







Article

Vision-Based Construction Safety Monitoring Utilizing Temporal Analysis to Reduce False Alarms

Syed Farhan Alam Zaidi [†], Jaehun Yang [†], Muhammad Sibtain Abbas , Rahat Hussain , Doyeop Lee 
and Chansik Park ^{*}

Department of Architectural Engineering, Chung-Ang University, 84, Heukseok-ro, Dongjak-gu, Seoul 06974, Republic of Korea; farhanzaidi@cau.ac.kr (S.F.A.Z.); yjhoon11@cau.ac.kr (J.Y.); muhammadsibtain@cau.ac.kr (M.S.A.); rahathussain@cau.ac.kr (R.H.); doyeop@cau.ac.kr (D.L.)

^{*} Correspondence: cpark@cau.ac.kr

[†] These authors contributed equally to this work.

Abstract: Construction safety requires real-time monitoring due to its hazardous nature. Existing vision-based monitoring systems classify each frame to identify safe or unsafe scenes, often triggering false alarms due to object misdetection or false detection, which reduces the overall monitoring system's performance. To overcome this problem, this research introduces a safety monitoring system that leverages a novel temporal-analysis-based algorithm to reduce false alarms. The proposed system comprises three main modules: object detection, rule compliance, and temporal analysis. The system employs a coordination correlation technique to verify personal protective equipment (PPE), even with partially visible workers, overcoming a common monitoring challenge on job sites. The temporal-analysis module is the key component that evaluates multiple frames within a time window, triggering alarms when the hazard threshold is exceeded, thus reducing false alarms. The experimental results demonstrate 95% accuracy and an F1-score in scene classification, with a notable 2.03% average decrease in false alarms during real-time monitoring across five test videos. This study advances knowledge in safety monitoring by introducing and validating a temporal-analysis-based algorithm. This approach not only improves the reliability of safety-rule-compliance checks but also addresses challenges of misdetection and false alarms, thereby enhancing safety management protocols in hazardous environments.



Citation: Zaidi, S.F.A.; Yang, J.; Abbas, M.S.; Hussain, R.; Lee, D.; Park, C. Vision-Based Construction Safety Monitoring Utilizing Temporal Analysis to Reduce False Alarms. *Buildings* **2024**, *14*, 1878. <https://doi.org/10.3390/buildings14061878>

Academic Editor: Irem Dikmen

Received: 14 May 2024

Revised: 17 June 2024

Accepted: 19 June 2024

Published: 20 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: construction safety; computer vision; temporal analysis; false alarm; personal protective equipment (PPE); real-time monitoring

1. Introduction

Globally, the construction industry experiences an excessive number of fatal and non-fatal accidents. Based on a report from the International Labor Organization, construction workers in developed nations face a nearly four-fold higher likelihood of experiencing fatal accidents compared to workers in other industries. Similarly, their counterparts in less-developed countries are at a nearly six-fold higher risk compared to workers in various other sectors [1]. Fatalities and cases of permanent disability occur alarmingly often in the construction industry [2–4]. Although the construction sector employs about 7% of the global workforce, it is responsible for a significant 30% to 40% of occupational deaths in many countries [5–7]. For instance, South Korea had the highest average mortality rate of 17.9 compared to the United States of 9.4 and China of 5.3 [8]. Renowned for its complexity, the construction industry sector heavily relies on manual labor for supervision and oversight. However, the involvement of human intervention in maintenance and safety monitoring has proven to be a costly, time-consuming, and error-prone process, resulting in inefficiency. Nevertheless, the utmost responsibility of construction companies lies in ensuring the safety of their workers. Consequently, it is crucial to speed up processes, improve productivity, and deal with safety problems in the construction industry promptly [9–13].

Previous research on construction safety has highlighted the significant number of hazards present in the industry due to dynamic and temporary on-site activities. Factors contributing to the high accident rates include personnel factors, company factors, and immediate factors directly related to accidents [14]. Recently, computer vision (CV) technology applications have expanded in various industries, including construction, where CV plays a crucial role in enhancing safety and monitoring operations [15]. Zheng and Li reviewed studies between 1991 and 2021, indicating the significance of information technology tools and software such as computer vision, virtual reality, and simulation in enhancing hazard awareness and safety practices in construction [14]. Researchers have developed CV-based detectors and systems to prevent accidents and ensure compliance with safety regulations at construction sites [16]. Significant contributions have been made by proposing personal protective equipment (PPE) detectors to ensure compliance with safety regulations [17–22].

However, these methods, despite their effectiveness in controlled environments, have yet to be deployed and assessed on actual construction sites in real-time [23]. Some researchers have proposed practical applications of their CV techniques in real construction-site scenarios, but these models commonly exhibit limited generalizability, performing well only within the specific conditions of their training datasets. Furthermore, these methods do not consistently detect each frame accurately, leading to false or missed detections due to occlusion or overlapping [24]. This issue results in false alarms, making automated monitoring systems impractical for construction sites. Additionally, Li et al. have highlighted the need for standardized methods to mitigate subjectivity and errors in safety risk decisions [25]. These approaches highlight the need for robust and reliable systems that can operate effectively in dynamic and complex environments.

To address these challenges, researchers have integrated CV technology with Internet of Things (IoT) sensors to minimize false alarms. Wang et al. [26] proposed a system that combines object-tracking algorithms with sensors to decrease false alarms in detecting unsafe-proximity situations. Talaat et al. [27] proposed a smart fire detection system that enhances accuracy and reduces false alarms. Similarly, optimized deep learning models and temporal-analysis techniques have been explored to improve detection accuracy and reduce false detection rates [28–33]. According to the bibliometric study by Luo et al. [34], various developed and developing countries have adopted AI technology for safety research on construction sites. Mostly, researchers have proposed new methods for rule compliance, inspection, and risk assessment and identification. However, classification or rule-compliance-based systems often fail during monitoring due to false detections or missed detections, triggering the alarm after detecting an unsafe event. Despite these advancements, no study has comprehensively addressed the challenges of false and missed detection in real-time construction-site safety monitoring, which triggers false alarms.

Therefore, this study aims to propose a CV-based approach that integrates time-based analysis of work construction activities. This integration aims to classify safe and unsafe actions effectively by considering the presence of both the objects and the temporal context of worker behavior. To do so, the proposed system incorporates a newly developed algorithm for temporal analysis to address the challenge of false alarms triggered in dynamic work environments. As a case study, the system focuses on validating the dynamic scenario of wearing a helmet, where worker behavior frequently changes throughout the workday. The system first utilizes an object correlation technique to ascertain whether the worker is wearing a helmet. Subsequently, the system leverages the temporal-analysis module to examine the time-based relationship between the detected objects. This analysis helps to reduce false alarms triggered by dynamic work environments and ensures consistent performance in the monitoring process.

The remainder of this paper is organized as follows. Section 2 presents related research on CV techniques for construction safety monitoring and existing methods for reducing false alarms during real-time monitoring. Next, Section 3 describes the proposed methodology and algorithms for implementing safety rules in real-time monitoring, incorporating temporal analysis to reduce false alarms. Then, Section 4 outlines the experimentation

details, the results of the proposed methodology, and the performance of the algorithms. Subsequently, Section 5 discusses the proposed method and its uniqueness and limitations. Finally, Section 6 summarizes the significant findings of the proposed technique and suggests avenues for future research.

2. Literature Review

2.1. Computer Vision Techniques for Construction Safety Monitoring

The integration of CV technology is driving a significant transformation in the field of construction safety management [35,36]. With the advent of deep learning, opportunities for CV-based data analysis have emerged, offering solutions to challenges associated with the manual observation and recording of unsafe behaviors. Researchers and industry professionals recognize the considerable potential of CV systems for conducting safety inspections and monitoring at construction sites [11,37]. This recognition has increased research activities aimed at exploring various methods and applications tailored to monitoring construction sites. Despite acknowledging their potential, the literature emphasizes the limited practical deployment and application of these techniques within the dynamic environment of construction sites [23]. This limitation underscores the necessity for further exploration and development in this area. This section highlights this gap by providing an overview of the various CV techniques and methodologies that have been proposed for construction safety monitoring.

Fang et al. [17] introduced a deep learning-based detector capable of identifying individuals without proper helmet protection. In another study, Fang et al. [18] focused on enhancing worker safety in aerial environments, detecting individuals with and without helmets, harnesses, and anchors, whether lined or not lined with webbing. Further, Huang et al. [19] and Han et al. [20] worked on detecting individuals wearing helmets. Further advancements have been made to detect diverse types of PPE. For example, Hung et al. [21] successfully detected PPE items, such as hard hats, shirts, gloves, belts, pants, and shoes. Furthermore, Wu et al. [22] extended the scope by identifying assorted colors of hard hats. These endeavors reflect the increasing importance of CV-based systems in promoting workplace safety and the proactive measures taken by researchers to address various aspects of PPE compliance.

Researchers have focused on PPE detection and proposed techniques to enhance worker safety through compliance with safety regulations regulated by the Occupational Safety and Health Administration (OSHA) or the Korean OSHA (KOSHA) for other construction equipment operations. For instance, Anjum et al. [38] presented a technique to check the safe working height of workers using A-type ladders. In addition, Fang et al. [39] introduced an automatic CV approach using the mask region-based convolutional neural network (Mask R-CNN) to detect individuals traversing structural supports during construction projects to identify unsafe behavior and prevent potential falls from heights. Moreover, Khan et al. [16] proposed a correlation-based approach for mobile scaffold safety monitoring in the construction industry, using the Mask R-CNN to identify and detect safe and unsafe worker behaviors.

In the construction domain, CV-based techniques are commonly employed across various tasks, including image classification, object detection [17], object segmentation [40], and pose estimation [41]. Image classification involves categorizing images into predefined classes or categories, enabling the recognition of specific objects, scenes, or activities relevant to construction safety. For example, Seong et al. used classification techniques to provide an evaluation of safety vest detection using color information in construction-site images [42]. Object detection aims to locate and classify objects within an image or video frame. This technique is employed to detect equipment, machinery, workers, or potential hazards in real-time [17,22]. For instance, Fang et al. utilized Faster R-CNN to detect workers and identify harnesses for safety during falls from heights [43]. Wang et al. used surveillance cameras to track and classify workers and equipment using a deep region-based convolutional neural network (R-CNN). Subsequently, trajectories were derived

from another CNN-based model to analyze spatial-temporal relationships and identify danger zones [44].

Instance segmentation involves locating specific objects or regions of interest within an image. Few studies in construction have used segmentation techniques or drawn polygons when training the model on real-world data. This method achieves high tracking accuracy and precision, demonstrating its effectiveness in tracking multiple workers despite occlusions and scale variations. For instance, Xiao et al. proposed a vision-based method for tracking workers in off-site construction that integrates the Mask R-CNN algorithm to apply instance segmentation and the Kalman filter to accomplish instance association [40]. Similarly, Khan et al. used this technique for safety-rule correlation for mobile scaffold monitoring [16].

Pose estimation refers to estimating the spatial orientation or pose of objects or individuals within an image or video frame. Establishing the pose estimation of objects (e.g., machines or workers) poses challenges due to such factors as viewpoint, illumination, and contextual backdrop, which introduce noise. Different action recognition techniques were established for recording construction worker actions [41]. Yang et al. introduced a scene-parsing system using semantic information to enhance the action recognition of workers [45]. Although CV-based worker monitoring systems have been deployed across various construction-site scenarios, employing various techniques for multiple tasks, the effectiveness of these methods diminishes with environmental changes and variations in training data specific to certain conditions, leading to missed and false detection [46].

2.2. Reducing False Alarms during Real-Time Monitoring

False detection, false positives, and missed detection contribute significantly to false alarms in real-time monitoring systems. Achieving accurate detection relies on developing precise and generalized models capable of effectively identifying target objects [47]. Borowski et al. addressed the challenge of high false-alarm rates within intensive care unit monitoring systems, primarily stemming from irrelevant noise and outliers in the time series of the sensor data. Their study introduced two online signal filters based on robust repeated median regression within moving windows of varying widths. These filters aimed to differentiate relevant signals from noise and outliers in real time, enabling comparisons between signal estimations and alarm limits rather than raw measurements [48].

Similarly, Yu et al. addressed the problem of false alarms in fire detector sensors within structures by introducing a multidetector fire detection model. This model leveraged asynchronous spatiotemporal signal similarity among detectors, calculating correlation coefficients using the Pearson-derivative dynamic-time-warping method. Furthermore, they proposed a calculation rule for the correlation coefficients of the multidetector signals and determined alarm threshold values using the support vector classification algorithm. The model provided early fire warnings by constructing a real-time detection model employing these correlation coefficients [49].

The integration of camera sensors with IoT sensors is also a notable technique presented in the literature. For instance, Sudhakar et al. addressed the ecological degradation caused by forest fires by employing uncrewed aerial vehicles (UAVs) for continuous monitoring and fire hotspot detection. Their method focused on reducing false alarms by developing and implementing reliable and accurate forest fire detection algorithms explicitly tailored to UAVs. This approach involved enhancing signal processing techniques, optimizing sensor data fusion, and improving algorithm adaptability to diverse environmental conditions [50]. However, this study has numerous limitations, such as outdoor navigation obstacles, cost, safety concerns associated with UAV testing, and difficulties in detecting defined marks for navigation, which posed significant hurdles. Additionally, inaccuracy in width estimation influenced by the pitch angle precision was also encountered.

Moreover, Talaat et al. introduced the smart fire detection system, which employed the improved YOLOv8 algorithm for real-time fire detection in smart cities. This system aimed to enhance accuracy, reduce false alarms, and offer cost-effective scalability for

detecting various urban hazards. Their framework integrated fog, cloud, and IoT layers and facilitated rapid data processing and responses to mitigate property damage and safeguard lives during fire emergencies [27].

