buildings

MDPI

Article

Dynamic Collision Alert System for Collaboration of Construction Equipment and Workers

Ren-Jye Dzeng [1], Binghui Fan [2,*] and Tian-Lin Hsieh [1]

[1] Department of Civil Engineering, National Yang Ming Chiao Tung University, Hsinchu 30010, Taiwan; rjdzeng@nycu.edu.tw (R.-J.D.); a993355light@gmail.com (T.-L.H.)
[2] College of Civil Engineering, Fuzhou University, Fuzhou 350108, China
* Correspondence: fanbinghui@fzu.edu.cn

Abstract: The construction industry is considered one of the most hazardous industries. The accidents associated with construction equipment are a leading cause of fatalities in the U.S., with one-quarter of all fatalities in the construction industry due to equipment-related incidents, including collisions, struck-by events, and rollovers. While close collaboration among multiple equipment and humans is common, conventional collision alert mechanisms for equipment usually rely on distance sensors with static thresholds, often resulting in too many false alarms, causing drivers' ignorance. Considering the collaborative operation scenario, this research proposes and develops a dynamic-threshold alert system by recognizing hazardous events based on the types of nearby objects with their orientation or postures and their distances to the system carrier equipment based on image-based recognition and Sim2Real techniques. Two experiments were conducted, and the results show that the system successfully reduced a large number of false near-collision alarms for the collaboration scenarios. Although the accuracy of object recognition and image-based distance estimation is feasible for practical use, it is also easily degraded in the self-obstruction scenario or for equipment with large and movable parts due to incorrect recognition of the bounding boxes of the target objects.

Keywords: equipment collision; YOLO; image recognition; Sim2Real

check for updates

Academic Editor: Paulo Santos

Received: 20 November 2024
Revised: 22 December 2024
Accepted: 23 December 2024
Published: 31 December 2024

Citation: Dzeng, R.-J.; Fan, B.; Hsieh, T.-L. Dynamic Collision Alert System for Collaboration of Construction Equipment and Workers. *Buildings* **2025**, *15*, 110. https://doi.org/10.3390/buildings15010110

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/).

1. Introduction

The construction industry is considered one of the most hazardous industries [1]. In the U.S., according to the Occupational Safety and Health Administration of U.S., out of 5333 worker fatalities in 2019, 1061, or approximately 20%, occurred in the construction industry [2]. Among the various types of work-related injuries and fatalities in the construction industry, those associated with construction equipment are the leading cause [3]. Vahdatikhaki et al. [4] also identified that incidents involving construction equipment are a primary cause of the various types of work-related injuries and fatalities in the construction sector. Hinze and Teizer [1] reported that one-quarter of all fatalities in the construction industry are due to equipment-related incidents, including collisions, struck-by events, and rollovers.

Similarly, in the EU [5] in 2022, the largest share of fatal accidents at work in all industries occurred while people were driving or aboard transport or handling equipment (18.4%). In the construction industry, equipment operation-related accidents also had the highest share in non-fatal accidents and the third in fatal accidents, with falling-related accidents being the highest.

In Japan, industrial accidents caused by large machines, cranes, conveying machines, and other equipment alone already accounted for 43.2% of reported accidents (56,679 of 131,156 cases) between 1999 and 2020, and up to 59.5% if only fatal accidents are considered [6]. In China, within the construction industry, Zhang et al. [7] collected 2758 accident event reports available from the emergency administrations of the provinces nationwide from 2013 to 2023. They reported that 1083 events were related to the construction industry, and falls from height occurred most frequently (478 cases), followed by collapses (227 cases), being struck by objects (133 cases), hoisting operation (66 cases), poisoning and suffocation (40 cases), machinery injuries (36 cases), etc. Among the accidents, objects striking, hoisting, and machinery may also be related to equipment operation.

1.1. Proximity Detection for Construction Site

Given the importance of safe equipment operation, several studies have been conducted on monitoring the safety of the construction environment, such as hillsides or unsteady slopes ([8,9]). Some investigated the causes of construction equipment-related accidents ([10,11]). Some proposed technologies, such as personal instruction systems [12], predict the movements of onsite equipment and workers [13] or vision-based excavator detection and tracking [14] to help prevent accidents.

Collaboration among multiple equipment and humans is common, and the key to intelligent construction lies in the efficient collaboration between equipment [14]. The authors have not yet found any research on detecting near-collisions specifically for construction equipment collaboration scenarios. However, some studies have utilized various techniques to detect objects entering dangerous proximity zones around equipment. For example, Bohn and Teizer [15] used video cameras and time-lapse photography to assess projects' overall daily safety status, providing non-real-time references for safety improvements. Fullerton et al. [16] employed LiDAR-generated point clouds to identify whether objects were present in the blind spots of machinery operators, offering real-time warnings to the drivers. Teizer et al. [17] first used LiDAR to scan the blind spots of commonly used heavy equipment (e.g., trucks, excavators, and graders) to establish blind-spot maps. They then developed sensing devices to alert both operators and personnel entering these blind spots with sound and vibration when they were in dangerous proximity while also measuring the distance between them. Kim et al. [18] used drones equipped with cameras to capture aerial images of construction sites. They applied YOLO-V3 deep neural networks to locate specific targets and developed image correction methods that allowed direct measurement of actual distances in 2D images without performing 3D inference. These studies focused on the accuracy and effectiveness of detecting the proximity of objects to equipment, whether through unidirectional passive sensing (where surrounding objects do not carry sensors) or bidirectional active sensing. However, they all employed fixed alert distances for detection alerts.

Conventional alert mechanisms for heavy equipment usually rely on distance sensors, such as ultrasonic, infrared light, laser technologies, or proximity sensors. These sensors provide enough accuracy for collision alerts. However, conventional alert mechanisms generate a collision alarm whenever and with whatever object is close to the subject equipment, within a static threshold of distance range. Such a simple alert mechanism often creates too many false alarms during collaborative operation scenarios, which may result in the ignorance of equipment drivers. The frequent false alarms generated by current proximity avoidance systems have limited their implementation in realistic construction environments [19]. Wang and Razavi [20] reviewed the prevalent proximity warning systems at the time using technologies such as ultrasonic, radio frequency, laser, radar, and vision and also pointed out that one standard limitation of those methods was the

high possibility of generating false alarms. Ruff's application of radar technology in proximity detection and his 7-day test of 618 times reverse of a dump truck (without considering the collaborative situation) [21] shows that only 235 alarms (41%) actually required immediate action or extra caution. This research aims to improve the safety of collaborative construction equipment operations by proposing a dynamic collision-alert threshold mechanism based on deep-learning automatic recognition of image events of cameras attached to the equipment.

1.2. Sim2Real

While adopting automatic image recognition of near-collision events, this research faced the challenges of obtaining a high volume of images related to equipment near-collision events. To learn effective collision prediction models, the robot or self-driving vehicles must see several real-world collisions during training, which comes at a high cost [22]. Similarly, creating collision or near-collision video clips of human-driving construction equipment is costly and unsafe for the participating drivers.

Sim2Real (Simulation-to-Reality) refers to the technique and field of applying learning results from virtual simulation environments to the real world. It is an important research direction in machine learning, aiming to address the domain adaptation differences and transfer learning challenges between virtual simulation environments and the real world. In Sim2Real, researchers often use computer simulations and virtual environments to train and optimize machine learning models. These models can be deep neural network models for tasks like visual perception, control, path planning, or other machine learning algorithm models. For example, in robotics research (e.g., [22,23]), simulation environments can be computer-generated settings that mimic real-world scenarios. These environments provide a safe, cost-effective, and easily controllable platform for developing and testing algorithms, learning policies, or training models before deploying them in real-world applications. However, the results obtained from learning in virtual environments may not be directly applied to the real world due to the differences between virtual and real worlds, often referred to as the reality gap. This gap can result from differences in physical properties, sensor noise, or noises from the uncontrolled real environment, leading to a decline in performance when transferring a model from simulation to reality.

Safety and cost are probably why researchers adopt Sim2Real approaches instead of collecting data and training machines directly in the real world. For example, Kang et al. [24] used real-world data and simulated data from virtual environments to train models on aerodynamics and movement, allowing unmanned aerial vehicles to avoid collisions in new environments with different lighting conditions and geometries. This research adopts the Sim2Real approach for the same reasons: collision or near-collision image data on construction sites among equipment and workers are hard to collect. Setting up an experiment for the equipment and workers to create collision or near-collision scenarios is also unsafe and costly.

The goal of Sim2Real is to develop methods and techniques that enable models trained in virtual environments to perform well in the real world. Depending on the research problem and objective, several techniques have been used to help achieve this purpose and decrease the reality gap. For example, domain randomization involves randomizing physical parameters (e.g., gravity, friction) and objects (e.g., colors, texture) in a virtual environment. The goal is to generate data with various parameters in a virtual world through randomization to increase the chance of covering those in the real world. For example, Tobin et al. [25] created various objects with non-realistic textures in the virtual environment. By randomly rendering the textures, they used this as training data for the model, enabling it to detect objects in the real world.

Domain adaptation involves mapping data from virtual and real environments, which belong to the same category but have different distributions, into the same feature space. The goal is to minimize the difference between them so that models trained in the virtual environment can be directly applied in the real world. For example, Long et al. [26] proposed an algorithm to reduce the difference between virtual and real data in the feature space, allowing the model to be used in the real world.

System identification involves constructing a mathematical or simulation model in the virtual environment that closely resembles the real-world environment, allowing models trained in the virtual environment to be directly transferred to real-world applications. The challenge of this approach lies in creating ultra-realistic image rendering in the virtual environment. For example, Yu et al. [27] first sampled system parameters (e.g., friction) and then designed a control strategy to handle the sampled parameters. They used this control strategy to manage the system parameters within the sampled range. A simple neural network was then built using the system parameters' output by the control strategy as input, allowing the neural network to approximate the control strategy's output parameters.

