information

MDPI

Article

A Real-Time Advisory Tool for Supporting the Use of Helmets in Construction Sites

Ümit Işıkdağ [1], Handan Aş Çemrek [1], Seda Sönmez [1], Yaren Aydın [2], Gebrail Bekdaş [2,*] and Zong Woo Geem [3,*]

[1] Department of Architecture, Mimar Sinan Fine Arts University, 34427 Istanbul, Turkey; umit.isikdag@msgsu.edu.tr (Ü.I.); 20242109002@ogr.msgsu.edu.tr (H.A.Ç.); 20242109001@ogr.msgsu.edu.tr (S.S.)
[2] Department of Civil Engineering, Istanbul University-Cerrahpaşa, 34320 Istanbul, Turkey; yaren.aydin@iuc.edu.tr
[3] Department of Smart City, Gachon University, Seongnam 13120, Republic of Korea
* Correspondence: bekdas@iuc.edu.tr (G.B.); geem@gachon.ac.kr (Z.W.G.)

Abstract

In the construction industry, occupational health and safety plays a critical role in preventing occupational accidents and increasing productivity. In recent years, computer vision and artificial intelligence-based systems have made significant contributions to improving these processes through automatic detection and tracking of objects. The aim of this study was to fine-tune object detection models and integrate them with Large Language Models for (i). accurate detection of personal protective equipment (PPE) by specifically focusing on helmets and (ii). providing real-time recommendations based on the detections for supporting the use of helmets in construction sites. For achieving the first objective of the study, large YOLOv8/v11/v12 models were trained using a helmet dataset consisting of 16,867 images. The dataset was divided into two classes: "Head (No Helmet)" and "Helmet". The model, once trained, was able to analyze an image from a construction site and detect and count the people with and without helmets. A tool with the aim of providing advice to workers in real time was developed to fulfil the second objective of the study. The developed tool provides the counts of the people based on video feeds or analyzing a series of images and provides recommendations on occupational safety (based on the detections from the video feed and images) through an OpenAI GPT-3.5-turbo Large Language Model and with a Streamlit-based GUI. The use of YOLO enables quick and accurate detections; in addition, the use of the OpenAI model API serves the exact same purpose. The combination of the YOLO model and OpenAI model API enables near-real-time responses to the user over the web. The paper elaborates on the fine tuning of the detection model with the helmet dataset and the development of the real-time advisory tool.

Keywords: intelligent recommendation system; deep learning; YOLO; safety analysis; steel quantity

check for updates

Academic Editor: Gholamreza Anbarjafari (Shahab)

Received: 4 July 2025
Revised: 10 September 2025
Accepted: 22 September 2025
Published: 24 September 2025

Citation: Işıkdağ, Ü.; Çemrek, H.A.; Sönmez, S.; Aydın, Y.; Bekdaş, G.; Geem, Z.W. A Real-Time Advisory Tool for Supporting the Use of Helmets in Construction Sites. *Information* **2025**, *16*, 824. https://doi.org/10.3390/info16100824

Copyright: © 2025 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

With advances in technology, computer vision systems have become widely used in all industries. Similarly, computer vision systems are also utilized in various ways in the construction sector [1]. Computer vision systems, initially used to save time and reduce costs, have also begun to be used in the construction industry to manage health and safety issues using technological advances.

Computer vision is a branch of image processing. It is the process of extracting features from the pixels of an image. Object detection is a subfield of computer vision [2]. With the advancement of deep learning (DL) architectures, object detection can now be performed more accurately and quickly [3]. Recent studies show that object detection is becoming increasingly popular and facilitates processes in many areas [4].

Occupational health and safety (OHS) is a systematic set of measures aimed at protecting employees, the production process and equipment from potential risks. OHS not only ensures employee safety but also guarantees production continuity and product quality [5]. A helmet is a piece of personal protective equipment (PPE) that protects the worker against falling objects, bumps and impacts, as well as securing the head area against risks such as electric shock, molten metal splashes and burns.

The application prepared in this study was developed with a focus on the construction sector/construction sites and was focused on occupational health and safety. The purpose of the developed application was to analyze an image taken from a construction site to detect the status of the personal protective equipment of the employees and to provide information about possible occupational safety risks if the equipment was not present.

There are many studies in the literature on occupational health and safety and construction site management using computer vision. The following paragraph elaborates on some key examples. For example, Xiao and Kang [6] developed the Alberta Construction Image Dataset (ACID) specifically for construction machinery, collecting and annotating 10,000 images of 10 machine types. The dataset was tested with the YOLO-v3, Inception-SSD, R-FCN-ResNet101 and Faster-RCNN-ResNet101 algorithms, and the highest mAP was obtained by Faster-RCNN-ResNet101 with 89.2%. Araya-Aliaga et al. [7] trained the RetinaNet, Faster R-CNN and YOLOv5 models on a dataset involving the identification of formwork and reinforcement during construction using robotic process automation (RPA) and generative AI techniques. Among them, YOLOv5 outperformed the others and achieved the highest F1 score (63.7%) and precision (66.8%). Kazaz et al. [8] used object detection based on deep learning to overcome challenges in construction stormwater inspections. The system includes data preparation with aerial inspections, model training, validation and testing. Trained with 800 aerial images, the model detected four different applications with 100% accuracy and minimal false positives. The results showed that UAV imagery is effective in object detection and provides accurate results in site plan comparisons. Seth and Sivagami [9] improved the training of YOLOv8 for object detection and helmet detection in worker images using a Test Time Augmentation (TTA)-based approach. Using image transformations such as Histogram Equalization, they achieved a precision of 1. Barlybayev et al. [10] used YOLOv8, a fast object detection model, to detect personal protective equipment (PPE) worn by the worker and created the Color Helmet and Vest (CHV) and Safety HELmet dataset (SHEL5K) with 5K images consisting of eight object classes such as helmets, vests and goggles. YOLOv8m (medium) correctly classified the objects, showing an evaluation score of 0.929. Bai et al. [11] utilized the YOLOv8-seg (segmentation) model to address issues such as construction machine uncertainty and surface similarities. The YOLOv8n-seg model achieved an mAP value of 0.866. Jiao et al. [12] used unmanned aerial vehicles (UAVs) combined with an optimized YOLOv8 model to flexibly capture construction site images and used transfer learning to annotate and train labels for "person" and "helmet". The mAP@0.5 of all classes obtained by their method was 0.975. Biswas and Hoque [13] proposed a real-time computer vision system based on YOLOv8 to detect PPE, hard hats, masks, hard shoes, excavators and trucks at construction sites using a dataset of 1026 images. As a result of the study, the system achieved mAP values of 95.4% for Mask, 82.1% for Excavator, 83.8% for Hard Hat, 80% for Truck, 75.2% for No Hard Hat, 55% for PPE and 54.8% for No PPE. El-Kafrawy and Seddik [14] evaluated YOLOv12 for real-time

detection of PPE violations. In their study, where they trained on a dataset of 2047 images, the results showed that YOLOv12 provided the highest accuracy (box precision = 0.798, mAP50 = 0.553, and recall = 0.466) in detecting small and distant objects.

The growth of the global construction industry in recent years has emphasized the importance of safety, yet accidents continue to occur. This paper introduces a novel approach using the YOLOv8, YOLOv11 and YOLOv12 models for personal protective equipment (PPE) detection. The work is based on the integration of data processing with autonomous systems and object detection. YOLO is an algorithm that can perform better than previous YOLO versions and perform real-time object detection [15].

2. Materials and Methods

Figure 1 shows the flow of this study. The study started with the acquisition of an open-source helmet detection dataset; following this, three YOLO models were fine-tuned using the acquired dataset, and finally an interactive Streamlit application that utilizes the integrated YOLO and LLM was developed. The details of this process are elaborated below.