However, the integration of IoT sensors and CV for decision-making can cause delays and may be unsuitable for real-time monitoring. Thus, some researchers have opted to rely solely on camera sensors for real-time monitoring using CV techniques. For instance, De Venâncio et al. emphasized the significance of early fire detection and highlighted the limitations of human-based surveillance in open areas. They proposed an automatic fire detection method combining spatial (visual) and temporal patterns, leveraging convolutional neural networks (CNNs) [51].

In addition, Abdulghafoor et al. addressed challenges encountered by surveillance systems in real-time object detection and tracking, presenting an algorithm to overcome these obstacles. By integrating the principal component analysis and deep learning networks, the algorithm efficiently detected multiple moving objects in natural scenes. Its adaptability between the two approaches optimizes performance, as demonstrated by the experimental results achieving superior detection and classification accuracy compared to existing systems. This approach promises advancements in security and surveillance applications [52].

In addition, CV researchers have optimized deep learning models by modifying their architecture and optimizing the hyperparameters through automated machine learning to reduce false alarms and false positives. For example, Zhu et al. optimized the head of an object-detection model to enhance accuracy and generalizability [28]. Their optimized head effectively classifies and localizes small objects.

Similarly, Chen et al. employed transformers for object detection alongside a residual network (ResNet-101), resulting in improved detection accuracy and a missed detection rate reduced by up to 3.1% [30]. In another study, Chen et al. proposed an attention mechanism-based deformable convolution to enhance the feature pyramid network, achieving a detection accuracy of 87.9% for complex scenes [29]. Although these studies have demonstrated enhanced detection accuracy and lower missed and false detection rates, the models still lack sufficient generalization and produce false alarms.

Temporal analysis in object detection involves analyzing the changes and movements of objects over time in frame or video sequences [53]. To mitigate false detections and false alarms, researchers have employed temporal-analysis techniques. For example, Kong et al. employed logistic regression for the classification of scenes with and without fire and applied temporal smoothing to reduce false-alarm rates [32].

Similarly, De Venâncio et al. proposed a two-dimensional deep CNN that integrated object detection with tracking to analyze temporal behavior and decrease false alarms from objects, such as clouds and car lights. Their approach reduced the 60% false positive rate [33]. Temporal-analysis methods and false-alarm reduction techniques are primarily applied in fire detection systems.

Scarce studies have been conducted on reducing false alarms in the construction domain. Wang et al. presented an unsafe-proximity detection model focused on minimizing false alarms in construction sites. By considering the position, heading, and speed attributes, the model achieved accurate identification through a state tracking and safety-rule module. Evaluation through simulation and field experiments demonstrated promising results, indicating the potential for enhancing construction safety and mobility while reducing false alarms and disruptions [26].

Additionally, Chow et al. proposed an anomaly-detection approach for the inspection of concrete defects in civil infrastructure. This approach integrated anomaly detection, extraction, and defect classification, significantly reducing the search space for defects by at least 60% with an average hit rate of up to 88.7% and a false-alarm rate of up to 14.2% [54]. However, occlusion or object overlapping further complicates accurate detection [55].

Construction sites encompass a diverse spectrum of worker activities and scenarios, which can be broadly categorized into two distinct types: static and dynamic. In static

scenarios, the work environment and nature of the activities remain relatively stable and predictable. Construction workers engage in tasks that demand precision and unwavering attention, with constant, uniform conditions over extended periods. For instance, we consider the scenario of a worker operating on a mobile scaffold. In such a static scenario, a monitoring system primarily focuses on inspecting and ensuring the structural integrity of the mobile scaffold, including such aspects as the presence of outriggers [16]. However, in dynamic scenarios, such as scenarios where workers consistently wear required PPE, reliability becomes crucial. Dynamic scenarios at construction sites are characterized by a constant state of flux, demanding workers to adapt and frequently relocate. While recent advanced techniques can initially accurately detect hard hats, maintaining this accuracy over time presents challenges. Instances of missed or false detection can result in frequent false alarms, compromising reliability and disrupting operations. An exemplary instance of a dynamic scenario involves the continuous surveillance of workers to ascertain their strict adherence to safety protocols while actively employing a mobile scaffold. This heightened level of surveillance becomes essential because workers may need to adjust or temporarily remove their safety gear. This paper addresses the dynamic scenario of wearing a hard hat as a case study to evaluate the proposed system.

3. Research Method

This study employs a comprehensive approach to real-time construction safety monitoring, integrating both CNN-based single-stage object-detection models such as the YOLO series [56], and temporal-analysis techniques, specifically the sliding window approach [57], to enhance object detection and mitigate false alarms. The primary objective of this method is to automatically identify the scene as safe or unsafe by ensuring compliance with safety rules and regulations. The research method is depicted in Figure 1. The initial stage involves the conceptualization of the study. This includes identifying insufficient safety measures and conducting an exhaustive review of existing literature to pinpoint gaps and evaluate current problem-resolution methodologies. For preliminary validation, a straightforward use case involving PPE checks—specifically, verifying the use of hard hats by workers—is examined. Following this, a dataset is compiled and utilized to train a CV model.

A CV-based safety monitoring process consists of two core modules: an object-detection module, deploying algorithms to efficiently identify and track diverse objects and individuals at the construction site, and a rule-compliance module, assiduously enforcing safety protocols by evaluating the collected data or frames from real-time streaming against pre-established rules. A meticulous sequence of steps is adhered to for the object-detection module to establish an effective real-time monitoring system. This process encompasses data collection and preparation, including annotation and preprocessing, followed by model training, rigorous validation, and deployment. After validating the performance, the trained model is seamlessly integrated into the monitoring system, enabling immediate detection of safety violation (as safe or unsafe scene) in real time and facilitating the generation of alerts or alarm to immediate corrective action. In addition, the temporal-analysis module is added to the existing monitoring system, using a unique temporal-analysis technique to reduce false alarms. In this context, a hypothesis is formulated to determine the difference in accuracy of the monitoring system in generating false alarms with and without integrating the proposed temporal-analysis module.

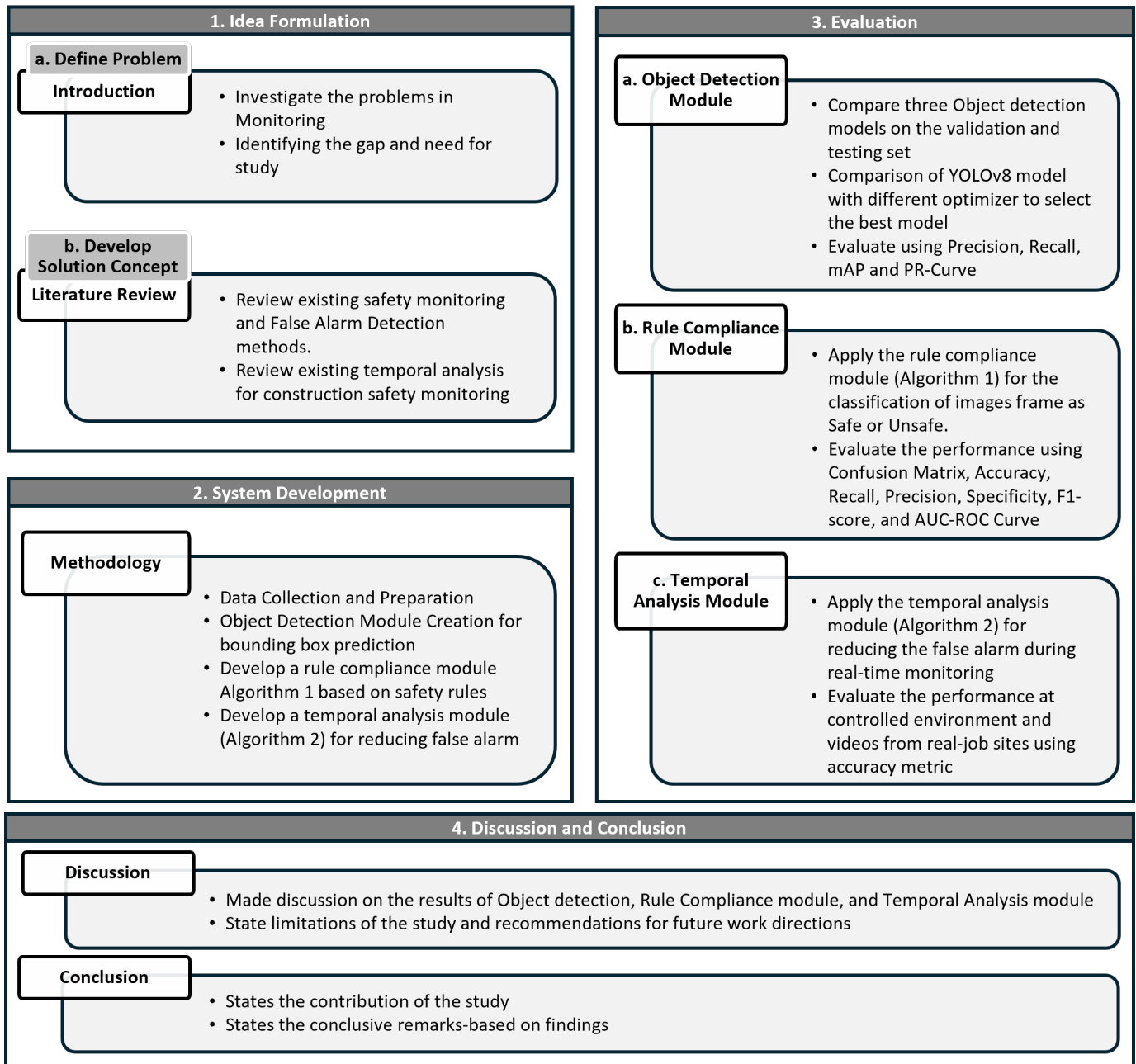


Figure 1. Research method.

3.1. Data Collection and Preparation

The dataset encompasses a comprehensive repository of 5903 images depicting a myriad of instances involving safety gear objects. These objects were systematically classified into five classes: person, hard hat, boots, vest, and robodog. A heterogeneous approach was employed to acquire images from various sources (e.g., job sites, and publicly accessible dataset [58]) in the compilation of data. The dataset adeptly captures authentic scenarios prevalent in diverse occupational settings. The dataset was partitioned into distinct subsets, with 80% allocated for training, 10% for validation, and 10% for testing to facilitate model training and evaluation because the 80:10:10 data split ratio is the best for optimizing the learning [59]. Furthermore, various studies have used the stated data split ratio for training deep learning models and dataset construction [60–62]. This multifaceted strategy was instrumental in infusing the dataset with a broad spectrum of contextual diversity. Significantly, the annotations for each object in the dataset were carefully generated using

Roboflow, incorporating bounding box annotations to enhance the utility of the dataset for machine learning tasks.

The training subset was augmented by applying a diverse array of augmentation techniques to enhance the diversity and resilience of the dataset. These techniques comprise various operations, such as horizontal flipping, saturation and brightness adjustments, exposure variations, blur, noise addition, controlled darkening, and shear transformations. These augmentation strategies enrich the training process by introducing deviations in lighting conditions, spatial perspectives, and intrinsic parameters commonly encountered in real-world scenarios. This augmentation initiative yielded a substantial three-fold expansion of the original training dataset scale. The augmented dataset encapsulates a broader range of variations, contributing to the robustness and adaptability of the machine learning model trained on these augmented data.

Furthermore, a testing set comprising 214 images was collected from online data (<https://universe.roboflow.com/universe-datasets/hard-hat-universe-0dy7t/dataset/26>, accessed on 4 December 2023) for classification. Subsequently, each picture was manually examined and classified as either safe or unsafe to establish the ground truth during the selection of the images. Also, some images with partial objects at the border of the image were removed because these images are not suitable for fair classification. Thus, the final classification dataset contains 111 safe images and 103 unsafe images.

3.2. Object-Detection Module

After preparing the dataset, the subsequent stage involves training the detection model. These detection models are categorized into two categories according to the architecture: single- and two-stage detectors. The domain of two-stage detectors encompasses prominent algorithms, such as the R-CNN [63], Fast R-CNN [64], Faster R-CNN [65], and Mask R-CNN [66]. In contrast, one-stage detectors, exemplified by the YOLO series [67] and the single-shot multibox detector [68], employ a single CNN for predicting class labels and positional offsets without necessitating proposal generation. One-stage detectors are oriented toward real-time object detection, prioritizing swift inferences over attaining maximal detection precision [56].

Among the real-time object detectors, YOLO emerges as a preeminent selection due to its lightweight network architecture, adept feature fusion techniques, and improved detection performance. Notably, YOLOv5 and YOLOv7 have garnered extensive adoption for their efficacy in real-time and resource-efficient object detection tasks. However, YOLOv5 may exhibit constraints in detecting diminutive objects and densely clustered object scenarios [69]. Conversely, the performance of YOLOv7 might be susceptible to degradation attributed to various factors, including data availability, model architecture intricacies, and hyperparameter settings [70]. Ultralytics introduced YOLOv8 in 2023, aiming to combine the strengths of various real-time object detectors [71]. It offers excellent extensibility and achieves a 1% higher accuracy than that of YOLOv5, making it the most accurate detector to date [70]. The comparison of YOLOv5, YOLOv7, and YOLOv8 is shown in Table 1.

In addition, YOLOv8 has emerged as a potent and versatile object-detection model with broad applications. The architecture of YOLOv8 comprises two principal components: the backbone and the head. The backbone extracts features from the input image [72]. Notably, YOLOv8 introduces modifications to the YOLOv5 backbone architecture. The conventional C3 module of YOLOv5 is replaced with the C2f module, and the initial convolutional layer employs a 3×3 kernel configuration, diverging from the prior 6×6 specification [27,73].