Of course, many other approaches are still available, such as transfer learning [28] and knowledge distillation [29]. The Sim2Real method used in this research is similar to the system identification. To overcome the challenges of collecting collision and near-collision images from the real world due to safety and cost concerns, this research chose to rebuild a realistic virtual world based on Unity's physics engine to mimic the collaborative operations of equipment on construction sites and generate video images necessary for training machine learning model.

1.3. Vision-Based Object Recognition

With regards to the techniques in the recognition of objects in an image, Girshick et al. [30] first proposed the well-known Fast R-CNN model, an architecture that uses Convolutional Neural Networks (CNNs) for object detection and semantic segmentation in an image. It performs object detection by generating candidate regions and applying convolution operations. Girshick introduced a state-of-the-art CNN-based object detection training process at the time. However, he also noted that, while R-CNN significantly outperformed other object detection models in speed, the training process was multi-stage and required fine-tuning hyperparameters for specific classifications and high hardware requirements. Ren [31] also introduced a one-stage training algorithm that simultaneously learns to classify object proposals and refine the spatial location of objects. This approach led to models capable of training very deep object detection networks (e.g., VGG16), which were 9 times faster than R-CNN.

Redmon et al. [32] introduced the YOLO (You Only Look Once) model, an efficient object detection method capable of fast and accurate real-time detection. The advantage of YOLO over previous object detection models is its one-stage training approach, which means detection is completed in a single pass. Even using a smaller version, Fast YOLO, the mean Average Precision (mAP) is twice that of other real-time detectors. YOLO operates at 45 frames per second for the regular version, while Fast YOLO achieves an impressive 155 frames per second.

2. Problem Statement and Conceptual Model

The objective of this research is to develop a system to generate appropriate alarms under the collaborative operation scenario by recognizing hazardous events based on the types of nearby objects (e.g., counter equipment and workers) and their distances to the system carrier (i.e., the subject equipment with the developed system). Since collecting real-life near-collision images of human-operating equipment is infeasible, this research

adopts the Sim2Real approach and creates virtual simulated images to train the model. In addition, this research also automates laborious conventional manual work during the generation and labeling of virtual images.

Figure 1 outlines the workflow of this research. The development of the object recognition model is on the left-hand side, and on the right-hand side, with a light-green background, are the steps for developing and evaluating the proposed system. After surveying the literature to find suitable vision-based models for detecting construction equipment and estimating the distance between them, we searched for construction projects as references for building corresponding virtual reality models. Humans operated the equipment in the virtual world to generate and annotate video images of the equipment's collaborative operation and near-collision situations. The annotated video images were then used to train Mask-RCNN and YOLO models.

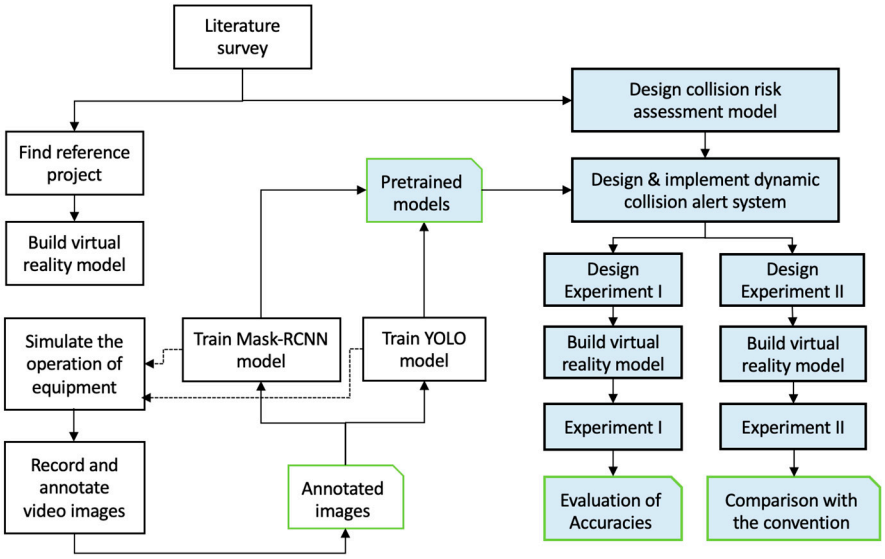


Figure 1. Research workflow of this research.

The proposed dynamic alert system was designed and implemented with a collision risk assessment model and the pre-trained model. Two experiments were designed and conducted to evaluate the system's recognition accuracy and compare the reduction in false alarms compared to the conventional static threshold mechanism.

The following first describes the framework of the proposed dynamic collision alert system, the details in the primary research steps and modules of the system, including the training image generation and automatic labeling, distance estimation, and most importantly, the Collision Risk Assessment Model, which makes dynamic alert possible.

2.1. Dynamic Collision Alert System

This research proposes a system called the Dynamic Collision Alert System (DCAS), which evaluates the collision risk from the perspective of the recording camera attached to the subject equipment based on image recognition of the type of target objects in the image and their distance from the subject. Within a close range, the near-collision alarms are only generated when the evaluated risk level exceeds the prescribed threshold during the collaborative operation.

As shown in the bottom of Figure 2, consider a collaborative operation scenario that involves subject equipment (e.g., excavator), and nearby counter equipment (e.g., forklift or another excavator) or workers, termed target objects (i.e., objects to be identified and determined if a hazardous situation occurs). Cameras are attached to appropriate parts of the subject equipment to capture an image of the environment continuously from the

camera's perspective. Conventional alert mechanisms for heavy equipment usually rely on distance sensors, which provide enough accuracy for collision alerts. However, as depicted by the blue dashed lines in Figure 2, conventional alert mechanisms generate a collision alarm whenever and with whatever object is close to the subject equipment, within a static threshold of distance range. Such a simple alert mechanism often creates too many false alarms during collaborative operation scenarios.

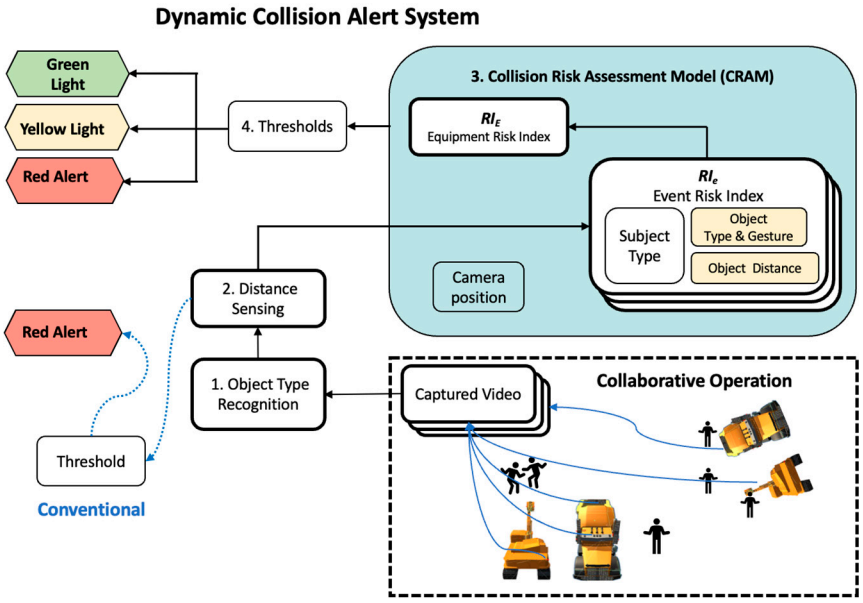


Figure 2. Dynamic Collision Alert System and CRAM.

DCAS aims to reduce false or unnecessary near-collision alarms in collaborative operations. The objective is to distinguish true near-collision scenarios from the proximity scenarios necessary for collaboration. DCAS adopts a dynamic proximity concept that considers the subject's type, the target objects' types, and their orientations or gestures, along with their distances to the subject and the attached position of the camera. For example, with all other things the same, an operating excavator imposes more collision risk than a dump truck. An operating excavator imposes more hazards on a nearby worker than on another excavator. Also, a worker facing away from the back of an operating excavator is exposed to more collision risk than when the worker is facing the front (driver) of the excavator. Lastly, with respect to a nearby worker at the same distance, the worker captured by a camera attached to the arm joint or cabin-side of an excavator indicates a more collision-risky situation than the image captured by the one attached to the front of the cabin.

As shown in Figure 2, DCAS consists of four continuous iteration phases. Given an image of a video clip from a camera attached to a subject equipment, the first step is to recognize the target objects in the image, including their types (e.g., excavator, worker) and their orientation or gestures (e.g., equipment's front, side, or back; worker's walking and facing, or kneeling back-facing). The second phase estimates the distance of each target object to the subject according to the assumption of their sizes. The third phase evaluates the collision risk of the equipment based on the proposed Collision Risk Assessment Model (CRAM). The last phase generates red light alarms for near-collision, yellow lights for caution of approaching for collaboration, and green lights for safety. Note that when the first author proposed CRAM, two different systems with different deep learning models and training cases were implemented. The first system, presented in Chinese [33], adopted the Mask-RCNN model. The DCAS presented in this paper adopted the YOLO model to

reduce the image recognition time so that the system can be applied more readily to the construction site for instant alerts.

2.2. Image Generation

DCAS's evaluation of collision risk relies on the correct recognition of the type of target objects and their distances to the subject equipment, the training of which depends on high volumes of appropriate images. DCAS's strategy for reducing false collision alarms is to fire alarms only for true near-collisions and only show caution with yellow lights for the objects reasonably approaching during collaboration. Thus, images of true near-collisions (or even collisions), reasonable proximity with caution, and proximity that need no caution should be collected to train the machine learning model.

