



MCX-YOLOv5: efficient helmet detection in complex power warehouse scenarios

Hongchao Xu¹ · Zhenyu Wu¹

Received: 13 August 2023 / Accepted: 21 December 2023 / Published online: 30 January 2024
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2024

Abstract

The accuracy of the original YOLOv5 algorithm in detecting whether power company employees are wearing helmets is low due to the complex monitoring scenarios in the power warehouse and the small size of the helmets. As a result, it cannot be applied to actual operations. To address this issue, we developed the MCX-YOLOv5 helmet detection algorithm. Our model utilizes the YOLOv5 architecture and integrates a Coordinate-Spatial Attention Module (CSAM) to effectively filter the spatiotemporal data of the feature inputs. Additionally, we implement a Multi-scale Asymmetric Convolutions (MAConv) downsampling module to improve the algorithm's sensitivity to feature scale variations. To address the challenge of task information cross-coupling in coupled heads, we propose a decoupled head that is less heavy than YOLOv6 as a substitute. Our enhanced model achieved a 2.7% rise in the mean Average Precision at 50 (mAP50) and a 4.9% improvement in mAP75 on our self-developed database through multiple experiments, with just a minimal increase in parameters. Our model has yielded significant performance improvements on the Kaggle open-source Hard Hat Workers Detection dataset (HHWD), the public Safety Helmet Wearing Dataset (SHWD), and the PASCAL Visual Object Classes (VOC) dataset. These results highlight the effectiveness of our proposed algorithm in achieving higher accuracy for safety helmet wear detection in storage scenarios.

Keywords YOLOv5 · Decoupled head · Helmet detection · Attention mechanism · Downsampling

1 Introduction

Ensuring that workers wear safety helmets while performing tasks at heights or in the presence of falling objects is of paramount importance. Safety helmets effectively reduce and disperse impact forces, thereby safeguarding the lives of workers in hazardous environments. Traditionally, supervisors relied on manual oversight to determine whether workers were wearing helmets, which proved to be a time-consuming and labor-intensive process. However, with the rapid advancements in computer vision technology, target detection has emerged as a crucial solution to address this issue. By applying target detection technology intelligently, the recognition of individuals wearing safety helmets can

significantly enhance safety and convenience within power companies.

Through the implementation of computer vision, efficient monitoring of helmet compliance among workers can be achieved, simultaneously reducing the associated labor costs related to safety helmet detection. By leveraging intelligent surveillance systems based on target detection technology, power companies can ensure more effective adherence to safety helmet usage, thereby mitigating the risk of potential accidents, enhancing overall work safety, and optimizing the allocation of human resources and costs for the company.

Traditional object detection methods primarily use sliding windows to build candidate boxes on images and extract features using techniques such as Scale-Invariant Feature transformation (SIFT) [1], Harr-like features (HLF) [2], and Histogram of Oriented Gradients (HOG) [3]. Then, template matching algorithms are used for target matching, or classification is performed using methods like Support Vector Machine (SVM) [4]. While these methods are effective in certain scenarios, they suffer from high complexity and poor robustness, making them unsuitable for current

✉ Zhenyu Wu
xhc192393xhc@gmail.com

Hongchao Xu
13957346098@163.com

¹ School of Mechanical Engineering, Zhejiang Sci-Tech University, Hangzhou 310018, China

object detection tasks. In recent years, deep learning-based methods have become the mainstream approach for helmet-wearing detection and have made significant progress. However, they still face challenges in maintaining robust detection results, particularly in complex environments with small targets and surveillance scenarios.

In this study, we adopt the YOLOv5-7.0 algorithm as the primary solution, tailored to address the aforementioned limitations. YOLOv5 has proven to be effective in achieving high detection accuracy and real-time processing capabilities. Building upon this foundation, we propose further enhancements to create an efficient helmet-wearing detection algorithm catered to the specific needs of power company personnel. The main improvements in this paper are as follows:

1. In this study, we report the introduction of a multi-scale fusion downsampling module known as Multi-scale Asymmetric Convolutions (MAConv), which draws inspiration from the theoretical framework proposed in FasterNet [5]. This module utilizes various downsampling techniques internally, enabling the acquisition of comprehensive feature information while simultaneously decreasing the model's parameters and computing complexity. The module additionally employs the technique of combining numerous modules and adjusting their weights adaptively in order to dynamically choose the most suitable way for aggregating features.
2. The Coordinate-Spatial Attention Module (CSAM) was utilized in our study to incorporate both coordinate attention and spatial attention. This module incorporates information from both coordinate and spatial positions to dynamically modify the weights of features. This adjustment process enhances the model's emphasis on and significance for the target regions. The implemented design facilitates enhanced identification of essential characteristics of the intended items, resulting in heightened precision in detection and aiding in the precise identification of safety helmets in power storage environments.
3. We adopted a more lightweight decoupled head structure called VXDetect, which features a more compact design and lower computational complexity compared to the decoupled head in YOLOv6 [6] while maintaining stable accuracy.
4. A dataset specifically collected for the purpose of detecting safety helmets among employees of power firms has been developed. This dataset has a total of 4000 photos that have been classified into four distinct categories.

The subsequent sections of this work are structured in the following manner: In the subsequent section, an examination of the most recent scholarly investigations

pertaining to the domain of safety helmet detection will be conducted. The third part will detail the network architecture of the basic YOLOv5 model and its performance advantages in single-stage detection models. Section 4 presents an elaborate exposition of the model structure and module particulars pertaining to MCX-YOLOv5. In the fifth section, we provide an overview of the training environment and give the findings of the experimental comparison. In conclusion, Sect. 6 provides a comprehensive summary of the entirety of the study.

2 Related work

With the continuous advancement of technology, researchers are gradually shifting their focus from traditional image processing methods to deep learning, aiming to address the multifaceted challenges in the field of safety helmet detection. Despite the notable achievements of conventional algorithms in past studies, their robustness in complex environments remains constrained. In this context, deep learning methods have emerged, bringing heightened accuracy and adaptability to the domain of safety helmet detection. Subsequently, we will delve into two key aspects: safety helmet detection based on traditional algorithms and safety helmet detection based on deep learning. Through detailed discussions of these methods, we aim to unveil why they have become the focal point of current research.

2.1 Safety helmet detection based on traditional algorithms

Initially, scholars employed conventional image-processing methodologies for the purpose of helmet detection. For example, Park et al. employed the HOG technique to detect individuals wearing safety helmets. They accomplished this by identifying the existence of safety helmets in the region of the head through the utilization of color histograms [7]. Rubaiyat et al. used color information and the Hough transform method to find safety helmets by combining frequency domain data from images with human detection algorithms [8]. The safety helmet detection method that Du et al. proposed integrates temporal imagery and machine learning techniques [9]. Despite the excellent outcomes attained in these investigations, they are nonetheless confronted with specific challenges and constraints. These investigations utilize conventional object detection techniques that mainly depend on human feature extraction, leading to very limited robustness of the derived features, especially in intricate settings.

2.2 Safety helmet detection based on deep learning

Deep learning-based object identification approaches can be classified into three main categories: two-stage object detection, one-stage object detection, and transformer-based object detection. Nevertheless, transformer-based object identification approaches are not well-suited for small datasets and edge deployment, primarily because of the inherent properties of self-attention. The extraordinary accuracy of two-stage object detection algorithms, such as Region-based Convolutional Neural Network (RCNN) [10], Fast RCNN [11], and Faster R-CNN [12], is widely recognized. However, these methods are characterized by slower detection speeds and higher computing complexity. In contrast, there exist one-stage object identification approaches that have been developed to achieve a trade-off between detection accuracy and computational efficiency. Notable examples include the You Only Look Once (YOLO) series [13–17], Single Shot Multi-box Detector (SSD) [18], and Center-Net [19]. Nevertheless, it is important to acknowledge that both one-stage and two-stage algorithms possess certain limits when it comes to effectively recognizing smaller safety helmet targets.

To obtain multi-scale global information, an author introduced self-attention methods into the Faster R-CNN framework in Reference [20]. The incorporation of this integration enables the model to effectively capture more intricate details by increasing its receptive field. Two-stage object detection algorithms frequently require a lot of memory resources, despite the excellent accuracy they achieve. The authors of Reference [21] recommended the incorporation of a coordinate attention module into the YOLOv5 design. The Res2NetBlock structure's residual block was used to replace the C3 residual block. This was done to improve the backward gradient flow and the model's ability to include fine-grained features. In a prior investigation, the researchers (Reference [22]) proposed an advanced algorithmic model known as YOLOv5+. The proposed approach integrates a specialized detection layer that is specifically designed to enhance the accuracy of identifying small objects, thereby improving the overall performance of object detection. The methodology given by Reference [23] presents an approach for object recognition that leverages the SSD as its foundation. The proposed methodology integrates cross-layer attention mechanisms to improve the effectiveness of feature extraction and feature pyramids. Additionally, it employs multi-scale perception modules to tackle the issue of low precision in detecting the presence of safety helmets. At present, the detection of safety helmets mostly depends on detection methods based on deep learning.

3 YOLOv5 network

The YOLOv5 model has been widely adopted in the field of object detection and offers four different versions with varying complexities and parameter sizes. Subsequently, researchers have proposed the YOLOv6, YOLOv7, and YOLOv8 models, each with its own improvements and advancements over the previous versions. YOLOv6 introduces model pruning and other techniques to enhance accuracy, making it more suitable for practical model deployment. YOLOv7 focuses on further enhancing model accuracy by incorporating modules like ELAN, resulting in higher precision. Although YOLOv7-tiny has the smallest number of parameters and computational complexity, it is prone to overfitting. YOLOv8 is an integrated algorithm model specifically designed to facilitate practical deployment in real-world scenarios.