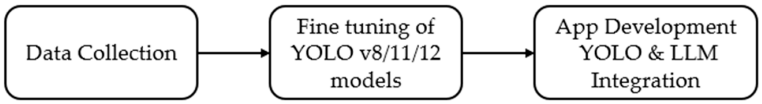


Figure 1. Study flow diagram.

2.1. The Dataset

In deep learning algorithms, the size and diversity of the dataset directly affects the training performance of the algorithm. In this study, open-source images were used to create the training dataset. The original version of the Hard Hat Workers Computer Vision Project dataset [16] used in the study contained 7035 images of 416 × 416 size. The split of the original dataset was 75/25% for train/test. Sample images of three classes from the dataset are shown in Figure 2.



Figure 2. Dataset examples (people wearing helmets, head-only images and full-body images (Helmet)) [16].

In the training of the computer vision models used in this study, the v13 of the dataset generated with a 70/20/10% train/valid/test split with preprocessing and augmentation was utilized. The dataset [16] contained 16867 images in total. The v13 dataset contained two classes: Helmet and Head (No Helmet).

2.2. The Models

The integration of artificial intelligence into the construction industry processes increases productivity in the field. In the applications of artificial intelligence (AI), visual data and sensor information are processed together to increase the environmental awareness of machines, while computer vision systems improve equipment tracking and downtime analysis, reducing operator errors. Artificial intelligence is a meta-concept that encompasses concepts such as machine learning and deep learning.

Machine learning (ML) is widely used in civil engineering such as the prediction of the compressive and splitting tensile strength of ceramic waste aggregate concrete [17], compressive strength in fly ash-modified recycled aggregate concrete [18], cooling load of tropical buildings [19], compressive strength of Ultra-High-Performance Concrete [20], compressive strength of high-strength concrete [21], splitting tensile strength of Basalt Fiber-Reinforced Concrete [22], modeling soil behavior [23], and soil classification [24,25]. With the development of computer vision technology, object detection applications are becoming increasingly common and deep learning is the most widely used method, thanks to its observed success in processing images and extracting features from images [26].

Deep learning is a subfield of artificial intelligence (AI). Deep learning is a method of training a neural network that can predict outputs on datasets. Neural networks are divided into three groups: input, hidden and output layers. Data is received in the input layer, mathematical operations are performed in the hidden layers and predictions are generated in the output layer. The connections within the network are expressed in weights and are initially assigned randomly. These weights determine the importance of the inputs, while activation functions standardize the outputs. The term "deep" refers to the presence of more than one hidden layer. For deep learning to be successful, large datasets and hardware with high processing power are needed. Convolutional Neural Networks (CNNs) are widely used deep learning models for computer vision tasks.

Computer vision tasks involve key tasks such as object detection, object classification and object tracking [27]. In this study, different versions of the YOLO model were utilized and tested along with a LLM.

2.2.1. The You Only Look Once (YOLO) Model

You Only Look Once (YOLO) is an ANN architecture that performs real-time object detection by reducing object detection to a single regression problem. YOLO was first introduced in the paper "You Only Look Once: Unified, Real-Time Object Detection," published on 9 May 2016. YOLO treats object detection as a single-step regression problem, from image pixels to bounding boxes and class probabilities [28]. YOLOv8 was developed by Ultralytics on 10 January 2023, and offers higher accuracy, speed, and flexibility compared to previous versions [29,30]. YOLOv11 is a YOLO model released by Ultralytics in December 2024. It is an improvement over YOLOv8 and YOLOv9, offering advances in speed, accuracy, and efficiency. Its backbone consists of C3K2 blocks and an SPPF module, its neck features FPN-PAN structures and a C2PSA mechanism and its head has a headless and separate head structure. The model produces multi-scale predictions and utilizes BCE, CIoU and DFL loss functions [31]. Released in 2025, YOLOv12 offers state-of-the-art technology in real-time object detection and increases efficiency in complex environments with small target recognition. Built on the YOLO architecture, the model combines four innovations: Attention, Multi-Scale Aggregation, Dynamic Anchor Assignment and Residual Efficient Layer Aggregation Networks (R-ELANs).

Table 1 gives a comparison of mAP50-95 values (Detection) for YOLO models trained on the Common Objects in Context (COCO) dataset. The COCO dataset consists of 330,000 images; 200,000 of them include object detection, segmentation, and annotations. It has 80 categories and covers both common and specialized objects. Performance is measured using mAP and mAR. Mosaicking is used during training to improve model generalization [32]. In the COCO benchmark, YOLOv12m achieves 52.5% Map [33]. The "n" is the fastest and lightest model and is suitable for real-time applications. The "m" version offers a balanced option between speed and accuracy and performs better in more complex object detection tasks [34].

Table 1. Comparison of mAP50-95 values (Detection) for YOLO models trained on the COCO dataset, including 80 pre-trained classes [33].

Model	mAP50-95
YOLOv8n	37.3
YOLOv11n	39.5
YOLOv12n	40.6
YOLOv11m	51.5
YOLOv12m	52.5

In the training process, a desktop computer with an NVIDIA GeForce RTX 3060 12 GB graphics card, Intel i7 13700 processor and Ubuntu 22.04 operating system were used to train the YOLO models with the v13 of the hard hat dataset. The results of the training process are provided in the following section.

The performance of the YOLO models was evaluated using widely adopted metrics in classification and object detection, including standard evaluation metrics (accuracy, recall, precision, F1 score and Mean Average Precision (mAP) [35]), training losses (box_loss, cls_loss, dfl_loss) [36], validation losses (box_loss, cls_loss, dfl_loss) and learning rates [37].

2.2.2. Large Language Models

Large Language Models (LLMs) are deep learning models based on transformer architectures residing in the domain of Natural Language Processing (NLP). In NLP, the use of LLMs is common in question answering. LLMs consist of an encoder, a decoder or both. LLMs are trained with a large dataset and understand the complexities in natural language, such as Bidirectional Encoder Representations from Transformers (BERT); OpenAI's Generative Pre-trained Transformer (GPT) series has attracted a lot of attention as a LLM and gained popularity in recent years [38,39]. Today, many commercial LLM developers provide API access to their models for business use. The "gpt-3.5-turbo" model of OpenAI was used with OpenAI API as the LLM for this study.

3. Results

As shown in Figure 3, the study started with the recruitment and preparation of the PPE dataset. Following this, YOLO models were fine-tuned to ensure that the model could detect PPE in images with high accuracy. After the training phase, a Real-Time Advisory Tool was developed as a proof-of-concept test environment to investigate the integration of the trained YOLO models together with a Large Language Model (i). for the detection of persons with and without a helmet at a construction site, and based on this detection, (ii). for providing a real-time safety assessment and advice to the users.

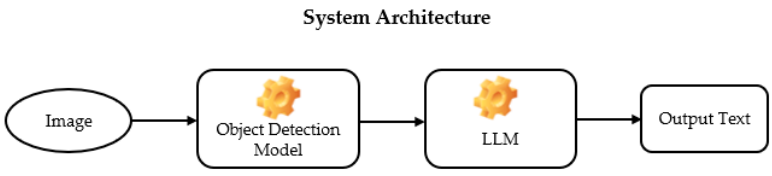


Figure 3. System architecture.

The following sections provide detailed information on development stages and the developed tool.

3.1. The Training Phase Outputs

The application is focused on occupational health and safety in construction sites. The purpose of the developed application was supporting the use of PPE (specifically

helmets) in construction sites. For this purpose, a computer vision model was fine-tuned to analyze an image taken from the construction site to provide information about the current status and also about the possible occupational safety risks if workers do not wear their helmets. In this context, firstly, YOLO models were trained/fine-tuned on the Hard Hat Workers dataset [16]. The dataset [16] contained 16,867 images in total. The dataset contained two classes: Helmet and Head (No Helmet). The training setup used a 70/20/10% train/valid/test split of the dataset.