Conversely, the head component predicts essential parameters, including bounding boxes, objectness scores, and class probabilities relevant to objects identified within the image. A feature that sets YOLOv8 apart is its deliberate adoption of the anchor-free detection paradigm [27]. This approach eliminates the necessity of a priori anchor box definitions and instead directly predicts the central coordinates of the detected objects [74].

This approach deviates from the conventional approach of calculating offsets with respect to predefined anchor boxes. This paradigm shift in the detection methodology offers a two-fold advantage. First, it significantly reduces the magnitude of box predictions, mitigating computational complexity [75]. Second, it enhances the efficacy of the subsequent nonmaximum suppression phase, an intricate postprocessing procedure tailored to refining candidate detection in inference [76].

Table 1. Comparison of YOLOv5, YOLOv7, and YOLOv8 [24,56,69,72,77,78].

Feature	YOLOv5	YOLOv7	YOLOv8
Backbone	CSPDarknet53	Extended efficient layer aggregation network (E-ELAN)	Modified CSPDarknet53 backbone (C2f module replaces the CSPLayer used in YOLOv5)
Architecture	Efficient and Lightweight	More complex with additional layers	Optimized for better accuracy and speed
Detection	Anchor-based	Anchor-based	Anchor-free
Loss Function	<ul style="list-style-type: none"> Cross Entropy Loss CIoU (Complete Intersection over Union) Loss 	Focal Loss	Focal Loss
Small Object Detection	No	No	Yes (YOLOv8 solves the occlusion issue and small object detection, introduces a new augmentation technique for improving the overall detection accuracy)
Dynamic Environment Adaptation	Good for real-time performance, robust to variations	Better adaptation with enhanced network design	Superior adaptability and generalization across environments

The exploration of object-detection model training has emphasized its critical role in achieving accuracy and efficiency within detection systems. A profound analysis of real-time object detectors has illuminated the strengths and limitations of prominent models, with particular attention paid to YOLOv8. The selection of YOLOv8 as the methodology for the detection framework is due to its compelling attributes, including adept feature fusion techniques, and superior detection performance [27]. As revealed in the following section, YOLOv8 holds great promise for achieving precise and standard-compliant object detection within occupational safety contexts.

3.3. Rule-Compliance Module

This module is designed to ensure compliance with the safety rules and regulations of the OSHA of various countries. After object detection, the bounding box coordinates of the detected objects can be used to check for rule compliance and conduct safety assessments. Additionally, other CV postprocessing techniques and depth-estimation techniques can be applied at this step for safety-rule compliance. Previous studies have proposed vision-based methods for ensuring construction worker safety. Ahmed et al. [79] performed PPE detection; however, the safe and unsafe scene classification was not conducted to check rule compliance, requiring manual monitoring to identify individuals not wearing helmets. Similarly, Gallo et al. [80] proposed a smart system for detecting PPE, such as helmets and vests, without performing classification.

Classifying the scene as being compliant with safety rules (safe) or noncompliant (unsafe) is crucial for monitoring and triggering alarms in hazardous situations. Some researchers have addressed rule compliance; for instance, Isailovic et al. [81] developed an algorithm for head-mounted industrial PPE compliance using deep learning. However, it was evaluated only in a laboratory environment with one person in one frame. Moreover, Lee et al. [82] developed an algorithm for monitoring the wearing of personal equipment

on construction sites, checking whether the hard hat and vest are within the worker area. The algorithm classifies the situation as safe or unsafe accordingly.

In this study, the assessment of worker behavior at a construction site centers around the crucial criterion of hard hat compliance. The methodology employs an object-detection module that independently identifies workers and hard hats within the construction-site environment. Determining whether a worker is wearing a hard hat depends on the precise coordinates of each detected object, adhering to a coordinate-based system. The functionality of the algorithm, detailed in Algorithm 1, encapsulates this process. The algorithm involves detecting and categorizing objects in the input frame, drawing bounding boxes around them, and examining the coordinates of each object in a coordinate-based system. The scenarios for evaluating hard hat compliance are depicted in Figures 2 and 3.

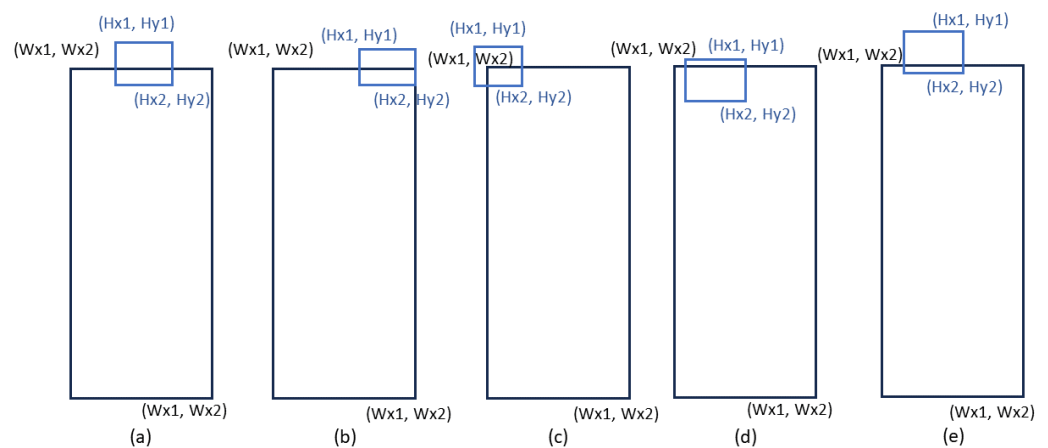


Figure 2. Scenarios for checking wearing hard hats with a coordination-based system when the full-length person bounding box is detected. The subfigures (a–e) show the various positions of the hardhat intersecting with the upper line of the person’s bounding box, confirming that the hardhat is worn.

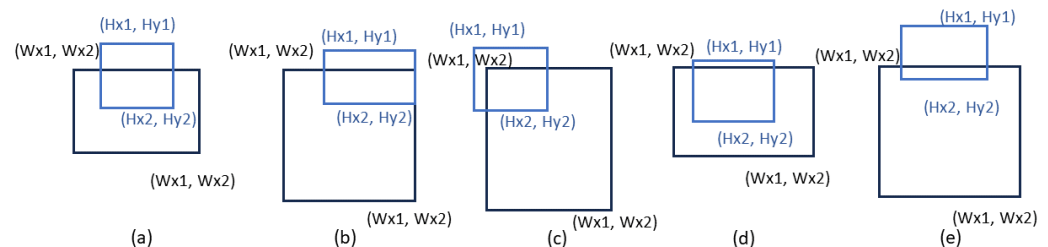


Figure 3. Scenarios for checking wearing hard hats with a coordination-based system when a partial person bounding box is detected. The subfigures (a–e) show the different positions of the hardhat’s bounding box detected when a partial person is detected.

Initially, Algorithm 1 assumes that every worker is detected without wearing a hard hat, and the flag i is set to *False*. The algorithm then calculates the size of the hard hat and worker using Equations (1) and (2):

$$\text{size_of_worker} \leftarrow W^{y2} - W^{y1} \quad (1)$$

$$\text{size_of_HardHat} \leftarrow H^{y2} - H^{y1} \quad (2)$$

After calculating the sizes, the algorithm determines whether the worker size is twice the size of the hard hat, confirming the cases shown in Figure 2. If the condition is not satisfied, the algorithm indicates that the worker is occluded or the bounding of the worker is partially drawn, as depicted in Figure 3. If the first scenario is confirmed, the algorithm checks the conditions ($W^{x1} < H^{x1}$ and $W^{x2} > H^{x2}$ and $(W^{y1} + W^{y2})/2 > H^{y2}$). If this condition is true, it confirms that the detected worker is wearing the hard hat, and flag i

changes to true. Otherwise, flag i remains false. In the second case, the algorithm checks the conditions ($W^{x_1} < H^{x_1}$ and $W^{x_2} > H^{x_2}$ and $(W^{y_1} + W^{y_2})/2 > H^{y_1} + H^{y_2}/2$). If true, the worker is wearing the helmet, and flag i is set to true. Otherwise, flag i remains false.

Additionally, the algorithm can classify unsafe scenarios, as depicted in Figure 4, such as a hard hat in the middle of the worker, a hard hat in the hand, or a hard hat not appearing in the head area of the worker. In these cases, the algorithm classifies the situation as the worker not wearing a hard hat, which is an unsafe event.

Algorithm 1 Rule-Compliance Module

```

1: Input: model (PPE Trained Model) and F (Frame or image)
2: Output: F' (detected frame) and safety_status (Safe or Unsafe)
3: function RULECOMPLIANCEMODULE(model, F)
4:   Step 1: Load the image
5:   Step 2: Initialize variables:
6:     hats  $\leftarrow$  [] ▷ empty list to store hat details
7:     Workers  $\leftarrow$  [] ▷ empty list to store person details
8:     safety_status  $\leftarrow$  Safe
9:   Step 3: Inference the model:
10:    recognitions  $\leftarrow$  model(F) ▷
    pass the image into a loaded model to obtain detection results
11:   Step 4:
12:   for each res in recognitions.getItem() do:
13:     label  $\leftarrow$  res.label_name
14:     bounding  $\leftarrow$  res.bounding_box_info
15:     if label == "Hat":
16:       hats.add(res)
17:     else if label == "Worker":
18:       Workers.add(res)
19:     F'  $\leftarrow$  drawBBox(F, label, BBox) ▷ draw bounding box on image
20:   end for
21:   Step 5: Logic to check worker with helmet:
22:   for each W in Workers.getItem() do:
23:     i  $\leftarrow$  false ▷ flag variable to check worker with helmet
24:     W_bounding  $\leftarrow$  W.bounding_box_info
25:     for each H in hats.getItem() do:
26:       ▷ Calculate the size of the helmet and person in pixels
27:       size_of_worker  $\leftarrow$   $W^{y_2} - W^{y_1}$ 
28:       size_of_helmet  $\leftarrow$   $H^{y_2} - H^{y_1}$ 
29:       ▷ Checking if a worker is wearing a helmet and the worker's body is
       occluded or partial Worker shown
30:       if size_of_worker > size_of_helmet  $\times$  2 then
31:         if  $W^{x_1} < H^{x_1}$  and  $W^{x_2} > H^{x_2}$  and  $\frac{(W^{y_1} + W^{y_2})}{2} > H^{y_2}$  then
32:           i  $\leftarrow$  true ▷ per instance (worker) with a helmet
33:         end if
34:       else
35:         if  $W^{x_1} < H^{x_1}$  and  $W^{x_2} > H^{x_2}$  and  $\frac{(W^{y_1} + W^{y_2})}{2} > \frac{(H^{y_1} + H^{y_2})}{2}$  then
36:           i  $\leftarrow$  true ▷ per instance (worker) with a helmet
37:         end if
38:       end if
39:     end for ▷ Check flag variable status of each instance of Worker
40:     if i == false then
41:       {safety_status  $\leftarrow$  "UnSafe"}
42:     end if
43:   end for
44:   return F', safety_status
45: end function

```

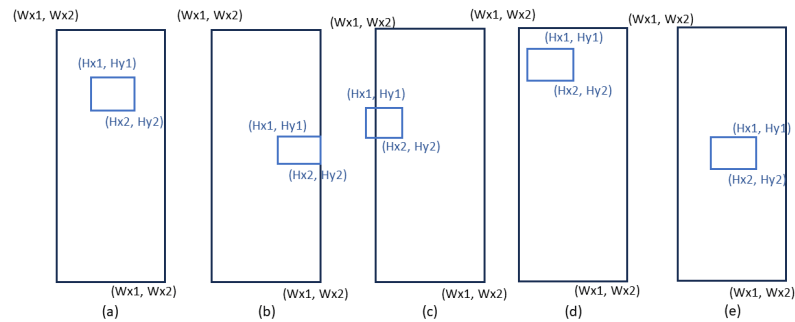


Figure 4. Unsafe scenarios for checking for wearing hard hats with a coordination-based system when the full-length bounding boxes of the person and the hard hat are detected. (a) When the hardhat is overlapped by the person’s bounding box and is positioned above the midpoint of the person but does not intersect with the upper line of the person’s bounding box. (b) The hardhat is positioned at the midpoint of the bounding box and intersects with the right side of the person’s bounding box. (c) The hardhat is above the midpoint of the person’s bounding box but partially intersects with the person’s bounding box. (d) The hardhat bounding box is near the upper line of the person’s bounding box and completely overlapped. (e) The hardhat is completely overlapped and positioned at the midpoint of the person’s bounding box.

3.4. Temporal-Analysis Module for Real-Time Monitoring

Real-time monitoring systems require object detection and scene classification for each frame or image. Relying solely on object detection or segmentation in real-time monitoring requires manual intervention for decision-making and action. However, categorizing a frame as depicting a safe or unsafe scene involves postprocessing, as discussed in Section 3.3. This postprocessing automates decisions and initiates actions. Additionally, achieving 100% accuracy in classification for each frame is challenging or impossible, leading to false alarms because consecutive frames of the same scene may experience misdetection or false detection.

Temporal analysis is a technique that involves examining data and events over time [83]. The output of the rule-compliance module is an event that can be safe or unsafe. To mitigate the false alarms, this study proposes and develops an algorithm to reduce false alarms during real-time monitoring through the temporal analysis of consecutive frames. Specifically, the sliding window approach is adopted for the temporal analysis that involves dividing the time series (or sequence of frames in our case) into overlapping or non-overlapping segments of a fixed length and analyzing each segment separately [57,84].