Considering the available options of YOLOv5s, YOLOv6s, YOLOv7-tiny, and YOLOv8s, this paper selects YOLOv5s as the foundational model for safety helmet detection based on several considerations, including Params and Floating Point Operations (FLOPs). And The larger the number of FLOPs, the more computing resources will be consumed and the slower the speed of recognition will be. Table 1 provides a comparison of these models in terms of their respective parameters and computational demands, affirming the suitability of YOLOv5s for our research on safety helmet detection. And YOLOv7-tiny is too easy to overfit due to its positive and negative sample allocation strategies.

YOLOv5s stands out by maintaining a high level of accuracy while having a more lightweight architecture compared to other models. This characteristic makes it well-suited for deployment in resource-constrained environments where computational resources are limited.

The YOLOv5s model is composed of four main components: input, backbone, neck, and prediction. In the input stage, various data augmentation techniques are employed to effectively increase the diversity of image samples, enhancing the model's ability to generalize to different scenarios. Moreover, an adaptive anchor box design is utilized to initialize multi-scale anchor boxes using clustering algorithms. This approach addresses the

Table 1 Parameter comparison

Method	Params (M)	FLOPs (B)
YOLOv5s	7.20	16.5
YOLOv6s	18.5	45.3
YOLOv7-tiny	6.2	13.7
YOLOv8s	11.2	28.6

issue of scale variations present in the detection targets within the dataset.

The overall model structure is illustrated in Fig. 1, where the C3 module represents a residual module that facilitates gradient flow across different layers, mitigating the risk of gradient vanishing and improving the model's training stability. The Neck component incorporates Spatial Pyramid Pooling Fusion (SPPF) pyramid pooling, which integrates feature scales to capture information at different levels of granularity. The output branch consists of three branches dedicated to detecting and recognizing large, medium, and small objects, respectively. With these innovative methods and components, YOLOv5 demonstrates state-of-the-art performance on some datasets, showcasing its effectiveness in object detection tasks.

4 MCX-YOLOv5 network

In practical scenarios, the detection of safety helmet wear is widely recognized as a challenging task, particularly due to the small size of the objects involved. This paper proposes an improved model called MCX-YOLOv5, which is built upon the YOLOv5 framework. The overall structure of the MCX-YOLOv5 model is depicted in Fig. 2, which illustrates how these improvements are integrated within the model architecture.

In the MCX-YOLOv5 model, we introduce the CSAM before the SPPF module. The SPPF module allows the model to handle input images of various sizes, but it may lead to information loss across different channels due to multiple pooling operations. By incorporating attention before the SPPF layer in the backbone network, we effectively improve the feature distribution, enabling better capture of contextual information, feature selection, and accurate object localization. This enhancement significantly improves the model's performance and inference capabilities.

As for the MAConv module, it involves splitting higher-level feature channels, down-sampling, and aggregating information. However, the splitting approach may not be suitable for shallow network layers with relatively low information repetition. Therefore, in the MCX-YOLOv5 model, we apply the MAConv module only to the intermediate three layers of the CBS module. This allows for more effective information aggregation without compromising the performance of the shallow layers.

4.1 MAConv structure

The convolution process generates multiple output channels, with each channel representing a distinct feature representation. However, it is common to observe redundancy in feature extraction, where multiple channels capture similar or overlapping features. Figure 3 illustrates this phenomenon,

Fig. 1 YOLOv5 network structure

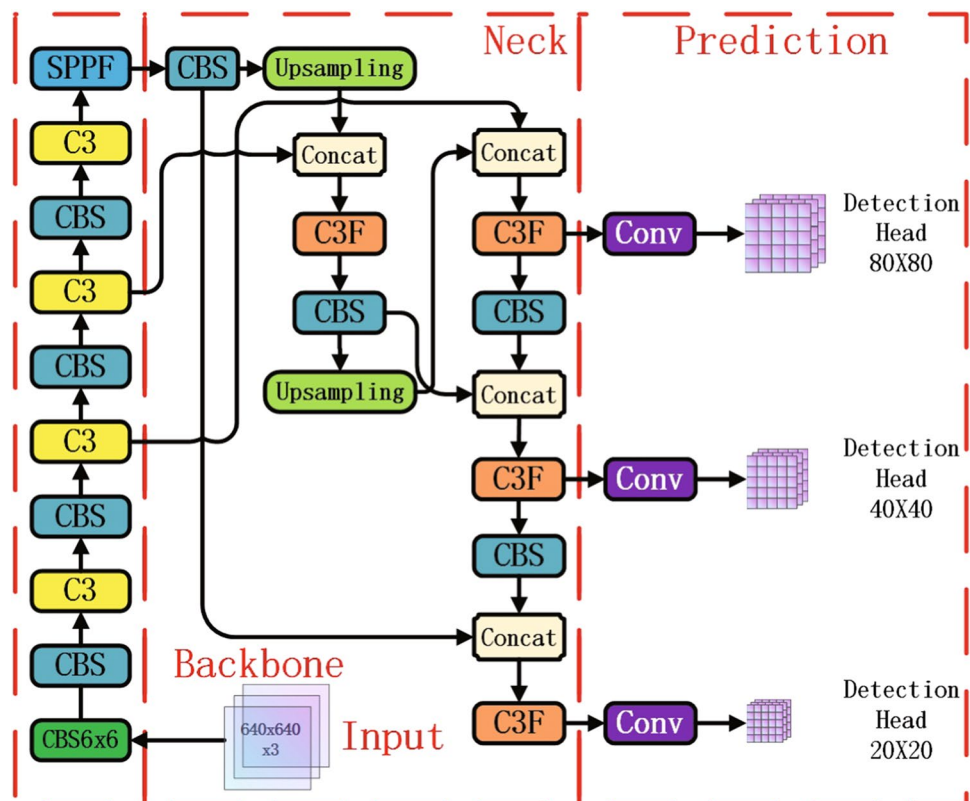


Fig. 2 MCX-YOLOv5 network structure

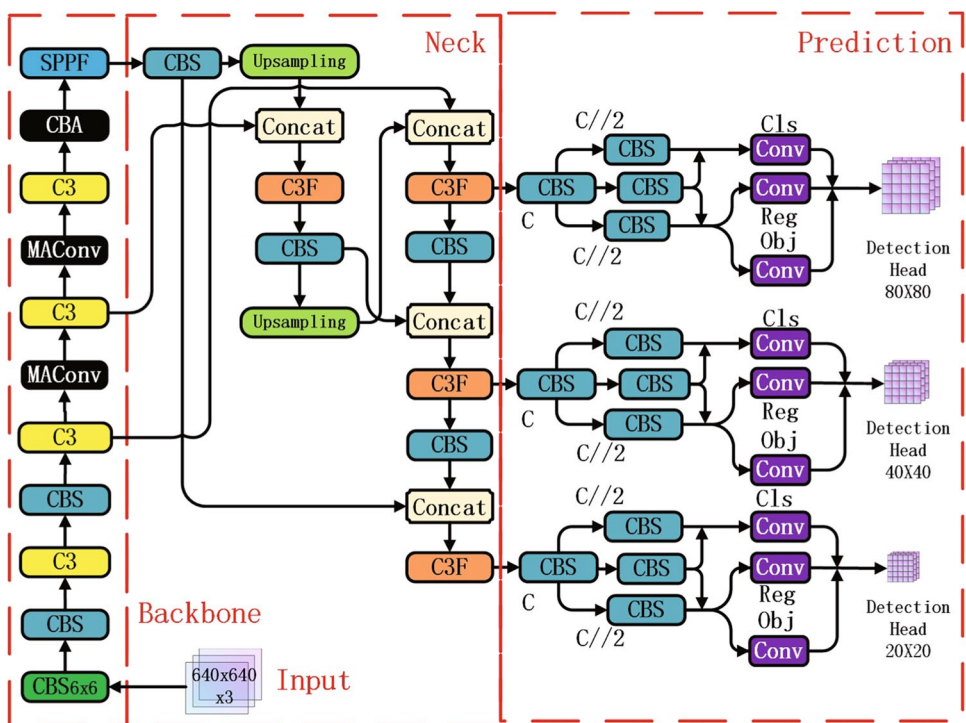
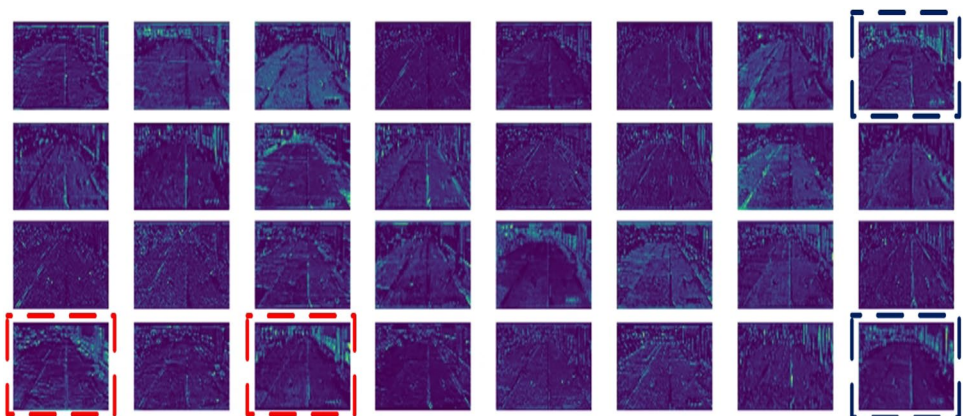


Fig. 3 The feature maps of the convolutional layer before SPPF



showing high similarity among the 32-channel feature maps, suggesting that certain features are redundant within the overall context. A comparison between the four feature maps highlighted by the red dashed box and the black dashed box reveals substantial similarity between them.

