Deep Learning-Based Automatic Safety Helmet Detection System for Construction Safety

Ahatsham Hayat 1,2,* and Fernando Morgado-Dias 1,2,*

1 Madeira Interactive Technologies Institute, University of Madeira, 9000-082 Funchal, Portugal
2 Interactive Technologies Institute (ITI/LARSyS and ARDITI), 9020-105 Funchal, Portugal
* Correspondence: ahatsham@iti.larsys.pt (A.H.); morgado@staff.uma.pt (F.M.-D.)

Abstract: Worker safety at construction sites is a growing concern for many construction industries. Wearing safety helmets can reduce injuries to workers at construction sites, but due to various reasons, safety helmets are not always worn properly. Hence, a computer vision-based automatic safety helmet detection system is extremely important. Many researchers have developed machine and deep learning-based helmet detection systems, but few have focused on helmet detection at construction sites. This paper presents a You Only Look Once (YOLO)-based real-time computer vision-based automatic safety helmet detection system at a construction site. YOLO architecture is high-speed and can process 45 frames per second, making YOLO-based architectures feasible to use in real-time safety helmet detection. A benchmark dataset containing 5000 images of hard hats was used in this study, which was further divided in a ratio of 60:20:20 (%) for training, testing, and validation, respectively. The experimental results showed that the YOLOv5x architecture achieved the best mean average precision (mAP) of 92.44%, thereby showing excellent results in detecting safety helmets even in low-light conditions.

Keywords: computer vision; safety helmet detection; You Only Look Once (YOLO); deep learning

1. Introduction

Workplace safety has become a significant concern for many industries due to the effect of unsafe environments on productivity, and the resulting loss of workers. In the United States, most workers labor in dangerous conditions, and many die each year. In 2012, there were 4383 fatal occupational injuries in the United States, averaging 89 deaths per week and nearly 12 per day [1]. Construction is a high-risk sector, with construction workers frequently injuring themselves while on the job. In fact, the construction business in the United States has the largest number of fatalities of any industry, accounting for one out of every five worker deaths in the private sector in 2014 [2]. According to accident reports issued by the state administration of work safety from 2015 to 2018, 53 construction accidents occurred due to the improper wearing of helmets, accounting for 67.95% of the total accidents [3]. Some developing nations have a substantially higher mortality rate than developed nations. For example, the mortality rate in the construction business in the Republic of Korea is more than double that in the United States [4]. Construction managers are concerned about the greater rate of construction deaths in emerging nations.

According to an International Labor Organization (ILO) report, the construction industry has a higher accident rate than any other industry [5]. Construction often entails high-risk operations that require workers to work in dangerous surroundings and risk their lives. According to the U.S. Bureau of Labor Statistics, the number of fatalities has steadily climbed from 985 in 2015 to 1034 in 2020, with an annual increase of 2% [2]. In China, 840 workers died while working on construction projects in 2018, with 52.2% dying after falling from a high vantage point [6]. Similarly, the United Kingdom (U.K.) Health and Safety Executive (HSE) revealed that 142 employees died in fatal accidents in 2020/2021 [7].
Figure 1 shows the data for the leading types of fatal accidents in the U.K. from 2016 to 2021. Falling, slips, being struck by equipment, electrocution, and getting entangled between equipment were the significant causes of construction site fatalities [8]. Fall-related deaths in construction jobs accounted for 34.6% of overall construction deaths [8], up from 49.9% in the 1980s and the first half of the 1990s [9].

It is critical to monitor construction workers’ safety. Protection equipment use monitoring is part of construction site safety management. In most falling accidents, workers fall from heights and smash their heads on hard floors. Safety helmets can absorb and diffuse the impact of falling, reducing the risk of injury to workers who fall from heights. Hard helmets are made to withstand shock, object penetration, and contact with electrical hazards. Half of all fatalities from accidental falls and a considerable number of fatalities from slips, trips, and being struck by falling items may be reduced if employees wore hard helmets properly [10]. Previous studies have shown that wearing a safety helmet can reduce severe brain damage by up to 95% [11].