The algorithm detects every sequential frame and performs classifications based on the rule-compliance module. The algorithm analyzes each frame, maintains streaming results in a buffer, and generates a list categorizing frames’ status as safe or unsafe. The proposed algorithm evaluates the classification results for x sec window. If $z\%$ of frames within this time frame (x s) are classified as unsafe, an alarm is triggered, and the streaming data, including the buffer from the preceding y min, is stored as an unsafe event.

Mathematical Model and Algorithm

We let F_i denote the frames of real-time streaming, R_i is the result of classification for frame i , B_i represents the buffer for temporary event recording, and T denotes the duration of evaluating the classification results (x s). Moreover, z denotes the threshold percentage for classifying frames as unsafe, and N denotes the total number of frames within the time duration T . Further, U is the number of frames classified as unsafe within time T , and $Alarm$ is the binary variable indicating whether an alarm is triggered. The six critical elements of the algorithm are mathematically represented as follows.

Frame analysis and classification: This step performs detection and classification for each frame in video_streaming (Equation (3)):

$$F'_i, status_i = RuleComplianceModule(F_i). \quad (3)$$

Buffer update: After classification, each frame is stored in a buffer for temporal analysis. The results of frame i are stored with the previous buffered frames (Equation (4)).

$$B_i = \text{UpdateBuffer}(B_{i-1}, F_i). \quad (4)$$

Temporal analysis: Initially, the total number of frames N is estimated in time T using Equation (5). Then, the total unsafe and safe frames are calculated using Equations (6) and (7). The symbol \mathbb{I} represents the indicator function, which is used to define a function that takes the value of 1 if a specified condition is true and is 0 otherwise:

$$N = \lfloor \frac{T}{\text{frame_duration}} \rfloor \quad (5)$$

$$U = \sum_{i=1}^N \mathbb{I}(\text{status}_i = \text{"Unsafe"}) \quad (6)$$

$$S = \sum_{i=1}^N \mathbb{I}(\text{status}_i = \text{"Safe"}) \quad (7)$$

$$\text{Alarm} \leftarrow \begin{cases} 1 & \text{if } \frac{U}{N} \geq \frac{z}{100} \\ 0 & \text{if } \frac{S}{N} \geq \frac{z}{100} \end{cases} \quad (8)$$

The total unsafe and safe instances in T are calculated to confirm unsafe and safe events and trigger alarms, and the decision is made according to threshold z (Equation (8)). If $\frac{U}{N} \geq \frac{z}{100}$, it triggers the alarm and starts recording the video for y min, including the frames in T , and resets the buffer and variables after recording the y min video. In contrast, if $\frac{S}{N} \geq \frac{z}{100}$, the buffer and variables are initialized and analyzed against the new frames for time T . The Algorithm 2 presents the overall process and workflow of the temporal-analysis-based real-time monitoring system to reduce false alarms.

Algorithm 2 Temporal-Analysis-based Real-time Monitoring Algorithm

Input: Streaming/Frame sequence $F = \{F_1, F_2, \dots, F_n\}$, Rule-Compliance Module, z threshold, and y is the time duration for recording unsafe event in minutes

2: **function** TEMPORALANALYSIS($model, F, T, z, y$)

Initialize buffer: $B \leftarrow \{\}$

4: $N \leftarrow \lfloor \frac{T}{\text{frame_duration}} \rfloor$

$U \leftarrow 0$

6: $S \leftarrow 0$

for $i \leftarrow 1$ to n **do**

8: $F'_i, \text{status}_i \leftarrow \text{RuleComplianceModule}(model, F_i)$ ▷ Frame Analysis and Classification

$B \leftarrow \text{UpdateBuffer}(B, F'_i)$ ▷ Buffer Update

10: $U \leftarrow U + \mathbb{I}(\text{status}_i = \text{"Unsafe"})$ ▷ Classification Evaluation

$S \leftarrow S + \mathbb{I}(\text{status}_i = \text{"Safe"})$ ▷ Classification Evaluation

12: $\text{Alarm} \leftarrow \begin{cases} 1 & \text{if } \frac{U}{N} \geq \frac{z}{100} \\ 0 & \text{if } \frac{S}{N} \geq \frac{z}{100} \end{cases}$

if $\text{Alarm} == 1$ **then**

14: $\text{Trigger alarm for } 5 \text{ s}$ ▷ Alarm will ring for 5 s as a background process

Store streaming data for y minutes, including a time buffer T from the preceding y minutes, as an unsafe event.

16: **if** y minutes video recorded **then**

$B \leftarrow \{\}$ ▷ Reset Buffer

18: $U \leftarrow 0$ ▷ Reset Unsafe counter

$S \leftarrow 0$ ▷ Reset Safe counter

20: **end if**

else ▷ Reset buffer and other variables because most frames are classified as Safe

22: $B \leftarrow \{\}$ ▷ Reset Buffer

$U \leftarrow 0$ ▷ Reset Unsafe counter

24: $S \leftarrow 0$ ▷ Reset Safe counter

end if

26: **end for**

end function

4. Experimentation, Results, and Evaluation

4.1. Model Training and Experimental Setup

The object-detection model underwent training on a robust system featuring an i9-10900 central processing unit clocked at 2.80 GHz with 10 cores, 32 GB of RAM, and an NVIDIA RTX 3090 graphics processing unit (GPU) with 24 GB of dedicated memory. The YOLOv8 framework employs a suite of optimizers, including Adam, AdamW, AdaMax, NAdam, RAdam, RMSprop, and stochastic gradient descent (SGD) [71]. However, no optimizer is universally applicable to all machine learning tasks. The choice of the optimizer that yields optimal performance depends on the specific dataset, model architecture, and hardware configuration [85,86]. This choice underscores the importance of experimenting with various optimizers to determine the one that best suits the task, highlighting the essential role of experience and the iterative process of trial and error in optimizing machine learning models.

Seven experiments were conducted to achieve this goal, with each experimental session using one of the mentioned optimizers. A consistent batch size of 16 was maintained, and training continued for 300 epochs in each experiment. Furthermore, default hyperparameters, such as the learning rate, momentum, and decay, were fixed during model training. For comparative purposes, the larger YOLOv5 and YOLOv7 models were also trained under similar conditions. The YOLOv5 and YOLOv8 models underwent training with default hyperparameter settings and a batch size of 16 for 300 epochs.

4.2. Evaluation Metrics

As mentioned in the methodology, the proposed method comprises two primary modules: the object-detection and rule-compliance modules. Two distinct validation approaches were employed for each to validate these modules experimentally. This choice stems from the nature of the modules. The first module focuses on object detection, whereas the second module classifies safe and unsafe scenes. In the evaluation of the object-detection module, established evaluation metrics were employed to quantify performance. These metrics encompass the precision (P), recall (R), mean average precision (mAP), and the precision-recall (PR) curve. These metrics are calculated as follows [87]:

$$P = \frac{TP}{TP + FP} \quad (9)$$

$$R = \frac{TP}{TP + FN} \quad (10)$$

$$AP = \sum_{n=0}^1 P_n (R_n - R_{n-1}) \quad (11)$$

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (12)$$

where TP denotes true positive, TN indicates true negative, FP represents false positives, and FN denotes false negatives in Equations (9) and (10). In Equation (11), n is the threshold level belonging to real numbers, and the values are in the range of 0 to 1, whereas N denotes the total number of classes. The average precision (AP) and mAP values are fundamental for object detection, offering a comprehensive view of the ability of an algorithm to identify and localize objects accurately within images [88], accounting for precision and recall trade-offs [89].

In contrast, the AP and mAP were not computed for the rule-compliance module. Typically, these metrics are employed to evaluate the object-detection module. Instead, the evaluation relied on the metrics of accuracy ($Acc.$) and the F1-score ($F1$) to quantify the performance of the rule-compliance module, classifying scenes into safe and unsafe categories. Accuracy is essential for classifying objects or scenes correctly, providing a straightforward measure of overall correctness. The F1-score strikes a balance between pre-

cision and recall, making it valuable when finding the equilibrium between false positives and negatives is critical [87,90].

Furthermore, sensitivity, precision, and specificity were computed to evaluate the classification performance of the rule-compliance module. Sensitivity (also recognized as the recall or true positive rate) ensures the model correctly identifies positive instances. Precision (also known as positive predictive value) ensures the accuracy of the positive predictions, and specificity (commonly referred to as the true negative rate) evaluates the ability of the model to identify negative instances correctly [91]. The precision and recall were calculated using the formulas expressed in Equations (9) and (10). The remaining metrics were calculated as follows [87]:

$$Acc. = \frac{TP + TN}{TP + TN + FP + FN} \quad (13)$$

$$F1 = \frac{2 \cdot P \cdot R}{P + R} \quad (14)$$

$$Specificity = \frac{TN}{TN + FP} \quad (15)$$

4.3. Comparison of YOLOv5, YOLOv7 and YOLOv8

This section assesses the performance of three versions of the YOLO object-detection model: YOLOv5, YOLOv7, and YOLOv8. Although, it is a known fact that the YOLOv8 is better than YOLOv5 and YOLOv7 based on the detection results on MS-COCO dataset which is a huge dataset with 80 classes and 330k images. However, the dataset used in this study is smaller with different characteristics. Furthermore, the effectiveness of the deep learning models can vary significantly depending on the characteristics of the dataset [92,93]. The evaluation is conducted on a validation set and testing set, using an intersection over union threshold of 0.5. Table 2 presents detailed results. The analysis of the results of the validation set reveals several noteworthy observations. First, YOLOv7 exhibited superior performance compared to YOLOv5 and YOLOv8 in terms of precision and recall. Specifically, YOLOv7 achieved a precision of 0.922 and a recall of 0.800, indicating its ability to identify objects accurately and locate a high percentage of relevant objects in the validation dataset. In contrast, YOLOv5 and YOLOv8 achieved slightly lower precision and recall scores; YOLOv5 and YOLOv8 demonstrated mAP scores of 0.859 and 0.867, respectively, on the validation set, signifying a similar overall object detection performance for these two models on this particular dataset.

Table 2. Comparison of YOLOv5, YOLOv7, and YOLOv8 on the validation and testing sets with a threshold of 0.5.

Dataset	Images	Technique	Precision	Recall	mAP@0.5	mAP@0.95
Validation Set	676	YOLOv5	0.899	0.798	0.859	0.583
		YOLOv7	0.922	0.800	0.785	0.523
		YOLOv8	0.908	0.806	0.867	0.601
Test Set	676	YOLOv5	0.803	0.645	0.743	0.440
		YOLOv7	0.855	0.649	0.626	0.361
		YOLOv8	0.836	0.659	0.752	0.450

The evaluation of the testing set provides further insight into the model's capabilities and ability to generalize the model to unseen data. Moreover, YOLOv8 emerged as the top performer on the testing set, surpassing YOLOv5 and YOLOv7 in precision, recall, and mAP. Notably, YOLOv8 achieved a precision score of 0.836, recall score of 0.659, and mAP score of 0.752, demonstrating its robustness and accuracy when applied to new, unseen data. Although YOLOv5 demonstrated superior recall compared to YOLOv8 on the testing set, highlighting its ability to locate relevant objects, its precision was slightly

lower. Further, the performance of YOLOv7 significantly decreased on the testing set compared to the validation set, indicating potential sensitivity to the training dataset and suggesting overfitting.

The observed decline in performance on the testing set, particularly in the case of YOLOv7, raises concerns regarding overfitting. Overfitting occurs when a model becomes overly tailored to the training and validation data, compromising its ability to generalize to new and unseen data. The contrasting performance of YOLOv7 between the two datasets highlights the importance of addressing overfitting problems during model development and training.

4.4. Experimental Validation

This section presents the performance of all three modules of the proposed methods, including the validation of the object-detection module, safety-rule-compliance module, and temporal-analysis-based real-time monitoring module.

4.4.1. Validation of Object-Detection Module

The YOLOv8 model outperformed YOLOv5 and YOLOv7, as illustrated in Table 2. Consequently, YOLOv8 was selected for further experimentation and analysis. Experiments were conducted with different optimizers to determine the best model, as detailed in Table 3. Among the six tested optimizers, YOLOv8 with SGD and YOLOv8 with AdamW demonstrated superior performance on the validation set, achieving a 0.86 mAP. However, YOLOv8 with SGD performed best on the testing set. Table 3 provides further details.

Table 3. Comparison of the YOLOv8 model with optimizers on the validation and testing sets with a 0.5 threshold.

Dataset	Images	Optimizer	Precision	Recall	mAP@0.5	mAP@0.95
Validation Set	676	SGD	0.908	0.806	0.867	0.601
		Adam	0.844	0.737	0.801	0.504
		AdamW	0.876	0.805	0.860	0.546
		AdaMax	0.869	0.801	0.855	0.548
		NAdam	0.862	0.743	0.812	0.513
		RAdam	0.857	0.752	0.815	0.514
Test Set	676	SGD	0.836	0.659	0.752	0.450
		Adam	0.819	0.669	0.731	0.387
		AdamW	0.775	0.697	0.73	0.396
		AdaMax	0.898	0.524	0.719	0.448
		NAdam	0.777	0.652	0.722	0.379
		RAdam	0.845	0.628	0.726	0.394

Adam and AdamW displayed competitive performance in terms of the mAP@0.5 scores, with AdamW achieving the highest recall score on the test set; however, SGD demonstrated superior scores of mAP@0.5 and precision on the testing set. The high precision of the SGD-based YOLOv8 model indicates that a high proportion of the objects it retrieves is relevant. This result suggests the model is good at avoiding false positives, ensuring that the items it returns are primarily relevant. Moreover, the high mAP implies that the model performs well across various levels of recall, indicating a robust performance across the thresholds. Therefore, the results suggest that SGD might be more effective at capturing a higher proportion of true-positive instances in the dataset than Adam and AdamW. However, optimization algorithm selection depends on various factors, including dataset characteristics, convergence speed, and computational resources. The advantage of the SGD over the Adam optimizer in object detection likely arises from the stable convergence [85] (attributed to its fixed learning rate [86]) compared to potentially aggressive updates for the Adam optimizer due to the adaptive learning rates. Therefore, based on these results, the YOLOv8 model trained with the SGD optimizer was best and was selected

for scene classification. Figure 5 depicts the PR curves of the YOLOv8 model trained with various optimizers.