The approach presented in reference [24] tackles the issue of feature redundancy by partitioning the input feature map into representative and uncertainly redundant segments. The representative part is subjected to computationally intensive operations to extract essential information, while the uncertain redundant part, which contains minor hidden details, is processed using lightweight operations.

The MAConv structure involves duplicating the input feature map X , resulting in two sets of feature inputs, X_a and X_b . By utilizing downsampling with channel increase

and reducing the size of feature maps in the backbone network architecture, X_a and X_b are further divided into X_{a1} , X_{a2} , X_{b1} , and X_{b2} . Each half of the input features undergoes different downsampling operations, facilitating better information integration at multiple scales.

$$X_{ai} = X_{bi} = \frac{1}{2}X, \tag{1}$$

where X represents the input feature map. The subscript I distinguishes subsets within the two input feature sets (X_a and X_b), indexing different portions. For example, X_{a1} and X_{b1} may denote the first subset, while X_{a2} and X_{b2} represent the second. The I is used to differentiate components or indices within the two sets.

The module structure depicted in the diagram above (Fig. 4) illustrates the components of MAConv, which comprises four types of downsampling modules. The maximum pooling operation reduces the spatial dimension of the data while preserving crucial features. The average pooling helps to smooth out noise and disturbances in the input data, enhancing the network’s robustness. Compared to maximum pooling, average pooling exhibits better stability as it is less affected by outliers in the data. Asymmetric convolution (AC) increases the effective size of the convolutional kernel in one direction, thereby expanding the model’s receptive field and improving its ability to capture spatial features in the input signal.

Since the different downsampling modules extract features of varying importance, it becomes necessary to allocate the results obtained from multi-scale sampling. Prior to feature concatenation, a dynamic weight calculation based on softmax is performed on the four types of downsampling modules. This weight adjustment process considers the importance of the downsampling of feature information and optimizes the aggregation of features. By adaptively assigning weights, the model can effectively prioritize relevant feature information and achieve optimized feature aggregation.

$$\theta_j = \frac{e^{FP_j}}{e^{FP_1} + e^{FP_2} + e^{FP_3} + e^{FP_4}} \tag{2}$$

$$\text{Out} = Y_1 \times \theta_1 + Y_2 \times \theta_2 + Y_3 \times \theta_3 + Y_4 \times \theta_4. \tag{3}$$

In the Eq. (2), the j represents the index for different weights, θ_j represents the corresponding feature weight (weight1-4) and FP_j denotes the aggregated single-channel feature after channel dimension reduction. Y_j represents the corresponding feature maps generated by the four types of downsampling modules, and Out is the module output after weight adjustment. The convolution operation before the softmax function maps the aggregation results of the four

modules onto four channels. The computed weight values are multiplied inversely, and finally, the channels are combined to obtain the feature map with adaptively adjusted weights.

4.2 Coordinate-spatial attention

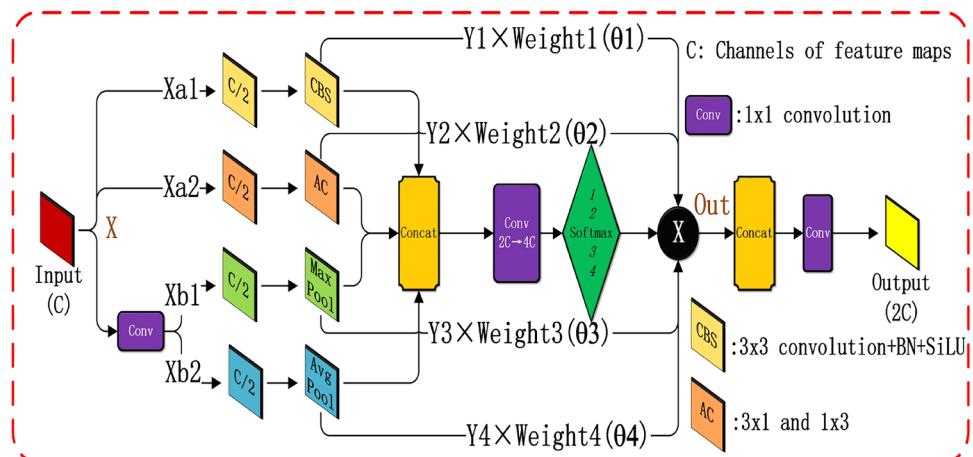
The incorporation of attention mechanisms into neural network models is of paramount importance. Squeeze-and-Excitation Networks (SE-Net) have emerged as prominent models in the field of channel attention processes [25]. Equation (4) can be employed to offer a more comprehensive depiction of the computational procedure employed by the channel attention module. The Convolutional Block Attention Module (CBAM) is widely recognized as a prominent model in the field of spatial attention mechanisms [26]. The CBAM integrates both channel attention and spatial attention. The channel attention module is designed to dynamically modify channel-wise attributes in order to enhance the significance of relevant channels. Simultaneously, the spatial attention module performs the task of recalibrating features in a spatial manner by recording the interdependencies that exist among various spatial locations.

$$M_c(F) = \sigma(\text{MLP}(\text{AvgPool}(F)) + \text{MLP}(\text{MaxPool}(F))), \tag{4}$$

where the symbol σ denotes the sigmoid activation function, the MLP signifies a fully linked layer, AvgPool and MaxPool correspond to average pooling and maximum pooling processes, and F denotes the input feature map.

The spatial attention module is an attention mechanism applied to the spatial dimension of the feature map. It aims to select locally dominant features by aggregating the most salient features within each channel, allowing the network to focus more on important local features. As illustrated in Fig. 5, this module considers both feature similarity and

Fig. 4 MAConv structure



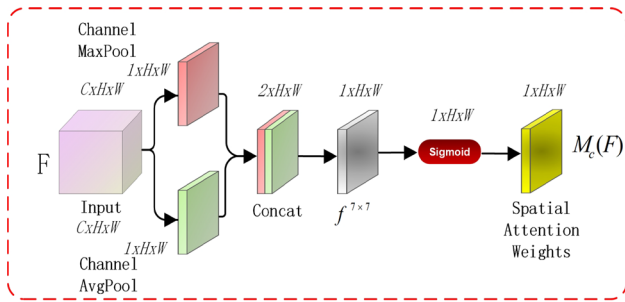


Fig. 5 Spatial attention

spatial distribution. Equation (5) presents the calculation method of the spatial attention module. And in Figs. 5, 6, and 7, the C , W , and H , respectively, denote the number of channels, width, and height of the feature map.

$$M_c(F) = \sigma(f^{7 \times 7} \begin{bmatrix} \text{Concat}(\text{Channel_MaxPool}(F)), \\ \text{Channel_AvgPool}(F) \end{bmatrix}) \quad (5)$$

where $f^{7 \times 7}$ represents a 7×7 convolutional kernel, Concat denotes channel concatenation, the symbol σ denotes the sigmoid activation function, Channel_MaxPool refers to the maximum pooling operation along the channel dimension, and Channel_AvgPool refers to the average pooling operation along the channel dimension. The F denotes the input feature map. $M_c(F)$ is a feature map containing the importance weights for each spatial position.

The coordinate attention (CA) [27], as illustrated in Fig. 6, is a form of channel attention mechanism that specifically targets the spatial dimensions. The CA integrates two-dimensional features by encoding the features along the X (width) and Y (height) directions, capturing long-range dependencies within each spatial direction. This approach helps preserve positional relationships

Fig. 6 CA structure

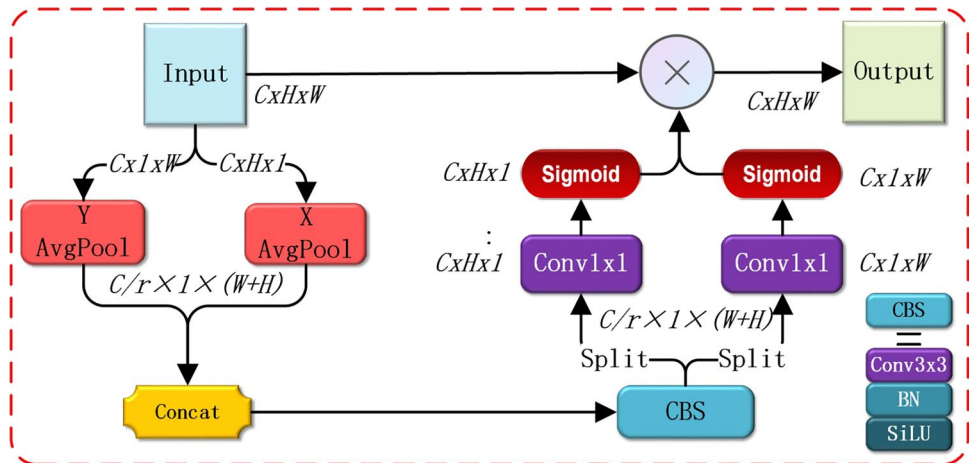
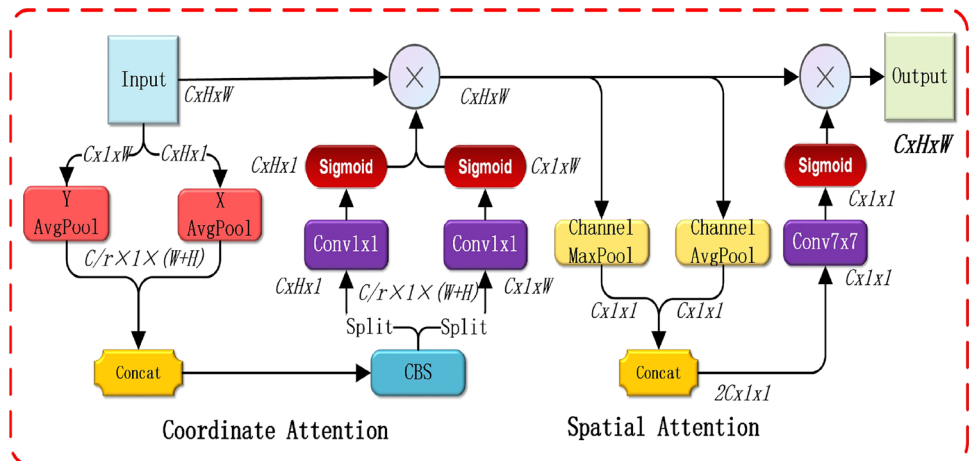


Fig. 7 CSAM structure



that would typically be lost during traditional pooling-based aggregation. By incorporating the CA, the model can effectively capture spatial information and enhance its understanding of the overall context.

