scientific reports

Check for updates

OPEN

Safety helmet detection methods in heavy machinery factory

Liu Baoju[1], Wei Xiangqian[1,2], Chen Qingshan[1,2], Liu Jiaqi[1,2], Chen Ye[1,2], Yu Peng[3], Lei shi[4] & Hu Yongfeng[5]✉

In heavy machinery factories, accurately detecting whether workers correctly wear safety helmets is important to their well-being. Since manual inspection and video surveillance are prone to misjudgment and omission, designing a fast and intelligent algorithm essential for modern factory safety management. The YOLO series, a popular object location and detection method, offers an excellent balance between detection speed and accuracy, drawing wide attention from industry scholars. In light of this, this paper presents an improved model based on YOLOv10 to achieve safety helmet identification. Firstly, it replaces Conv convolution with distributed shift DSConv convolution in YOLOv10. This boosts memory efficiency in the convolutional layer and ensures small object identification accuracy. Secondly, the Dysample module is incorporated to cut computational load, enhance sampling, and improve model generalizability. Additionally, the WIoU loss function is introduced to accelerate convergence and increase adaptability. When compared with mainstream object recognition algorithms such as SSD, Faster RCNN, and various YOLO versions, the optimized model shows its superiority. Compared to the original YOLOv10, its average accuracy rises by 0.5%, while floating-point computation and model size decrease by 7.1% and 1.4% respectively. Finally, the optimized model is deployed on the Atlas200I DK A2 computing box to validate its usability on IoT edge devices.

Keywords YOLOv10, Helmet detection, DSConv, Dysample, WIoU

There are various safety risks in the heavy machinery factory. For instance, cranemen frequently need to hoist raw materials such as steel, semi-finished products, and other items within the workplace. Given that sling chains or spreaders are prone to cracking, there exists a significant safety hazard when workers are walking beneath the construction area. During ascend operations, construction workers face the risk of falling from heights. In limited space scenarios, operators may sustain collision injuries due to spatial constraints. With the continuous expansion of the production scale of modern smart factories, the workplace area and the number of factory branches have witnessed a rapid increase, accompanied by a sharp rise in the number of workers. Ensuring worker safety in the workplace is of utmost importance to enterprises. Thus, workers must adhere to safety regulations during actual operations, including wearing safety helmets during working hours. It has been proven that wearing safety helmets can effectively prevent most accidental risks, such as falls from heights, collisions, electric shocks, and other head-related injuries. However, in actual operations, some accidents still occur due to workers ignoring the rules and not wearing safety helmets, or for other reasons. Therefore, monitoring whether workers are wearing safety helmets in the workplace is crucial. Nevertheless, manual inspection of safety helmet wearing is challenging to achieve full coverage of the workplace due to worker mobility and the large distances between different branch factories. As a result, traditional manual inspection is prone to misjudgments and missed detection. Video surveillance, on the other hand, can cause fatigue when monitoring objects for extended periods in complex backgrounds. Moreover, it has drawbacks such as high labor costs, subjective monitoring, and slow response times. It cannot also effectively predict and judge various potential safety risks, making it difficult to meet the requirements for real-time detection in modern factory settings.

In contrast, deep learning algorithms, particularly the YOLO (You Only Look Once) series, are representative methods for achieving real-time image identification and positioning. These algorithms can predict the classification and locations of multiple objects in an image through a single forward propagation, significantly accelerating the detection speed and accuracy[1]. They have been widely applied in the fields of autonomous

[1]School of Information and Engineering, Pingdingshan University, Pingdingshan, China. [2]International Joint Laboratory of Machine Vision and Intelligent Systems of Henan Province, Pingdingshan 467000, China. [3]School of Computer Science, Beijing University of Posts and Telecommunications, Beijing 100876, China. [4]Carlow Institute of Technology, Kilkenny Road, Carlow, Ireland. [5]Pingdingshan Coal Mine Machinery Co., Ltd, Pingdingshan 467000, China. ✉email: 3212@pdsu.edu.cn

driving, security monitoring, industrial quality inspection[2], and medical services[3],. Currently, previous YOLO series still struggle to accurately locate and identify safety helmets in complex environments with small targets and monochromatic features. This is because, during safety helmet detection, they encounter interfering objects like overlapping or shielding. YOLOv10 is the latest real-time and end-to-end object detection technique. Its overall network architecture is similar to that of YOLOv8 but with some improvements in model details. Notably, deep learning technology enables the learning and extraction of complex visual features, ensuring the robust implementation of helmet image detection tasks even in resource-constrained scenarios. However, YOLOv10 still has limitations in detecting small objects, and its efficiency in recognizing safety helmets can be further enhanced. In light of this, exploring more intelligent algorithms for detecting individuals wearing safety helmets is highly significant for the management of modern heavy machinery factories.

To address the above mentioned issues, we propose an improved image detection algorithm named IYA, which is based on YOLOv10. The goal is to enhance the image detection ability in complex scenarios and improve the precision and accuracy of small object detection on a lightweight model. Generally, improvements are made in the convolution transformation, sampling model, and loss function on the convolution layer. The dataset is sourced from the publicly available Safety Helmet Wearing Detection (SHWD). The simulation results verify the effectiveness and practicality of the proposed algorithm. The main innovations are as follows:

(1) This paper presents the YOLOv10-IYA model. This model integrates three key improvements: the adoption of distributed shift DSConv convolution instead of Conv convolution, the utilization of the Dysample module, and the WIoU loss function. These enhancements significantly promote the detection performance and computational cost in complex backgrounds with small targets.
(2) The proposed model is verified on the SHWD dataset compared to various other methods. Extensive experimental results demonstrate that it outperforms other algorithms in small object detection.
(3) IYA is deployed on the Atlas200I DK A2 edge-computing box to further validate its effectiveness. The algorithm also exhibits higher accuracy and precision in real-world scenarios.