The confusion matrices obtained at the end of training are presented in Figure 3. The confusion matrix is a table showing the relationship between the labels classified by the model and the actual labels [40]. The True Positives (diagonal) of the matrix indicate the correctly detected instances, Head–Head indicates correctly identified unsafe heads without helmets and Helmet–Helmet indicates correctly identified people wearing helmets. When we check the False Negatives (when we read down columns), Head–Helmet shows the number of bare heads misclassified as wearing a helmet (i.e., safety risk), Head–Background shows the number of bare heads completely missed (i.e., safety risk), Helmet–Head indicates a helmeted person misclassified as not wearing one (i.e., false alarm) and Helmet–Background shows a helmeted person completely missed (not a critical error but reduces detection rate). When we check the False Positives (reading across columns), Background–Head indicates where the Background is mistakenly identified as a bare head (i.e., false alarm), and Background–Helmet shows the number of instances where the Background is mistakenly identified as a helmeted person (i.e., false alarm).

Figure 4 shows the confusion matrices for the detection studies performed with YOLOv11m and YOLOv12m, respectively. According to the confusion matrix obtained with YOLOv11m, the model was quite successful for Helmet. This is because 3795 samples were correctly classified (67.07%). The performance for Head is good, but there are some mix-ups with Background; 1275 examples were correct (22.53%) and 161 examples were incorrectly predicted as Background (2.85%). Background, on the other hand, was not classified correctly at all; most examples were incorrectly predicted as Helmet (106 examples). The YOLOv12m model correctly classified 1279 examples as Head (22.54%) and misclassified only 10 examples as Helmet and 146 examples as Background. Helmet was recognized quite well by the model: 3815 examples were correctly classified (67.22%), and only 15 examples were misclassified as Head and 277 examples as Background. The confusion matrices were drawn using the tool developed by Perri et al. [40].

Validation Set

TRUE / PREDICT	Head	Helmet	Background	SUM
Head	1275 22.53%	12 0.21%	161 2.85%	1448 88.05% 11.95%
Helmet	19 0.34%	3795 67.07%	245 4.33%	4059 93.50% 6.50%
Background	45 0.80%	106 1.87%	0 0.00%	151 0.00% 100.00%
SUM	1339 95.22% 4.78%	3913 96.98% 3.02%	406 0.00% 100.00%	5070 / 5658 89.61% 10.39%

(a)

Validation Set

TRUE / PREDICT	Head	Helmet	Background	SUM
Head	1279 22.54%	10 0.18%	146 2.57%	1435 89.13% 10.87%
Helmet	15 0.26%	3815 67.22%	277 4.88%	4107 92.89% 7.11%
Background	45 0.79%	88 1.55%	0 0.00%	133 0.00% 100.00%
SUM	1339 95.52% 4.48%	3913 97.50% 2.50%	423 0.00% 100.00%	5094 / 5675 89.76% 10.24%

(b)

Figure 4. Confusion matrices of the hard hat dataset: (**a**) YOLOv11m; (**b**) YOLOv12m.

The confusion matrix illustrates that the accuracy rate for the validation set is 89.61% for YOLOv11m, and 89.76% for YOLOv12m. Table 2 shows the Basic Evaluation Metrics for the validation set.

The key evaluation metrics such as accuracy, precision, recall and F1 score were explained in the previous section. The Misclassification Rate is 1-Accuracy; in the calculation of Macro F1, F1 score is calculated independently for each class and then the average is taken for treating all classes equally regardless of their size. Weighted F1 calculates the F1 score for each class but takes a weighted average, where the weights are proportional to the number of true instances of each class.

Table 3 shows the performance metrics obtained during the training process of the model, such as the training and validation losses (loss), accuracy, precision, recall, mAP50, mAP50-95 and learning rate (lr) of the model at different epochs.

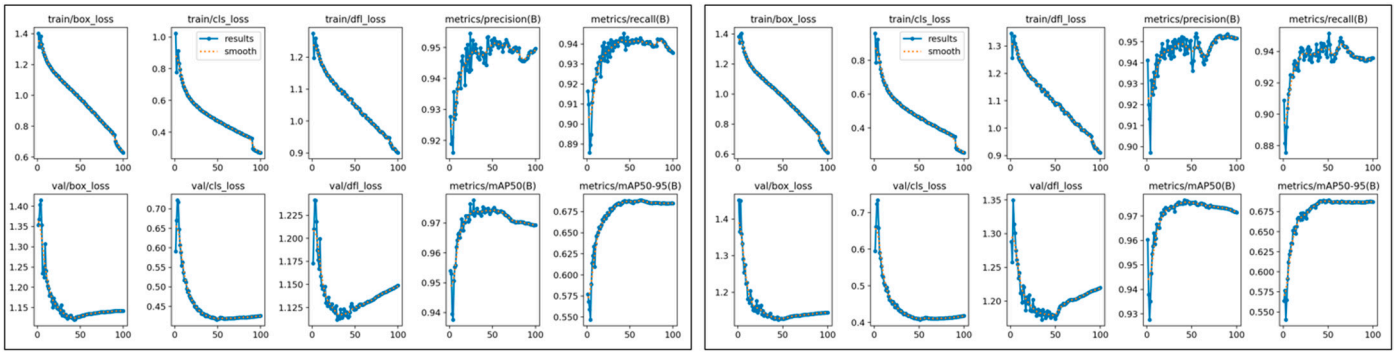
The metrics indicate that the training loss values decreased and the precision, recall, mAP50 and mAP50-95 improved during the training process. In particular, the mAP50 value increased from 0.9313 at the beginning to 0.9679 at the end of the training process. This supports that the object detection performance of the model has improved for helmets with the fine tuning of the model.

All versions of the YOLO models used were trained and tested on the Hard Hat Workers dataset [16], and the results obtained are shown in Table 4. Upon examining the results, it is observed that the highest performance is achieved with the YOLOv11m and YOLOv12m models. These models perform well in detection tasks because they have more parameters and a more complex structure.

Figure 5 shows the performance and loss graphs obtained during the training and validation of the YOLOv11m and YOLOv12m algorithms. As shown in Figure 6, after 50 epochs, the improvements in the model were slow and low; this is related to both the training strategy with a variable learning rate, which decreases in the later stages of the training, and the saturation (i.e., amount of knowledge obtained) of the model, which increases in the later stages.

The object detections obtained from the training and validation datasets are presented in Figures 6 and 7.

In order to provide more detailed information about the details of the training, the graphs with the measurement results (training metrics) are presented in Figures 8–11. Figure 8 is the Recall–Confidence Curve, Figure 9 is the Precision–Confidence Curve, Figure 10 is the F1–Confidence Curve and Figure 11 is the Precision–Recall Curve.



(a) (b)

Figure 5. Results for hard hat dataset: (**a**) YOLOv11m; (**b**) YOLOv12m.

Table 2. Basic Evaluation Metrics (hard hat dataset).

Model	Class Name	Precision	1-Precision	Recall	False Negative Rate (FNR)	F1 Score	Specificity (TNR)	False Positive Rate (FPR)	Accuracy	Misclassification Rate	Macro-F1	Weighted-F1
YOLOv11m	Head	0.9522	0.0478	0.8805	0.1195	0.9150	0.9848	0.0152	0.8961	0.1039	0.6223	0.9172
	Helmet	0.9698	0.0302	0.9350	0.0650	0.9521	0.9262	0.0738				
	Background	0.0000	1.0000	0.0000	1.0000	0.0000	0.9263	0.0737				
YOLOv12m	Head	0.9552	0.0448	0.8913	0.1087	0.9221	0.9858	0.0142	0.8976	0.1024	0.6245	0.9217
	Helmet	0.9750	0.0250	0.9289	0.0711	0.9514	0.9375	0.0625				
	Background	0.0000	1.0000	0.0000	1.0000	0.0000	0.9237	0.0763				

Table 3. The model training logs for each YOLO model.