The objective of the CA mechanism is to generate an attention vector, denoted as attention (X, Y) , for each position (X, Y) in the input feature map, represented as X with dimensions H, W , and C (height, width, and number of channels). The attention vector, Attention (X, Y) , is computed based on the spatial coordinates of the position (X, Y) and can be expressed as follows:

$$\text{Attention}(X, Y) = \text{Sigmoid}(f(X, Y)). \quad (6)$$

The feature vector $f(X, Y)$ is obtained by applying two convolutional layers at the position (X, Y) in the feature map. One-dimensional average pooling is performed separately along the X and Y directions, decomposing the global average pooling operation. The resulting one-dimensional vectors in the two directions are then convolved to fuse the information. Finally, the fused vector is split into two sets of position coordinates in the H and W directions. This enhances the model's ability to perceive input features in different directions, allowing it to handle directional features more effectively and improve overall performance. The representation is as follows:

$$Z_c^h(h) = \frac{1}{W} \sum_{0 \leq k \leq W} x_c(h, k) \quad (7)$$

$$Z_c^w(w) = \frac{1}{H} \sum_{0 \leq l \leq H} x_c(w, l) \quad (8)$$

$$f^{h-w} = \delta(F_1([Z_c^w, Z_c^h])) \quad (9)$$

$$g^w = \sigma(F_w(f^{h-w})) \quad (10)$$

$$g^h = \sigma(F_h(f^{h-w})). \quad (11)$$

In Eq. 7, $x_c(h, k)$ denotes the value of the k th element at channel c and position h in the input feature map. while $Z_c^h(h)$ represents the output of channel c at position h . Here, H signifies a specific position along the height dimension, and W denotes the width at this position on the feature map. Equation 8 parallels Eq. 7, differing only in the spatial direction. In Eq. 9, f^{h-w} is the result of applying the 1×1 convolution F_1 to the concatenation of Z_c^w and Z_c^h . Typically, δ denotes the Hard-Swish activation function. In Eqs. (10) and (11), g^w and g^h are the outputs obtained by applying the sigmoid activation function (σ) to the results of the respective 1×1 convolutions $F_w(f^{h-w})$ and $F_h(f^{h-w})$.

The CSAM combines the spatial filtering capability of the CBAM and channel filtering with position preservation from the CA. As illustrated in Fig. 7, the CA establishes long-range dependencies in the X and Y directions, ensuring a basic receptive field. Subsequently, the spatial attention module extracts local advantageous information based on this foundation. This integration of both spatial and channel filtering allows the CSAM to capture both long-range dependencies and local details, contributing to the model's enhanced perception and feature representation.

4.3 VXDetect decoupled head

In object detection, there are two different approaches to network design known as the coupled head and the decoupled head. The coupled head involves connecting the object detection head and the classification head in the network, allowing them to share the features extracted during the feature extraction process. In object detection, both localization and classification tasks are performed simultaneously. However, the inherent differences between these two tasks can lead to an averaging effect when the features are fused.

On the other hand, the decoupled detection head performs the classification and regression tasks in parallel, with separate feature extraction parts for each task. While this approach offers flexibility and modularity, a drawback is that it requires more computational resources and a larger network size. This is because it involves training the object detection and classification heads independently. Therefore, adopting a lightweight, information-fused, decoupled approach becomes necessary.

In this paper, the VXDetect decoupling approach is built upon the decoupling approach in YOLOv6 and further reduces the computational parameters. As shown in Fig. 8b, in VXDetect, the two 3×3 convolution channels are halved, and the other half uses a 1×1 convolution. The 1×1 convolution is shared between the classification and regression tasks. During the gradient backpropagation process, VXDetect employs a gradient fusion approach in the heads to share some underlying information and representation capacity, as opposed to the complete decoupling shown in Fig. 8a. By sharing the feature representation, different tasks can influence each other and learn shared features. This approach strikes a balance between efficiency and performance in object detection tasks.

5 Experiments and analysis

In this section, we will start by giving a concise description of the experimental setup and parameters used. Then, we will introduce the dataset utilized for our experiments and the evaluation metrics employed to assess the model's

Fig. 8 Decoupled head structure. **a** V6Detect and **b** VXDetect

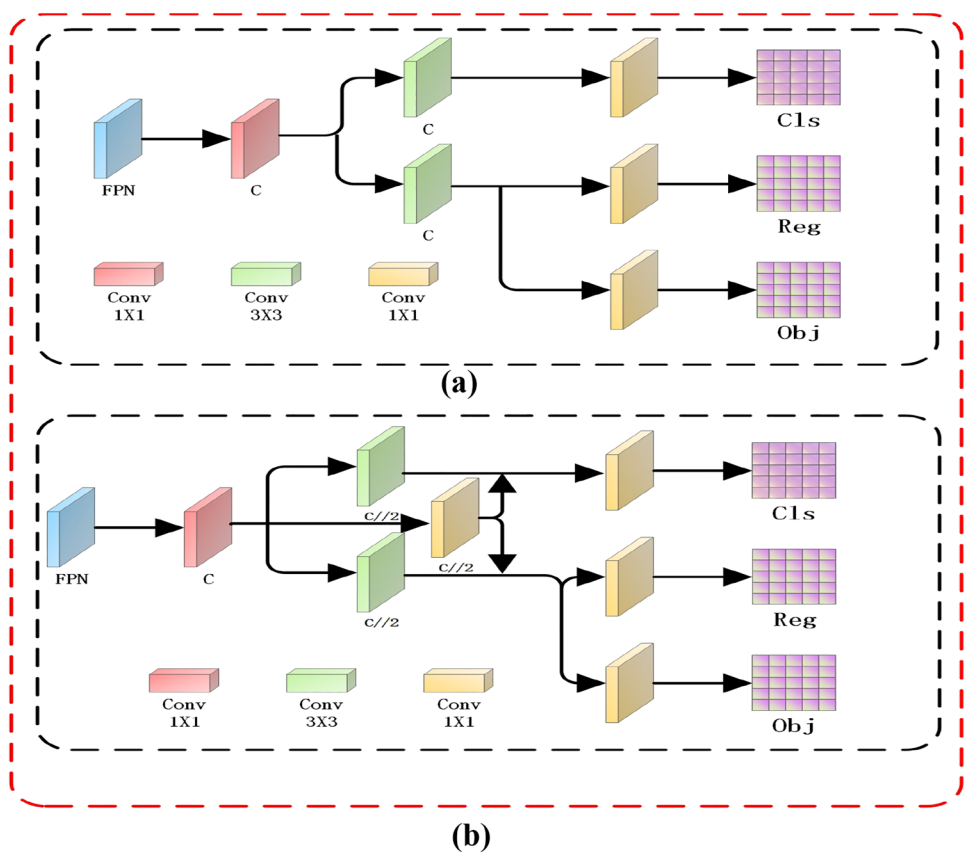


Table 2 Experimental environment

Lab environment	Environment configuration
system	Windows10
CPU	12th Gen Intel (R) Core (TM) i9-12900F
GPU	NVIDIA GeForce RTX 3090
RAM	24G
Framework	Pytorch1.13.0
CUDA version	11.6
Python version	3.9

performance. Finally, we will present a detailed analysis of the results obtained from our experiments.

5.1 Experimental environment

The environment used in this experiment is shown in Table 2.

5.2 Experimental parameters and experimental evaluation

Before training, the automatic anchor box adaptation feature is utilized to adjust the sizes of the prior boxes. To ensure consistency in the experiments, no pre-trained weights are utilized.

The optimization algorithm employed is Stochastic Gradient Descent (SGD), with a batch size of 24. The total number of training iterations is set at 400. The initial learning rate is 0.01, and the final learning rate is 0.001. A cosine annealing learning rate adjustment strategy is implemented, with a momentum value of 0.937.

The evaluation metric employed in the experiments is the mean Average Precision (mAP), which combines precision and recall metrics. Precision (P) represents the percentage of correctly predicted positive samples among all predicted positive samples, while recall (R) represents the percentage of correctly predicted positive samples among all actual positive samples. True Positive (TP) is the number of correctly predicted positive samples. False Positive (FP) is the number of incorrectly predicted positive samples. False Negative (FN) is the number of incorrectly predicted negative samples. Average Precision (AP) is the area under the precision-recall curve. The $P(r)$ represents the P at a given recall rate r . The mAP is the average AP across all classes. The specific calculation formulas are as follows:

$$P = \frac{TP}{TP + FP} \tag{12}$$

$$R = \frac{TP}{TP + FN} \quad (13)$$

$$AP = \int_0^1 P(r)dr \quad (14)$$

$$mAP = \frac{\sum_{n=1}^m AP_n}{m}, \quad (15)$$

where m represents the number of detection classes and n represents the index from 1 to m .

