X. Zhao, "Applying sensor-based technology to improve construction safety management", *Sensors*, vol. 17, no. 8, 2017. DOI: 10.3390/s17081841
- [6] M. M. Alateeq, F. R. P.P., and M. A. S. Ali, "Construction site hazards identification using deep learning and computer vision", *Sustainability*, vol. 15, no. 3, p. 2358, 2023. DOI: 10.3390/su15032358
- [7] M. Arabshahi et al., "Review on sensing technology adoption in the construction industry", *Sensors*, vol. 21, no. 24, 2021. DOI: 10.3390/s21248307
- [8] C. Boje, A. Guerriero, S. Kubicki, and Y. Rezgui, "Towards a semantic construction digital twin: Directions for future research", *Automation in Construction*, vol. 114, p. 103 179, 2020. DOI: 10.1016/j.autcon.2020.103179
- [9] J. Karlapudi, K. Röhl, M. Di'Pasquale, and K. Menzel, "Improving safety on construction sites by exploiting iot, digital twins, and data fusion", in *Smart and Sustainable Collaborative Networks*, L. M. Camarinha-Matos, editor, Valencia: Springer, 2023. DOI: 10.1007/978-3-031-42622-3_46
- [10] M. Sohan, S. Ram, and C. V. Reddy, "A review on yolov8 and its advancements", in *Lecture Notes in Electrical Engineering*, 2024, pp. 529–545. DOI: 10.1007/978-981-99-7962-2_39
- [11] M. Talib, A. H. Y. Al-Noori, and J. Suad, "Yolov8-cab: Improved yolov8 for real-time object detection", *Karbala International Journal of Modern Science*, vol. 10, no. 1, 2024. DOI: 10.33640/2405-609X.3339
- [12] D. Parkash, T. Kundu, and P. Kaur, "The rfid technology and its applications: A review", *International Journal of Electronics and Computer Science Engineering*, vol. 1, no. 3, pp. 1447–1455, 2012.
- [13] S. Hossen, "Analysis and mitigation of hazard situations and locations at construction sites", Masterarbeit, TU Dresden, 2024.