Model	Epoch	Train/Box_loss	Train/Cls_loss	Train/dfl_loss	Metrics/ Precision(B)	Metrics/ Recall(B)	Metrics/ mAP50(B)	Metrics/ mAP50-95(B)	Val/Box_loss	Val/Cls_loss	Val/Dfl_loss	lr/pg0	lr/pg1	lr/pg2
YOLOv8n	1	1.4789	1.6877	1.2887	0.89417	0.87535	0.9313	0.49118	1.5729	0.94295	1.2703	0.00333	0.00333	0.00333
	50	1.1169	0.57672	1.0933	0.94847	0.93062	0.96811	0.66174	1.1725	0.45508	1.1052	0.005149	0.005149	0.005149
	100	0.92405	0.38676	1.0114	0.9519	0.93127	0.96793	0.66252	1.1737	0.44413	1.1072	0.000199	0.000199	0.000199
YOLOv11n	1	65.3311	1.47569	1.68443	1.33289	0.90634	0.86088	0.92357	0.49152	1.54324	0.95522	1.28459	0.00333	0.00333
	50	2773.16	1.12328	0.58735	1.09226	0.9461	0.93719	0.9714	0.67527	1.13352	0.44897	1.08096	0.005149	0.005149
	100	5670.2	0.92176	0.38898	1.00745	0.94832	0.93784	0.97037	0.68013	1.13031	0.43515	1.08172	0.000199	0.000199
YOLOv12n	1	223.192	1.46226	1.65184	1.30184	0.91392	0.87746	0.93669	0.53639	1.42353	0.88957	1.26034	0.00333	0.00333
	50	9078.57	1.10491	0.56182	1.14365	0.94452	0.9459	0.97142	0.67258	1.14483	0.43819	1.15867	0.005149	0.005149
	100	18220.3	0.88344	0.35808	1.03329	0.94337	0.9412	0.96677	0.67527	1.14635	0.42833	1.16127	0.000199	0.000199
YOLOv11m	1	199.879	1.40227	1.02382	1.27468	0.92764	0.91636	0.954	0.57687	1.35352	0.59099	1.17274	0.00333	0.00333
	50	10143.7	0.97952	0.47022	1.04803	0.95047	0.94135	0.97429	0.68721	1.1287	0.41706	1.12316	0.005149	0.005149
	100	33267.5	0.62764	0.26969	0.90073	0.94964	0.93551	0.96936	0.68489	1.14201	0.42561	1.14905	0.000199	0.000199
YOLOv12m	1	590.136	1.38181	0.96074	1.34625	0.94118	0.9088	0.96037	0.56379	1.45156	0.59458	1.28809	0.00333	0.00333
	50	24731.6	0.98494	0.46164	1.08454	0.93998	0.94636	0.97528	0.68883	1.12704	0.40958	1.17363	0.005149	0.005149
	100	37824.2	0.6086	0.25221	0.91	0.9517	0.93576	0.97147	0.68701	1.14465	0.41815	1.21958	0.000199	0.000199

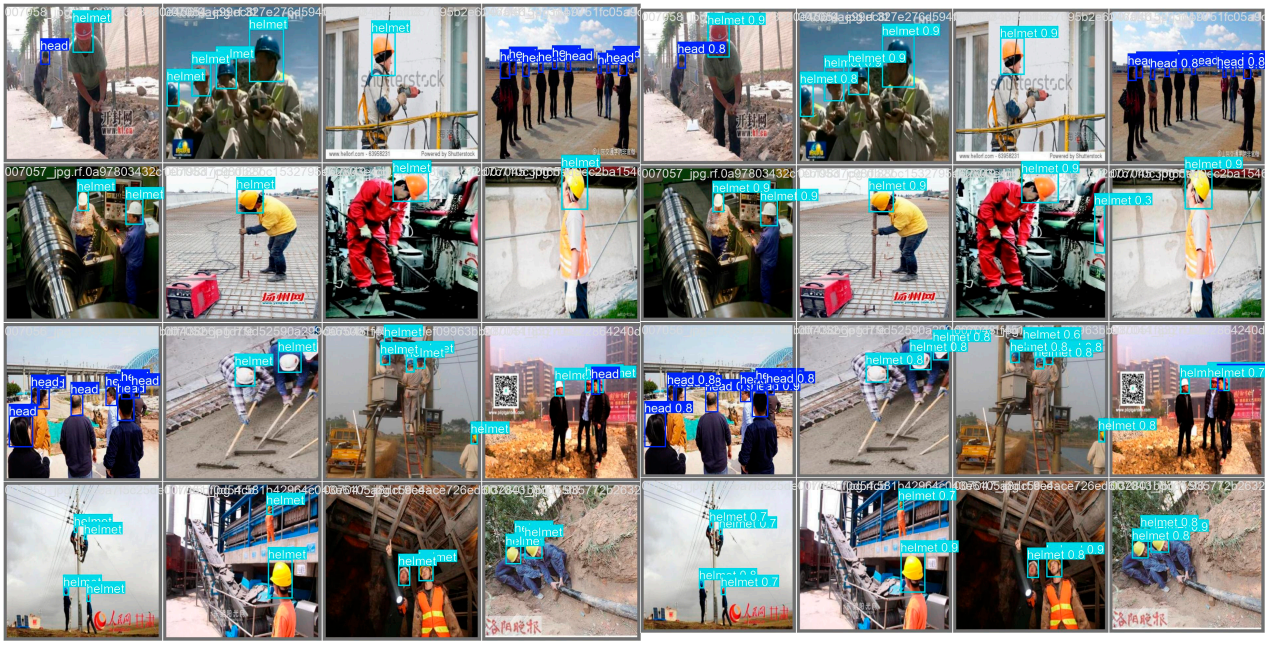
Table 4. Comparison of YOLO models based on performance metrics and best epochs.

Model	Metrics/mAP50(B)-Epoch no.	Metrics/mAP50-95(B)-Epoch no.
YOLOv8n	0.96936-55	0.66398-60
YOLOv11n	0.97198-51	0.68013-100
YOLOv12n	0.97142-50	0.67527-100
YOLOv11m	0.97766-28	0.68853-62
YOLOv12m	0.97648-42	0.68919-51



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

Figure 6. The object detections obtained from the training dataset (hard hat dataset) using YOLOv12m. (**a**) train_batch0; (**b**) train_batch1.



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

Figure 7. The object detections obtained from the validation dataset (hard hat dataset) using YOLOv12m. (**a**) val_batch0_labels; (**b**) val_batch0_pred.

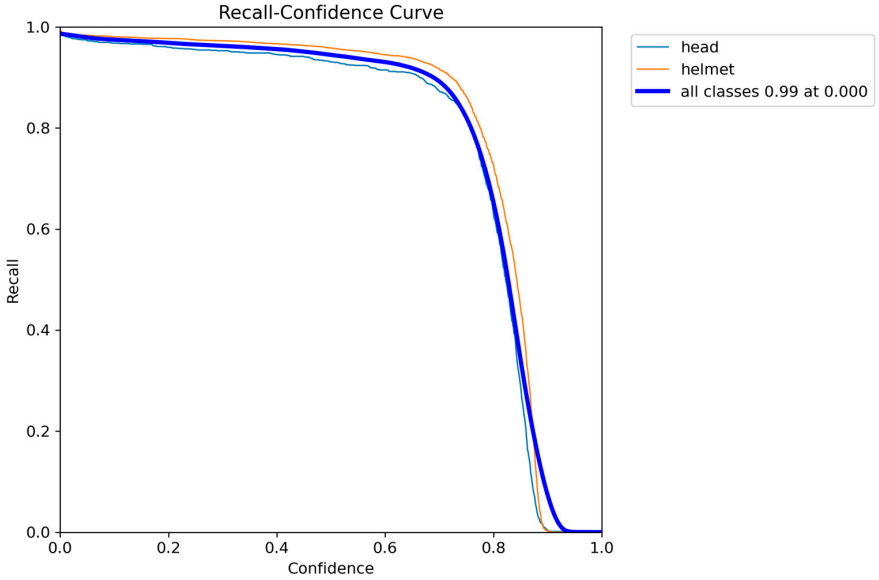


Figure 8. Recall–Confidence curve (YOLOv12m).