Related works and background

Related works

The YOLO series[4] models, R-CNN[5] (Regions with Convolutional Neural Network), and SSD[6] (Single Shot MultiBox Detector)based on single-stage detectors are typical object detection algorithms. The YOLO series models are one of the research hotspots of industry and academia due to the simpler frame and deployment, faster architecture, better training technique, better model scalability, and generalizability. It is widely used in production services, security management, traffic management, intelligent retail, agricultural monitoring, environmental research, map production[7–10], etc. There is some research based on deep learning to conduct image detection, Chen et al.[11] proposed a lightweight YOLOv4 model for helmet fit detection in construction sites, which solves the problems of redundant parameters, slow detection speed and insufficient target localization accuracy in complex scenes in the original model, enhances the generalization ability, and improves the detection accuracy and detection speed. Wang et al.[12] designed a lightweight and improved YOLOv5s model for detecting safety helmets in underground mines, which solves the problems of high leakage detection rate of safety helmets, difficulty in recognizing small targets and model redundancy in complex scenes (low-light, dust interference, and target occlusion) in mines, and improves the average accuracy and detection speed. The number of parameters is reduced.Song et al.[13] proposed a road small target detection method based on the improved YOLO v5 algorithm for small target detection vehicle and pedestrian detection on the road, which solves the problem of difficult detection of small targets on the traffic road, low accuracy, and prone to misdetection and leakage detection. It improves the model's learning ability for small targets, feature expression ability, bounding box localization accuracy, small target detection accuracy and average recognition accuracy. Lu et al.[14] proposed a dynamic feature point elimination method combining YOLO detection and geometric constraints to solve the problem of attitude estimation bias caused by dynamic objects, and to improve the localization and map building accuracy in dynamic scenes, in order to address the problem of SLAM accuracy being interfered by feature points in dynamic environments. Min et al.[15] proposed a machine tool image recognition algorithm based on improved YOLO v5 for machine tool image recognition, which solves the problems of fewer applications of machine tool classification, complex preprocessing, small scope of target detection and low recognition accuracy in the existing methods, and improves the accuracy and speed of machine tool detection. Wang et al.[16] proposed a lightweight remote sensing vehicle detection algorithm, AMEA-YOLO, based on the attention mechanism and efficient architecture for high-resolution remote sensing vehicle detection, which solves the problems of high computational complexity and degradation of model performance in traditional lightweight networks, enhances real-time performance, improves the resolution of vehicle images and mAP value, enhanced the processing ability of high-resolution images, greatly reduced the number of parameters, and effectively balanced the model lightweight and detection accuracy. Chen et al.[17] designed a face detector based on the YOLOv3 framework for real-time face detection to improve the performance of face detection. Their innovative approach not only dramatically improves the accuracy of face detection, but also ensures that a relatively fast detection speed is maintained. This result is significant in real-world applications such as surveillance systems and biometric authentication, where both high accuracy and fast response time are critical. Han et al.[18] proposed a helmet detection model based on super-resolution reconstruction-driven YOLOv5 for helmet detection in construction sites, which solves the problem of resolution degradation due to compression during the transmission of images from construction sites and improves the detection accuracy and speed. In addition, Chen et al.[19] proposed an improved convolutional neural network model YOLOv7-WFD for helmet detection in high-risk workplaces. It solves the problem of insufficient feature extraction ability of the traditional model, improves the model's ability to learn target features, enhances the model's ability to reconstruct details and structural information during

image up-sampling, and improves the model's generalization ability and detection accuracy.To make it more clearly, we listed the main innovation of the above works as shown in Table 1.

YOLOv10

YOLOv10 builds upon and advances the features of the overall network structure found in previous YOLO versions. Through a series of innovative modules, it achieves a significantly faster detection speed and enhanced detection accuracy. These innovative modules can be elaborated as follows[20]: (1) The Spatial-channel decoupled downsampling (SCDown) module replaces traditional standard convolution with a combination of point convolution and depth convolution. This operation not only effectively reduces the computational burden and the quantities of parameters but also boosts the efficiency of feature extraction. By pre-adjusting the channel dimensions before performing spatial downsampling, the SCDown module can better preserve spatial information. As a result, it reduces latency and enhances the overall competitiveness of the model. This unique approach allows for more efficient processing of input data, enabling the model to quickly and accurately capture relevant features while minimizing resource consumption. (2) The C2fUIB module analyzes the redundancy at each stage of the network through intrinsic rank analysis. It then adopts a rank-based block allocation strategy and an inverted block structure. This innovative approach greatly optimizes the redundancy and complexity within the network. In contrast to traditional block designs, the implementation of C2fUIB ensures more efficient model operation without sacrificing performance. By intelligently managing the allocation of network resources, the C2fUIB module allows the model to operate more smoothly and effectively, making the most of the available computational power.(3) One of the key innovations in YOLOv10 is the removal of the non-maximum suppression step, which is typically time-consuming and complex. This is accomplished by optimizing the detection strategy and enhancing the accuracy of the detection frame. By re-engineering the way the model makes detection decisions, YOLOv10 can streamline the detection process, eliminating a potential bottleneck in real-time applications. This not only speeds up the overall detection process but also simplifies the underlying algorithms, making the model more accessible and easier to implement in various scenarios.