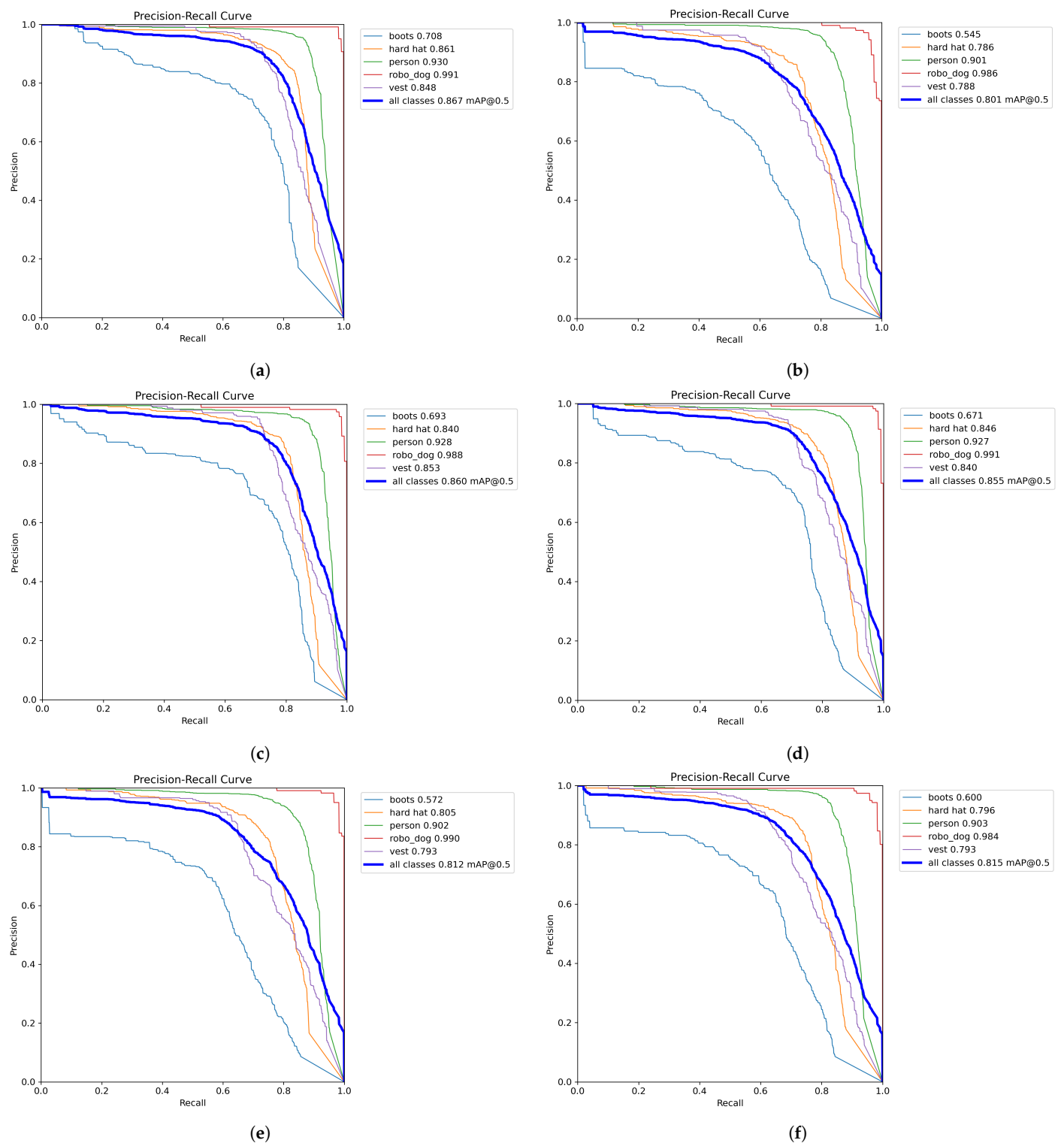


Figure 5. Precision-recall curves of the YOLOv8 model with (a) SGD, (b) Adam, (c) AdamW, (d) AdaMax, (e) NAdam, and (f) RAdam optimizers.

4.4.2. Validation of Rule-Compliance Module

The algorithm presented in Algorithm 1 is applied for scene classification in images, determining whether safety rules are adhered to within the scene. The algorithm categorizes

images as either safe or unsafe based on this criterion. The dataset for classification consists of 214 images, as outlined in Section 3.1. By employing the YOLOv8 detection model with a rule-compliance module, the model achieved high precision scores of 0.93 for the unsafe class and 0.98 for the safe class. These scores indicate that 93% and 98% of predictions for unsafe and safe instances, respectively, were accurate. The recall scores were commendable, with the models correctly identifying 98% of actual unsafe instances and 93% of actual safe instances. The F1-score, the harmonic mean of precision and recall, was equally impressive for both classes, at 0.95. The macro and weighted averages for the precision, recall, and F1-score were consistently high, further affirming model robustness. With an overall accuracy of 95%, the model effectively classified instances, which is crucial for applications where safety is paramount. These findings suggest that the model has promise for practical deployment in scenarios where distinguishing between safe and unsafe conditions is imperative, contributing to enhancing safety measures and risk-mitigation strategies. Figure 6 and Table 4 present the confusion matrix and detailed macro average results, respectively.

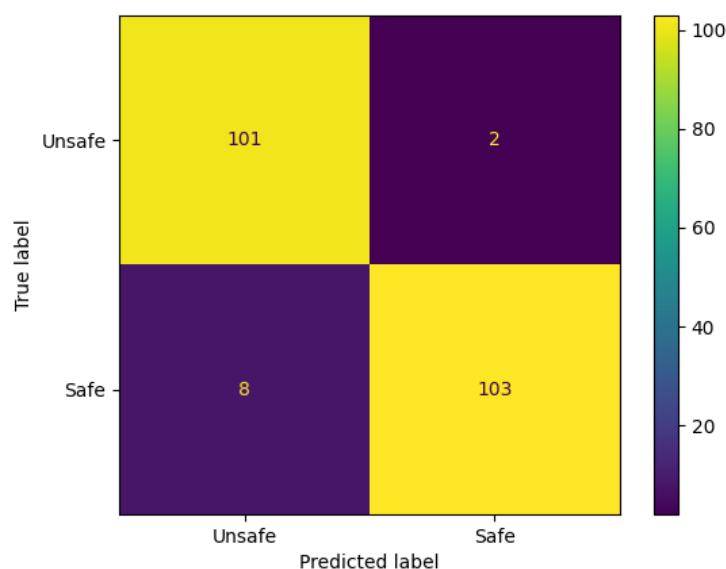


Figure 6. Confusion matrix of scene classification using the rule-compliance module.

Table 4. Macro average scene classification results employing the rule-compliance module.

Class	Precision	Recall	F1-Score	Specificity	Accuracy
Unsafe	0.93	0.98	0.95	0.98	0.95
Safe	0.98	0.93	0.95	0.93	
Macro Average	0.95	0.95	0.95	0.95	0.95

In addition to the classification results, the area under the curve (AUC) for the receiver operating characteristics (ROC) curve was calculated at 0.95. Figure 7 illustrates the AUC of the ROC, indicating that the ability of the model to distinguish between safe and unsafe instances is excellent, with a high probability that the model ranks a randomly chosen positive instance higher than a randomly chosen negative instance. The ROC-AUC visually represents the trade-off between the true (sensitivity) and false positive rates (1 – specificity) across threshold settings. Its availability further underscores the model performance, displaying a curve that approaches the upper-left corner of the plot, indicative of excellent discrimination between the two classes. This combined evaluation reinforces the efficacy and reliability of the model in accurately classifying instances, strengthening its potential for practical deployment in real-world scenarios where safety assessment is critical.

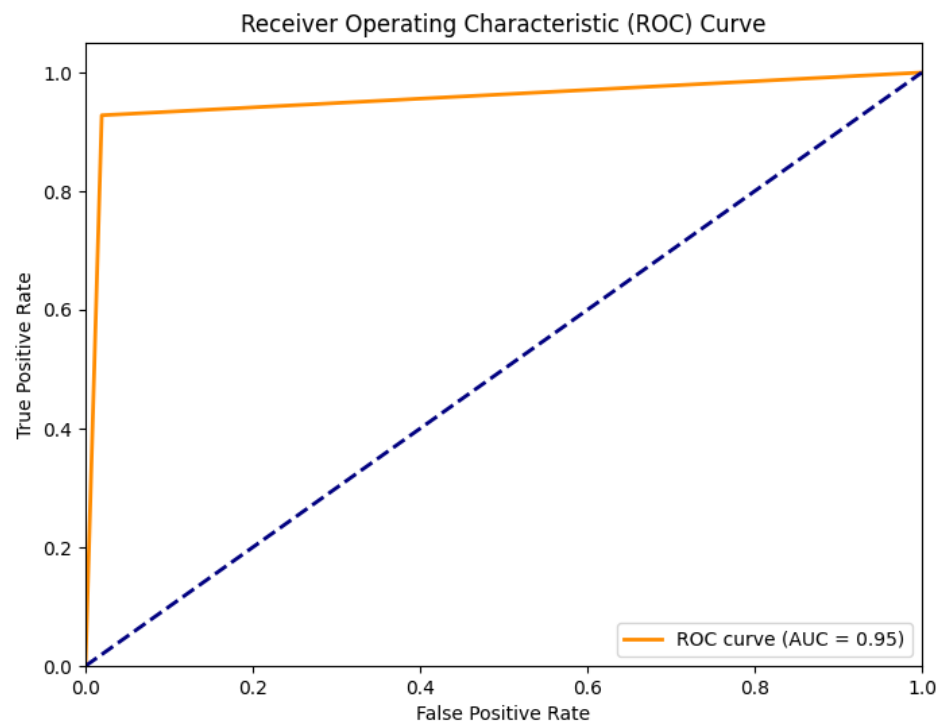


Figure 7. Area under the receiver operating characteristic curve of the YOLOv8 model with the rule-compliance module on an unseen classification dataset.

Figure 8 depicts the visual results of the rule-compliance module. The proposed system employs the rule-compliance module, wherein any violation results in the classification of a frame or scene as unsafe. Workers who are compliant with safety rules, such as those wearing helmets in this case, are indicated with green bounding boxes, signifying safe behavior. Conversely, workers not wearing hard hats (who are violating the rules) are marked with red bounding boxes accompanied by the label unsafe overlaid on the frame or image.



Figure 8. Visual representation of the rule-compliance module outcomes. Compliant workers wearing helmets are indicated by green bounding boxes; noncompliant workers without helmets are marked with red bounding boxes labeled unsafe on the frame.

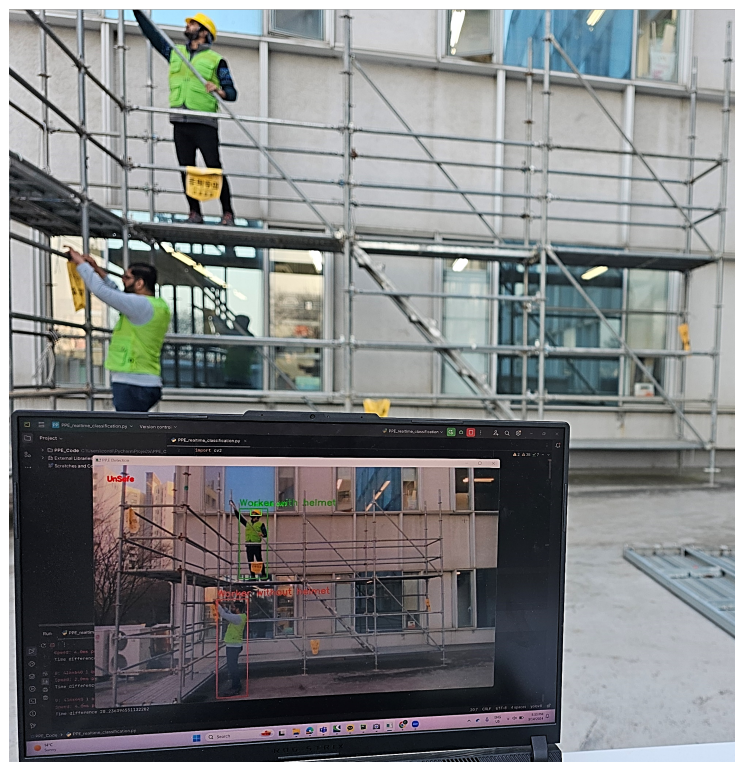
4.5. Evaluation of Temporal-Analysis Module for Real-Time Monitoring

To assess the effectiveness of the temporal-analysis module for real-time monitoring, we conducted experiments in the controlled environment of the Construction Technology

Innovation Laboratory at Chung-Ang University in Seoul, South Korea. For further evaluation, the proposed monitoring system was employed on an actual construction site, where short videos of each category (safe and unsafe) were collected for analysis. The performance of the monitoring system was compared with and without the temporal-analysis module by measuring the number of false alarms generated. For the experiment, we used a laptop equipped with an NVIDIA 3080Ti GPU and 16 GB of dedicated RAM, with live streaming and input facilitated through a closed-circuit television. Figure 9 illustrates the experimental setup and results for real-time testing in the controlled environment.