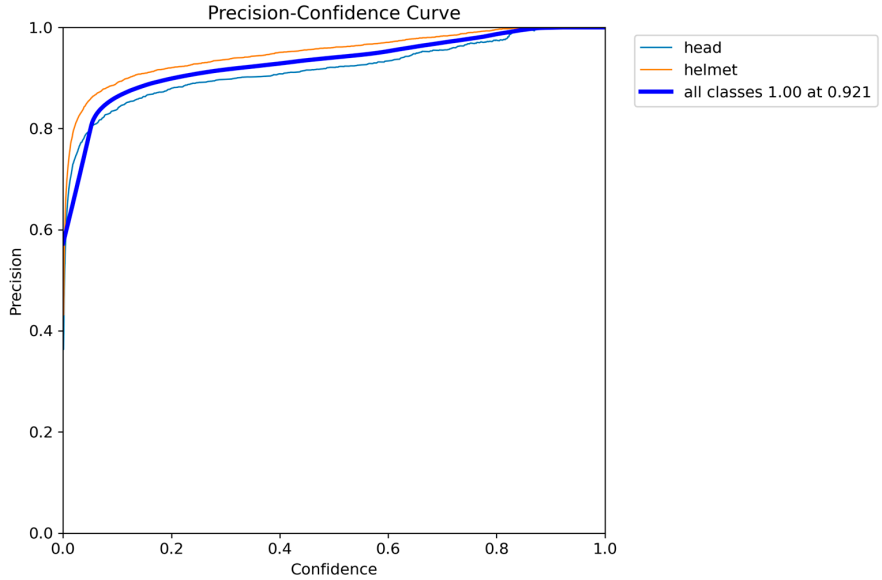


Figure 9. Precision–Confidence curve (YOLOv12m).

The Recall–Confidence plot helps in choosing the appropriate threshold for confidence level. The recall is very high (near 1.0) until reaching a confidence level of 0.8. Between confidence levels of 0.8 and 0.9 the recall drops significantly. Based on the curve, it is apparent that setting a confidence threshold above 0.8 during inference would significantly reduce the number of detections.

When the Precision–Confidence curve is examined, it is observed that the precision increases with the increase in confidence threshold for both the head and helmet classes. The graph indicates that perfect precision can be achieved when the confidence level is ~0.85–0.90.

When Figure 10 is examined, it is observed that the best F1 score (i.e., 0.94) can be obtained when the confidence threshold is selected as 0.457. This confidence threshold serves for balancing the correct detections (recall) with accuracy in detections (precision).

The average precision value is indicated in the legend of Figure 11. The average precision value of 0.958 shows the very strong performance of the model for detecting the "head" class. The detection performance of the model for the "helmet" class is even better

with an average precision value of 0.979. The mAP50 (or mAP@0.5) of all classes is 0.969, indicating a very good object detection performance of the model.

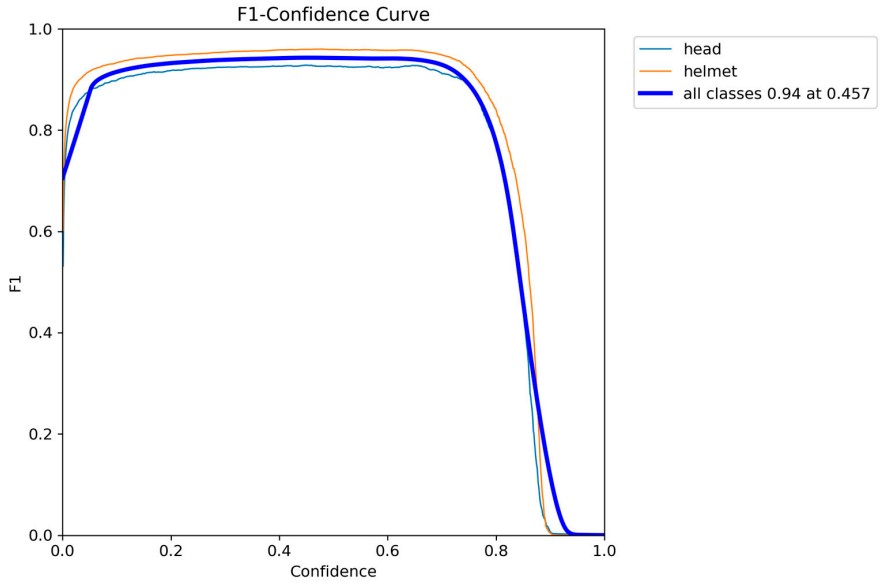


Figure 10. F1–Confidence curve (YOLOv12m).

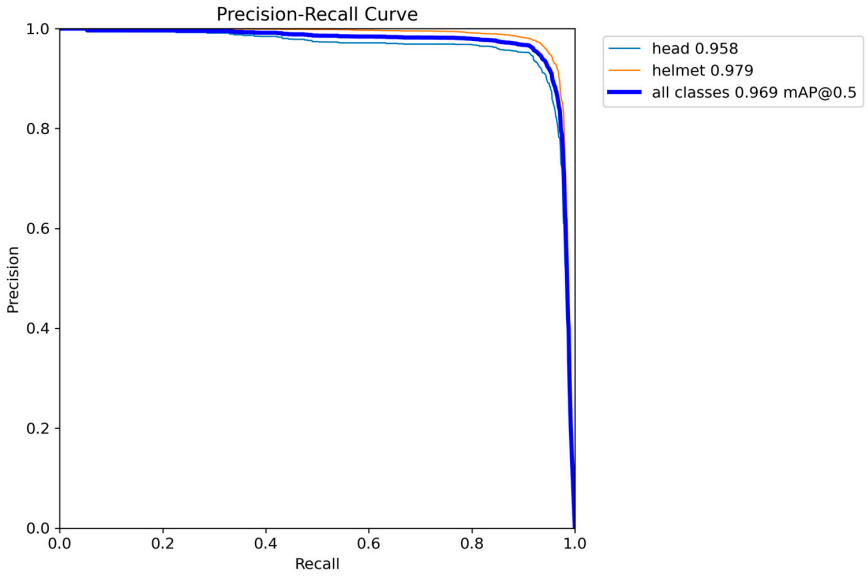


Figure 11. Precision–Recall curve (YOLOv12m).

3.2. The Real Time Advisory Tool

After the training phase, the Real Time Advisory Tool is developed as a proof-of-concept test environment to investigate the integration of the trained YOLO models together with a Large Language Model (i). for the detection of persons with and without a helmet at a construction site, and based on this detection, (ii). for providing a real-time safety assessment and advice to the users. The tool is a Streamlit application working on the server side. It acquires a live view from a web or IP camera, and can analyze a recorded video feed, image directory or single images uploaded by the user. If the data source is a moving image (either a live or recorded video) feed or a set of images, an image from the feed or set is captured every 5 s sequentially, the inference of helmet and head class counts is carried out using the fine-tuned YOLO models and then these counts are sent to the Large Language Model (LLM) through the API of the LLM (OpenAI gpt3.5-turbo

model). Figure 12 provides the system and user prompts sent to the LLM to acquire the safety assessment and advice.

```
system_prompt = let the total number of people with helmets = h and without helmets = n, never mention these variable names h, n in your response, never mention about the conditions explained. condition1: h > n, condition2: h < n, condition3: h = n, respond first by providing the number of people with helmet and number of people without helmets. if condition1 or condition3 is true change your tone to relaxed explain why h people are safer, also explain why n people need to wear helmets in a construction site with maximum of 250 words then stop your response, if condition2 is true change your tone to serious and explain the consequences with maximum of 250 words then stop your response.

user_promt = f"people with helmets {helmet_count} people without helmets {head_count} \n"
```

Figure 12. The system and user prompts sent to the LLM.