5.3 Experimental dataset

Due to the absence of specific datasets tailored for safety helmet detection in the power warehousing scenario, we compiled a dataset comprising 4000 images sourced from historical monitoring data within the power warehousing industry. We named this dataset “Electric Warehousing Helmet Detection” (EWHHD). Given the relatively limited number of images, we partitioned the dataset into training, validation, and test sets, maintaining an 8:1:1 ratio. Assigning a larger proportion of samples to the training set facilitates enhanced generalization of the model in the presence of data scarcity.

Furthermore, we conducted supplementary experiments to assess the performance of the MCX-YOLOv5 model in detecting safety helmet usage across general scenarios. To achieve this, we obtained a dataset of 5000 safety helmet detection images from the Kaggle platform, which we named “Hard Hat Workers Detection” (HHWD). The allocation ratio for this supplementary experiment mirrored that of the self-collected dataset. Both datasets encompass the following class labels: “head_with_helmet” (0), “head_no_helmet” (1), “person_with_helmet” (2), and “person_no_helmet” (3).

A graphical depiction of the sample label instances and class ratios for both datasets is presented in Fig. 9.

In contrast to the HHWD dataset shown in Fig. 9b, the self-collected dataset presented in Fig. 9a demonstrates a more balanced distribution of label categories. However, an inherent limitation of the self-collected dataset is the overrepresentation of small objects as targets. To ensure a comprehensive evaluation of the model's detection capabilities across scenes featuring objects of different sizes (small, medium, and large), we employed the PASCAL Visual Object Classes 2012 (VOC2012) and VOC2007 [28] datasets to establish a novel dataset. Additionally, we utilized the widely adopted Safety Helmet Wearing Dataset (SHWD) dataset to compare our research outcomes with those of other researchers for performance validation.

The new dataset configuration involved combining the training sets from both VOC 2012 and VOC 2007, resulting in a total of 8218 images for the training set. The validation set consisted of the test set from VOC 2012, encompassing 5823 images. Finally, the test set comprised the validation set from VOC 2007, consisting of 2510 images. By adopting this dataset configuration, we aimed to evaluate the generalization performance of the proposed model, ensuring its effectiveness across diverse scenarios.

5.4 Results and analysis

The algorithm suggested in this study was subjected to module ablation tests, in which the assessment metrics employed were mAP and model parameter count. The mAP metric is commonly used in evaluating the accuracy of algorithms. Additionally, the parameter count offers valuable information regarding the size of the model. The results of the ablation experiments are displayed in Table 3, where various model configurations are identified as M-YOLOv5, MC-YOLOv5, and MCX-YOLOv5.

The M-YOLOv5 configuration integrates the MAConv architecture, whereas the MC-YOLOv5 configuration

Fig. 9 Sample distribution. **a** EWHHD dataset and **b** HHWD dataset

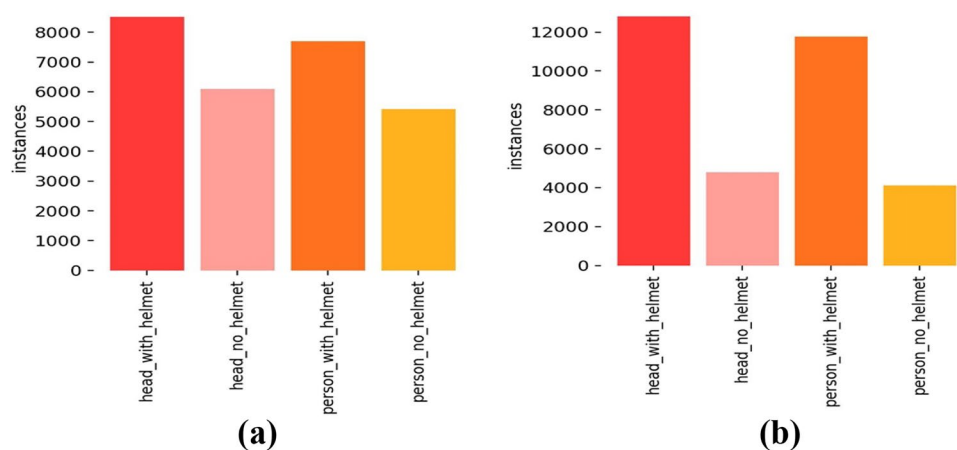


Table 3 Results of ablation experiments on the test set in the EWHHD dataset

Method	P (%)	R (%)	mAP50 (%)
YOLOv5s	86.2	79.0	84.4
M-YOLOv5	86.3	79.2	84.8
MC-YOLOv5	86.9	79.8	85.5
MCX-YOLOv5	89.0	81.6	87.1
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
48.5	47.7	7.02	15.8
48.0	48.1	6.32	15.2
49.5	48.5	6.35	15.2
53.4	51.0	8.30	19.0

includes both the MAConv architecture and the CSAM. The MCX-YOLOv5 configuration incorporates the utilization of MAConv, CSAM, and VXDetect.

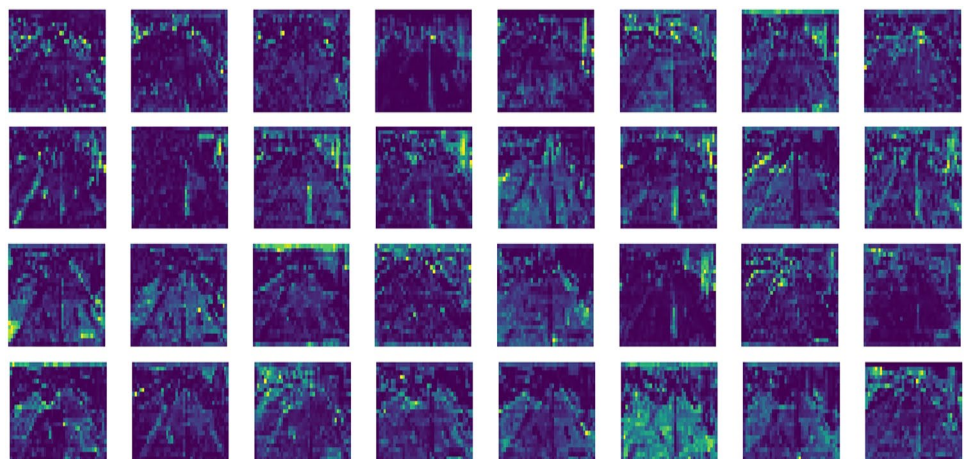
The results of the module ablation tests are presented in Table 3. These experiments were conducted to evaluate the impact of different modules on the performance of the proposed algorithm. The findings demonstrate the effectiveness of the lightweight module, MAConv, as it achieved a 0.4% improvement in mAP at 50% intersection over union (IoU) [29] while reducing the parameter count by 0.7 million. This indicates that the implemented module effectively enhances the model's performance while reducing its complexity.

Figure 10's visual representation demonstrates the benefits of using the multi-scale sampling module. The reduction in similar features and increased utilization of redundant features lead to better testing results. Furthermore, by integrating the CSAM into the model, MC-YOLOv5 showed significant improvements compared to the baseline YOLOv5 model. It achieved a 1.1% enhancement in mAP at 50% and a 1.0% increase in mAP at 75%.

The class activation map (CAM) display shows how the CSAM has improved the weight distribution of the model.

The visual investigation depicted in Fig. 11 examines the impact of the CSAM on the recognition of Person A. The input image is depicted in Fig. 11a, showcasing the detected target A. The heatmap representation in Fig. 11b demonstrates the situation wherein the model exhibits a deficiency in attention, leading to dispersed attention weights on the target. Figure 11c illustrates the CAM obtained by applying the CSAM to participant A. Clearly, the introduction of the CSAM improves the model's weight allocation, leading to a more concentrated and confident detection of the target object.

By incorporating the VXDetect decoupled head, the model's detection performance was further improved. The mAP increased by 2.7% at the IoU threshold of 0.5 (mAP50), and the mAP at the IoU threshold of 75 (mAP75) showed a substantial improvement of 4.9%. Table 4 presents a comprehensive comparison of the experimental results obtained using the three different detection heads. V6Detect represents the decoupled head utilized in YOLOv6, while Decoupled Detect corresponds to the decoupled head employed in YOLOX [30]. It is evident from Table 4 that the VXDetect decoupled head achieved comparable accuracy while

Fig. 10 Visualization of the feature map of the layer before SPPF

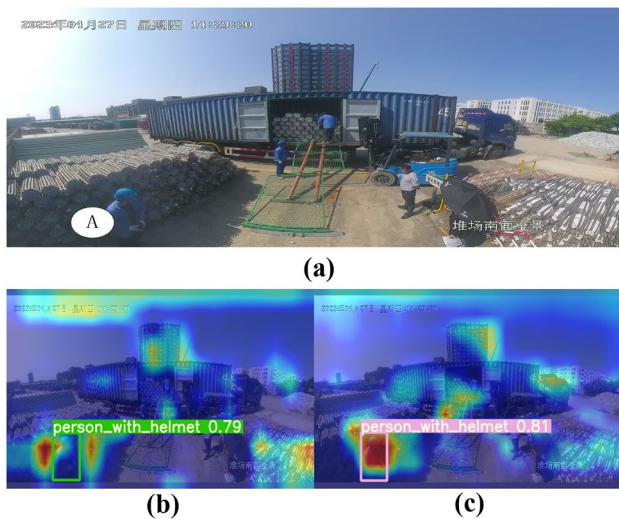


Fig. 11 Visualization results. **a** Target A, **b** without CSAM visualization, and **c** CSAM visualization

exhibiting significantly fewer parameters and computations compared to both V6Detect and Decoupled Detect.