To generate images necessary to train the YOLO model, a 3D virtual world of the scenarios with equipment on construction sites must be built and operable with the possibility of near-collision occurrences using platforms such as Unity 3D. Operable equipment, such as excavators and forklifts, and workers with different actions are the primary target objects in the scenario. To closely simulate reality so that realistic images can be created, the equipment objects should be built with realistic colors and textures and according to the specifications (e.g., sizes, moving parts, maximum rotation angles) of the equipment in reality. Workers with different actions (e.g., working, walking, kneeling, facing, or facing away) should also be provided when applicable.

Virtual camera objects should also be created and attached to the desirable parts of the equipment objects to record images during the simulated operation. Cameras and their attached positions on the equipment should also be designed according to the relevant specifications and feasibilities. Taking an excavator as an example, a camera attached to the front or side of the cabin or even to the front of the arm is feasible. However, a camera attached to the bucket or bottom rollers is not feasible because it can be easily muddied or damaged during the excavator's operation.

During the simulated operation of equipment and workers, the attached cameras record images necessary to train deep learning models, including safe collaborations, near-collisions, and possible collisions. To prepare for the next step of automatic labeling, three types of images are required, all of which are recorded via the cameras attached to the subject equipment. First, the cameras record images of target objects (e.g., counter equipment, workers) in empty spaces without background or with a single-color background. Second, the cameras record the backgrounds without counter equipment or workers. Both should be stored as photo images in .png format for easy image overlay in the later process. Third, the cameras record video images of normal operation, with or without target objects in spaces with backgrounds. The video may be stored in .mp4 format.

2.3. Automatic Labeling

Conventionally, for image recognition tasks, image data need to be labeled before being used for model training. Taking the YOLO model as an example, the image data for YOLO typically require manual labeling using tools such as Roboflow [34], where objects of interest in the image are manually enclosed within bounding boxes. The coordinates of the bounding boxes within the entire image are then calculated. The correct categories of the objects, along with the box coordinate values, are subsequently used as labels for the image data before being fed into the YOLO model for training. Manual enclosure of the bounding box and categorization of the target objects in the image data are time-consuming and laborious. This study proposes an automatic labeling process for generating an object detection bounding box on the target objects in the images to save preparation time for training data.

The advantage of manual image labeling is that it allows for accurate object identification without noise interference from backgrounds or other overlayed objects. However, when dealing with large datasets, manual labeling not only becomes time-consuming but also results in human errors. These errors could result in bounding boxes either exceeding or not fully covering the intended objects, leading to unsatisfactory recognition performance during model training.

It is comparatively complex to automatically identify and label the boundary of the target object in a real-life image. However, in this study, since virtual environments are used to generate image data, the background can be arbitrarily set to a single color initially during image generation. Thus, the object's boundary can be easily and automatically detected and labeled based on image binarization. This approach offers the advantage of labeling objects based on pixel values, ensuring that the bounding boxes precisely enclose the target objects, which is generally more precise than human labeling. Additionally, since the process is automated, large image datasets can be labeled quickly without the imprecision associated with manual labeling.

However, this approach also has a significant drawback: the overly simplistic background can cause the trained model to mistakenly classify background elements as target objects, as shown later in the experiment section. Thus, after labeling, the images should be further overlayed with a variety of realistic backgrounds so that the trained model may better recognize target objects in complex backgrounds accurately.

2.4. Image Recognition

Bharati and Pramanik [35] reviewed several convolutional neural network models and concluded that, in general, the speed performance of YOLO was the fastest, and the average precision of Mask R-CNN was the highest among the tested techniques. Sumit et al. [36] compared the performance of YOLO and Mask R-CNN, which unveils the inability of Mask R-CNN to detect tiny human figures among other prominent human images and illustrates that YOLO was successful in detecting most of the human figures in an image with higher accuracy. Both Mask-RCNN and YOLO appear to be suitable for this research.

2.5. Distance Estimation

Wu et al. [37] integrated digital twins, deep learning, and mixed reality (MR) to develop a real-time visual alert system. The system aims to recognize workers and surrounding hazard alerts through image recognition and relay this information to personnel wearing MR devices. In their study, a fixed camera's field of view was used, and the coordinates of this field were defined. When personnel entered the camera's range, their coordinates were matched with the camera's defined coordinates to calculate the distance between the personnel and the camera. However, compared to this study, they focused solely on worker models, without consideration of equipment or other objects. Additionally, their system relied only on fixed cameras, which could lead to occlusion or personnel moving out of the camera's range in this study's context.

Waranusast et al. [38] proposed an algorithm for estimating the distance between a camera and a vehicle from an image based on object detection and machine learning. The algorithm started by collecting vehicle images of different types and measuring their distances from the camera with a laser measuring device. Natanael et al. [39] estimated the distance to a car that drives in front of the subject's car, with possibly relative speed with respect to it. The image, taken with a front camera, is classified using YOLO, and after the searched object is detected, the distance is estimated by counting the number of pixels in a bounding box that fits the detected object. The distance is further corrected by using Canny edge detection and HSV color space. Liang [40] also used YOLO to develop a

safety distance warning system for motorcycles based on image recognition. He estimated the object distance based on the principles of optics, which requires knowledge of the size or height of the target objects, the camera's focal length, and sensor size. Similarly, this study also applies the optical principle to calculate the distance between recognized objects and the camera in images. However, unlike Liang's method, this study's camera can be mounted on different parts of different construction machines. Moreover, the method used to calculate the error between the actual and recognized distances is performed while the machinery is in motion, rather than relying on a fixed shooting position for recognition.

The distance of an object can be calculated based on its size, image size, and focal length of the camera used, using the definition of focal length and magnification formulas, as shown in Equation (1) where D represents the target's distance to the lens and d is the distance of the lens to the image sensor. H is the target's assumed height, and h is the target's projected height in the image. Figure 3 visualizes the equation example with the same object at two different distances based on the principle of optics with a 50 mm focal length lens. Note that the automatic labeling of the bounding box is at the pixel level, and a conversion of the size at pixels to the mm unit is necessary.

$$d\ (\text{mm}) = \frac{D(\text{m}) \times h\ (\text{mm})}{H(\text{m})} \tag{1}$$

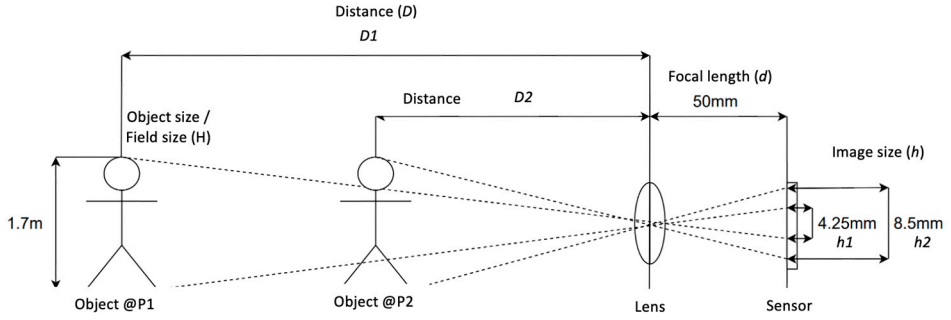


Figure 3. Example of projected image size of an object at two different distances.

2.6. Collision Risk Assessment Model

Given the known camera attached to the subject equipment and the types of identified target objects, CRAM first calculates the Risk Index for the event (RI_e), which represents the collision risk from the perspective of the specific camera of the subject for each target object in the single image captured at some moment. As shown in Equation (2), RI_e is the sum of the risk levels of the subject and each target object (R_s and R_o) within a prescribed distance range. If necessary, one may increase the weight of the assigned risk level for certain orientations or postures of the object (w_o) if the orientation or posture poses more risk to the object (e.g., due to limited view scope, slower response). The value of R_s and R_o is in the range of 0 to 2 to match the idea of three levels of alerts, i.e., green, yellow, and red lights, representing safe, caution, and near-collision statuses. Note that R_s, R_o, and w_o are arbitrarily assessed by safety experts for each type of subject and object with different orientations or postures. For example, for an excavator, R_o may be assigned a value of 2 if the target is a worker and 1 if the target is a forklift. A value of 0.5 for w_o may be assigned as the added weight to the risk for the situation when a worker is facing away or kneeling.

When an image consists of multiple target objects (i.e., multiple events), the Risk Index for the image (RI_i) is the maximum of all RI_e of the image (Equation (3)). When multiple cameras are attached to different parts of the subject equipment, the Risk Index for the equipment (RI_E) is the weighted sum of RI_i of all cameras attached to the equipment (Equation (4)). The RI_E is then used to determine if an alarm should be generated based on

prescribed thresholds at that moment. We suggested three levels of alarms: green, yellow, and red lights, representing safety, caution for objects approaching in collaboration, and near-collision dangers, respectively. Any moment when RI_i or RI_E exceeding 2 occurs can be considered a red light situation, and an alarm sound is fired. Similarly, the total collision risk of the entire operation of related equipment, termed RI, can be measured by the weighted sum of RI_E of all equipment involved in the operation. For customization of some cameras at special positions, a red-light threshold can be adjusted from the default value of 2 by considering the collision risk pertaining to the subject equipment and the camera attachment position.

$$RI_e = R_s + (1 + w_o) \times R_o \tag{2}$$

R_s = 0~2, risk level of the subject and camera location;
w_o = 0~1, added weight for the object's orientation or posture;
R_o = 0~2, risk level of the target object.

$$RI_i = Max\ (RI_e),\ e = 0\sim m,\ e \in m \text{ target objects in image } i; \tag{3}$$

$$RI_E = \sum (w_i \times RI_i),\ i = 0\sim n,\ i \in n \text{ cameras attached to equipment } E; \tag{4}$$

3. Prototype Implementation

This section describes the prototype implementation, including setting up the operable virtual construction site, capturing operation images for training YOLO, and conducting experiments to evaluate the system's performance.