Ref.	Datasets	Method used	Results	Originality	Limitations
[11]	SHWD	Improved Yolov4	Accuracy:92.98%, model size :41.88 M, Detection speed: 43 frame/second	Introducing depth-separable convolution to reduce the model parameters; embedding the coordinate attention mechanism module to enhance the feature information; Designing the PB module to fuse the target information; and using the SIoU loss function to replace the CIoU loss function to improve the accuracy and speed of helmet detection, and reduce the size of the model.	Has the situation of missed detection of small targets at long range.
[12]	CUMT-HelmeT	Improved YOLOv5s	Average Precision:87.5%	Fusing the attention mechanism CBAM and YOLOv5s to improve the accuracy; designing the P2 small target detection layer to increase the multi-scale sensing field of the model; replacing the CIoU loss function with the EIoU loss to ensure the accuracy of the regression frame; replacing the ordinary convolutional Conv in the backbone network with the ShuffleNetV2 to realize the lightweight network model.	The situations of missed and misdirected tests still exist
[13]	KITTI	Improved YOLO v5	Average recognition accuracy:95.2%	A 160 × 160 small target detection head is added to improve shallow feature retention, a deformable convolutional network V2 (DCN V2) is introduced to improve the learning ability of small moving targets, a context augmentation module (CAM) is added to improve the detection of small targets at long distances, the loss function is replaced by EIoU to improve the accuracy of bounding box localization, and the SPPCSPC_group module is adopted to improve multi-scale feature fusion.	More hardware processing power and memory are required
[14]	TUM	Improved ORB-SLAM with YOLO	Target detection accuracy :99.3%	Combining the optical flow method with geometrical constraints for secondary judgment, and static feature points are utilized for position estimation; target tracking algorithm is used for inter-frame detection correction for the image blurring problem.	Weak robustness and insufficient computational resources in highly dynamic and complex scenarios.
[15]	Proprietary dataset	Improved YOLO v5	Precision:98.88% recall:94.82% mAP : 98.13%	CBAM attention module is added to the convolutional neural network feature extraction layer to enhance important features and suppress useless features; CARAFE is added to the feature fusion layer to be able to dynamically generate adaptable kernels. Improved precision and accuracy of machine tool recognition.	Weak complex environment adaptability and lower computational resource efficiency
[16]	VisDrone, VEDAI	AMEA-YOLO	mAP:43.4% Parameters size :10.4 M GFLOPs:23.7 mAP :66.2%	The lightweight network Ghostone is designed as the backbone network and combined with FasterNet to accelerate the model training; the enhanced second-order channel attention module EnhancedSOCA is utilized to improve the high resolution of the image; the GC3 module is designed by introducing the SimAM attention mechanism to further lightweight the model; and the HardSwish activation function is used	Weak generalization capacity
[18]	Proprietary dataset	Improved YOLOv5	PSNR: 29.420 SSIM:0.855 The average accuracy AP:79.1%	A double residual channel Super-Resolution (SR) reconstruction module was designed to improve the image resolution. A new CSP module of YOLOv5 is proposed to reduce information loss and gradient confusion. An end-to-end safety helmet detection model based on SR reconstruction network and YOLOv5 is constructed.	Weak generality and real-time performance
[19]	SHEL5K	YOLOv7-WFD	mAP:92.6% FPS:79%	The DBS composed of deformable convolution, batch normalization layer and SiLU activation function to enhance the feature extraction ability of the model was proposed. The CARAFE module was introduced for feature upsampling to improve the reconstruction ability of model details and structural information. The Wise-IoU loss function is used to calculate the localization loss to enhance the generalization ability.	Weak computational efficiency, hardware compatibility, furhter optimize dataset balance, and expand range of application scenarios

Table 1. Summary of different methods derived from up-to date literature.

(4) This enhancement concerning improved detection frame rate not only simplifies the post-processing procedures of the model but also significantly improves the performance of real-time object detection. The innovative structures of the PSA (presumably another key component, though not elaborated in the original text) and SCDown further reduce the computational load and the number of parameters. They achieve this by optimizing the downsampling process while maximizing the retention of information during downsampling. This dual-optimization approach ensures that the model can operate at a high frame rate, making it suitable for applications where real-time response is crucial, such as video surveillance and autonomous driving. The network structure of YOLOv10 is illustrated in Fig. 1.

Methods

Detection model

An improved object recognition method based on the YOLOv10 algorithm is designed in this section to enhance accuracy and precision. Firstly, the Distributed Shift Convolution (DSConv) is used to promote memory efficiency and detection accuracy in the convolution layer. Dysample is then leveraged to improve the efficiency and accuracy of the model and reduce the computational load. Finally, the WIoU loss function serves to improve the ability of small object detection, meanwhile accelerating the convergence speed and enhancing its adaptability. The optimized YOLOv10 network structure is provided in Fig. 2, and the enhancements have been marked in red dotted lines.