The curves depicted in Fig. 12 visually represent the performance of the model in terms of loss and accuracy. It is clear that the proposed model outperforms the YOLOv5s model in both accuracy and convergence speed. The proposed method exhibits higher accuracy and more efficient convergence, demonstrating its effectiveness in object detection tasks.

The proposed model in this study exhibits notable improvements in accuracy while incurring only a marginal increase in computational overhead compared to the original model. The detection results of various models on the validation set under identical parameter configurations are presented in Table 5. It is evident that the model proposed in this paper outperforms other models in terms of accuracy while maintaining a comparable parameter count. Figure 13 presents a comparative analysis of detection results between two images. The left image highlights instances where certain objects are subject to detection challenges, including

partial occlusion and difficulties in detecting objects at medium to long distances. However, through optimization efforts, the right image demonstrates improved detection performance, effectively addressing the aforementioned challenges. The detection results in Fig. 13a have been significantly enhanced and refined in Fig. 13b.

To assess the efficacy of the helmet-wearing detection model across different settings, we carried out tests utilizing the HHWD dataset. The approach for data segmentation, hyperparameter selection, and training strategies adhered to the same methodology as that employed for the EWHD dataset.

Table 6 displays the experimental findings, wherein the performance of the model is compared between the HHWD dataset and the EWHD dataset. It is important to acknowledge that the observed enhancement in performance is significantly diminished due to disparities in data features between the two datasets. On average, there is a 1.5% gain in precision across various IoU thresholds.

However, examining Table 7, which provides a comprehensive evaluation of several models, it is evident that our model exhibits superior performance in comparison to yolov6s, yolov7-tiny, and yolov8s on the HHWD dataset. This observation serves as evidence that the modifications implemented in this study have led to improvements in performance.

Figure 14 depicts the accuracy/loss curve of the ablated model employed on the HHWD dataset.

On the SHWD dataset, which has been widely used in various studies, our proposed model remains highly competitive. Table 8 shows that our model has a more balanced advantage when comparing the models by reproducing the four most recent papers published in journals. The models in references 1 and 2 improve the detection accuracy by dramatically increasing the FLOPs while ignoring the limitations of detection speed and computational resources. The model in reference 3 employs a lightweight architecture to improve inference speed. However, this leads to a significant decrease in model accuracy. Reference 4 has a similar size to the model in this paper but also has a slightly lower

Table 4 Comparison of the detection results of the different decoupled heads on the test set in the EWHD dataset

Method	<i>P</i> (%)	<i>R</i> (%)	mAP50 (%)
V6Detect	88.6	81.6	87.1
Decoupled Detect	87.6	81.7	86.6
VX Detect	89.0	81.6	87.1
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
53.2	51.2	12.89	27.2
53.1	51.0	13.66	55.6
53.4	51.0	8.30	19.0

Fig. 12 Variation of performance evaluation metrics with the number of iterations for different groups in the EWHD dataset. **a** mAP@0.5, and **b** validation set loss

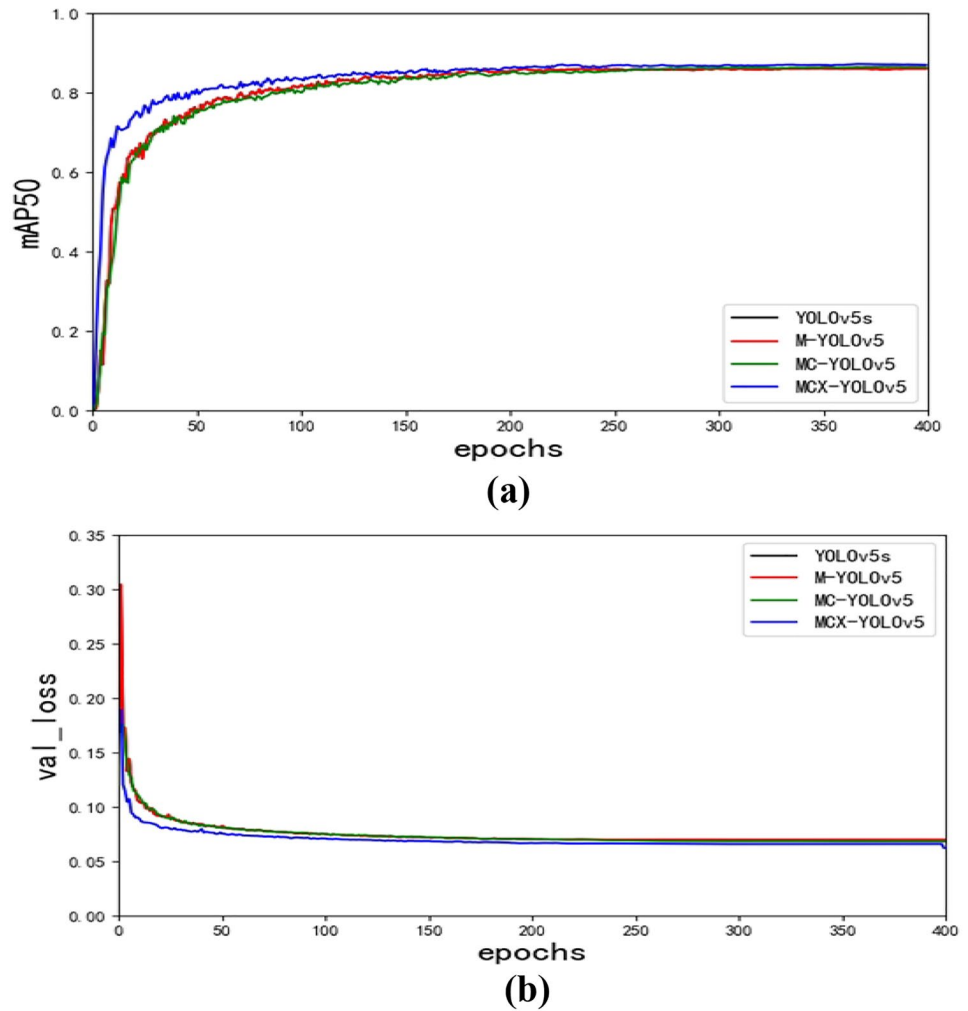


Table 5 Comparison of the detection results of the five models on the test dataset in the EWHD dataset

Method	P (%)	R (%)	mAP50 (%)
YOLOv5s	86.2	79.0	84.4
YOLOv6s	80.5	51.1	83.1
YOLOv7-tiny	87.5	79.7	85.2
YOLOv8s	87.3	79.6	86.1
MCX-YOLOv5	89.0	81.6	87.1
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
48.5	47.7	7.02	15.8
46.8	46.9	18.50	45.17
47.6	47.2	6.01	13.0
54.5	51.3	11.13	28.4
53.4	51.0	8.30	19.0

refinement performance than our model on the SHWD dataset. Moreover, with respect to the results of the three datasets mentioned above, the detection capability of MCX-YOLOv5 in the field of helmet-wearing detection is also comparable

to that of current state-of-the-art single-stage detection models and requires less computational resources.

The validation experiments conducted on a subset of the VOC dataset confirmed the generalizability and excellent

Fig. 13 Experimental detection comparison results. **a** YOLOv5s, and **b** MCX-YOLOv5

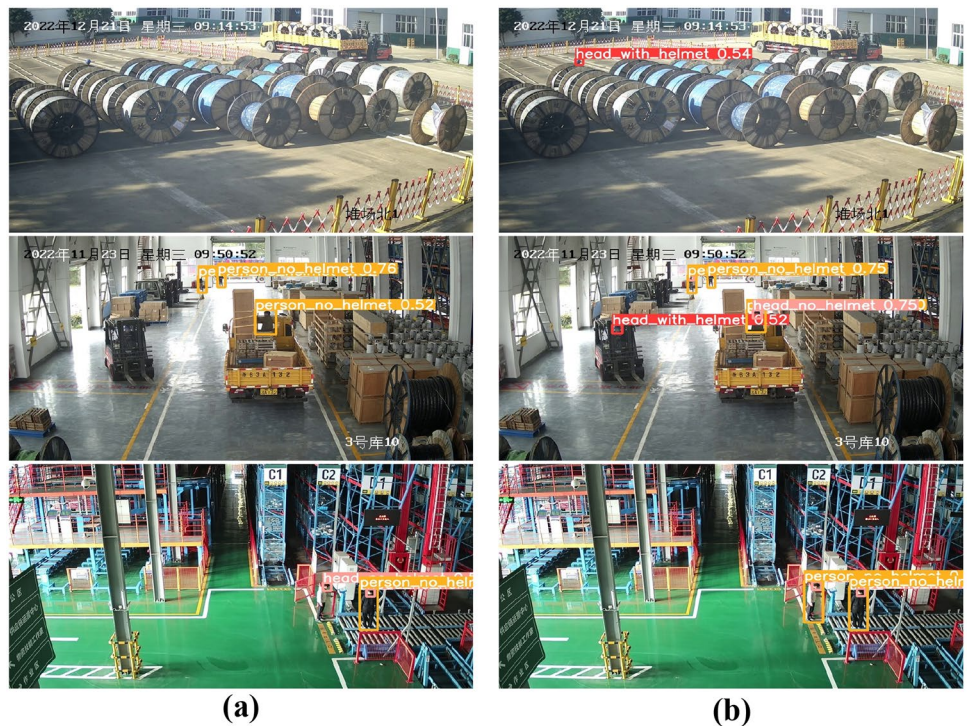


Table 6 Results of ablation experiments on the test set in the HHWD dataset

Method	<i>P</i> (%)	<i>R</i> (%)	mAP50 (%)
YOLOv5s	90.8	84.7	91.7
M-YOLOv5	90.3	85.6	91.9
MC-YOLOv5	90.2	85.9	92.1
MCX-YOLOv5	91.1	87.0	92.5
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
73.0	62.5	7.02	15.8
73.4	62.5	6.32	15.2
73.9	63.2	6.35	15.2
75.7	64.8	8.30	19.0

performance of our proposed model in other detection tasks. The hyperparameters used in the training process remained unchanged, and the number of training iterations was extended to 500. We trained and evaluated four other models, namely YOLOv5s, YOLOv7-tiny, YOLOv6s, and YOLOv8s, separately and compared their performance with our proposed model. As shown in Table 9, the detection results on the test set clearly indicate that our proposed model outperforms other models significantly at the IoU threshold of 0.5. Although it may not have an advantage at higher thresholds, in practical use, IoU threshold 0.5 is the most commonly used design threshold.