In order to ensure worker safety, various countries have imposed industrial safety regulations. The U.S. government has created an organization called the Occupational Safety and Health Administration (OSHA), which aims to develop and impose rules and regulations (like wearing a safety helmet, glasses, etc.) at construction sites to reduce injuries. However, for reasons such as simple negligence or misinformation about safety helmets, workers at construction sites do not follow the OSHA guidelines. Manually monitoring violations of safety helmet regulations is not a feasible solution, especially at large construction sites. Therefore, automatic detection of safety helmet use is extremely important.

Automatic helmet detection is basically an object detection problem and can be solved using deep learning and computer vision-based approaches. Due to its computational method and precision in the field of object detection, deep learning and its applications in computer vision have achieved a breakthrough [12]. The object identification method has been a research hotspot in the field of computer vision in recent years [13]. Two types of state-of-the-art deep learning methods for object detection are currently available: the R-CNN (Convolutional Neural Network)-based target detection algorithm, which generates candidate regions first and then performs classification or regression [14], and the You Only Look Once [15,16] (YOLO) and Single Shot MultiBox Detector (SSD) algorithms [17], which perform classification or regression using only one CNN. R-CNN-based approaches achieved relatively higher accuracy with the demerit of longer execution time, making them unsuitable for real-time scenarios. The SSD algorithm runs faster but faces problems
in detecting small objects, which could be problematic in automatic helmet detection \[18\]. Therefore, YOLO, with its different architecture, was used in this study to detect safety helmets on construction sites automatically.

In this paper, YOLOv5 architectures automatically detect safety helmets at construction sites. YOLOv5 is the latest YOLO architecture, and has different models based on size. For this study, YOLOv5x is used, which is the largest model of YOLOv5 architecture. The performance of the proposed YOLOv5x based model was also compared with other YOLO versions, i.e., YOLOv3 and YOLOv4.

This paper is organized as follows. Section 2 focuses on recent studies on safety helmet detection using deep learning approaches. Section 3 describes the dataset, the methodology used for safety helmet detection, and the different performance evaluation criteria for the methods. Section 4 shows the experimental results of this study. Section 5 concludes the paper and discusses the study’s future scope and limitations.

2. Materials and Methods

Several studies have been conducted in the past to detect safety helmets for the workers at construction sites. This literature review focuses on three major techniques used for safety helmet detection: sensor-based, machine learning-based, and deep learning-based detection. A total of 84 papers were found to be relevant for this study, of which 64 were excluded after abstract screening, resulting in the inclusion of 20 papers in this study. These papers present models for detecting safety helmets using the above-mentioned three techniques.

Sensor-based techniques usually try to track the safety helmet and worker. Barro-Torres et al. \[19\] combined the Zigbee and Radio Frequency Identification (RFID) technologies to detect personal protective equipment (PPE). The workers wear a microcontroller-based device to detect the PPE and send information to a central unit which will generate an alert if the worker is not wearing PPE properly. Similarly, Kelm et al. \[1\] and Zhang et al. \[11\] used RFID technology to detect PPE at construction sites. Kelm et al. \[1\] designed a mobile-based RFID model that detects workers wearing PPE while passing through the checking gate. Zhang et al. \[11\] created a real time IoT-based smart hard-hat system, which combines RFID technology and Internet of Things (IoT) to detect PPE. However, RFID-based technology cannot confirm that a worker is wearing PPE or safety helmet properly. Furthermore, while sensor-based technology can be a good reliable solution, these methods always rely on external equipment, which makes this technology very difficult to implement, especially on very large construction sites. Moreover, sensor-based methods are expensive \[20\].

To overcome the problem of device dependence, machine learning-based object detection methods have been widely used. Moreover, these methods offer increased feasibility and higher detection accuracy. Waranusast et al. \[21\] used vertical and horizontal projection methods for head segmentation and combines them with the K-Nearest Neighbor (KNN) method for helmet detection. Doungmala et al. \[22\] combine two techniques for helmet detection. First, Haar-like features were used to detect the helmet region \[23\], and then Circle Hough Transform (CHT) was applied to detect half and full helmets. Similarly, Rubaiyat et al. \[24\] proposed an automatic helmet detection method using Histograms of Oriented Gradient (HOG) features to detect workers, and then applying the CHT technique for safety helmet detection. Park et al. \[25\] and Zhu et al. \[26\] also used HOG features for segmentation followed by utilizing conventional machine learning methods for detecting safety helmets. Du et al. \[27\] used Haar-like features for face detection; then, in order to reduce false positive cases, they detected the motion of the worker, and used color information to detect the helmet. Shrestha et al. \[28\] developed a safety framework for workers at construction sites using segmentation and edge detection for detecting safety helmets. Kang et al. \[29\] in their work first applied a visual background extractor (ViBe) algorithm for the segmentation of moving objects, followed by the C4 algorithm for pedestrian classification, and finally the color feature discrimination (CFD) algorithm for safety
helmet detection. Víctor et al. [12] used a Microsoft Kinect sensor to collect color and depth data, and then applied machine learning techniques to detect workers and their actions at construction sites.