In the tool (Figure 13), after a photo of the construction site is uploaded or acquired automatically from each 5 s from the video feed or image directory, the application automatically detects the number of people wearing and not wearing helmets in the relevant image, and then provides the detection results and safety assessment and advice to the users in real time with the text generated by the LLM (Figure 14). The real-time image acquisition (from the Live Camera Stream) and object detection functionality of the system was tested using a 4 K webcam connected to a PC.

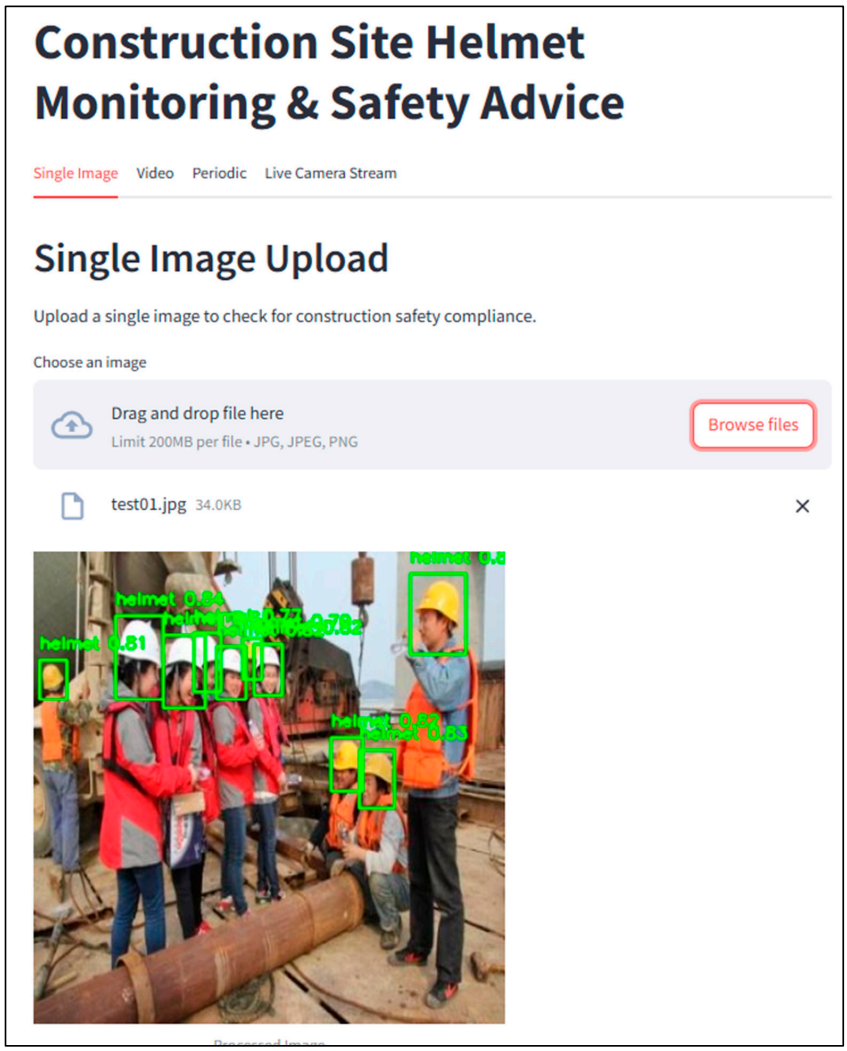


Figure 13. Occupational safety analysis and recommendation tool user interface.

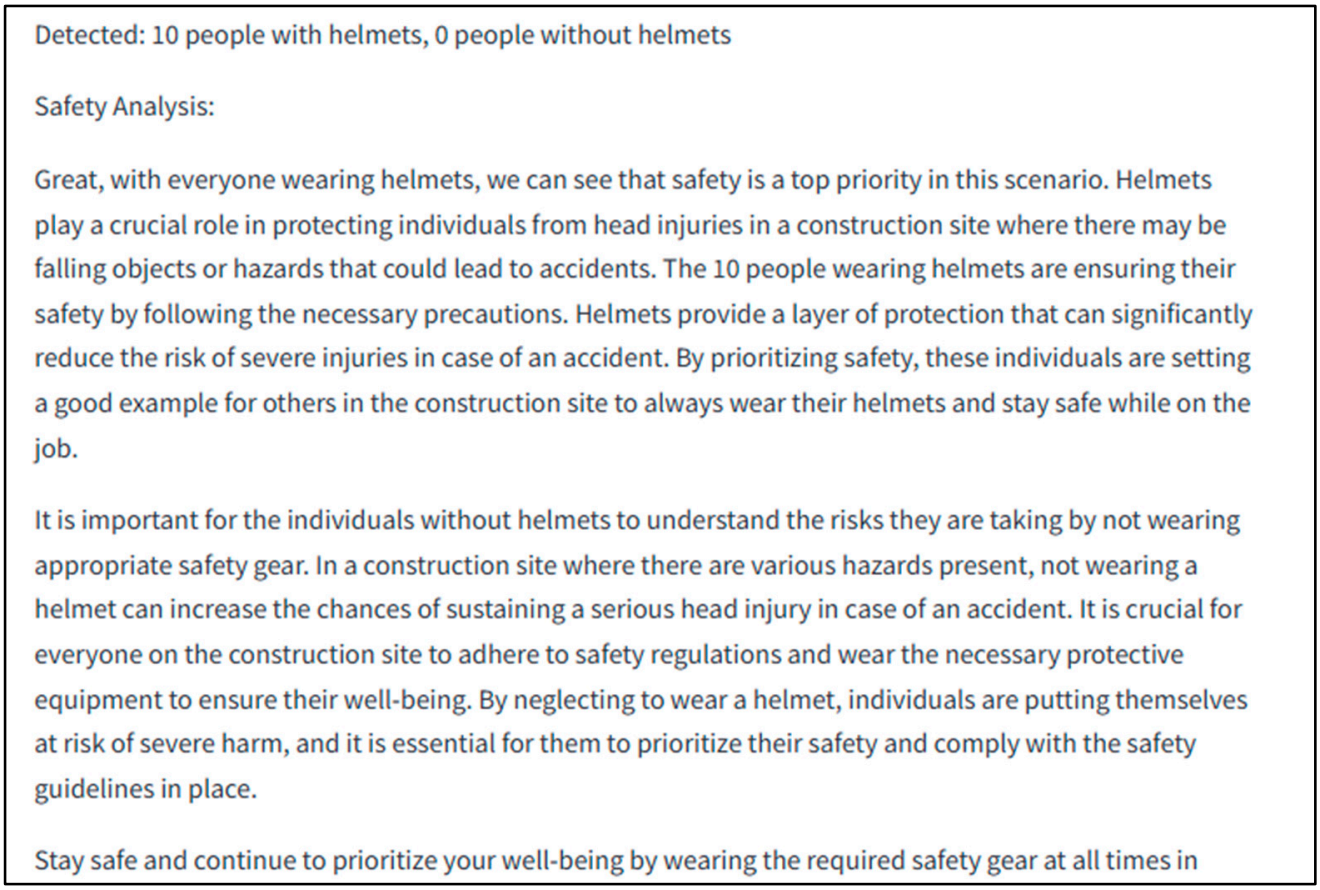


Figure 14. A sample LLM output presented to the user through the user interface.