3.1. Virtual Construction Site

Two construction operation task scenarios were designed and prepared, as shown in Figure 4. The first task involves two excavators working collaboratively in close range in a limited area for an excavation job that requires a speed-up to meet the job deadline. The second scenario involves a forklift stacking and unloading construction materials while a worker performs other inspections. Both scenarios have collision risks during the task.



Figure 4. Virtual construction site with collaborative excavators (**left**), forklift, and worker (**right**).

All three pieces of equipment were designed according to the specifications. For example, the excavator is based on the specifications of PC130-11 manufactured by Komatsu Ltd. (Tokyo, Japan) [41]. The equipment will be operated by the participants in the following experiment. Virtual workers are designed to perform their tasks automatically according to the prescribed script with three main postures, walking, sitting, and kneeling, as shown in Figure 5.

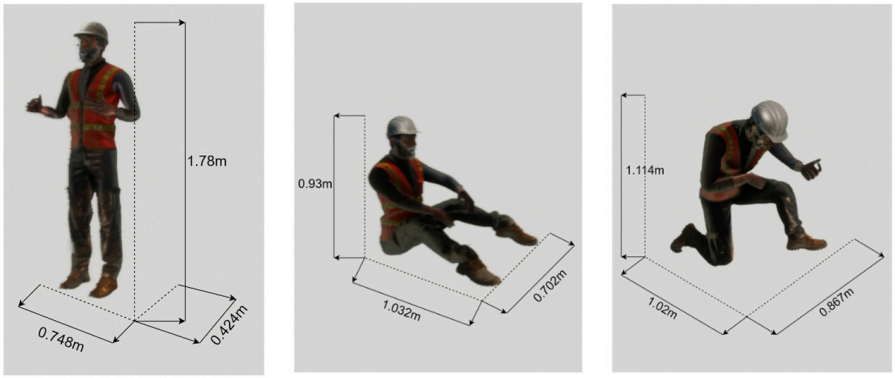


Figure 5. Postures of walking, sitting, and kneeling for virtual workers.

3.2. Training of Recognition Model

Mask-RCNN was first used to develop the recognition module for this research. Later on, given the emergent nature of the hazardous events to be identified in this study and the model training conducted on a laptop in the field with limited computational power, a one-stage algorithm was chosen to reduce the required time and the load on computing. YOLO, being a one-stage model, was selected due to its excellent performance in both mAP and detection speed, making it suitable for this study's image recognition tasks.

The YOLOv7 was built using Python. Various data augmentations were adopted to prepare the images for the training of YOLO, including FlipLR, Mosaic, Mixup, and Scale. The FlipLR setting was configured at 50%, meaning there was a 50% chance of flipping an image horizontally to simulate different angles captured by cameras during operation. The Mosaic setting was at 100%, combining four random images into one mosaic-style training sample to help the model handle complex scenarios, like overlapping or hidden objects. The Mixup setting was set to 50%, meaning pairs of images and their labels were combined 50% of the time to improve the model's ability to recognize variations and detect similar objects. The Scale setting was also 50%, resizing input images randomly to help the model adapt to objects of varying sizes, mimicking real-world conditions where objects appear at different distances. For hyperparameters, the configurations were as follows: Epochs = 100, Batch_size = 8, Optimizer = Adam, Learning Rate = 0.0005, and LR_Decay = Linear_Decay.

3.3. Object Recognition Result

Two training datasets were used to train the model. One is the images of target objects without backgrounds, and the other is the images of target objects with backgrounds, which will be referred to as the target-only case and target-with-background case, respectively, in the following presentation of the model performance. In both cases, the trained models were used to test their recognition accuracies for the target objects with backgrounds.

Figure 6 shows the confusion matrix of the trained model's result, which includes three user-defined object categories: Forklift, Excavator, and Man, and the Background_FP category, i.e., non-target objects. The matrix on the left represents the precision matrix for the target-only case, while the right represents the precision for the tar-get-with-background case. The accuracy rates are 100% for the objects of Forklift, Excavator, and Man in both cases. However, notice that, without backgrounds, there was a 50% chance that both Excavator and Man were treated as backgrounds. With the noise of backgrounds, there was a 23% and 77% chance that Excavator and Man were treated as backgrounds, respectively, indicating the identification of *Man* in backgrounds was easier to miss.

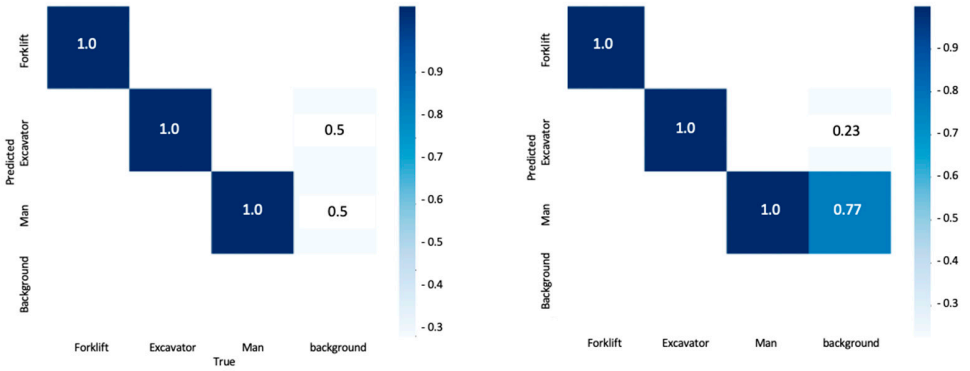


Figure 6. Confusion matrixes for the target-only case (**left**) and target-with-background case (**right**).

Figure 7 is the PR_curve (Precision-Recall Curve) result for the model, a plot generated from precision and recall values. The curve shows the trade-offs between precision and recall at varied thresholds. It shows that the model performs quite well in both cases, with mAP (mean Average Precision) of 0.998 and 0.997 for the target-only and target-with-background cases, respectively.

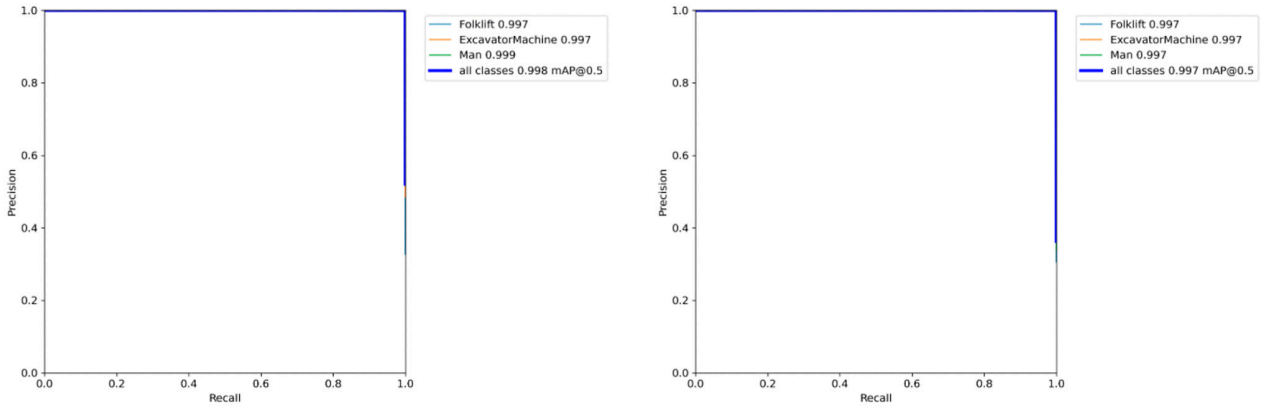


Figure 7. PR-Curves for the target-only case (**left**) and target-with-background case (**right**).

The F1 Score is the harmonic mean of precision and recall, providing a balanced assessment of a model's performance while considering both false positives and false negatives. Figure 8 shows the F1-Curve for the model in both the cases with and without background, which is a plot generated based on the F1-Score at various thresholds. It can be observed that the F1-Score reaches F1 = 1.0 at 0.736 and 0.690 for the target-only and target-with-background cases, respectively.

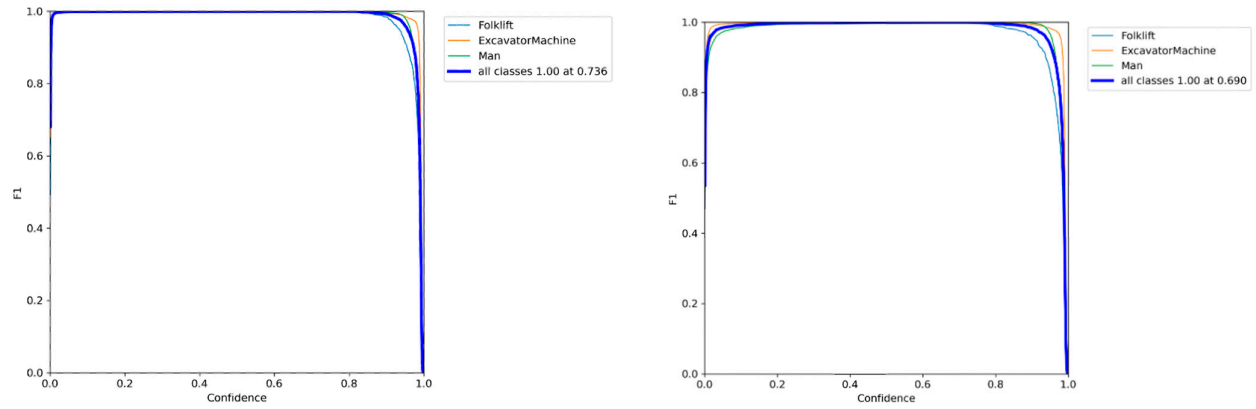


Figure 8. F1-curves the target-only case (**left**) and target-with-background case (**right**).