Machine learning-based technologies use hand-crafted features for safety helmet detection that could lead to poor generalization in complex environments like bad weather or at big construction sites. With the recent developments in the area of deep learning-based object detection, many researchers have used deep learning strategies for safety helmet detection. Wang et al. [30] used different YOLO architectures to detect four different color safety helmets, persons, and vests. Among all the architectures, YOLOv5x gives the best precision and YOLOv5s has the fastest speed. Similarly, Geng et al. [31] and Nath et al. [32] also used YOLO-based architecture for safety helmet detection. Geng et al. [31] used YOLOv3 architecture to detect safety helmets for the unbalanced dataset. They have improved the accuracy of YOLOv3 by using the Gaussian blurring method to deal with the associated data imbalance problems. Wu et al. [20] proposed a single shot-based CNN model to automatically detect hard hats and identify the corresponding color. They used SSD [33] to detect safety helmets and achieved a mAP of 83.89%. Similarly, Li et al. [18] and Han et al. [34] used SSD for helmet detection. Li et al. [18] proposed a deep learning-based method for real-time detection of safety helmets. An SSD-MobileNet algorithm was used in this study and presented good precision and recall. However, the model was not performing well for smaller images and complex backgrounds, resulting in very low mAP. Shen et al. [35] used bounding-box regression and transfer learning to detect safety helmets. They have used DenseNet-based strategies to improve the efficiency of the model and achieved an excellent accuracy of 94.47%. One of the major limitations of their work is the use of a face detection-based approach, as the model fails when the worker is not facing the camera, which is very common on construction sites.

This literature review shows that many studies have been conducted on safety helmet detection on construction sites based on sensor, machine learning, and deep learning techniques. Most of them fail in detecting safety helmets in complex scenarios, such as sites with multiple workers. Furthermore, helmet detection in low-light conditions and with small object sizes needs significant improvements for real-time systems deployment. Therefore, this study addresses both scenarios and proposes a deep learning (YOLO algorithm)-based automated helmet detection system to ensure the safety of construction workers. Furthermore, the performance of the proposed method was also compared with the existing helmet detection methods using other YOLO variants (YOLOv3, YOLOv4).

3. Methodology

This work proposes a deep learning-based framework architecture to detect workers’ helmets at construction sites using a publicly available benchmark dataset. Power-law transformation was initially performed for image enhancement, followed by image rescaling. Finally, a computer vision system was developed using the YOLOv5 object detection algorithm to classify workers with or without a helmet. Figure 2 shows the general steps in safety helmet detection for the workers.

Figure 2. General Architecture for Worker Safety Helmet Detection using Deep Learning Framework.
3.1. Dataset Description

The Hard Hat worker image dataset published by MakeML was used in this paper for the detection of hard helmets used in the construction industry [36]. It contains 5000 images with bounding box annotation files. Initially, it has three classes: Helmet, Person, and Head, but this research focuses on only two classes, Helmet and Head, which makes this a binary classification problem. Figure 3 shows some examples of Helmet and Head images.

![Figure 3. Examples of Construction Site images of people (a) Wearing Helmets (b) Not Wearing Helmets.](image)

The data is further divided into three sets (training, testing, and validation) with the split of 60% for the training, 20% for the testing, and 20% for the validation. Therefore, the training set contains 3000 images, the testing set contains 1000 images, and the validation dataset contains 1000 images. The training dataset was used to train the deep learning model, while the validation dataset was used for early stopping criteria. Finally, the fine-tuned deep learning model was tested on an independent test dataset.