DSConv depth separable convolution

The Standard convolution has the advantage of parallel computing on high-performance GPUs and can simultaneously deal with spatial and channel information. However, it is weak in the context of limited computational load and memory, such as mobile devices and embedded systems. In contrast, the traditional convolutional kernel is divided into two modules: the Variable Kernel (VQK) and the Distribution Offset by DSConv. It enables to achieve higher speed and lower usage of memory by only storing integer values in the VQK. At the same time, the operations of the Distribution Offset based on both kernel and channels guarantee the same output as the original convolution. Furthermore, it splits the standard convolution into deep convolution and point-by-point convolution. The deep convolution only performs convolution operation on a single channel and adopts a distinct convolution kernel to each channel. The point-by-point convolution exploits a convolution kernel on all channels to combine the results of deep convolution. The advantage of DSConv is that it leverages a learnable convolution kernel to further improve the model performance compared to deep separable convolution. The overall goal of DSConv is to simulate the behavior of convolutional layers by using the methods of quantization and distribution shifts, and hence it is more memory efficient than the traditional convolution. The operations of DSConv are shown in Fig. 3.

Here, the original convolution tensor is denoted as (ch_o, ch_i, k, k), where ch_o and ch_i are the number of channels in the next layer and the current layer, respectively. k is the width and height of the kernel, and BLK is the size

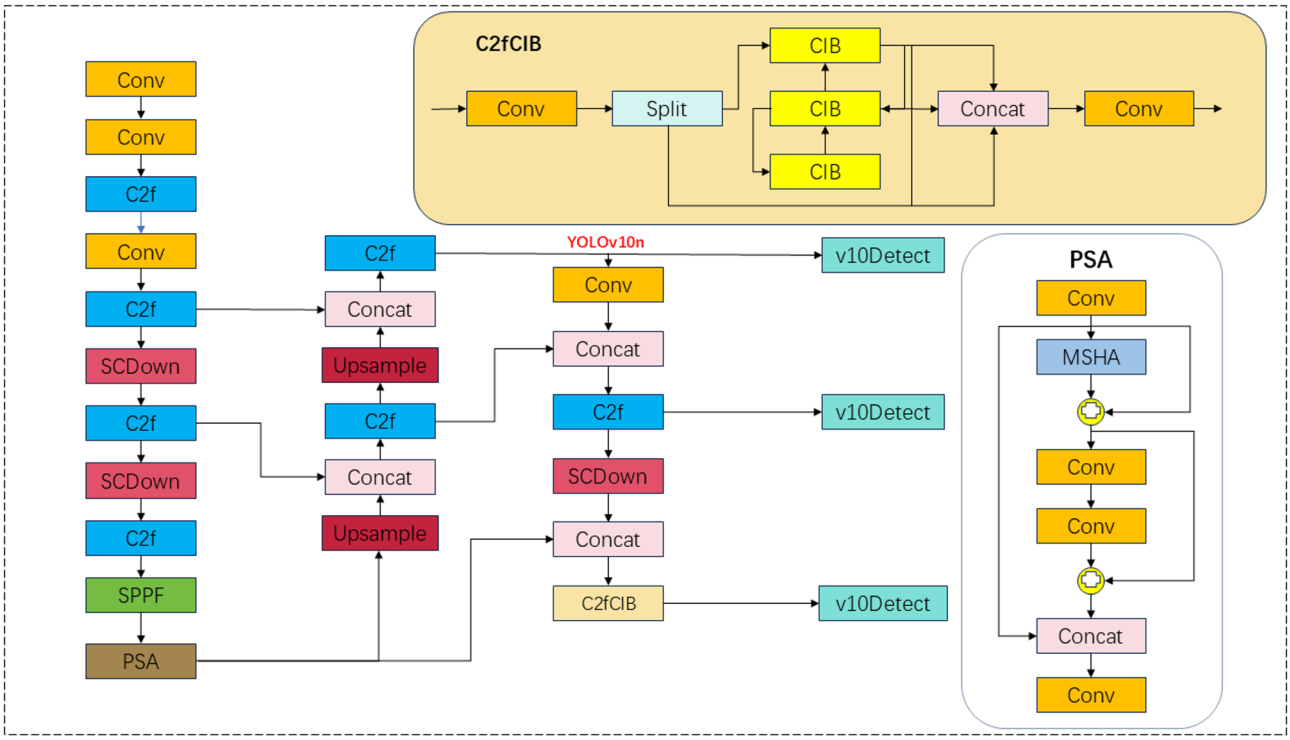

Fig. 1. YOLOv10 Network Architecture.