(a)



(b)

Figure 9. Real-time monitoring: (a) Experimental setup for real-time monitoring and (b) results of the temporal-analysis-based real-time monitoring.

The real-time results demonstrate a notable decrease in false alarms with the implementation of the temporal-analysis module, underscoring its effectiveness, as no false alarms were detected. Conversely, experiments conducted without the temporal-analysis module exhibited instances of missed and false detection, leading to false alarms. Despite varying the threshold z (in Algorithm 2) to 60%, 70%, and 80%, the proposed technique yielded similar results. The selection of threshold z is contingent upon the mAP of the detection algorithm and the accuracy of the rule-compliance module. With the rule-compliance module achieving 95% accuracy on the testing dataset (unseen data), it outperformed across all thresholds of 60%, 70%, and 80%. However, as the number of classes or objects in the dataset increases, the accuracy and mAP of the model decrease [94]. Consequently, for the experiments, we set $z = 70\%$ in the algorithm, where Algorithm 2 triggers the alarm when 70% of the frames within a 5 s window are identified as unsafe, subsequently saving the event video as proof.

For the quantitative analysis, we collected five videos to evaluate the temporal-analysis module and false alarms. Of the collected videos, three were classified as safe and two as unsafe. All safe videos contained safe frames, whereas unsafe videos contained unsafe frames. Two types of ground truth were used: one with and one without the proposed temporal-analysis method for real-time monitoring. Table 5 provides details of the videos and a comprehensive overview of the video dataset for real-time monitoring, comparing outcomes with and without the temporal-analysis module. Each video segment is classified as either safe or unsafe, with corresponding durations measured in seconds and frame rates specified in frames per second (FPS). The total number of frames in each segment is also recorded.

Table 5 is divided into three main sections: one for video data information and two for presenting results with and without the temporal-analysis module. In the section detailing the results without the temporal-analysis module, ground truth alarms and generated alarms are presented for each video segment, alongside the corresponding accuracy percentages. Conversely, in the section covering the results with the temporal-analysis module, the same metrics are provided, revealing the influence of the module on alarm generation and accuracy. The results indicate that the real-time monitoring system without temporal analysis triggered false alarms for Videos 1, 2, 4, and 5. The average accuracy of the real-time monitoring system without the temporal analysis is 97.97%. However, the real-time monitoring system with temporal analysis displayed 100% accuracy, indicating the efficiency of the system. Hence, our hypothesis that the monitoring system with the proposed temporal-analysis module would show better accuracy results compared to the system without temporal analysis is supported by the findings. By showing 2.03% higher accuracy compared to the system without temporal analysis, the monitoring system with the temporal-analysis module demonstrates its effectiveness in enhancing performance and reducing false alarms. While miss and false detection occurred during testing, the proposed temporal-analysis module successfully avoided generating alarms, highlighting its capability to enhance system reliability and minimize unnecessary alerts.

Table 5. Video dataset information for real-time monitoring and results with and without temporal-analysis module.

	Video 1	Video 2	Video 3	Video 4	Video 5	Avg. Acc. %
Classification	Safe	Unsafe	Safe	Unsafe	Safe	-
Duration in seconds	24	14	10	21.25	43.6	-
Fps	30	30	30	24	30	-
Total Frames	720	420	300	510	1308	-
GT Alarms for without TA	720	420	300	510	1308	-
Results without TA	52	419	0	509	32	-
Accuracy %	92.78	99.76	100	99.80	97.55	97.97
GT Alarms with TA	0	3	0	4	0	-
Results with TA	0	3	0	4	0	-
Accuracy %	100	100	100	100	100	100

GT: ground truth, TA: temporal analysis, FPS: frames/s, Avg. Acc.: average accuracy.

5. Discussion

This study proposes an algorithmic framework for the real-time monitoring of construction sites, addressing the challenge of false alarms triggered by inaccurate detection. This study employs a systematic approach that integrates three critical modules (object detection, rule compliance, and temporal-analysis monitoring), focusing on mitigating safety risks and enhancing overall monitoring efficiency to achieve the research objective.

By leveraging a PPE detection model and spatial analysis techniques, the system accurately identifies workers and evaluates whether they are wearing appropriate safety gear, particularly helmets. The rule-compliance module uses the coordinate system, which checks the correlation of bounding boxes between workers and hard hats. This approach adds a layer of sophistication to safety compliance assessment, allowing for more accurate detection of safety gear and consideration of various factors, such as occlusion and partial visibility. The experimental evaluation of the algorithm for the rule-compliance module demonstrated promising results, with high accuracy and reliability in detecting safety violations while minimizing false alarms. The integration of the rule-compliance module significantly enhances the ability of the algorithm to discern genuine safety breaches from false positives, improving the overall effectiveness of the monitoring system.

Moreover, the accuracy of the classification model is intrinsically linked to the precision of the object-detection model, as Algorithm 1 relies on the bounding box information provided by the object-detection model. During the classification process, these bounding boxes are drawn and filtered based on a confidence score threshold. For instance, if an object-detection model operates effectively with a threshold of 0.5, any adjustment to a higher or lower threshold value will influence the classification outcomes, potentially leading to a decrease in the system's accuracy and F1-score. For validation purposes, we employed the use case of hard hat detection, which yielded promising results. However, it is important to note that the accuracy of the model may vary with different datasets, potentially increasing or decreasing. This variability underscores the importance of selecting appropriate thresholds and datasets to maintain high-performance levels in diverse real-world scenarios.

The temporal-analysis module, when tested in controlled environments, reduced false alarms to zero. Without the temporal-analysis module, the average accuracy was 97.97%, with a 2.03% false-alarm rate. The critical strengths of the algorithm include adaptability to environmental conditions and scene complexities. By integrating the rule-compliance module and setting appropriate thresholds, the algorithm effectively identifies unsafe events while minimizing false alarms. The buffering mechanism captures temporal context, allowing the algorithm to discern patterns of unsafe behavior over time. It triggers alarms and stores data for a specified duration when detecting unsafe events, facilitating post-event analysis and compliance verification. The ability of the algorithm to reset buffers and counters after processing each frame sequence enhances its efficiency and adaptability. Its performance in maintaining zero false alarms in controlled environments underscores its practical utility and reliability in real-world applications, especially in safety-critical settings.

Temporal analysis in object detection involves tracking techniques or filtering methods [26]. These methods leverage information from neighboring frames to improve the accuracy and robustness of object detection over time. The proposed temporal-analysis method for real-time monitoring represents a departure from conventional tracking techniques in the domain of object detection and scene classification. The involvement of a rule-compliance module for classifying the frame or scene as safe or unsafe in real-time tracking is challenging and requires more computation. Unlike tracking methods, which inherently prioritize the continuation of identified objects, the proposed method evaluates the collective behavior of objects within a temporal context. By analyzing frames within a 5 s time window (the time window is flexible, but a 5 s window is used in this study for experimentation), we transcend the limitations of tracking, which may falter in scenarios involving rapid object movement or occlusion.

Moreover, the proposed method introduces a novel threshold criterion for event declaration, where an event is classified as unsafe if at least 70% (the threshold is flexible and can be changed, but 70% is used in this study) of the frames within the time window exhibit characteristics indicative of unsafe conditions. This approach enables the identification of potentially hazardous situations with a higher degree of certainty, avoiding the ambiguity often associated with traditional tracking methods. Furthermore, the proposed method incorporates the feature of streaming and recording concurrent with event detection. This functionality facilitates post-event analysis for forensic purposes and serves as a robust means of data validation and verification, augmenting the reliability and transparency of the proposed methodology.

The object-detection model serves as the foundation to evaluate the adherence to safety protocols. The best model can be suitable for other classification tasks and the use cases because the object-detection model is trained on dataset that has various classes such as hardhat, vest, boot, and person. In this study, Algorithm 1 is developed to assess the safety compliance that uses the bounding box information provided by the object-detection model; if a worker is found to be violating any safety rule within a frame, that frame is classified as unsafe. Algorithm 1 can be expanded or modified to include additional safety regulations, enabling its application to various classification tasks or use cases. Then, the updated Algorithm 1 can classify the image as safe and unsafe and check more safety rules according to the use case. Additionally, the proposed Algorithm 2 focuses on reducing false alarms, remains applicable across various use cases, and is the primary focus of this research. Furthermore, Algorithms 1 and 2 can be used with any object detection algorithm because these algorithms only require bounding box information from the results of object-detection model. Its deployment is intended to enhance the reliability of real-time safety monitoring systems.

5.1. Limitations

While the proposed algorithm exhibits excellent promise, several limitations and areas for future research remain. Optimizing algorithms for real-world deployment involves addressing challenges such as data quality dependency, parameter sensitivity, and scalability to complex scenes, particularly in the case of trained computer object detection reliant on extensive, costly, and time-consuming photo datasets [25]. Typically, real-time monitoring requires 30 FPS for high-quality, smooth streaming [95]. One limitation of the study lies in the variability of the FPS performance observed in the temporal-analysis real-time monitoring system, ranging between 19 and 25 FPS (with a 640×640 image size) during the experiments. The achieved FPS was notably influenced by such factors as the resolution of the input image and the number of detected objects in the scene. Notably, scenes with fewer objects maintained a relatively high FPS, whereas the FPS decreased as the number of objects and frame complexity increased. This variability highlights the challenge of maintaining consistent real-time performance across diverse construction-site scenarios, affecting system reliability and its effectiveness in dynamic environments.

Figure 10 illustrates the results of the missed detection and classification. In Figure 10a, a worker without a hard hat is not detected due to occlusion. However, the scene classification remains accurate because another worker without a hard hat is detected in the frame. Figure 10b depicts a worker mistakenly classified as wearing a hard hat, likely due to the detected hard hat of an occluded worker behind the worker. Figure 10c depicts a worker with an occluded and partially visible hard hat, resulting in the misclassification of the person and the scene as unsafe. These types of false detection and classification occur because conventional cameras lack depth perception, leading to the bounding box of the hard hat being associated with the worker. These limitations highlight the challenges in deploying the model in real construction domains. Similarly, detecting occluded objects remains a significant challenge in CV, representing a limitation of the current object-detection modules.

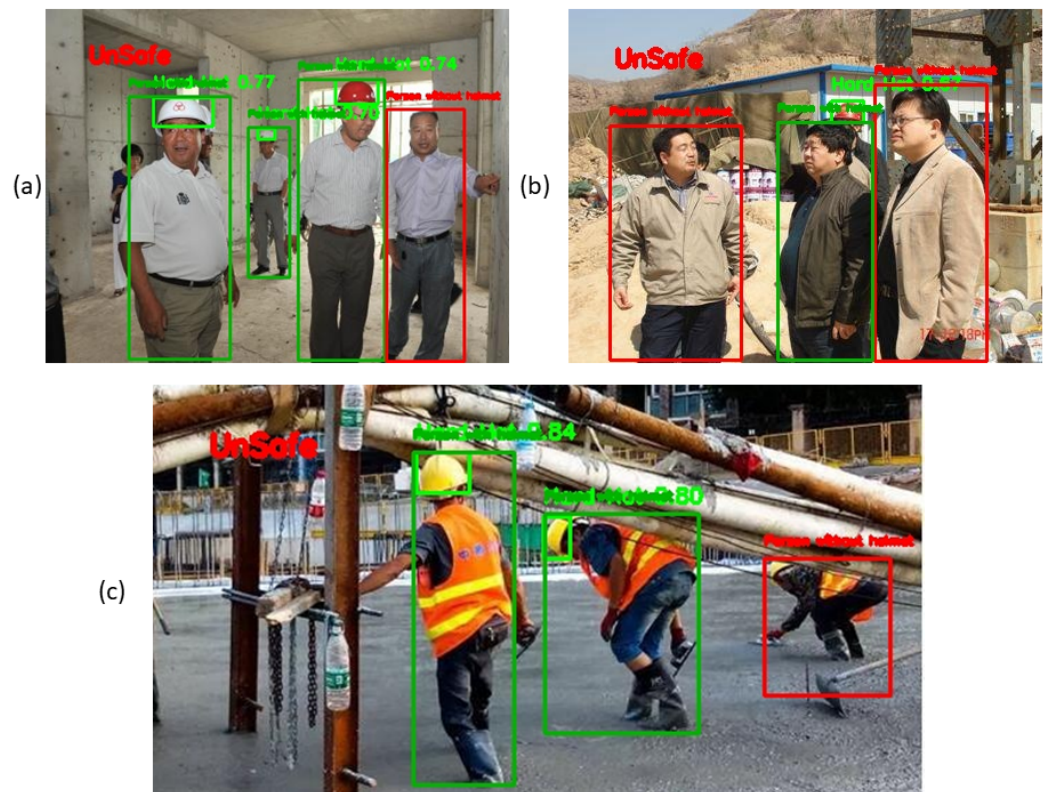


Figure 10. Missed and false detection and classification results. (a) Missed detection of the occluded person in the image. (b) Wrongly classified worker with green bounding due to the detected hard hat of the occluded worker. (c) Wrong scene classification due to the missed detection of a hard hat.

5.2. Recommendations

This research lays the foundation for further exploration and development in real-time construction-site monitoring. Ongoing research focuses on leveraging depth camera solutions and multicamera tracking techniques to overcome the challenges of occluded objects. Furthermore, there is a pressing need for research into more advanced algorithms capable of detecting small and occluded objects with higher accuracy and reliability. Additionally, further validation and testing in diverse construction-site environments is essential to ensure robustness and reliability across scenarios.

Additionally, optimizing performance for sustained real-time monitoring capabilities remains a crucial focus. Furthermore, exploring the integration of blockchain technology has promise, particularly regarding its potential to leverage system data for worker performance evaluation. This data-driven approach could inform the development of incentive schemes, where workers are rewarded based on their performance, enhancing productivity and safety practices. Future work may involve expanding the scope of the algorithm to encompass additional safety parameters and engaging industry stakeholders for real-world validation and deployment.