4. Discussion

The use of artificial intelligence in the construction industry offers solutions that increase efficiency and reduce error rates, enabling faster, cost-effective and reliable project management. The use of computer vision models offers critical advantages such as increasing occupational safety on construction sites, providing efficient resource management and maintaining the health of the structure in the long term. In this context, the YOLO algorithms, which are at the forefront of real-time object detection, were used for training and testing for object detection in occupational health and safety and construction site management. A summary of the recent studies from the literature review on the use of deep learning algorithms for object detection in the construction industry is presented in Table 5.

Comparing the evaluation metrics provided in the previous research and in the current study, the current study provided a mAP50 value of 0.9679, with 87.85% classification accuracy for detected objects, as result of training 100 epochs for our dataset with two classes. The result indicates that the success rates in object detection and classification are similar or even better when compared with previous studies. This indicates that the object detection model is working efficiently in the developed tool. The Real Time Advisory Tool is the unique contribution of this study, as it allows the detection of objects from a video feed in real time, and provides feedback by utilizing a LLM, based on information fed by the object detection model.

Table 5. Comparison of the classification performance of the models in this study with different studies in the literature.

Study	Model	Dataset	Class Number	Metrics
Kang et al. [6]	YOLO-v3, Inception-SSD, R-FCN-ResNet101 and Faster-RCNN-ResNet101	Alberta Construction Image Dataset (ACID) (10,000 images)	10	mAP = 89.2%
Araya-Aliaga et al. [7]	trained RetinaNet, Faster R-CNN and YOLOv5	High-quality synthetic images	4	F1 score = 63.7%, precision = 66.8%
Kazaz et al. [8]	VGG-16	Dataset consisting of 800 aerial images collected using unmanned aerial vehicles (UAVs)	-	accuracy = 100%
Seth and Sivagami [9]	YOLOv8	Open-source dataset contains 5000 images	3	precision = 1
Barlybayev et al. [10]	YOLOv8m	Color Helmet and Vest (CHV) and Safety HELmet dataset (SHEL5K) with 5K	4	evaluation score = 0.929
Bai et al. [11]	YOLOv8-seg	-	8	mAP = 0.866
Jiao et al. [12]	YOLOv8	1584 images	2	mAP = 0.975
Biswas and Hoque [13]	YOLOv8	1026 images	7	mAP = 95.4
El-Kafrawy and Seddik [14]	YOLOv8, YOLOv9, YOLOv11 and YOLOv12	2047 images	7	box precision = 0.798, mAP50 = 0.553
This study	YOLOv8, YOLOv11, YOLOv12 (OpenAI GPT-3.5-turbo Large Language Model and with a Streamlit-based GUI)	Open-source dataset contains 7035 images [16]	2	mAP50 = 0.97766

5. Conclusions

In parallel with the advancement of technology, it is predicted that deep learning models will be able to perform image analysis and inference from images autonomously with high accuracy and speed. The aim of this study was to fine-tune object detection models and integrate them with Large Language Models for (i). accurate detection of personal protective equipment (PPE) by specifically focusing on helmets and (ii). providing real-time recommendations based on the detections for supporting the use of helmets in construction sites. In this context, following the fine tuning of well-known object detection models (YOLOv8/v11/v12), a tool with the aim of providing advice to workers in real time was developed. The developed tool provides the counts of the people based on video feeds or analyzing a series of images and provides recommendations on occupational safety (based on the detections from the video feed and images) through an OpenAI GPT-3.5-turbo Large Language Model with a Streamlit-based GUI. The two main contributions of the study are as follows: The fine tuning of the YOLO models proved that well-known object detection models can be used to detect PPE in construction sites with high accuracy. The development of the real-time advisory tool has demonstrated that a pipeline for detecting PPE from images and videos in real time and transferring information on detections into LLMs to reason and provide real-time advice about site safety can be successfully built and implemented using a web-based user interface.

The primary purpose of the developed LLM-based advisory tool was providing real-time safety advice based on knowledge about the current situation of safety risks of workers based on their current PPE usage states, which can be detected by tools such as the one explained in this paper. This approach was developed to demonstrate the applicability

of LLMs with image understanding and a real-time advice-providing capacity as a tool to aid the daily safety procedures in construction sites, particularly to support the correct use of personal protective equipment (PPE). The novelty of the approach is the demonstration of a tool which can use LLMs with image understanding to provide real-time advice in the construction sites.

In this study, the system demonstrated its feasibility as a proof of concept based solely on helmet detection. In the future we aim to implement and test the system for real-world applications. In this regard, the following extensions are planned: (i) detection of different personal protective equipment types such as reflective vests and gloves, (ii) integration of the tool with access control systems, (iii) extraction of individual compliance profiles based on employees' PPE usage history and (iv) analysis of risk areas based on the frequency of rule violations. The future research will focus on the fine-tuning of LLMs to enhance the accuracy and reliability of the advice provided by the models. Also, in future studies, we plan to investigate the performance of the network under different lighting conditions for outdoor applications and to evaluate the effects of different environmental factors such as rain, snow or fog on image quality.

Author Contributions: Conceptualization and methodology were carried out by Ü.I., H.A.Ç. and S.S. Software development was conducted by Ü.I. Validation, formal analysis, and investigation were performed by H.A.Ç., S.S. and Y.A. Data curation was handled by Ü.I. The original draft of the manuscript was prepared by H.A.Ç., S.S. and Y.A. Review and editing were performed by G.B. and Z.W.G. Visualization materials were created by H.A.Ç. and S.S. Supervision was provided by Ü.I. and G.B., while project administration was managed by G.B. and Z.W.G. All authors have read and approved the final version of the manuscript.

Funding: This research was partially funded by the BAP Program of Mimar Sinan Fine Arts University (Project No: 2023-33) and by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry & Energy, Republic of Korea (RS-2024-00441420).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding authors.

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

References

1. Xu, S.; Wang, J.; Shou, W.; Ngo, T.; Sadick, A.-M.; Wang, X. Computer Vision Techniques in Construction: A Critical Review. *Arch. Comput. Methods Eng.* **2021**, *28*, 3383–3397. [CrossRef]
2. Tsirtsakis, P.; Zacharis, G.; Maraslidis, G.S.; Fragulis, G.F. Deep learning for object recognition: A comprehensive review of models and algorithms. *Int. J. Cogn. Comput. Eng.* **2025**, *6*, 298–312. [CrossRef]
3. Islam, S.U.; Zaib, S.; Ferraioli, G.; Pascazio, V.; Schirinzi, G.; Husnain, G. Enhanced Deep Learning Architecture for Rapid and Accurate Tomato Plant Disease Diagnosis. *Agriengineering* **2024**, *6*, 375–395. [CrossRef]
4. Sun, Y.; Sun, Z.; Chen, W. The evolution of object detection methods. *Eng. Appl. Artif. Intell.* **2024**, *133*, 108458. [CrossRef]
5. Kineber, A.F.; Antwi-Afari, M.F.; Elghaish, F.; Zamil, A.M.A.; Alhusban, M.; Qaralleh, T.J.O. Benefits of Implementing Occupational Health and Safety Management Systems for the Sustainable Construction Industry: A Systematic Literature Review. *Sustainability* **2023**, *15*, 12697. [CrossRef]
6. 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]
7. Araya-Aliaga, E.; Atencio, E.; Lozano, F.; Lozano-Galant, J. Automating Dataset Generation for Object Detection in the Construction Industry with AI and Robotic Process Automation (RPA). *Buildings* **2025**, *15*, 410. [CrossRef]
8. Kazaz, B.; Poddar, S.; Arabi, S.; Perez, M.A.; Sharma, A.; Whitman, J.B. Deep Learning-Based Object Detection for Unmanned Aerial Systems (UASs)-Based Inspections of Construction Stormwater Practices. *Sensors* **2021**, *21*, 2834. [CrossRef]
9. Seth, Y.; Sivagami, M. Enhanced YOLOv8 Object Detection Model for Construction Worker Safety Using Image Transformations. *IEEE Access* **2025**, *13*, 10582–10594. [CrossRef]