Figures 9 and 10 show the class-wise breakdown of performance metrics, e.g., Box, Objectness, Classification, for the target-only and target-with-background cases, respectively. Notice that overfitting was observed based on the increasing Val_Objectness curve, which measures how well the model identifies an object in a proposed region of interest. This is because this study continued training the model based on the pre-trained YOLO model, which already had pre-trained weights. In such a case, over-fitting occurred easily.

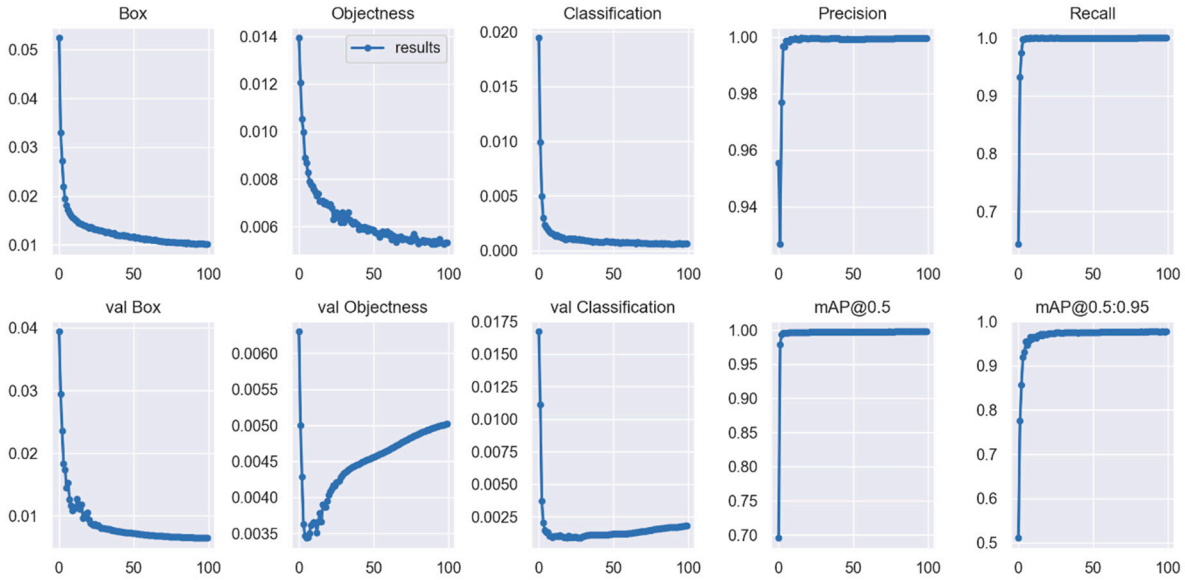


Figure 9. Class-wise performance metrics for the target-only case.

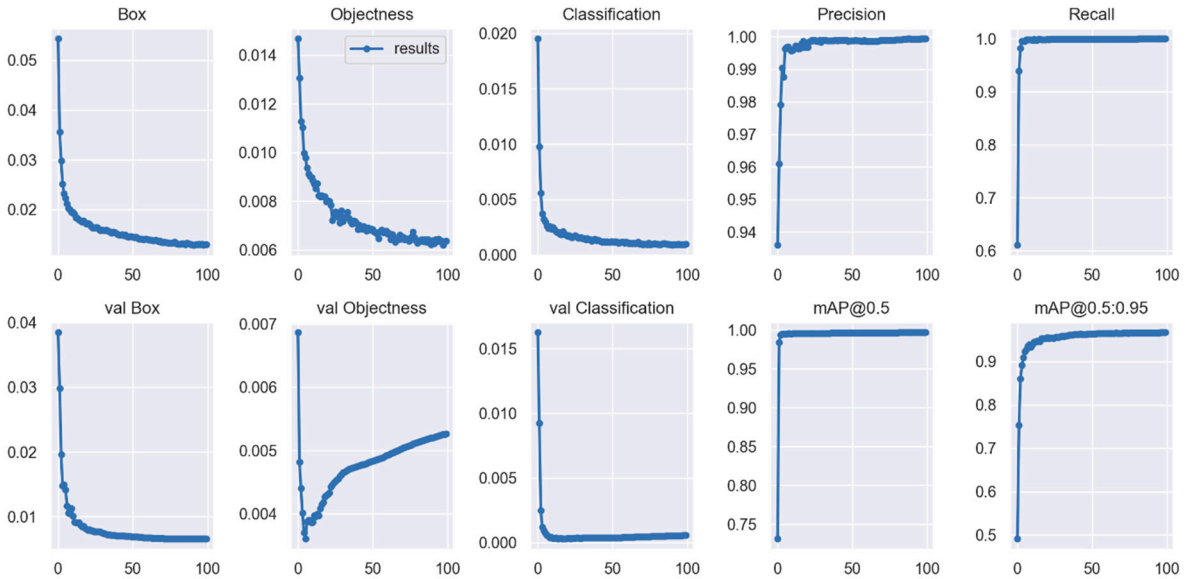


Figure 10. Class-wise performance metrics for the target-with-background case.

To illustrate the indication of the performance indexes presented previously, Figures 11 and 12 are the model's recognition examples in the target-only case. Most target objects can be correctly identified, but sometimes with offset bounding boxes. In addition, some backgrounds are also misidentified as targets. As for comparison, Figures 13 and 14 are the model's recognition examples in the target-with-background case, no backgrounds are misidentified as targets.



Figure 11. Recognition example in the target-only case (1).



Figure 12. Recognition example in the target-only case (2).



Figure 13. Recognition example in the target-with-background case (1).



Figure 14. Recognition example in the target-with-background case (2).

3.4. Alert Distance Threshold

Different types of equipment require different alert distance thresholds because their operation may impose different risks when a collision occurs. Currently, there are no local regulations regarding the operation safety range for each type of equipment. Article 8 of the Safety Standard of Machinery, Equipment, and Tools [42] only regulates the stopping performance of the two-hand control safety device or sensor-based safety device specified. Therefore, the potential collision range, considering the equipment's maximum operation range and risk buffer, needs to be determined. For brevity, this subsection uses only the excavator as an example to illustrate how appropriate thresholds can be determined.

The first step is to determine the distance range within which collision may occur. Using the Komatsu PC130-11 excavator as an example, as shown in Figure 15, the maximum digging radius is 12.86 m, which can be measured from the tip of the bucket to the center of the cabin's rotational axis when the arm and bucket are extended to a horizontal position. For the camera attached at the front of the cabin, 10.44 m is the distance range within which target objects may have collisions, which can be measured based on the Unity model. Additionally, the distance from the outermost edge of the rear of the excavator to the center of the cabin's rotational axis is measured at 3.05 m, making the rear operating radius 3.05 m. For the camera attached to the rear of the cabin, which is 2.51 m from the rotation center, 0.54 m (=3.05 − 2.51) is the collision distance range.

The second step is determining the safety buffer distances that should be added to the distance range as the thresholds for the red and yellow alerts. The determination of threshold values for various equipment and camera positions, in addition to considering the actual distance range where collisions may occur, should also account for factors such as the severity of potential injury caused by a collision, the driver's visibility, and reaction time. These considerations help establish an appropriate safety buffer distance, which can be used to set the threshold values for red-light collision warnings and yellow-light proximity cooperation signals. Table 1 provides comparative examples of threshold settings for the cameras on forklifts and excavators, both front and rear.

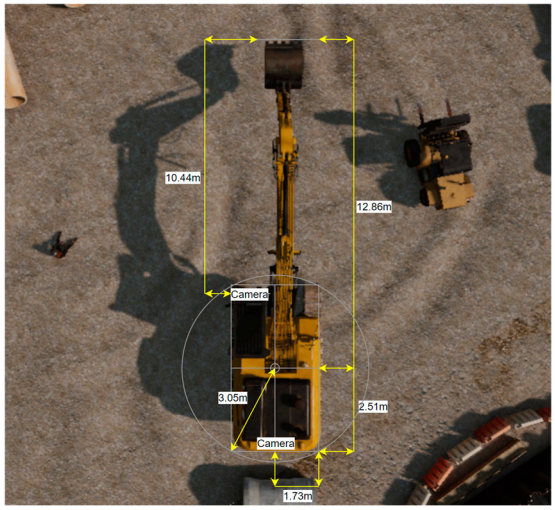


Figure 15. Determination of collision distance range.

Table 1. Thresholds for different cameras for the forklift and excavator when facing a worker.

				Unit: m
Camera Location	**Collison Distance**	**Buffer**	**Thresholds**	
			Near-Collision	**Caution**
Excavator (Front)	10.44	1.0	11.44	14.4
Excavator (Rear)	0.54	1.5	2.04	5.0
Forklift (Front)	1.59	0.5	2.09	4.1
Forklift (Rear)	0.51	1.0	1.51	3.5

3.5. Assessment of Risk Index for CRAM

When an image of an object approaching the subject equipment, DCAS detects its type and distance. If the object is within the corresponding alert distance threshold, determined as described in the previous subsection, the conventional static-threshold mechanism will generate an alert at this time. In contrast, DCAS will use CRAM to determine if an alert needs to be generated and what type of alert is appropriate. CRAM provides a quantifiable mechanism for assessing collision risk dynamically, given a scenario considering the types of subject equipment and objects, their heading or gesture, and distances in between. The following uses the excavator as an example to describe how the parameters were determined.

Three experts, including a safety and health manager, an excavator, and a dump truck driver, helped us determine the level of risk for the distances and the parameters required by CRAM. The experts assessed a risk level from three levels, i.e., 0 (no-risk), 1 (caution), and 2 (risky alert), for each type of subject equipment (i.e., excavator, dump truck, and forklift) with different object situations.