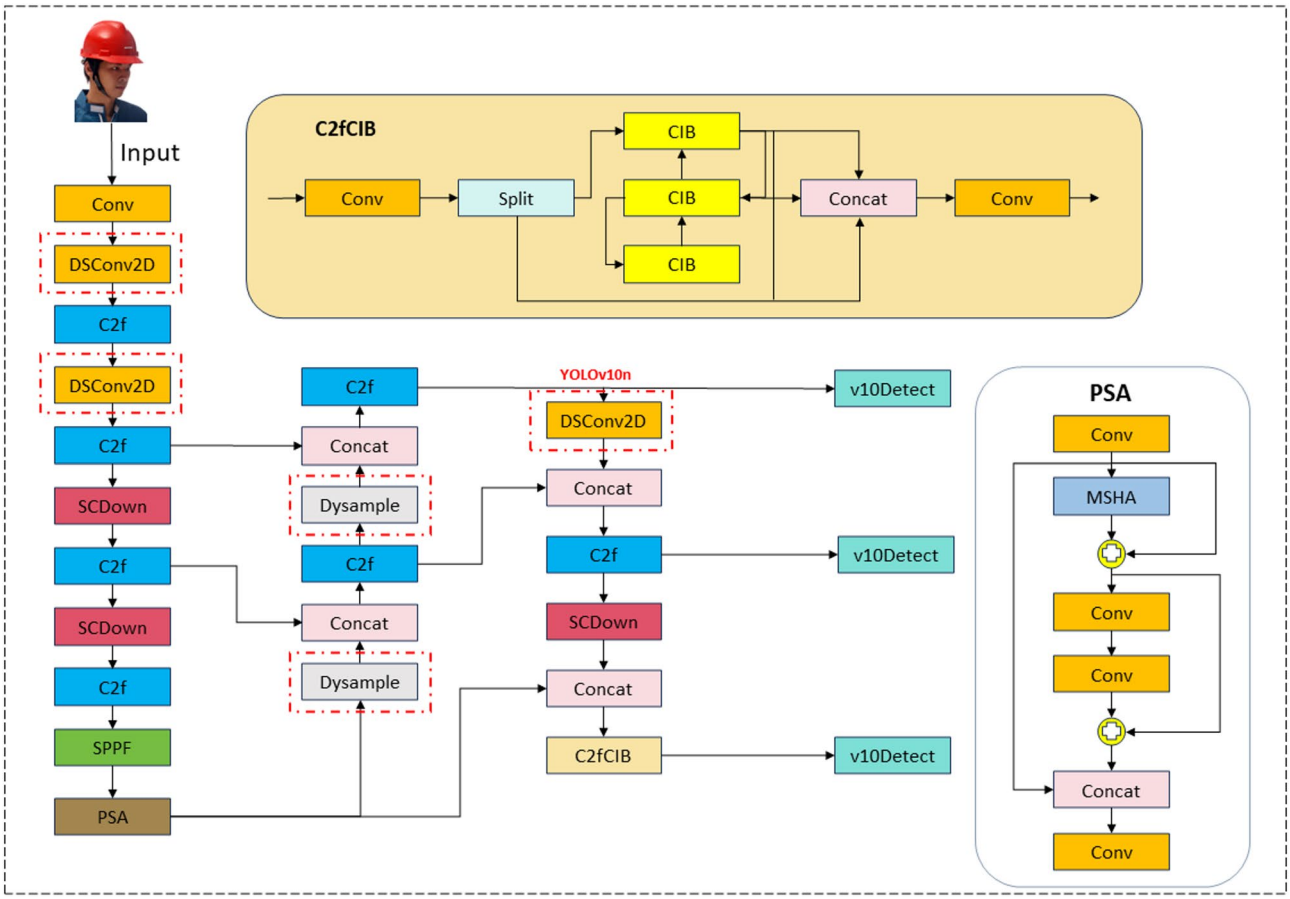

Fig. 2. Improved YOLOv10 network architecture.

of the given block, The role of the *CELL( )* function is to compute the distributional offset tensor based on the distribution of floating point weights in the pre-trained network. The Eqs. 1–2 computes the memory saved ratio of the generalized convolution to DSConv, respectively:

$$p = \frac{ch_0 \cdot k^2 \left(bit + 2 \cdot 32 \cdot CELL\left(\frac{ch_i}{BLK}\right)\right) + 2 \cdot 32 \cdot ch_0}{32 \cdot (ch_0 \cdot ch_i \cdot k^2)} \tag{1}$$

$$p = \frac{bit}{32} + \frac{2 \cdot CELL(\frac{ch_i}{BLK})}{ch_i} + \frac{2}{ch_i \cdot k^2} \tag{2}$$

The above result can be approximately calculated in Eq. (3) when the deviation in the shifted value is negligible:

$$p \approx \frac{bit}{32} \tag{3}$$

Dynamic upsampler

The UpSample modules are mainly used to perform up-sampling operations on input data. It executes up-sampling data processing with multiple channels based on one-dimensional time series, two-dimensional spatial images, and three-dimensional volumetric data. The critical role of this module is to achieve data enlargement and increase the data resolution while maintaining or transforming the features.

The Dysample is an innovative resource-efficient dynamic upsampler. Kernel-based dynamic upsamplers offer large performance gains. However, they impose a large workload mainly due to time-consuming dynamic convolution and additional subnetworks used to generate dynamic kernels. The Dysample employs the way of point sampling[20] to avoid the high computational load and resource consumption required by traditional kernel-based dynamic upsamplers. Therefore, it is superior to other approaches in multiple intensive prediction tasks. Meanwhile, as an advanced dynamic upsampler, the Dysample can be effortlessly implemented by leveraging the standard built-in functions in PyTorch. Compared to kernel-based dynamic upsamplers, this resource-efficient sampling does not require customized CUDA packages and has fewer parameters, FLOPs, GPU memory, and latency.

Fig. 3. DSConv Operation.