3.2. Data Pre-Processing

3.2.1. Image Enhancement

The dataset used for this study contains ample low-light images, which are challenging for object detection. To overcome this problem, power-law transformation or gamma correction-based image enhancement was applied to enhance the images [37]. Equation (1) shows how power-law transformation enhanced the visibility of images.

\[ s = c \times r^\gamma \]  

where “s” represents the output pixel value, “r” represents the input pixel value, and “c” and “\( \gamma \)” are the positive constants. A large \( \gamma \) value leads to darker output images; for example, \( \gamma > 1 \) leads to darker output values against lighter input values. In this scenario, the value of c is taken as 1, and to get clear images, the \( \gamma \) value is taken as 0.5. Figure 4 shows a comparison of some images before and after power-law transformation depicting its effectiveness in enhancing low-light images.

3.2.2. Image Rescaling

Due to different image resolutions in the Hard Hat image dataset, all images were rescaled to a resolution of 640 x 640, adapting to the pixel resolution of the YOLO framework. The framework uses an input size of 32 pixels for training, testing, and validation.
3.3. YOLOv5

In the past few years, the YOLO framework has become very popular for object detection because of its single-stage deep learning nature, which detects an object in a single run with relatively better accuracy and speed than other deep learning models such as R-CNN, Faster R-CNN, etc. [38] Among different YOLO versions, YOLOv5 was selected for this study for helmet detection due to its higher accuracy and faster execution as compared to other YOLO variants. Furthermore, YOLOv5x comprised the largest architecture of all versions of YOLOv5 architecture [39]. It uses CSPDarknet53 [40] as a backbone for feature extraction, which solves the problem of repetitive gradient information in the big backbone. Additionally, it integrates changes in the gradient with the feature map, increasing the accuracy and reducing the model size by decreasing the number of parameters.

The YOLO network consists of three networks: the backbone network, neck network, and detect or head network. Table 1 compares YOLOv3, YOLOv4, and YOLOv5 [39]. All the YOLO architectures use the same neural network type (Fully Connected) and head (YOLO layer), but the neck and backbone are different.

Table 1. Comparison between Different YOLO Architectures [39].

<table>
<thead>
<tr>
<th></th>
<th>YOLOv3</th>
<th>YOLOv4</th>
<th>YOLOv5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Type</td>
<td>Fully Connected</td>
<td>Fully Connected</td>
<td>Fully Connected</td>
</tr>
<tr>
<td>Backbone for Feature Extraction</td>
<td>Darknet-53</td>
<td>CSPDarknet53</td>
<td>CSPDarknet53</td>
</tr>
<tr>
<td>Neck</td>
<td>Feature Pyramid Network</td>
<td>Spatial pyramid pooling and Path Aggregation Network</td>
<td>Path Aggregation Network</td>
</tr>
<tr>
<td>Head</td>
<td>YOLO Layer</td>
<td>YOLO Layer</td>
<td>YOLO Layer</td>
</tr>
</tbody>
</table>

As discussed earlier, YOLOv5x was used because it uses CSPDarknet53 as the backbone for feature extraction, achieving better accuracy and reducing the model size. Further, to boost the information flow, it uses the Path Aggregation Network (PANet) [41] as the neck. For propagation of low-level features, PANet uses the Feature Pyramid Network (FPN) with an enhanced bottom-up path. PANet also enhances the use of efficient localization signals in lower layers, which can significantly improve object location accuracy. Finally, the head of YOLOv5x uses the YOLO layer, which generates three different sizes of feature maps to achieve multiple-scale prediction. Multi-scale detection enhances the accuracy of the model by predicting small to large objects efficiently. Figure 5 shows the general architecture of the YOLOv5 model; first, the images are fed to CSPDarknet53 to extract the essential features, then fed to PANet for feature fusion, and finally, the YOLO layer will generate the result (class, score, location, size) [40].
Figure 5. Architecture of YOLOv5 for Safety Helmet Detection.