3.5.1. Determination of Risk Level for Distance

Table 1 uses some examples to illustrate how the near-collision and caution distance thresholds were determined for different risk levels. Column 'Camera location' lists the cameras of different equipment and installation locations. The Column 'Collision distance' lists the potential physical distances that collision may occur during the equipment operation, determined using the method described in Section 3.4. The Column 'Buffer' lists the error buffers for different cameras. For example, if the actual collision risk distance for a forklift is 1.59 m when the target object is a worker, an additional buffer of 0.5 m is added to

the front camera (totaling 2.09 m), and a larger buffer of 1.0 m is added to the rear camera's collision distance of 0.51 m (totaling 1.51 m) due to the increased difficulty of the driver's visibility and distance judgment. The collision distances adjusted by the error buffers are the thresholds for the near-collision situation. The experts suggested an arbitrary buffer of an additional 3 m as the caution situation (i.e., range requiring caution).

Comparatively, the excavator's buffer distances are larger, with 1.0 m and 1.5 m for the front and rear cameras, because of its larger size and the potential for more severe damage when the arm rotates.

3.5.2. Determination of RI_e

When an object within the alert range is detected by an equipment camera, an appropriate alert (i.e., no-risk green, caution yellow alert, or risky red alert with alarm sound) is generated based on CRAM's evaluation of RI_e for the event. Table 2 illustrates the values of the parameters required to calculate RI_e for the excavator when facing different situations of a dump truck or a worker. For example, under Column 'Camera location', consider the rear camera, which is the camera attached to the rear side of the excavator's cabin and thus has an R_s of 2 because the rear side, especially the bottom of the rear side, is the area the driver cannot easily pay attention to all the time. The Columns 'Object type' and '$(1 + w_o)\ R_o$' represent the assessed weighted risk value when the rear camera detects each situation of the objects. When the excavator's rear faces a truck's front, rear, and side, the weighted risk values are 0, 2, and 1, respectively. When facing the worker's front and rear, the weighted risk values are 0 and 2, respectively. Both the fronts of the truck and the worker are assigned 0 because they can see the movement of the excavator and thus not a risky situation from this perspective. Both the rears of a truck and a worker are assigned 2 because it is difficult for them to see the movement of the excavator at all times.

The Column 'Static' represents the alert generated by the conventional static-distance threshold mechanism. Since the object has entered the operational range of the excavator in this table, the static mechanism fires an alarm sound for all situations. On the other hand, Column 'R_e' shows the aggregated R_e risk values for the same situations and results in different values, i.e., 2, 4, and 3 for the truck and 2, 4 for the worker. Based on the value of this column, a yellow light without an alert sound will be generated if R_e is 2, and a red light with an alert sound will be generated if R_e is 3 or 4.

Table 2. CRAM's parameters for Excavator (EXV) with Dump Truck (DT) and Worker (WK). R: Red light alert with alarm; Y: Yellow caution; G: Green safe.

Equip.	Camera Location	R_s	Object Type	$(1 + w_o)\ R_o$	Static	R_e	CRAM
EXV	Front	0	DT-Front	0	R	0	G
			DT-Rear	2	R	2	Y
			DT-Side	1	R	1	G
			WK-Front	0	R	0	G
			WK-Rear	2	R	2	Y
	Rear	2	DT-Front	0	R	2	Y
			DT-Rear	2	R	4	R
			DT-Side	1	R	3	R
			WK-Front	0	R	2	Y
			WK-Rear	2	R	4	R
	Left	1	DT-Front	0	R	1	G
			DT-Rear	2	R	3	R
			DT-Side	1	R	2	Y
			WK-Front	0	R	1	G
			WK-Rear	2	R	3	R

Table 2. *Cont.*

Equip.	Camera Location	R_s	Object Type	$(1 + w_o)\ R_o$	Static	R_e	CRAM
EXV	Right	1	DT-Front	0	R	1	G
			DT-Rear	2	R	3	R
			DT-Side	1	R	2	Y
			WK-Front	0	R	1	G
			WK-Rear	2	R	3	R
	Bucket Arm	2	DT-Front	0	R	2	Y
			DT-Rear	2	R	4	R
			DT-Side	1	R	3	R
			WK-Front	0	R	2	Y
			WK-Rear	2	R	4	R

4. Experiments

Two experiments were conducted for this study. The first experiment aims to test the distance measurement errors of the proposed image-based method. The experiment includes two scenes; one involves two excavators working collaboratively for excavation work on site, and the other involves a forklift stacking materials in an area while three workers work in a nearby area with different gestures. The second experiment aims to test how many false alarms can be reduced by DCAS compared to the conventional static alert mechanism. This experiment involves an excavator and a bulldozer for site work.

4.1. Experiment I

In Experiment I, cameras were attached to the front and rear of the cabins of both the excavator and forklift. Table 3 shows the average percentage of errors for the image-based distance measurement, assuming the distance between Unity's Collider bounding boxes between the subject and object is the true measurement.

Table 3. The average percentage of errors for distance estimation.

Camera Position	Target Object	Average Errors
Excavator @ front	Excavator	16.65%
Excavator @ rear	Excavator	13.39%
Forklift @ front	Worker standing	11.19%
	Worker kneeling	8.13%
	Worker sitting	7.36%
Forklift @ rear	Worker standing	6.33%
	Worker kneeling	7.94%
	Worker sitting	7.15%

From the experiment, it was observed that, apparently, larger distance errors occurred with the excavator's camera when identifying the target excavator compared to the forklift's scene. This is primarily because the front camera of the excavator might misinterpret the target excavator's object-bounding box due to the interference from simultaneous capture of the subject's own digging arm. Additionally, when the target object is an excavator, the adumbration change in the target is prone to result in misidentification, leading to errors in distance estimation. The following discusses the main causes of the observed errors.

(1) Obstruction interference from moving machine parts

When machinery has large, movable, and irregular parts (such as the excavator's joints connecting two long arms and the irregularly shaped bucket), the recognition is easily

affected by obstruction from its own components during movement. If the target possesses similar characteristics, it is also likely to be misidentified by the YOLO model as multiple objects, resulting in distance estimation errors. The interference from such obstruction may result in false, multiple identifications of the target, as the four red boxes shown on the left of Figure 16, or incorrect dimensions of the object bounding box for the target, as the reduced height of the box for the worker on the right of Figure 16. The reduced height of the object bounding box may cause the estimated distance to appear farther than the actual distance.



Figure 16. The target is obstructed by the subject's moving part.

(2) Perspective distortion from moving machine parts

When machinery with large, movable components approaches a camera, the perspective effect may cause errors in the dimension of the object-bounding box identified by YOLO. For example, in Figure 17a, the digging arm of the target excavator is elevated, increasing the height of the box and making the estimated distance appear closer than it actually is.



(**a**) (**b**)

Figure 17. (**a**) Perspective distortion interference. (**b**) Ground inclination interference.

(3) Inclination interference

The camera may be tilting due to inclined terrain or other reasons, which may result in incorrect pixel values of the object box captured by YOLO and lead to errors in distance estimation. As shown in Figure 17b, inclined ground caused the camera to tilt, distorting the object box for the target worker, resulting in the perceived distance being greater than the actual distance.

In summary, the estimation error caused by the perspective distortion tends to underestimate the target's distance from the subject and result in false alarms. As will be

explained later in the second experiment, considering the large number of false alarms CRAM can reduce, this type of error is acceptable. On the other hand, the error caused by the obstruction and inclination tends to overestimate the target's distance from the subject and may result in Type-II error, i.e., does not generate an alarm when a near-collision occurs. This type of error has the potential risk of missing the warning when the system should have generated it. Thus, a quick remedy is to enlarge the buffer for the distance threshold setting if these two causes exist in the scenarios at hand. In addition, changing the regular cameras to those with automatic level correction functionality may also reduce the error caused by inclination.

4.2. Experiment II

Experiment II aims to measure the percentage of reduction in false alarms by CRAM compared to the conventional static threshold mechanism [33]. Another collaborative operation scene involving an excavator and dump truck was built. The random movement of the two workers was animated according to some prescribed routes. Cameras were attached to the operable machines' front, rear, and sides, and an additional camera was attached to the arm joint of the excavator, as shown in Figure 18. Each task required two persons to operate the excavator and the truck to collaboratively complete the prescribed task of excavation and dumping of soil. Three groups of participants ran the tasks, each taking about 45 to 70 min. The experts helped determine the true red alert situation (i.e., near-collision or worthy of red-alert warning). The event was considered a red-alert event as long as at least one expert considered it a red-alert event. The static alarm threshold is set to 3 m for the excavators and 1.5 m for dump trucks for the conventional mechanism. Within these ranges, CRAM attempted to screen the conventional alarms to red, yellow, and green lights.

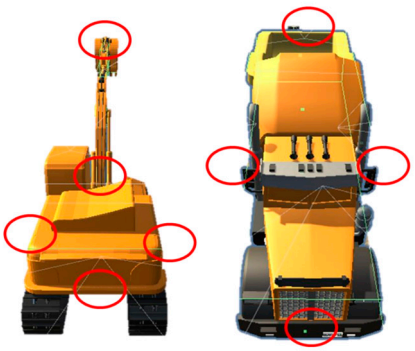


Figure 18. The camera locations are indicated within the red circles.

Throughout the experiment, only three collisions occurred; thus, most alarms were false. However, such a distinct separation could have been more practical because appropriate near-collision warnings are useful safety mechanisms for machine drivers. The objective should be to reduce the number of alarms generated on the conventional mechanism by distinguishing the near-collision alarms (i.e., red-light alarm) from the reasonable proximity required for collaboration (i.e., yellow light alarm). The near-collision alarms identified and labeled by the humans were assumed to be the true answers in the following performance evaluation.