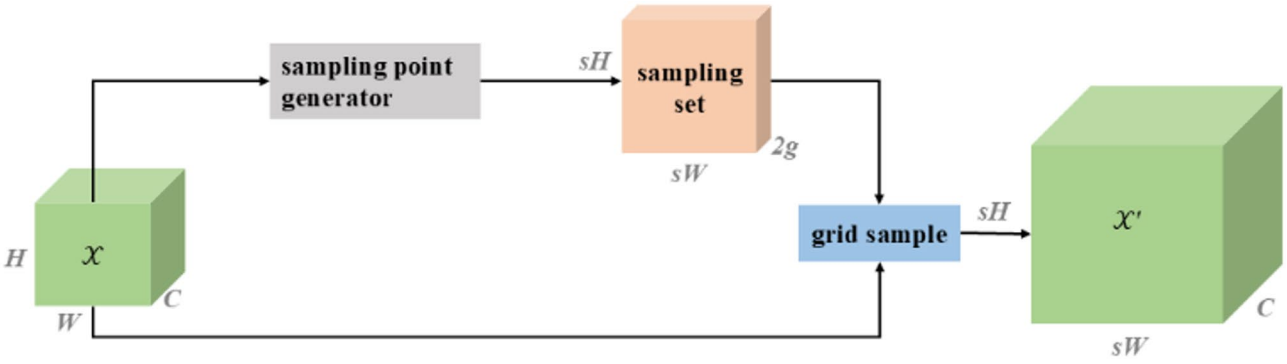

Fig. 4. Schematic diagram of Dysample structure.

The schematic diagram of the Dysample structure is given in Fig. 4. To the feature map X with the size of $C\times H_1\times W_1$ and the sampling set S of size $2\times H_2\times W_2$, where the first digit 2 denotes the coordinates of x and y. The *grid_sample* function resamples X, which is a hypothesized bilinearly interpolated X into X' of size $C\times H_2\times W_2$ using the positions in S. The process is formulated in Eq. (4):

$$X' = grid\_sample(X, S) \tag{4}$$

Based on an upsampling scale factor of s, and a feature map X size of $C\times H\times W$, an offset O of size $2s\times H\times W$ is generated using a linear layer, and the input and output channel numbers are C and $2s^2$, respectively. It can be reshaped into a size of $2\times sH\times sW$ by pixel transformation. The sampling set S is the sum of the offset O and the original sampling grid g, and this process can be denoted as follows.

$$o = linear(X) \tag{5}$$

$$S = g + O \tag{6}$$

where the shaping operation is omitted. Hence, the upsampled feature maps X with the size of $C\times sH\times sW$ could be generated.

According to the above steps, Dysample achieves dynamic up-sampling, in which the sampling points and the adjustment of their positions are dynamically determined in the light of the content of input feature maps. Consequently, it could ensure the efficiency and effectiveness of the up-sampling process.

Loss function optimization

The original YOLOv10 model uses CIoU as the loss function, and its formulas are presented in Eqs. 7–10:

$$L_{CIoU} = L_{IoU} + \frac{(x - x_{gt}^2) + (y - y_{gt}^2)}{(W_g^2 + H_g^2)} + \alpha\nu \tag{7}$$

$$\alpha = \frac{\nu}{L_{IoU} + \nu} \tag{8}$$

$$\nu = \frac{4}{\pi^2}(arctan\frac{w}{h} - arctan\frac{w_{gt}}{h_{gt}})^2 \tag{9}$$

$$L_{IoU} = \frac{W_i H_i}{wh + w_{gt}h_{gt} - W_i H_i} \tag{10}$$

where α is a weight function for balancing the parameters, and ν is the aspect ratio function for measuring the ratio consistency. The expression of IoU is shown in Fig. 5, where w, h, (x, y) denote the width and height dimensions of the predicted frame and the center coordinates, respectively, w_{gt}, h_{gt}, (x_{gt}, y_{gt}) describe the width and height dimensions of the real frame and the center coordinates. W_i, H_i describes the width and height of the intersection, respectively. W_g and H_g present the minimal border size of the width and the height, respectively. The CIoU loss function has some disadvantages, for example, slower convergence, lack of dynamic adjustment mechanism, and insensitive to small object detection, although it considers the factors of overlap area, center distance, and aspect ratio.

Therefore, the WIoU based on a dynamic nonmonotonic focusing mechanism is used to take the place of CIoU, which is computed in formulas (11)-(13)

$$L_{WIoU} = rR_{WIoU}L_{IoU} \tag{11}$$

$$r = \frac{\beta}{\delta\alpha^{\beta-\delta}} \tag{12}$$

$$R_{WIoU} = \exp(\frac{(x - x_{gt}) + (y - y_{gt})}{(W_g^2 + H_g^2)}) \tag{13}$$

where R_{WIoU} amplifies the L_{IoU} of ordinary-quality anchor frames, r is a nonmonotonic focusing coefficient used to focus on ordinary-quality anchor frames, α and δ are hyperparameters, respectively. r can dynamically enhance bounding-box gradient gain by decreasing the contribution of high-quality samples to the loss value, and thus reduce the harmful gradients generated by low-quality anchor frames during the training process. It focuses on ordinary-quality anchor frames to improve the model localization ability.

There is a weighting mechanism in the WIoU loss, which pays more attention to small object detection. The dynamic adjustment mechanism guides the model to focus on difficult samples or samples with large errors to promote the convergence of the model. On this basis, the WIoU loss can better adapt to objects with different sizes and shapes. In the workplace, the safety helmets may be blocked by some barriers or non-standard wearing by workers. Due to the complex environment of the workplace, we employ the WIoU loss function to replace the CIoU in the original model of YOLOv10 to accelerate the convergence speed and enhance the adaptability.