Figure 15 presents the performance improvement curves of the validation dataset during the training process of the

five models. Compared to the other models, our proposed model converges faster.

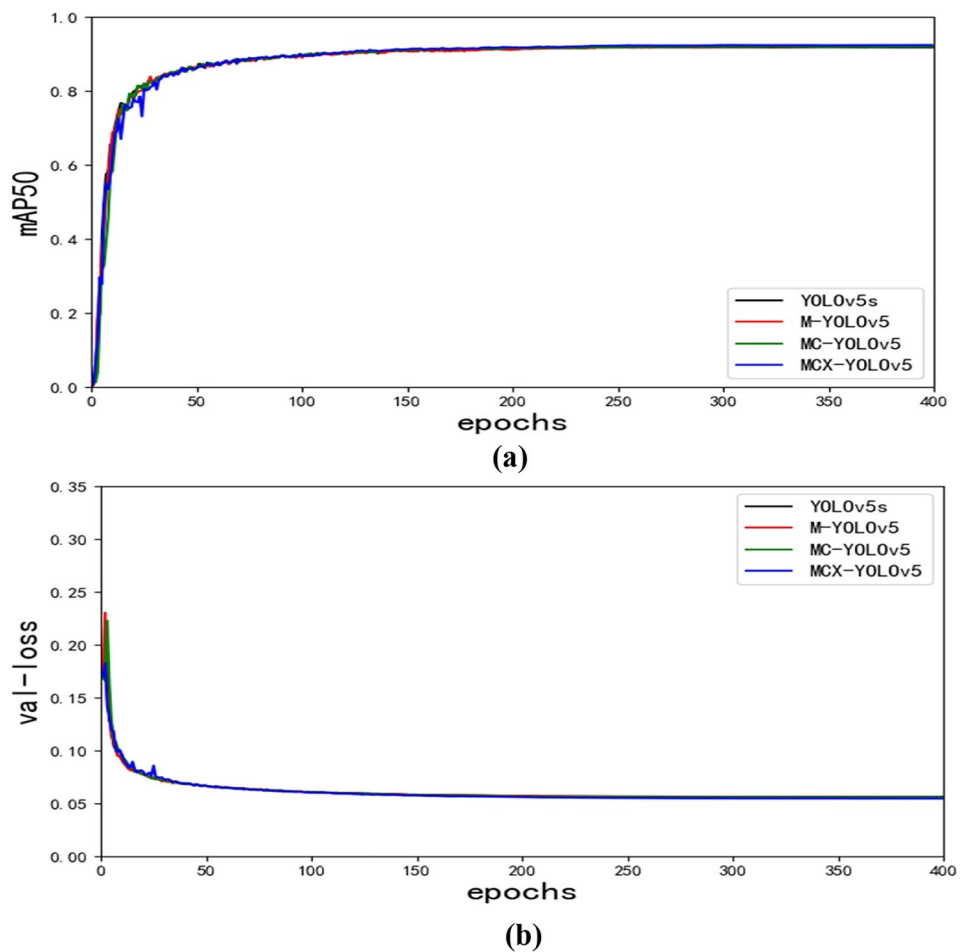
Upon evaluating the outcomes of the experiments and analyzing the performance improvement curves during the training process, our observations have determined that the suggested model presents resilient generalization abilities on various tasks and datasets. This signifies that said model not only attains outstanding performance in specific domains and datasets but also adjusts aptly to novel and unfamiliar circumstances, thus highlighting its vast scope of application.

It is important to note that during the comparative experiments, multiple prevalent object detection models were trained and evaluated. Out of these models, the

Table 7 Comparison of the detection results of the five models on the test set in the HHWD dataset

Method	<i>P</i> (%)	<i>R</i> (%)	mAP50 (%)
YOLOv5s	90.8	84.7	91.7
YOLOv6s	89.0	82.2	89.1
YOLOv7-tiny	88.8	87.3	92.1
YOLOv8s	89.3	86.3	91.9
MCX-YOLOv5	91.1	87.0	92.5
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
48.5	73.0	62.5	7.02
46.8	69.8	60.8	18.50
47.6	71.7	61.4	6.01
54.5	75.3	64.8	11.13
53.4	75.7	64.8	8.30

Fig. 14 Variation of performance evaluation metrics with the number of iterations for different groups in the HHWD dataset. **a** mAP@0.5, and **b** validation set loss



proposed model displays remarkable generalization performance due to the combined effect of the three optimization methods employed during the training process. The employed techniques enable the model to capture data patterns and features more effectively, leading to

improved generalization. Additionally, the model’s successful detection of small objects wearing safety helmets at long distances provides further evidence of its exceptional ability to handle complex and difficult scenarios. This feature is pivotal in meeting various needs in practical

Table 8 Comparison of the detection results of the nice models on the test set in the SHWD dataset

Method	<i>P</i> (%)	<i>R</i> (%)	mAP50 (%)
YOLOv5s	94.0	89.4	94.5
YOLOv6s	92.9	88.7	92.65
YOLOv7-tiny	93.2	89.3	94.2
YOLOv8s	93.7	91.0	95.0
Reference1 [31]	93.1	91.0	95.8
Reference2 [32]	92.8	91.2	95.5
Reference3 [33]	93.7	87.9	93.6
Reference4 [34]	93.5	90.0	95.2
MCX-YOLOv5	93.0	90.4	95.5
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
65.3	61.7	7.02	15.8
62.6	59.5	18.50	45.17
62.6	59.5	6.01	13.0
66.0	63.1	11.13	28.4
66.7	61.9	10.50	197.9
66.8	61.8	7.3	72.6
62.6	59.2	4.11	9.10
66.0	62.0	7.2	18.8
66.5	62.5	8.30	19.0

Table 9 Comparison of the detection results of the five models on the test set in the VOC dataset

Method	<i>P</i> (%)	<i>R</i> (%)	mAP50 (%)
YOLOv5s	75.0	63.0	70.2
YOLOv6s	78.0	62.0	70.5
YOLOv7-tiny	74.1	64.1	70.6
YOLOv8s	78.4	63.2	70.9
MCX-YOLOv5	75.8	65.7	72.9
mAP75 (%)	mAP50-95 (%)	Params (M)	FLOPs (B)
47.3	44.1	7.02	15.8
52.4	48.8	18.50	45.17
46.6	43.9	6.01	13.0
54.4	50.0	11.13	28.4
51.0	47.4	8.30	19.0

settings, particularly in the areas of surveillance, security, and industry.

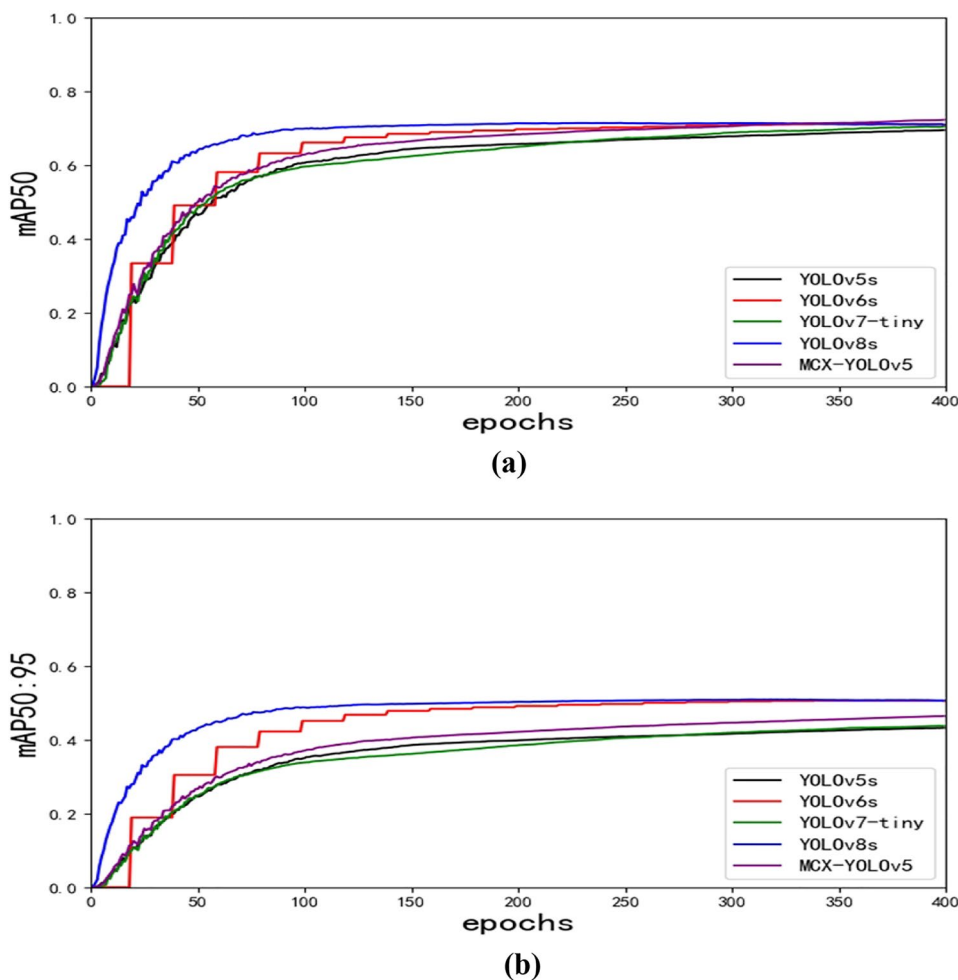
6 Conclusions

This paper presents a comprehensive investigation into the detection of small objects and the enforcement of safety helmet usage in warehousing scenarios. Conventional detection algorithms commonly encounter issues such as low detection accuracy, missed detections, and false alarms. To overcome