CRAM's performance was measured based on the percentage reduction in false alarms against the conventional static threshold mechanism in two ways: with and without the automatic event recognition of the system. Based on the human interpretation of the events, the number of red-light alarms was 1118 based on the static thresholds and reduced to only 284 red-light alarms based on CRAM, which treated the rest of the alarms as 757 yellow

and 77 green lights. Considering that only three collisions occurred, 1118 alarms apparently were too many and might cause the insensitivity and ignorance of machine drivers. The CRAM reduced the number of alarms from 1118 to 284, a false alarm reduction rate of 74.6% (=(1118 − 284)/1118). When the automatic event recognition was used, there were 951 alarms based on the static threshold mechanism, and CRAM reduced the alarms to only 229 red light alarms and treated the rest as 616 yellow and 106 green lights. CRAM's false alarm reduction rate is 75.9% (=(951 − 229)/951). Thus, CRAM significantly reduced the number of alarms and made the alarms more meaningful than those generated based on the static thresholds with or without automatic event recognition.

We further investigated the accuracy of the categorization of alarm lights for automatic event recognition. Figure 19 compares the number of green, yellow, and red lights generated based on the system (automatic recognition) and human interpretation with the accuracies of 106 vs. 77, 616 vs. 757, and 229 vs. 284, respectively. Figure 20 further breaks down the comparison numbers from the perspective of each camera position. In general, high numbers of identified events concentrated on the excavator's front, left, and rear cameras, as well as the bulldozer's front camera. This indicates that the excavator tended to face, turn to the left, and turn back to the front of the bulldozer during the collaborative operation. For the red lights, the number of alarms generated is not too much different for both auto mode and human, except for the excavator's right and rear cameras.

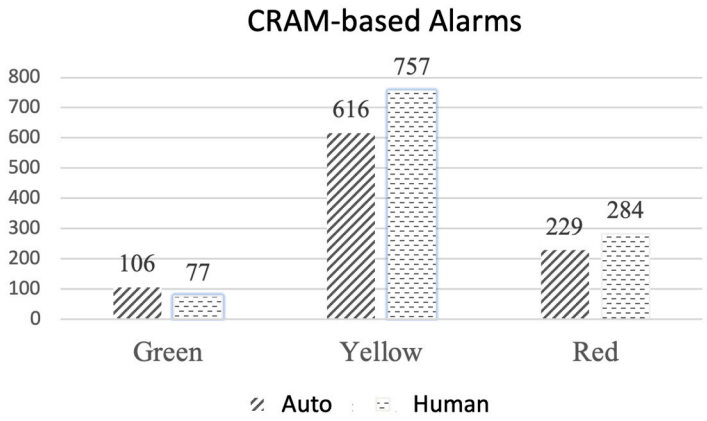


Figure 19. Comparison between auto and human event recognition based on CRAM.

Table 4 compares the accuracies of red alarms generated by different cameras. While most cameras had error rates under 25%, the excavator's right camera generated 12 red lights, while humans considered none of these were red light situations. If both yellow and red alarms were considered as a whole, the excavator's right camera generated 35 alarms (=23 + 12), while the human determined 38 alarms (=38 + 0). The number of alarms, then, was not too much different. Thus, in this situation, the system considered the 12 identified events to be red alarms according to CRAM's objective calculation, while the human considered the proximities that occurred in the 12 situations were still reasonable within the collaborative operation subjectively.

Table 4. Accuracy rates of red alarms for different camera positions.

Camera Position	Excav. (Front)	Excav. (Left)	Excav. (Right)	Excav. (Rear)	Excav. (Arm)	Dozer (Front)	Dozer (Left)	Dozer (Right)	Dozer (Rear)
Auto	0	28	12	68	22	0	50	19	30
Human	0	36	0	90	30	0	48	20	40
Error (%)	0%	22.2%	100.0%	24.4%	26.7%	0.0%	−4.2%	5.0%	25.0%

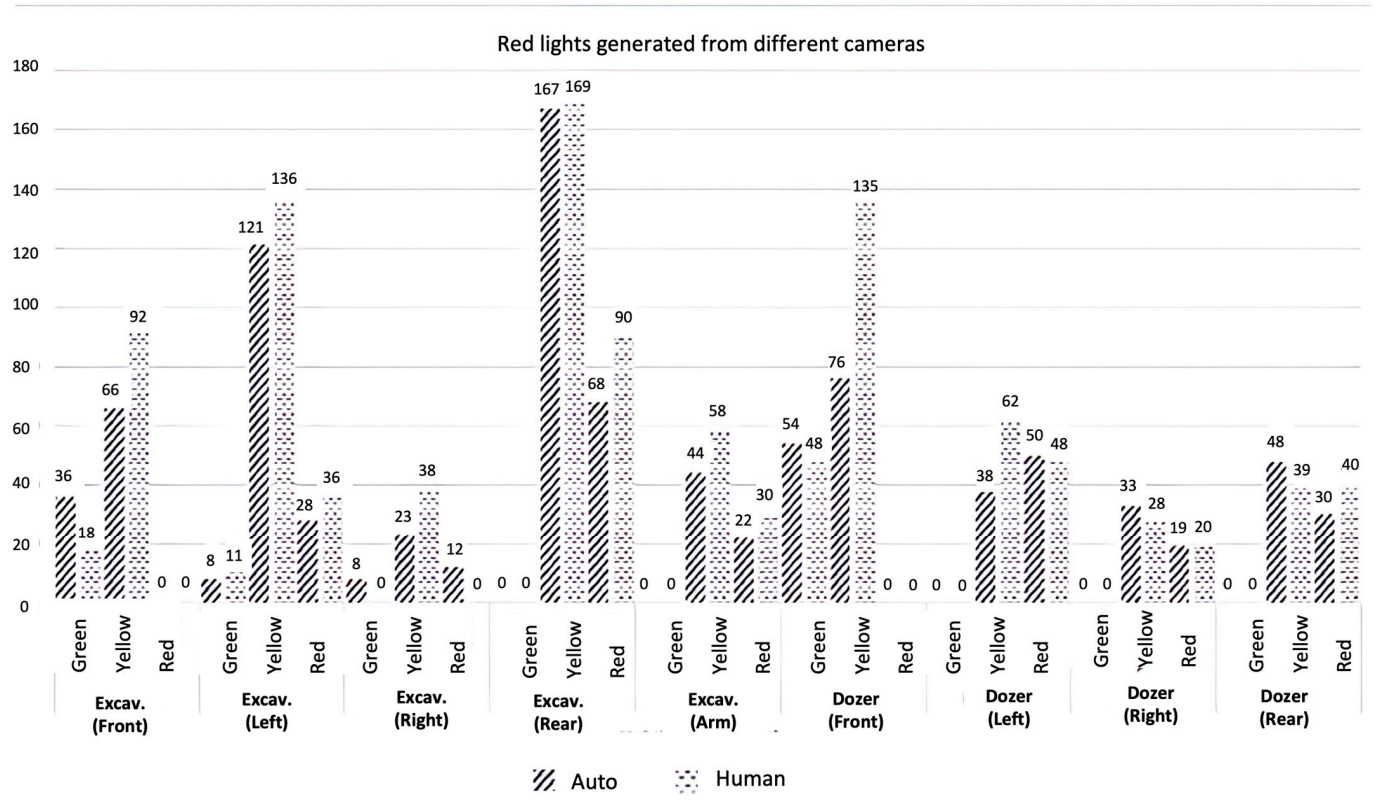


Figure 20. Comparison of red lights between auto and human event recognition based on CRAM.

5. Conclusions

Collaboration among multiple equipment and humans is common in the construction industry. Conventional static distance threshold mechanisms for near-collision alarms may result in too many false alarms during close collaboration and cause equipment drivers to be insensitive. This research aims to reduce false alarms by screening out necessary alarms from those within the static distance range and generating alarm sounds for only near-collision situations instead of collaborative proximity situations. This study proposes the DCAS that automatically recognizes the targets from the images of the cameras attached to the subject equipment, estimates their respective distance, and evaluates the collision risk based on the CRAM. CRAM allows a dynamic collision alarm mechanism for each type of subject equipment with respect to the camera positions and targets of different types of equipment and workers when they are approaching during collaboration. A near-collision alarm is only generated when the proximity risk of the subject and targets exceeds their respective thresholds based on their relative distances, which is also estimated based on image recognition and optics principles.

The result of the first experiment shows that the accuracy of the image-based distance estimation is feasible for practical use but may be easily degraded in the self-obstruction scenario or for the equipment with large and movable parts due to incorrect recognition of bounding boxes of target objects. Similar problems may still exist even if the system adopts precise distance measurement, such as laser-based techniques, while movable parts are out of laser scan coverage. Image-based measurement, although with less precision, may have more coverage than a single laser device with less cost.

This study also identifies three causes of distance estimation errors, namely obstruction, perspective distortion, and inclination. The estimation error caused by the perspective distortion tends to underestimate the target's distance from the subject but does not impose a serious problem. The errors caused by obstruction and inclination are comparatively

more serious because of possibly missing near-collision alarms. A quick remedy is to enlarge the buffer for the distance threshold setting and change the regular cameras to those with automatic level correction functionality. Nevertheless, to make the proposed technique practical, future research may still call for developing a model that can recognize not only the equipment but also its movable parts, and the correct bounding box for the inclined objects.

The result of the second experiment shows that CRAM successfully reduces a large number of false near-collision alarms for the equipment collaboration scenarios. Although a few cameras have large error rates compared to human interpretation and also, due to similar problems of self-obstruction and movable parts of equipment, most cameras maintain a similar number of alarms as human interpretation.

This research investigates and demonstrates the potential of using a vision-based model to automate collision risk detection and assessment for equipment collaboration. However, many challenges remain before it can be practically applied to construction sites. Future research will focus on minimizing factors interfering with object recognition and distance estimation, such as obstructions, perspective distortions, and inclinations. The reality gap will also be evaluated by comparing the current model with one trained on video footage of close-range collaborative equipment, which generally lacks collision or near-collision events. Lastly, the impact of large, crowded equipment collaboration must be studied and mitigated before the system is applied.