6. Conclusions

This study addresses the necessity of precise and reliable real-time monitoring systems in dynamic construction environments, with a paramount focus on ensuring worker safety. Through the introduction of a novel algorithmic framework grounded in temporal analysis, this research enhances real-time monitoring capabilities, particularly in verifying compliance with hard hat usage as a primary use case. The integration of techniques, including object detection, safety-rule-compliance assessment, and temporal analysis, empowers the system to identify safety breaches accurately while mitigating false alarms during real-time monitoring. Experimental validation confirms the effectiveness of the proposed

system for object detection and classification, achieving 95% accuracy and a 95% F1-score. The temporal-analysis for real-time monitoring algorithm reduces the overall false-alarm rate to 2.03%, with zero false negatives, enhancing efficiency.

Theoretically, this study contributes to the body of knowledge by demonstrating how temporal analysis can be effectively combined with object detection and rule compliance assessment to improve real-time monitoring systems. The framework's model-agnostic nature, as evidenced by the proposed Algorithms 1 and 2, allows for integration with any object-detection model, providing a versatile solution that can be adapted and expanded upon with future advancements in object detection technology.

Practically, the validated temporal-analysis module has been tested in both controlled construction environments and real job sites, confirming its robustness and reliability. This real-world applicability means that the framework can be deployed to improve worker safety across various construction projects, potentially reducing accidents and enhancing overall compliance with safety regulations. The reduction in the false-alarm rate can minimize unnecessary interruptions in real-time safety-rule-compliance checking. Its adaptability to diverse environmental conditions and scene complexities, particularly in the construction domain, underscores its practical utility in real-world applications. Additionally, incorporating a buffering mechanism enables post-event analysis and compliance validation, ensuring system reliability.

These endeavors aim to enhance safety practices, safeguard the well-being of construction workers, and drive innovation in construction-site monitoring technology. Ultimately, this framework has the potential to set new industry standards and inspire future advancements in safety monitoring systems across various high-risk industries.

Author Contributions: Conceptualization, S.F.A.Z., J.Y., D.L. and C.P.; methodology, S.F.A.Z., J.Y., and M.S.A.; software, S.F.A.Z.; validation, S.F.A.Z., and M.S.A.; formal analysis, S.F.A.Z., J.Y. and R.H.; investigation, S.F.A.Z., R.H. and D.L.; resources, C.P.; data curation, S.F.A.Z., and M.S.A.; writing—original draft preparation, S.F.A.Z. and J.Y.; writing—review and editing, R.H., D.L. and C.P.; visualization, S.F.A.Z. and M.S.A.; supervision, C.P. and D.L.; project administration, C.P.; funding acquisition, C.P., S.F.A.Z. and J.Y. contributed equally to this paper. Therefore, they both have the right to share the first authorship of the research paper. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Chung-Ang University Research Grants in 2022 and in part by the National Research and Development Project for Smart Construction Technology (No. RS-2020-KA156291) funded by the Korea Agency for Infrastructure Technology Advancement under the Ministry of Land, Infrastructure and Transport and managed by the Korea Expressway Corporation.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. International Labour Organization. Webinar: Safety and Health in the Construction Sector- Overcoming the Challenges. 2018. Available online: https://www.ilo.org/empent/Eventsandmeetings/WCMS_310993/lang--en/index.htm (accessed on 7 August 2023).
2. Jung, D.; Seo, Y.; Shin, S.; Kim, D. Analyzing the relationship between the critical safety management tasks and their effects for preventing construction accidents using IPA method. *Korean J. Constr. Eng. Manag.* **2022**, *23*, 77–86. [CrossRef]
3. Hussain, R.; Pedro, A.; Zaidi, S.F.A.; Abbas, M.S.; Soltani, M.; Park, C. Conceptual Framework for Safety Training for Migrant Construction Workers using Virtual Reality Techniques. In *Digitalization in Construction*; Routledge: London, UK, 2023; pp. 93–103.
4. Park, C.; Soltani, M.; Pedro, A.; Yang, J.; Lee, D.; Hussain, R. *Transforming Construction Site Safety with iSAFE: An Automated Safety Management Platform*; Routledge: London, UK, 2023; pp. 213–234. [CrossRef]
5. Koc, K.; Gurgun, A.P. Scenario-based automated data preprocessing to predict severity of construction accidents. *Autom. Constr.* **2022**, *140*, 104351. [CrossRef]
6. Choi, J.; Gu, B.; Chin, S.; Lee, J.S. Machine learning predictive model based on national data for fatal accidents of construction workers. *Autom. Constr.* **2020**, *110*, 102974. [CrossRef]

7. Hussain, R.; Sabir, A.; Lee, D.Y.; Zaidi, S.F.A.; Pedro, A.; Abbas, M.S.; Park, C. Conversational AI-based VR system to improve construction safety training of migrant workers. *Autom. Constr.* **2024**, *160*, 105315. [[CrossRef](#)]
8. Choi, S.D.; Guo, L.; Kim, J.; Xiong, S. Comparison of fatal occupational injuries in construction industry in the United States, South Korea, and China. *Int. J. Ind. Ergon.* **2019**, *71*, 64–74. [[CrossRef](#)]
9. Xiao, B.; Kang, S.C. Development of an Image Data Set of Construction Machines for Deep Learning Object Detection. *J. Comput. Civ. Eng.* **2021**, *35*, 05020005. [[CrossRef](#)]
10. Chen, C.; Gu, H.; Lian, S.; Zhao, Y.; Xiao, B. Investigation of Edge Computing in Computer Vision-Based Construction Resource Detection. *Buildings* **2022**, *12*, 2167. [[CrossRef](#)]
11. Suh, S. A Qualitative Study Understanding Unsafe Behaviors of Workers in Construction Sites. *Korean J. Constr. Eng. Manag.* **2023**, *24*, 91–98. [[CrossRef](#)]
12. Hussain, R.; Zaidi, S.F.A.; Pedro, A.; Lee, H.; Park, C. Exploring construction workers' attention and awareness in diverse virtual hazard scenarios to prevent struck-by accidents. *Saf. Sci.* **2024**, *175*, 106526. [[CrossRef](#)]
13. Soltani, M.; Pedro, A.; Yang, J.; Zaidi, S.F.A.; Lee, D.; Park, C. Isafeguard: A Proactive Solution for Construction Job Site Safety Monitoring. In *Smart & Sustainable Infrastructure: Building a Greener Tomorrow*; Banthia, N., Soleimani-Dashtaki, S., Mindess, S., Eds.; Springer: Cham, Switzerland, 2024; pp. 1150–1165.
14. Zeng, L.; Li, R.Y.M. Construction safety and health hazard awareness in Web of Science and Weibo between 1991 and 2021. *Saf. Sci.* **2022**, *152*, 105790. [[CrossRef](#)]
15. Wang, J.; Jiang, L.; Yu, H.; Feng, Z.; Castaño-Rosa, R.; Cao, S. Computer vision to advance the sensing and control of built environment towards occupant-centric sustainable development: A critical review. *Renew. Sustain. Energy Rev.* **2024**, *192*, 114165. [[CrossRef](#)]
16. Khan, N.; Saleem, M.R.; Lee, D.; Park, M.W.; Park, C. Utilizing safety rule correlation for mobile scaffolds monitoring leveraging deep convolution neural networks. *Comput. Ind.* **2021**, *129*, 103448. [[CrossRef](#)]
17. Fang, Q.; Li, H.; Luo, X.; Ding, L.; Luo, H.; Rose, T.M.; An, W. Detecting non-hardhat-use by a deep learning method from far-field surveillance videos. *Autom. Constr.* **2018**, *85*, 1–9. [[CrossRef](#)]
18. Fang, Q.; Li, H.; Luo, X.; Ding, L.; Luo, H.; Li, C. Computer vision aided inspection on falling prevention measures for steeplejacks in an aerial environment. *Autom. Constr.* **2018**, *93*, 148–164. [[CrossRef](#)]
19. Huang, L.; Fu, Q.; He, M.; Jiang, D.; Hao, Z. Detection algorithm of safety helmet wearing based on deep learning. *Concurr. Comput. Pract. Exp.* **2021**, *33*, e6234. [[CrossRef](#)]
20. Han, K.; Zeng, X. Deep Learning-Based Workers Safety Helmet Wearing Detection on Construction Sites Using Multi-Scale Features. *IEEE Access* **2022**, *10*, 718–729. [[CrossRef](#)]
21. Hung, H.; Lan, L.; Hong, H. A Deep Learning-Based Method for Real-Time Personal Protective Equipment Detection. *Le Quy Don Tech. Univ.-Sect. Inf. Commun. Technol. LQDTU-JICT* **2019**, *199*, 23–34.
22. Wu, J.; Cai, N.; Chen, W.; Wang, H.; Wang, G. Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset. *Autom. Constr.* **2019**, *106*, 102894. [[CrossRef](#)]
23. Khan, M.; Nnaji, C.; Khan, M.S.; Ibrahim, A.; Lee, D.; Park, C. Risk factors and emerging technologies for preventing falls from heights at construction sites. *Autom. Constr.* **2023**, *153*, 104955. [[CrossRef](#)]
24. Li, T.; Xu, H.; Han, Y.; Zhao, Y.; Yan, H. L-Yolov5: A multi-scale channel attention-based method for real-time safety helmet detection of electrical construction workers. In Proceedings of the 2023 International Joint Conference on Neural Networks (IJCNN), Gold Coast, Australia, 18–23 June 2023; pp. 1–8. [[CrossRef](#)]
25. Li, R.Y.M.; Chau, K.W.; Ho, D.C.W. AI Object Detection, Holographic Hybrid Reality and Haemodynamic Response to Construction Site Safety Risks. In *Current State of Art in Artificial Intelligence and Ubiquitous Cities*; Li, R.Y.M., Chau, K.W., Ho, D.C.W., Eds.; Springer Nature Singapore: Singapore, 2022; pp. 117–134. [[CrossRef](#)]
26. Wang, J.; Razavi, S.N. Low False Alarm Rate Model for Unsafe-Proximity Detection in Construction. *J. Comput. Civ. Eng.* **2016**, *30*, 04015005. [[CrossRef](#)]
27. Talaat, F.M.; ZainEldin, H. An improved fire detection approach based on YOLO-v8 for smart cities. *Neural Comput. Appl.* **2023**, *35*, 20939–20954. [[CrossRef](#)]
28. Zhu, Y.; Dong, E.; Tong, J.; Yang, S.; Zhang, Z.; Li, W. Deep Neural Network Based Object Detection Algorithm with optimized Detection Head for Small Targets. In Proceedings of the 2023 IEEE International Conference on Mechatronics and Automation (ICMA), Harbin, China, 6–9 August 2023; pp. 2378–2382. [[CrossRef](#)]
29. Chen, P.; Zhou, H.; Li, Y.; Liu, P.; Liu, B. A Novel Deep Learning Network with Deformable Convolution and Attention Mechanisms for Complex Scenes Ship Detection in SAR Images. *Remote Sens.* **2023**, *15*, 2589. [[CrossRef](#)]
30. Chen, H.; Guo, X. Multi-scale feature fusion pedestrian detection algorithm based on Transformer. In Proceedings of the 2023 4th International Conference on Computer Vision, Image and Deep Learning (CVIDL), Zhuhai, China, 12–14 May 2023; pp. 536–540. [[CrossRef](#)]
31. Shi, P.; Liu, Z.; Qi, H.; Yang, A. MFF-Net: Multimodal Feature Fusion Network for 3D Object Detection. *Comput. Mater. Contin.* **2023**, *75*, 5615–5637. [[CrossRef](#)]
32. Kong, S.G.; Jin, D.; Li, S.; Kim, H. Fast fire flame detection in surveillance video using logistic regression and temporal smoothing. *Fire Saf. J.* **2016**, *79*, 37–43. [[CrossRef](#)]