these challenges, we propose the integration of a CSAM, which effectively enhances the model's ability to attend to relevant regions. Moreover, we introduce a weighted down-sampling module, known as MACConv, specifically tailored for intermediate feature maps, thereby promoting greater diversity in lower-level features. Additionally, we replaced the coupled head with a lighter decoupled head, VXDetect, which effectively separates the classification and regression tasks.

After conducting a thorough analysis of the experimental outcomes, we confirm the exceptional efficacy of the

Fig. 15 Variation of performance evaluation metrics with the number of iterations for different groups in the VOC dataset. **a** mAP@0.5, and **b** mAP@0.5:0.95



suggested algorithm in identifying small objects and ensuring the implementation of safety helmets in warehouse settings. Significantly, we have observed the noteworthy adaptability of the model in dealing with varied data and practical scenarios. The improvements observed transcend specific datasets or scenarios and have been verified in diverse contexts, encompassing real-world situations and the VOC dataset. This implies that our model not only identifies patterns from particular training data but also generalizes proficiently to unobserved conditions, showcasing robust adaptability. The successful demonstration of this generalization capacity instills faith in the potential practical uses of our model. The model demonstrates strong adaptability across diverse environments, ranging from monitoring warehouses to industrial production lines, while maintaining a high level of detection accuracy. This affirms the model's superiority in specific scenarios and underscores its resilience in managing unknown situations and evolving data.

In future studies, we will further explore the generalization performance of the model while dealing with challenges across various industries and domains. We aim to strengthen the model's reliability and applicability in different practical

scenarios by conducting more tests on real-world applications to validate its generalization. Additionally, we will investigate methods that integrate object detection with object tracking and pedestrian re-identification techniques. We also intend to conduct in-depth research on lightweight methods, such as network pruning, to facilitate deployment on edge devices. In conclusion, our study offers not only an optimized detection method in warehouse scenarios but also highlights the model's strong generalization capabilities across a broad range of practical applications.

Author contributions HX: data curation, writing—original draft, writing—review and editing, software. ZW supervision, validation, conceptualization, and project administration. All authors reviewed the manuscript.

Data availability The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The authors declare no competing interests.

References

- Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
- Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. I-511–I-518, Kauai, HI, USA (2001). <https://doi.org/10.1109/CVPR.2001.990517>
- Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, pp. I-886–I-893. Institute of Electrical and Electronics Engineers (IEEE), San Diego, CA, USA (2005). <https://doi.org/10.1109/CVPR.2005.177>
- Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995). <https://doi.org/10.1007/BF00994018>
- Chen, J., Kao, S.-h., He, H., Zhuo, W., Wen, S., Lee, C.-H., Chan, S.-H.G.: Run, don't walk: chasing higher FLOPS for faster neural networks. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, BC, Canada, pp. I-12021–I-12031 (2023). <https://doi.org/10.1109/CVPR52729.2023.01157>
- Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., Wei, X.: YOLOv6: a single-stage object detection framework for industrial applications (2022). arXiv:2209.02976
- Park, M.-W., Brilakis, I.: Construction worker detection in video frames for initializing vision trackers. *Autom. Constr.* **28**(15), 15–25 (2012). <https://doi.org/10.1016/j.autcon.2012.06.001>
- Rubaiyat, A.H.M., Toma, T.T., Kalantari-Khandani, M., Rahman, S.A., Chen, L., Ye, Y., Pan, C.S.: Automatic detection of helmet uses for construction safety. In: *2016 IEEE/WIC/ACM International Conference on Web Intelligence Workshops (WIW)*, Omaha, NE, USA, pp. 135–142 (2016). <https://doi.org/10.1109/WIW.2016.045>
- Du, S., Shehata, M., Badawy, W.: Hard hat detection in video sequences based on face features, motion, and color information. In: *Proceedings of the 3rd International Conference on Computer Research and Development*, Shanghai, China, pp. 25–29 (2011). <https://doi.org/10.1109/ICCRD.2011.5763846>
- Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, pp. 580–587 (2014). <https://doi.org/10.1109/CVPR.2014.81>
- Girshick, R.: Fast R-CNN. In: *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, pp. 1440–1448 (2015). <https://doi.org/10.1109/ICCV.2015.169>
- 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.* **39**(6), 1137–1149 (2016). <https://doi.org/10.1109/TPAMI.2016.2577031>
- 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, pp. 779–788 (2016). <https://doi.org/10.1109/CVPR.2016.91>
- Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, pp. 6517–6525 (2017). <https://doi.org/10.1109/CVPR.2017.690>
- Redmon, J., Farhadi, A.: YOLOv3: An Incremental Improvement (2018). arXiv preprint [arXiv:1804.02767](https://arxiv.org/abs/1804.02767)
- Wang, A., Liu, A., Ouyang, W.: YOLOv4: optimal speed and accuracy of object detection. arXiv preprint [arXiv:2004.10934](https://arxiv.org/abs/2004.10934) (2020)
- Wang, C.-Y., Bochkovskiy, A., Liao, H.-Y.M.: YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv:2207.02696 (2022)
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: SSD: Single Shot MultiBox Detector, vol. 9905, pp. 21–37. Springer, Berlin (2016). https://doi.org/10.1007/978-3-319-46448-0_2
- Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., Tian, Q.: CenterNet: Keypoint Triplets for Object Detection. arXiv:1904.08189 (2019)
- Sun, G., Li, C., Zhang, H.: Safety helmet detection method with fusion of self-attention mechanism. *Comput. Eng. Appl.* **58**(20), 300–304 (2022)
- Song, X., Wu, Y., Liu, B., Zhang, Q.: Safety helmet detection with improved YOLOv5s algorithm. *Comput. Eng. Appl.* **59**(02), 194–201 (2023)
- Zhao, R., Liu, H., Liu, P.L., et al.: Safety helmet detection algorithm based on improved YOLOv5s. *J. Beijing Univ. Aeronaut. Astronaut.* **49**(8), 2050–2061 (2023)
- Han, G.: Method based on the cross-layer attention mechanism and multiscale perception for safety helmet-wearing detection. *Comput. Electr. Eng.* (2021). <https://doi.org/10.1016/j.compeleceng.2021.107458>
- Zhang, Q., Jiang, Z., Lu, Q., Han, J., Zeng, Z., Gao, S., Men, A.: Split to be slim: an overlooked redundancy in vanilla convolution. arXiv:2006.12085 (2020)
- Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-excitation networks. arXiv:1709.01507 (2019)
- Woo, S., Park, J., Lee, J.Y., Kweon, I.S.: CBAM: convolutional block attention module. In: *European Conference on Computer Vision*, pp. 3–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01234-2_1
- Hou, Q., Zhou, D., Feng, J.: Coordinate attention for efficient mobile network design. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13708–13717. IEEE (2021). <https://doi.org/10.1109/CVPR46437.2021.01350>
- Everingham, M., Eslami, S.M.A., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vis.* **111**(1), 98–136 (2015). <https://doi.org/10.1007/s11263-014-0733-5>
- Yu, J., Jiang, Y., Wang, Z., Cao, Z., Huang, T.: UnitBox: an advanced object detection network. *Proceedings of the 24th ACM International Conference on Multimedia*, pp. 516–520. (2016). <https://doi.org/10.1145/2964284.2967274>
- Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: YOLOX: exceeding YOLO series in 2021. arXiv:2107.08430 (2021)
- Zhao, L., Tohti, T., Hamdulla, A.: BDC-YOLOv5: a helmet detection model employs improved YOLOv5. *SIVIP* **17**, 4435–4445 (2023). <https://doi.org/10.1007/s11760-023-02677-x>
- Cao, K.-Y., Cui, X., Piao, J.-C.: Smaller target detection algorithms based on YOLOv5 in safety helmet wearing detection. In: *2022 4th International Conference on Robotics and Computer Vision (ICRCV)*, pp. 154–158. Wuhan, China (2022). <https://doi.org/10.1109/ICRCV55858.2022.9953233>
- Hou, G., Chen, Q., Yang, Z., Zhang, Y., Zhang, D., Li, H.: Safety helmet detection algorithm based on improved YOLOv5. *J. Eng. Sci.* **49**, 2050–2061 (2023)
- Qi, Z., Xu, Y.: Safety helmet wearing detection research based on improved YOLOv5s algorithm. *Comput. Eng. Appl.* **14**, 176–183 (2023)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the

author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.