10. Barlybayev, A.; Amangeldy, N.; Kurmetbek, B.; Krak, I.; Razakhova, B.; Tursynova, N.; Turebayeva, R. Personal protective equipment detection using YOLOv8 architecture on object detection benchmark datasets: A comparative study. *Cogent Eng.* **2024**, *11*, 2333209. [CrossRef]
11. Bai, R.; Wang, M.; Zhang, Z.; Lu, J.; Shen, F. Automated Construction Site Monitoring Based on Improved YOLOv8-seg Instance Segmentation Algorithm. *IEEE Access* **2023**, *11*, 139082–139096. [CrossRef]
12. Jiao, X.; Li, C.; Zhang, X.; Fan, J.; Cai, Z.; Zhou, Z.; Wang, Y. Detection Method for Safety Helmet Wearing on Construction Sites Based on UAV Images and YOLOv8. *Buildings* **2025**, *15*, 354. [CrossRef]
13. Biswas, M.; Hoque, R. Construction Site Risk Reduction via YOLOv8: Detection of PPE, Masks, and Heavy Vehicles. In Proceedings of the 2024 IEEE International Conference on Computing, Applications and Systems (COMPAS), Chattogram, Bangladesh, 25–26 September 2024; pp. 1–6. [CrossRef]
14. El-Kafrawy, A.M.; Seddik, E.H. Personal Protective Equipment (PPE) Monitoring for Construction Site Safety using YOLOv12. In Proceedings of the 2025 International Conference on Machine Intelligence and Smart Innovation (ICMISI), Alexandria, Egypt, 10–12 May 2025; pp. 456–459. [CrossRef]
15. Zhong, J.; Qian, H.; Wang, H.; Wang, W.; Zhou, Y. Improved real-time object detection method based on YOLOv8: A refined approach. *J. Real-Time Image Process.* **2025**, *22*, 4. [CrossRef]
16. Nelson, J. Hard Hat Workers Computer Vision Project. Available online: https://universe.roboflow.com/joseph-nelson/hard-hat-workers (accessed on 28 February 2025).
17. Ray, S.; Haque, M.; Rahman, M.; Sakib, N.; Al Rakib, K. Experimental investigation and SVM-based prediction of compressive and splitting tensile strength of ceramic waste aggregate concrete. *J. King Saud Univ.-Eng. Sci.* **2024**, *36*, 112–121. [CrossRef]
18. Omer, B.; Jaf, D.K.I.; Abdalla, A.; Mohammed, A.S.; Abdulrahman, P.I.; Kurda, R. Advanced modeling for predicting compressive strength in fly ash-modified recycled aggregate concrete: XGboost, MEP, MARS, and ANN approaches. *Innov. Infrastruct. Solut.* **2024**, *9*, 61. [CrossRef]
19. Bekdaş, G.; Aydın, Y.; Isıkdağ, Ü.; Sadeghifam, A.N.; Kim, S.; Geem, Z.W. Prediction of Cooling Load of Tropical Buildings with Machine Learning. *Sustainability* **2023**, *15*, 9061. [CrossRef]
20. Aydın, Y.; Cakiroglu, C.; Bekdaş, G.; Geem, Z.W. Explainable Ensemble Learning and Multilayer Perceptron Modeling for Compressive Strength Prediction of Ultra-High-Performance Concrete. *Biomimetics* **2024**, *9*, 544. [CrossRef]
21. Kumar, P.; Pratap, B. Feature engineering for predicting compressive strength of high-strength concrete with machine learning models. *Asian J. Civ. Eng.* **2024**, *25*, 723–736. [CrossRef]
22. Cakiroglu, C.; Aydın, Y.; Bekdaş, G.; Geem, Z.W. Interpretable Predictive Modelling of Basalt Fiber Reinforced Concrete Splitting Tensile Strength Using Ensemble Machine Learning Methods and SHAP Approach. *Materials* **2023**, *16*, 4578. [CrossRef]
23. Bekdaş, G.; Aydın, Y.; Nigdeli, S.M.; Ünver, I.S.; Kim, W.-W.; Geem, Z.W. Modeling Soil Behavior with Machine Learning: Static and Cyclic Properties of High Plasticity Clays Treated with Lime and Fly Ash. *Buildings* **2025**, *15*, 288. [CrossRef]
24. Aydın, Y.; Işıkdağ, Ü.; Bekdaş, G.; Nigdeli, S.M.; Geem, Z.W. Use of Machine Learning Techniques in Soil Classification. *Sustainability* **2023**, *15*, 2374. [CrossRef]
25. Aydin, Y.; Bekdaş, G.; Işikdağ, Ü.; Nigdeli, S.M.; Geem, Z.W. Optimizing artificial neural network architectures for enhanced soil type classification. *Geomech. Eng.* **2024**, *37*, 263–277.
26. Ji, Y.; Zhang, H.; Zhang, Z.; Liu, M. CNN-based encoder-decoder networks for salient object detection: A comprehensive review and recent advances. *Inf. Sci.* **2021**, *546*, 835–857. [CrossRef]
27. Mirzaei, B.; Nezamabadi-Pour, H.; Raoof, A.; Derakhshani, R. Small Object Detection and Tracking: A Comprehensive Review. *Sensors* **2023**, *23*, 6887. [CrossRef]
28. 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), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788. [CrossRef]
29. Karim, J.; Nahiduzzaman; Ahsan, M.; Haider, J. Development of an early detection and automatic targeting system for cotton weeds using an improved lightweight YOLOv8 architecture on an edge device. *Knowl.-Based Syst.* **2024**, *300*, 112204. [CrossRef]
30. Fakhrurroja, H.; Fashihullisan, A.A.; Bangkit, H.; Pramesti, D.; Ismail, N.; Mahardiono, N.A. A Vision-Based System: Detecting Traffic Law Violation Case Study of Red-Light Running Using Pre-Trained YOLOv8 Model and OpenCV. In Proceedings of the 2024 IEEE International Conference on Smart Mechatronics (ICSMech), Yogyakarta, Indonesia, 19–21 November 2024; pp. 1–6. [CrossRef]
31. Rasheed, A.F.; Zarkoosh, M. YOLOv11 optimization for efficient resource utilization. *J. Supercomput.* **2025**, *81*, 1085. [CrossRef]
32. Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft Coco: Common Objects in Context. In Proceedings of the Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, 6–12 September 2014; pp. 740–755. [CrossRef]
33. Jocher, G.; Qiu, J. Ultralytics YOLO Docs. Available online: https://docs.ultralytics.com/ (accessed on 9 September 2025).

34. Ayachi, R.; Said, Y.; Afif, M.; Alshammari, A.; Hleili, M.; Ben Abdelali, A. Assessing YOLO models for real-time object detection in urban environments for advanced driver-assistance systems (ADAS). *Alex. Eng. J.* **2025**, *123*, 530–549. [CrossRef]
35. Chicco, D.; Jurman, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genom.* **2020**, *21*, 6. [CrossRef]
36. He, L.-H.; Zhou, Y.-Z.; Liu, L.; Cao, W.; Ma, J.-H. Research on object detection and recognition in remote sensing images based on YOLOv11. *Sci. Rep.* **2025**, *15*, 14032. [CrossRef]
37. Tripathi, A.; Gohokar, V.; Kute, R. Comparative Analysis of YOLOv8 and YOLOv9 Models for Real-Time Plant Disease Detection in Hydroponics. *Eng. Technol. Appl. Sci. Res.* **2024**, *14*, 17269–17275. [CrossRef]
38. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
39. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
40. Perri, D.; Simonetti, M.; Gervasi, O. Synthetic Data Generation to Speed-Up the Object Recognition Pipeline. *Electronics* **2021**, *11*, 2. [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.