33. De Venâncio, P.V.A.B.; Rezende, T.M.; Lisboa, A.C.; Barbosa, A.V. Fire Detection based on a Two-Dimensional Convolutional Neural Network and Temporal Analysis. In Proceedings of the 2021 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Temuco, Chile, 2–4 November 2021; pp. 1–6. [\[CrossRef\]](#)
34. Luo, F.; Li, R.Y.M.; Crabbe, M.J.C.; Pu, R. Economic development and construction safety research: A bibliometrics approach. *Saf. Sci.* **2022**, *145*, 105519. [\[CrossRef\]](#)
35. Lee, J.; Lee, S. Construction Site Safety Management: A Computer Vision and Deep Learning Approach. *Sensors* **2023**, *23*, 944. [\[CrossRef\]](#) [\[PubMed\]](#)
36. Zhang, M.; Shi, R.; Yang, Z. A critical review of vision-based occupational health and safety monitoring of construction site workers. *Saf. Sci.* **2020**, *126*, 104658. [\[CrossRef\]](#)
37. Zaidi, S.F.A.; Hussain, R.; Abbas, M.S.; Yang, J.; Lee, D.; Park, C. iSafe Welding System: Computer Vision-Based Monitoring System for Safe Welding Work. In *CONVR 2023—Proceedings of the 23rd International Conference on Construction Applications of Virtual Reality, Florence, Italy, 13–16 November 2023*; Capone, P., Getuli, V., Rahimian, F.P., Dawood, N., Bruttini, A., Sorbi, T., Eds.; Firenze University Press: Florence, Italy, 2023; pp. 669–675. [\[CrossRef\]](#)
38. Anjum, S.; Khan, N.; Khalid, R.; Khan, M.; Lee, D.; Park, C. Fall Prevention From Ladders Utilizing a Deep Learning-Based Height Assessment Method. *IEEE Access* **2022**, *10*, 36725–36742. [\[CrossRef\]](#)
39. Fang, W.; Zhong, B.; Zhao, N.; Love, P.E.; Luo, H.; Xue, J.; Xu, S. A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network. *Adv. Eng. Inform.* **2019**, *39*, 170–177. [\[CrossRef\]](#)
40. Xiao, B.; Xiao, H.; Wang, J.; Chen, Y. Vision-based method for tracking workers by integrating deep learning instance segmentation in off-site construction. *Autom. Constr.* **2022**, *136*, 104148. [\[CrossRef\]](#)
41. Luo, H.; Wang, M.; Wong, P.K.Y.; Cheng, J.C. Full body pose estimation of construction equipment using computer vision and deep learning techniques. *Autom. Constr.* **2020**, *110*, 103016. [\[CrossRef\]](#)
42. Seong, H.; Son, H.; Kim, C. A Comparative Study of Machine Learning Classification for Color-based Safety Vest Detection on Construction-Site Images. *KSCE J. Civ. Eng.* **2018**, *22*, 4254–4262. [\[CrossRef\]](#)
43. Fang, W.; Ding, L.; Luo, H.; Love, P.E. Falls from heights: A computer vision-based approach for safety harness detection. *Autom. Constr.* **2018**, *91*, 53–61. [\[CrossRef\]](#)
44. Wang, M.; Wong, P.K.Y.; Luo, H.; Kumar, S.; Delhi, V.S.K.; Cheng, J.C.P. Predicting Safety Hazards Among Construction Workers and Equipment Using Computer Vision and Deep Learning Techniques. In Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC), Banff, AB, Canada, 21–24 May 2019; Al-Hussein, M., Ed.; The International Association for Automation and Robotics in Construction: Edinburgh, UK; pp. 399–406. [\[CrossRef\]](#)
45. Yang, J. Enhancing action recognition of construction workers using data-driven scene parsing. *J. Civ. Eng. Manag.* **2018**, *24*, 568–580. [\[CrossRef\]](#)
46. Kim, B.; Alawami, M.A.; Kim, E.; Oh, S.; Park, J.; Kim, H. A Comparative Study of Time Series Anomaly Detection Models for Industrial Control Systems. *Sensors* **2023**, *23*, 1310. [\[CrossRef\]](#) [\[PubMed\]](#)
47. Okumura, T.; Imai, K.; Misawa, M.; Kudo, S.e.; Hotta, K.; Ito, S.; Kishida, Y.; Takada, K.; Kawata, N.; Maeda, Y.; et al. Evaluating false-positive detection in a computer-aided detection system for colonoscopy. *J. Gastroenterol. Hepatol.* **2024**, *39*, 927–934. [\[CrossRef\]](#)
48. Borowski, M.; Siebig, S.; Wrede, C.; Imhoff, M. Reducing False Alarms of Intensive Care Online-Monitoring Systems: An Evaluation of Two Signal Extraction Algorithms. *Comput. Math. Methods Med.* **2011**, *2011*, 143480. [\[CrossRef\]](#) [\[PubMed\]](#)
49. Yu, M.; Yuan, H.; Li, K.; Wang, J. Research on multi-detector real-time fire alarm technology based on signal similarity. *Fire Saf. J.* **2023**, *136*, 103724. [\[CrossRef\]](#)
50. Sudhakar, S.; Vijayakumar, V.; Sathiya Kumar, C.; Priya, V.; Ravi, L.; Subramaniaswamy, V. Unmanned Aerial Vehicle (UAV) based Forest Fire Detection and monitoring for reducing false alarms in forest-fires. *Comput. Commun.* **2020**, *149*, 1–16. [\[CrossRef\]](#)
51. de Venâncio, P.V.A.B.; Campos, R.J.; Rezende, T.M.; Lisboa, A.C.; Barbosa, A.V. A hybrid method for fire detection based on spatial and temporal patterns. *Neural Comput. Appl.* **2023**, *35*, 9349–9361. [\[CrossRef\]](#)
52. Abdulghafoor, N.H.; Abdullah, H.N. A novel real-time multiple objects detection and tracking framework for different challenges. *Alex. Eng. J.* **2022**, *61*, 9637–9647. [\[CrossRef\]](#)
53. Ray, K.S.; Chakraborty, S. Object detection by spatio-temporal analysis and tracking of the detected objects in a video with variable background. *J. Vis. Commun. Image Represent.* **2019**, *58*, 662–674. [\[CrossRef\]](#)
54. Chow, J.K.; Su, Z.; Wu, J.; Li, Z.; Tan, P.S.; Liu, K.; Mao, X.; Wang, Y.H. Artificial intelligence-empowered pipeline for image-based inspection of concrete structures. *Autom. Constr.* **2020**, *120*, 103372. [\[CrossRef\]](#)
55. Naseer, A.; Alzahrani, H.A.; Almujally, N.A.; Nowaiser, K.A.; Mudawi, N.A.; Algarni, A.; Park, J. Efficient Multi-Object Recognition Using GMM Segmentation Feature Fusion Approach. *IEEE Access* **2024**, *12*, 37165–37178. [\[CrossRef\]](#)
56. Vijayakumar, A.; Vairavasundaram, S. YOLO-based Object Detection Models: A Review and its Applications. *Multimed. Tools Appl.* **2024**. [\[CrossRef\]](#)
57. Gaur, P.; Gupta, H.; Chowdhury, A.; McCreddie, K.; Pachori, R.B.; Wang, H. A Sliding Window Common Spatial Pattern for Enhancing Motor Imagery Classification in EEG-BCI. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–9. [\[CrossRef\]](#)
58. PPE Object Detection. PPE Dataset. 2022. Available online: <https://universe.roboflow.com/object-detection-ppe-0fljh/ppe-hc4lw> (accessed on 26 December 2023).

59. Pandey, R.K.; Kumar, A.; Mandal, A. A robust deep structured prediction model for petroleum reservoir characterization using pressure transient test data. *Pet. Res.* **2022**, *7*, 204–219. [[CrossRef](#)]
60. Mohanty, P.; Sahoo, J.P.; Nayak, A.K. Voiced Odia Digit Recognition Using Convolutional Neural Network. In *Advances in Distributed Computing and Machine Learning*; Sahoo, J.P., Tripathy, A.K., Mohanty, M., Li, K.C., Nayak, A.K., Eds.; Springer: Singapore, 2022; pp. 161–173.
61. Kim, D.; MacKinnon, T. Artificial intelligence in fracture detection: Transfer learning from deep convolutional neural networks. *Clin. Radiol.* **2018**, *73*, 439–445. [[CrossRef](#)]
62. Mukherjee, S.; Sahu, T.; Sai Chandra Teja, R.; Mittal, S. ConstructNet: A Deep Learning Object Detector for Construction Site Surveillance. In Proceedings of the 2024 IEEE Applied Sensing Conference (APSCON), Goa, India, 22–24 January 2024; pp. 1–4. [[CrossRef](#)]
63. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
64. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [[CrossRef](#)]
65. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)]
66. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 386–397. [[CrossRef](#)]
67. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, CA, USA, 27–30 June 2016; pp. 779–788. [[CrossRef](#)]
68. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *Proceedings of the Computer Vision—ECCV 2016, Amsterdam, The Netherlands, 11–14 October 2016*; Leibe, B., Matas, J., Sebe, N., Welling, M., Eds.; Springer: Cham, Switzerland, 2016; pp. 21–37.
69. Yu, W.; Xiang, Z.; Jiantong, S.; Minhua, L. YOLOv5-Based Dense Small Target Detection Algorithm for Aerial Images Using DIOU-NMS. *Radioengineering* **2024**, *33*, 12–23.
70. Lou, H.; Duan, X.; Guo, J.; Liu, H.; Gu, J.; Bi, L.; Chen, H. DC-YOLOv8: Small-Size Object Detection Algorithm Based on Camera Sensor. *Electronics* **2023**, *12*, 2323. [[CrossRef](#)]
71. Ultralytics. YOLOv8—Ultralytics | Revolutionizing the World of Vision AI. 2023. Available online: <https://ultralytics.com/yolov8> (accessed on 8 August 2023).
72. Terven, J.; Córdova-Esparza, D.M.; Romero-González, J.A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 1680–1716. [[CrossRef](#)]
73. Solawetz, J.; Francesco. What is YOLOv8? The Ultimate Guide. 2023. Available online: <https://blog.roboflow.com/whats-new-in-yolov8/> (accessed on 5 September 2023).
74. Lv, J.; Chen, J.; Huang, Z.; Wan, H.; Zhou, C.; Wang, D.; Wu, B.; Sun, L. An Anchor-Free Detection Algorithm for SAR Ship Targets with Deep Saliency Representation. *Remote Sens.* **2023**, *15*, 103. [[CrossRef](#)]
75. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: A Simple and Strong Anchor-Free Object Detector. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 1922–1933. [[CrossRef](#)] [[PubMed](#)]
76. Kong, T.; Sun, F.; Liu, H.; Jiang, Y.; Li, L.; Shi, J. FoveaBox: Beyond Anchor-Based Object Detection. *IEEE Trans. Image Process.* **2020**, *29*, 7389–7398. [[CrossRef](#)]
77. Olorunshola, O.E.; Irrehbude, M.E.; Ewwiekpaefe, A.E. A Comparative Study of YOLOv5 and YOLOv7 Object Detection Algorithms. *J. Comput. Soc. Inform.* **2023**, *2*, 1–12. [[CrossRef](#)]
78. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
79. Ahmed, M.I.B.; Saraireh, L.; Rahman, A.; Al-Qarawi, S.; Mhran, A.; Al-Jalaoud, J.; Al-Mudaifer, D.; Al-Haidar, F.; AlKhulaifi, D.; Youldash, M.; et al. Personal Protective Equipment Detection: A Deep-Learning-Based Sustainable Approach. *Sustainability* **2023**, *15*, 13990. [[CrossRef](#)]
80. Gallo, G.; Rienzo, F.D.; Garzelli, F.; Ducange, P.; Vallati, C. A Smart System for Personal Protective Equipment Detection in Industrial Environments Based on Deep Learning at the Edge. *IEEE Access* **2022**, *10*, 110862–110878. [[CrossRef](#)]
81. Isailovic, V.; Peulic, A.; Djapan, M.; Savkovic, M.; Vukicevic, A.M. The compliance of head-mounted industrial PPE by using deep learning object detectors. *Sci. Rep.* **2022**, *12*, 16347. [[CrossRef](#)]
82. Lee, Y.R.; Jung, S.H.; Kang, K.S.; Ryu, H.C.; Ryu, H.G. Deep learning-based framework for monitoring wearing personal protective equipment on construction sites. *J. Comput. Des. Eng.* **2023**, *10*, 905–917. [[CrossRef](#)]
83. Islam, M.; Mannering, F. A temporal analysis of driver-injury severities in crashes involving aggressive and non-aggressive driving. *Anal. Methods Accid. Res.* **2020**, *27*, 100128. [[CrossRef](#)]
84. Xiao, Z.; Xu, X.; Xing, H.; Luo, S.; Dai, P.; Zhan, D. RTFN: A robust temporal feature network for time series classification. *Inf. Sci.* **2021**, *571*, 65–86. [[CrossRef](#)]

85. Patel, V.; Zhang, S.; Tian, B. Global Convergence and Stability of Stochastic Gradient Descent. In *Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: Glasgow, UK, 2022; Volume 35, pp. 36014–36025.
86. Ziyin, L.; Li, B.; Simon, J.B.; Ueda, M. SGD with a Constant Large Learning Rate Can Converge to Local Maxima. *arXiv* **2021**, arXiv:2107.11774. [[CrossRef](#)]
87. Rainio, O.; Teuho, J.; Klén, R. Evaluation metrics and statistical tests for machine learning. *Sci. Rep.* **2024**, *14*, 6086. [[CrossRef](#)] [[PubMed](#)]
88. Quach, L.D.; Quoc, K.N.; Quynh, A.N.; Ngoc, H.T. Evaluating the effectiveness of YOLO models in different sized object detection and feature-based classification of small objects. *J. Adv. Inf. Technol.* **2023**, *14*, 907–917. [[CrossRef](#)]
89. Huang, J.; Rathod, V.; Sun, C.; Zhu, M.; Korattikara, A.; Fathi, A.; Fischer, I.; Wojna, Z.; Song, Y.; Guadarrama, S.; et al. Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
90. Miriam Steurer, R.J.H.; Pfeifer, N. Metrics for evaluating the performance of machine learning based automated valuation models. *J. Prop. Res.* **2021**, *38*, 99–129. [[CrossRef](#)]
91. Powers, D.M. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv* **2020**, arXiv:2010.16061.
92. Lei, S.; Zhang, H.; Wang, K.; Su, Z. How Training Data Affect the Accuracy and Robustness of Neural Networks for Image Classification. In Proceedings of the International Conference on Learning Representations 2019, New Orleans, LA, USA, 6–9 May 2019.
93. Drenkow, N.; Sani, N.; Shpitser, I.; Unberath, M. A Systematic Review of Robustness in Deep Learning for Computer Vision: Mind the gap? *arXiv* **2021**, arXiv:2112.0063. [[CrossRef](#)]
94. Zaidi, S.F.A.; Woo, H.; Lee, C.G. A Graph Convolution Network-Based Bug Triage System to Learn Heterogeneous Graph Representation of Bug Reports. *IEEE Access* **2022**, *10*, 20677–20689. [[CrossRef](#)]
95. Choi, J.; Lee, H. Real-Time Traffic Light Recognition with Lightweight State Recognition and Ratio-Preserving Zero Padding. *Electronics* **2024**, *13*, 615. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.