Model deployment

Edge computing has the advantage of quick system response and data processing, and it can provide services such as computing, storage, and network bandwidth near terminal users. It can better adapt to the requirements of modern factory management, such as real-time, accuracy and efficiency. Therefore, it is practical to carry out intelligent research on safety helmet detection under the edge computing architecture. The schematic diagram of the small object detection system is shown in Fig. 6. It is composed of the following parts: algorithm design, model training, algorithm deployment, and visualization display. To verify the effectiveness of the intelligent detection algorithm, it is deployed on the Atlas 200I DK A2 edge computing box, which is a high-performance

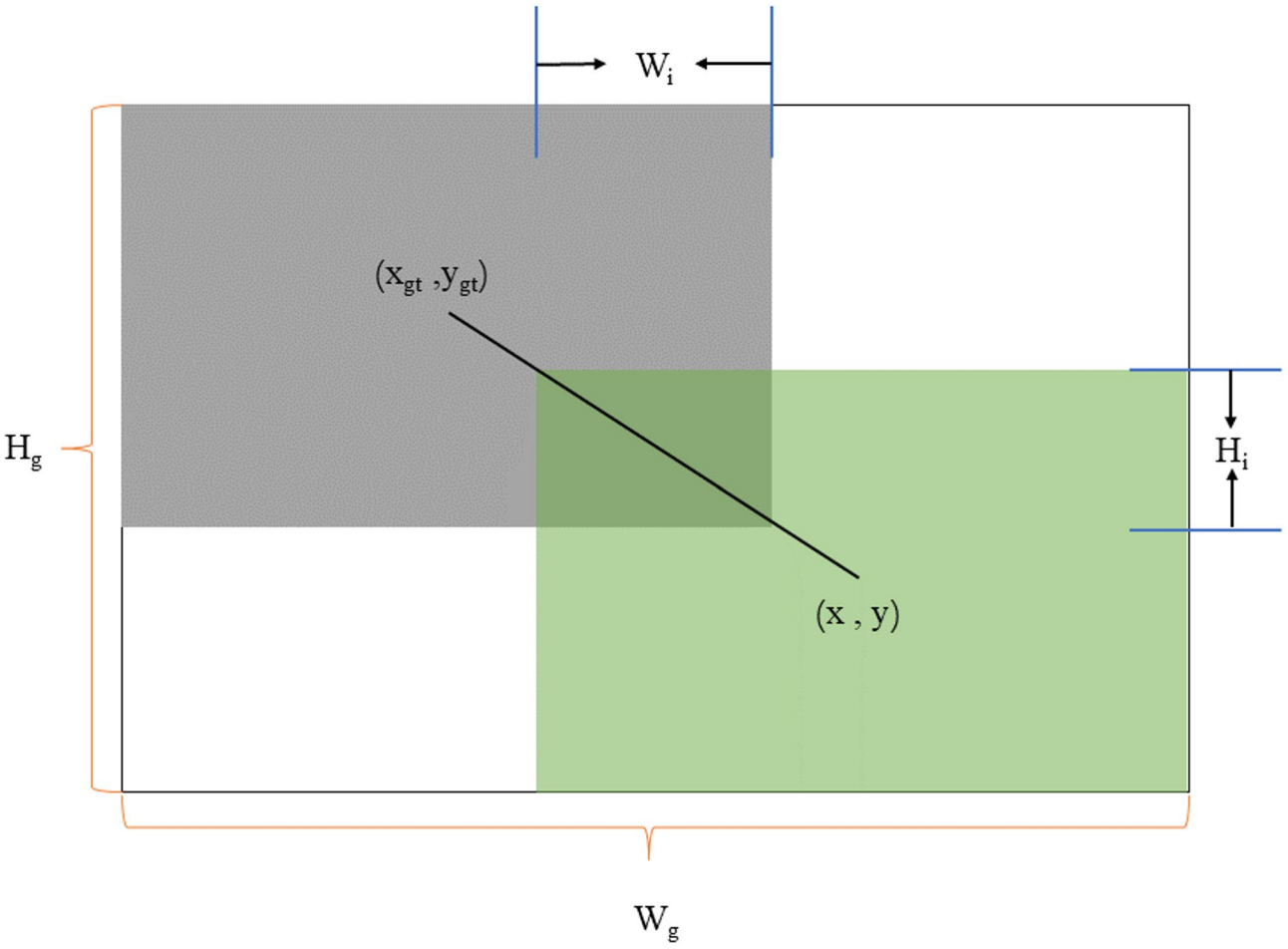

Fig. 5. Area of the intersection of real and predicted frames.