Author Contributions: Methodology, T.-L.H.; Software, T.-L.H.; Validation, T.-L.H.; Resources, R.-J.D.; Data curation, T.-L.H.; Writing—original draft, R.-J.D. and B.F.; Writing—review & editing, R.-J.D. and B.F.; Supervision, R.-J.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

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

References

1. Hinze, J.W.; Teizer, J. Visibility-related fatalities related to construction equipment. *Saf. Sci.* **2011**, *49*, 709–718. [CrossRef]
2. OSHA. OSHA Commonly Used Statics. Retrieved 30 December 2020. Available online: https://www.osha.gov/data/commonstats (accessed on 12 November 2024).
3. Jung, H.; Choi, B.; Kang, S.; Kang, Y. Temporal analysis of the frequency of accidents associated with construction equipment. *Saf. Sci.* **2022**, *153*, 105817. [CrossRef]
4. Vahdatikhaki, F.; Ammari, K.E.; Langroodi, A.K.; Miller, S.; Hammad, A.; Doree, A. Beyond data visualization: A context-realistic construction equipment training simulators. *Autom. Constr.* **2019**, *106*, 102853. [CrossRef]
5. Eurostat. Accidents at Work—Statistics on Causes and Circumstances, 2022. 2024. Available online: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Accidents_at_work_-_statistics_on_causes_and_circumstances (accessed on 5 November 2024).
6. JISHA. Fatal & Non-Fatal Accidents Caused by Large Causal Agents in Japan in 1999–2020, Retrieved 20 September 2024. Available online: https://www.jisha.or.jp/english/statistics/2020eCLA.html (accessed on 5 November 2024).
7. Zhang, Y.G.; Wang, W.S.; Mi, L.Y.; Qiao, L.J.; Tao, M. Multidimensional analysis and characterization of vulnerabilities among project managers in engineering safety accidents. *J. Saf. Environ.* **2024**, 1–12. (In Chinese)
8. Cheng, M.Y.; Ko, C.H. Computer-aided decision support system for hillside safety monitoring. *Autom. Constr.* **2002**, *11*, 453–466. [CrossRef]
9. Cheng, M.Y.; Ko, C.H. Automated safety monitoring and diagnosis system for unstable slopes. *Comput. Aided Civ. Infrastruct. Eng.* **2003**, *18*, 64–77. [CrossRef]
10. Rubio-Romero, J.C.; Rubio, M.C.; García-Hernández, C. Analysis of construction equipment safety in temporary work at height. *J. Constr. Eng. Manage.* **2013**, *139*, 9–14. [CrossRef]

11. Kazan, E.; Usmen, M.A. Worker safety and injury severity analysis of earthmoving equipment accidents. *J. Saf. Res.* **2018**, *65*, 73–81. [CrossRef] [PubMed]
12. Tang, N.; Hu, H.; Xu, F.; Zhu, F. Personalized safety instruction system for construction site based on internet technology. *Saf. Sci.* **2019**, *116*, 161–169. [CrossRef]
13. Zhu, Z.; Park, M.W.; Koch, C.; Soltani, M.; Hammad, A.; Davari, K. Predicting movements of onsite workers and mobile equipment for enhancing construction site safety. *Autom. Constr.* **2016**, *68*, 95–101. [CrossRef]
14. Yuan, C.; Li, S.; Cai, H. Vision-based excavator detection and tracking using hybrid kinematic shapes and key nodes. *J. Comput. Civil Eng.* **2017**, *31*, 04016038. [CrossRef]
15. Bohn, J.S.; Teizer, J. Benefits and barriers of construction project monitoring using hi-resolution automated cameras. *J. Constr. Eng. Manag.* **2010**, *136*, 632–640. [CrossRef]
16. Fullerton, C.E.; Allread, B.S.; Teizer, J. Pro-Active Real-Time Personnel Warning System. In Proceedings of the Construction Research Congress, Seattle, Washington, DC, USA, 5–7 April 2009; pp. 31–40.
17. Teizer, J.; Allread, B.S.; Fullerton, C.E.; Hinze, J. Autonomous pro-active real-time construction worker and equipment operator proximity safety alert system. *Autom. Constr.* **2010**, *19*, 630–640. [CrossRef]
18. Kim, D.; Liu, M.; Lee, S.; Kamat, V.R. Remote proximity monitoring between mobile construction resources using camera-mounted UAVs. *Autom. Constr.* **2019**, *99*, 168–182. [CrossRef]
19. Ruff, T.M. Overview of Proximity Warning and Technologies and Approaches. In Proceedings of the National Institute for Occupational Safety and Health (NIOSH) Workshop on Proximity Warning Systems for Mining Equipment, Washington, DC, USA, 15 September 2010.
20. Wang, J.; Razavi, S.N. Low false alarm rate model for unsafe proximity detection in construction. *J. Comput. Civ. Eng.* **2016**, *30*, 04015005. [CrossRef]
21. Ruff, T. Evaluation of a radar-based proximity warning system for off-highway dump trucks. *Accid. Anal. Prev.* **2006**, *38*, 92–98. [CrossRef] [PubMed]
22. Zhang, T.; Zhang, K.; Lin, J.; Louie, W.Y.G.; Huang, H. Sim2real learning of obstacle avoidance for robotic manipulators in uncertain environments. *IEEE Robot. Autom. Lett.* **2021**, *7*, 65–72. [CrossRef]
23. Horváth, D.; Erdős, G.; Istenes, Z.; Horváth, T.; Földi, S. Object detection using Sim2Real domain Randomization for robotic applications. *IEEE Trans. Robot.* **2022**, *39*, 1225–1243. [CrossRef]
24. Kang, K.; Belkhale, S.; Kahn, G.; Abbeel, P.; Levine, S. Generalization Through Simulation: Integrating Simulated and Real Data into Deep Reinforcement Learning for Vision-Based Autonomous Flight. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 6008–6014.
25. Tobin, J.; Fong, R.; Ray, A.; Schneider, J.; Zaremba, W.; Abbeel, P. Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 23–30.
26. Long, M.; Cao, Y.; Cao, Z.; Wang, J.; Jordan, M.I. Transferable representation learning with deep adaptation networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 3071–3085. [CrossRef]
27. Yu, W.; Tan, J.; Liu, C.K.; Turk, G. Preparing for the unknown: Learning a universal policy with online system identification. *arXiv* **2017**, arXiv:1702.02453.
28. Rusu, A.A.; Večerík, M.; Rothörl, T.; Heess, N.; Pascanu, R.; Hadsell, R. Sim-to-Real robot learning from pixels with progressive nets. In Proceedings of the 1st Annual Conference on Robot Learning, Zürich, Switzerland, 29–31 October 2018; Volume 78, pp. 262–270.
29. Traoré, R.; Caselles-Dupré, H.; Lesort, T.; Sun, T.; Cai, G.; Filliat, D.; Díaz-Rodríguez, N. Discorl: Continual Reinforcement Learning via Policy Distillation. In Proceedings of the NeurIPS Workshop on Deep Reinforcement Learning, Vancouver, BC, Canada, 8–14 December 2019.
30. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
31. 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] [PubMed]
32. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
33. Dzeng, R.J.; Syue, S.J.; Chen, Y.C. Data simulation for anti-collision detection in construction operations using virtual reality. *J. Chin. Inst. Civ. Hydraul. Eng.* **2023**, *35*, 779–791. (In Chinese) [CrossRef]
34. Roboflow, Inc. Roboflow Official Site. 2023. Available online: https://roboflow.com/ (accessed on 1 October 2024).
35. Bharati, P.; Pramanik, A. Deep Learning Techniques—R-CNN to Mask R-CNN: A Survey. In *Computing Intelligence in Pattern Recognition. Advances in Intelligent Systems and Computing*; Das, A., Nayak, J., Naik, B., Pati, S., Pelusi, D., Eds.; Springer: Singapore, 2020; Volume 999.

36. Sumit, S.S.; Watada, J.; Roy, A.; Rambli, D.R.A. In object detection deep learning methods, YOLO shows supremum to Mask R-CNN. *J. Physics Conf. Ser.* **2020**, *1529*, 042086. [CrossRef]
37. Wu, S.; Hou, L.; Zhang, G.K.; Chen, H. Real-time mixed reality-based visual warning for construction workforce safety. *Autom. Constr.* **2022**, *139*, 104252. [CrossRef]
38. Waranusast, R.; Riyamongkol, P.; Pattanathaburt, P. Distance Estimation Between Camera and Vehicles from an Image Using YOLO and Machine Learning. In Proceedings of the 2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Chiang Mai, Thailand, 7–10 November 2022; pp. 482–488.
39. Natanael, G.; Zet, C.; Foşalău, C. Estimating the Distance to an Object Based on Image Processing. In Proceedings of the 2018 International Conference and Exposition on Electrical and Power Engineering (EPE), Iasi, Romania, 18–19 October 2018; pp. 0211–0216.
40. Liang, W.Y. Smart Safety Distance Warning System for Motorcycles Based on YOLO v4 Image Recognition Technique. Master's Thesis, Department of Mechanic Engineering, Chung Yuan Christian University, Taoyuan City, Taiwan, 2022. (In Chinese)
41. Komatsu Ltd. PC130-11 Model Specifications, Last Visited on 21 September 2024. Available online: https://kcsj.komatsu/products/construction_machine/excavator_01/pc130-11 (accessed on 20 October 2024).
42. Ministry of Labor. Safety Standard of Machinery, Equipment, and Tools. 2022. Available online: https://law.moj.gov.tw/ENG/LawClass/LawAll.aspx?pcode=N0060034 (accessed on 20 October 2024).

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.