YOLOv5x uses Generalized Intersection over Union (GIoU)-based [40] loss function for the bounding box regression represented in Equation (3). The normal IoU loss function (Equation (2)) is limited to the cases where the predicted bounding box and target bounding box overlap with each other, but it does not work in non-overlapping cases. When dealing with non-overlapping situations, GIoU loss assists by gradually raising the predicted box’s size until it overlaps with the ground truth.

\[
\text{IoU} = \frac{|B \cap B'|}{|B \cup B'|} \tag{2}
\]

\[
\text{GIoU} = \text{IoU} - \frac{|C / (B \cup B')|}{|C|} \tag{3}
\]

where B and B’ are the predicted and ground truth bounding boxes, and C is the smallest bounding box covering both the B and B’. The loss function used for YOLOv5x is 1-GIoU which holds properties such as symmetry, triangular inequality, and imperceptible identity [42].

3.4. Evaluation Criteria

3.4.1. Evaluation Metrics

To measure the effectiveness of the YOLOv5x algorithm, five major metrics are used: accuracy, precision, recall, F1 score, and mAP. All performance metrics can be calculated using the four parameters of the confusion matrix, i.e., True Positive (T.P.), False Positive (F.P.), True Negative (T.N.), and False Negative (F.N.). T.P. represents the correct prediction of a person wearing a safety helmet, F.P. represents the incorrect prediction of a person wearing a safety helmet, T.N. represents the accurate prediction of a person without a safety helmet, and F.N. represents the inaccurate prediction of a person without a safety helmet.
helmet. Accuracy, precision, recall, F1 score, and mAP are mathematically explained in Equations (4)–(8). F1 score is the harmonic mean of precision and recall.

\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}
\]

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{5}
\]

\[
\text{Recall} = \frac{TP}{TP + FN} \tag{6}
\]

\[
\text{F1 score} = 2 \cdot \frac{(\text{Precision} \cdot \text{Recall})}{(\text{Precision} + \text{Recall})} \tag{7}
\]

\[
\text{mAP} = \frac{\sum_{c=1}^{C} \text{Average Precision (c)}}{C} \tag{8}
\]

where \( C \) is the total number of output classes. In this study \( C = 2 \) (Helmet and Head).

3.4.2. Training the Algorithm

For training, the different neural network Yolo architectures are used, and all other implementations are done using PyTorch [43]. As mentioned before, model training was performed using 3000 randomly sampled images, while for the validation and testing of the model, 1000 images each were utilized with a batch size of 32. Furthermore, stochastic gradient descent (SGD) was utilized for training the network using a learning rate of 0.001 and momentum of 0.8. The GIoU-based loss function was minimized using an early stopping condition based on the validation dataset.

4. Result and Discussion

In this paper, for a comparative study of the YOLOv5x model, two other models, YOLOv3 and YOLOv4, were also implemented for safety helmet detection at the construction site. Figure 6 shows the confusion matrix, which proved the descent performance of the proposed computer vision model.

![Confusion Matrix for Safety Helmet Detection using YOLOv5x.](image)

Figure 6 shows the average precision (A.P.) of each class and the mAP for each model. YOLOv5x achieved the best mAP of 92.44%, followed by YOLOv4 (90.64%) and YOLOv3 (85.78), respectively. On the other hand, YOLOv3 showed a balanced performance in detecting both classes, “Helmet” and “Head”, while YOLOv4 and YOLOv5x were relatively poor in detecting the “Head” class, especially the latter.

Table 2 shows the detailed comparison between all three YOLO models trained on the Hard Hat detection dataset, depicting the mean and standard deviation (\( \mu \pm \sigma \)) of accuracy, precision, recall, and F-1 score, respectively.
The YOLOv5x model showed higher accuracy, precision, recall, and F1 score value of almost 92%, 92%, 89%, and 91%, respectively, compared to YOLOv3 and YOLOv4, meaning that YOLOv5x performed best in detecting safety helmets. Furthermore, YOLOv4 and YOLOv5x showed significantly higher accuracy than YOLOv3, potentially due to the backbone feature layer. YOLOv4 and YOLOv5x use CSPDarknet53, and YOLOv3 uses Darknet53, which struggles when objects are small. Furthermore, YOLOv4 and YOLOv5x use mosaic data augmentation strategy internally [44], which also helped in improving their performance. Additionally, Table 3 compares all the above models on different metrics such as model weight, number of parameters, etc. Yolov5x uses CSPDarknet53 for feature extraction, which only extracts essential features from the image, followed by spatial pyramid pooling to avoid repeatedly computing the convolutional features. That is why YOLOv5x is the largest network with 86.7 million (M) parameters, and still has the lowest model weight of all three models.

![Figure 7. Mean Average Precision for Each Model.](image)

**Table 2. Performance Assessment of YOLO models.**

<table>
<thead>
<tr>
<th>Measure</th>
<th>YOLOv3 (%)</th>
<th>YOLOv4 (%)</th>
<th>YOLOv5x (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>82.12 ± 0.12</td>
<td>87.77 ± 0.14</td>
<td>92.00 ± 0.21</td>
</tr>
<tr>
<td>Precision</td>
<td>85.78 ± 0.28</td>
<td>90.63 ± 0.10</td>
<td>92.44 ± 0.11</td>
</tr>
<tr>
<td>Recall</td>
<td>81.87 ± 0.17</td>
<td>86.78 ± 0.21</td>
<td>89.24 ± 0.28</td>
</tr>
<tr>
<td>F1 score</td>
<td>83.77 ± 0.22</td>
<td>88.66 ± 0.16</td>
<td>90.81 ± 0.19</td>
</tr>
</tbody>
</table>

![Figure 8 shows the loss vs. epochs curve on the training and validation dataset for YOLOv5x architecture.](image)

**Table 3. Architectural Comparison between Different YOLO Models.**

<table>
<thead>
<tr>
<th>Model</th>
<th>Model Weight (M.B.)</th>
<th>Parameters (M)</th>
<th>GFLOPs (B)</th>
<th>Latency (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv3</td>
<td>212</td>
<td>63</td>
<td>157.3</td>
<td>11.78</td>
</tr>
<tr>
<td>YOLOv4</td>
<td>189</td>
<td>78.5</td>
<td>187.6</td>
<td>14.3</td>
</tr>
<tr>
<td>YOLOv5x</td>
<td>167</td>
<td>86.7</td>
<td>205.7</td>
<td>16.5</td>
</tr>
</tbody>
</table>

Figure 8 shows the loss vs. epochs curve on the training and validation dataset for YOLOv5x architecture. For the training process, the model runs for 50 epoch cycles with the patience value of 10 (complete 10 continuous epoch cycles without change in the loss by 0.01) to avoid the problem of overfitting. Figure 8a shows the loss curve for bounding box detection, Figure 8b shows the loss curve for class (Helmet and Head) detection, and 8c shows the loss curve for object detection. The loss curve for the training set is shown in Figure 8, which declines significantly at first and progressively converges as the number of
epochs grows, which indicates that the model was evolving well in the training process. All the losses were almost less than 0.03, illustrating the decent performance of the YOLOv5x-based safety helmet detection model in low-light or when workers are at a considerably larger distance (resulting in small objects for detection).

Figure 8. Training and Validation curve of loss for the YOLOv5x model. (a) Loss curve for bounding box, (b) Loss curve for class, (c) Loss curve for object.

Figure 9 shows the prediction results with a confidence score for both classes. In Figure 9a,b, almost all the possible scenarios (single object, very small object, multiple objects, object from the back) have been accurately identified, thereby showing the ability to detect small objects with higher confidence scores. Furthermore, existing studies have faced the problem of detecting safety helmets in low-light conditions and against complex backgrounds. On the contrary, the proposed YOLOv5x-based model showed significant results even in low-light conditions. Figure 10 presents the prediction results of YOLOV5x in low-light conditions. Therefore, the proposed model could potentially be implemented as a real-time safety helmet detection system at construction sites to ensure worker safety in cases of inevitable accidents.

Figure 9. Prediction results of the YOLOv5x model. (a) Helmet class, (b) Head class.
Despite the significantly better performance, the proposed helmet detection model has certain limitations. For example, the proposed model faces problems in scenarios such as where the helmet is behind an obstacle(s) or with objects identical to helmets. Figure 11 shows potential detection errors; Figure 11a,c illustrate the poor performance of the proposed model when the helmet is behind bars due to the inability of the model to detect the ground truth. Nevertheless, instead of missing all the objects, the model is able to partially see objects obfuscated by the bars. Similarly, Figure 11b represents a scenario where an object looks like a helmet, and due to low light and obstacle structure the model could not detect it as another object. These problems could potentially be addressed in future studies by proposing novel deep learning frameworks for the discussed scenarios.

The proposed model is also compared with some state-of-the-art models. Table 4 shows the detailed comparison. Wu et al. [20] and Han et al. [34] used SSD-based learning for safety helmet detection. Both studies have shown stable results, but they struggle to detect small-scale safety helmets. Wang et al. [30] and Nath et al. [32] used YOLO-based learning for personal protective equipment (safety helmet, vest, etc.) detection at
construction sites. They have used relatively small datasets, which could lead to the poor generalization of the models. One of the most significant limitations of their work was that they struggled to detect small helmets in low-light conditions. Shen et al. [35] used transfer learning based on DenseNet architecture and applied two strategies, feature extractor and fine-tuning, for safety helmet detection. Despite the excellent results shown in their study, it would not be feasible to use their method in real-time scenarios as it is based on face detection techniques. Therefore, their model cannot detect the workers with their back to the surveillance camera, and in real-time scenarios it is impossible to always capture the faces of all the workers. Our proposed study shows stable results in all the different scenarios. Figure 9 shows that the model can detect helmets even when workers have their back to the camera, and Figure 10 shows the model is very efficient in detecting smaller helmets even in low-light conditions.

Table 4. Comparison of Proposed Model with other state-of-the-art models Based on Deep Learning methods.

<table>
<thead>
<tr>
<th>Related Work</th>
<th>Precision (%)</th>
<th>Recall (%)</th>
<th>F1 Score (%)</th>
<th>mAP (%)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Li et al. [18]</td>
<td>95.00</td>
<td>77.00</td>
<td>85.05</td>
<td>36.82</td>
<td>X</td>
</tr>
<tr>
<td>Wu et al. [20]</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>83.89</td>
<td>X</td>
</tr>
<tr>
<td>Wang et al. [30]</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>86.55</td>
<td>X</td>
</tr>
<tr>
<td>Nath et al. [32]</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>72.30</td>
<td>X</td>
</tr>
<tr>
<td>Han et al. [34]</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>88.10</td>
<td>X</td>
</tr>
<tr>
<td>Shen et al. [35]</td>
<td>96.20</td>
<td>96.20</td>
<td>96.20</td>
<td>X</td>
<td>94.47</td>
</tr>
<tr>
<td>Proposed Study</td>
<td>92.44</td>
<td>89.24</td>
<td>90.81</td>
<td>92.44</td>
<td>92.00</td>
</tr>
</tbody>
</table>

5. Conclusions

As worker safety is a major concern on construction sites, this study considered helmet detection as a computer vision problem, and proposed a deep learning-based solution. Existing studies have struggled in detecting objects from low-light images and smaller objects (due to the larger distance between the camera and workers). Therefore, a YOLOv5x-based architecture for automatic detection of safety helmets on construction sites was proposed to ensure worker safety.

This study used different versions of YOLO architecture, YOLOv3, YOLOv4, and YOLOv5x, to detect safety helmets due to their proven accuracy in object detection tasks. Among them, YOLOv5x achieved the best mAP (92.44%) in detecting smaller objects and objects in low-light images, thereby showing its efficacy in safety helmet detection.

Despite the significant outcomes achieved by YOLOv5x-based architecture, it also possesses several limitations. The proposed deep learning model struggled to perform in some scenarios (e.g., with an obstacle in front of helmets, and objects identical to helmets). Training the model with more images, including the above-mentioned scenarios, could potentially increase the model’s efficacy. Moreover, in the future, more safety tools could be added for detection, such as vests, gloves, and glasses, to ensure greater safety for workers.

Author Contributions: Funding acquisition, F.M.-D.; Investigation, A.H.; Methodology, A.H.; Validation, A.H. and F.M.-D.; Writing—original draft, A.H.; Writing—review & editing, F.M.-D. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: The dataset used in this article is openly available in the MakeML at https://makeml.app/datasets/hard-hat-workers.
Acknowledgments: Acknowledgment to the Bolsa de Investigação (B.I.) within Project BASE: Banana Sensing (PRODERAM20-16.2.2-FEADER-1810). Acknowledgment to the LARSys (Projeto—UIDB/50009/2020) for funding this research.

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

References