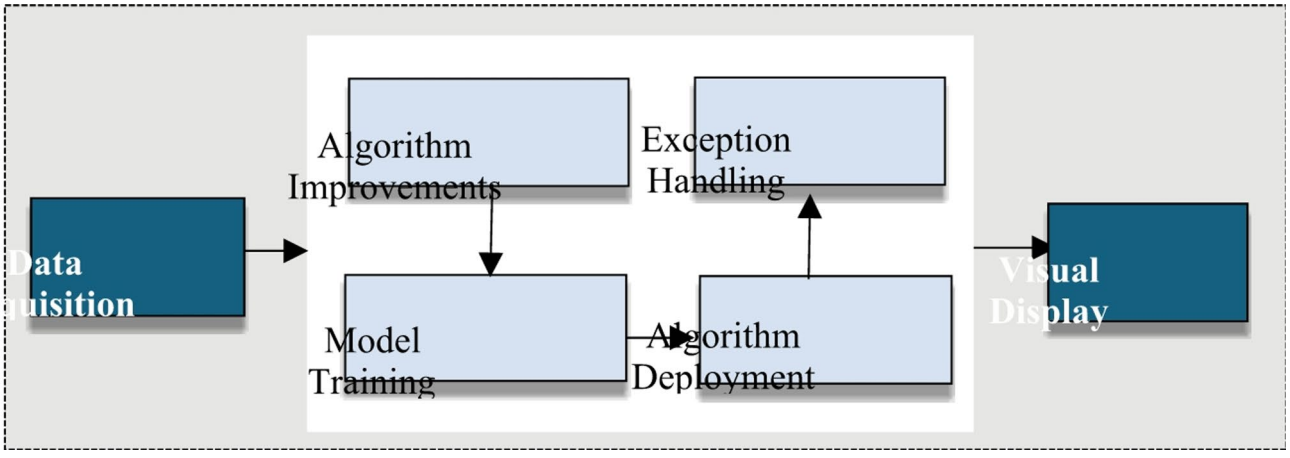

Fig. 6. System Schematic.

AI developer kit that provides 8TOPS INT8 computing power. It can realize a variety of data analyses and reasoning calculations such as images and video. The Rise 310 series of AI processors is the core component with powerful encoding and decoding ability, which guarantees the model training smoothly.

Experiments and discussion

Datasets and experimental settings

The choice of dataset is critical to the quality of the network model in deep learning research. In the experiments, we select the publicly available safety helmet-wearing detection dataset (SHWD) to evaluate the performance of the algorithm. The dataset contains 20,588 images in total, including 9,044 images wearing safety helmets and 11,514 normal head objects without wearing safety helmets. The SHWD dataset is partitioned into training set, validation set, and test set, respectively, and the ratio of them is 8:1:1.

The following metrics, Precision (*P*), Recall (*R*), Average Precision (*AP*), and the mean of Average Precision (*mAP*) are used to evaluate the performance of the improved model. Here, *P* is the proportion of correctly positive predictive detection in the sum of correct and incorrect positive predictions. *R* is the ratio of correctly positive predictive samples in the total predictions. The above two metrics are used to measure the accuracy of model detection. *AP* is the average of some classes, which depicts the area between the *P*-*R* curve and *x* axes. *mAP* describes the mean of *AP*s for all classes. As mentioned above, IOU presents the proximity of predictive and real bounding boxes. In the training process, different IOU thresholds are set to measure the accuracy of predicted results. As a confidence level, *mAP* is always computed with different IOU thresholds. For example, *mAP*@0.5 stands for detection results that are correct with *IOU* larger than 0.5. They are expressed in Eqs. (14) -(17), respectively.

$$P = \frac{TP}{TP + FP} \tag{14}$$

$$R = \frac{TP}{TP + FN} \tag{15}$$

$$AP = \int_0^1 p(r)\mathrm{d}r \tag{16}$$

$$mAP = \frac{\sum_{i=1}^{N} AP_i}{N} \tag{17}$$

Where *TP* stands for true positive which represents the number of correctly positive predictions in the positive detection. *FP* is short for false positive which depicts incorrect object detection.*FN* is short for false negative which represents the number of miss detected objects. $p(r)$ is a smooth precision-recall curve ranging from 0 to 1. *N* denotes the number of detection classes. The metrics of the Floating point Calculations (GFLOPs) and the model size aims to measure model complexity.

The experimental environment is configured as follows: the CPU is AMD Ryzen 77840Hw/ Radeon 780 M Graphics, the RAM is 16GB, and the graphics card model is NVIDIA GeForce RTX 4060 Laptop. The learning framework is Pytorch 2.2, Python 3.8, CUDA 12.1, and the operating system is Windows 11. The parameters used for model training are set in the Table 2.

The reasons for the above parameter selection are as follows. The initial learning rate is set to the classic benchmark value. In the YOLO series and most CNN target detection models, 0.01 is a typical initial learning rate of SGD optimizer, which can balance the convergence speed and stability. At the same time, due to the Anchor-Free design of YOLOv10 being more sensitive to the learning rate, setting a high learning rate, such as 0.1, will cause the bounding box regression to diverge. Conversely, using a low learning rate, like 0.001, will decelerate the process of feature fusion. The batchsize is set to 16, which is the limit of single-card training on an NVIDIA RTX4060 with 8GB memory due to GPU memory limitations. The Momentum is set to 0.937 which directly inherit from the default configuration of YOLOv5 and is validated by the training process. As to the parameter of training epoch, the model fits well when the number of training epochs reaches 100. With the increase of the number of training epochs, the value of mAP does not significantly increase and therefore the Training Epoch is selected as 100. The strategy of Cosine annealing decay can periodically enlarges the learning

Parameter name	Parameter value
Initial learning rate	0.01
Batchsize	16
Momentum	0.937
Training Epoch	100
Decay strategy	Cosine annealing decay
Final learning rate	0.0001
Weight decay	0.0005

Table 2. Experimental parameter settings.

DSConv	Dysample	WIoU	*P*/(%)	*R*/(%)	mAP/(%)	Floating point Calculations / GFLOPs	Model size/MB
×	×	×	90.6	87.7	93.4	8.4	5.61
√	×	×	91.6	87.6	93.5	7.8	5.85
×	√	×	90.8	87.7	93.6	8.4	5.64
×	×	√	96.2	97.0	93.6	8.4	5.61
√	√	√	96.3	97.0	93.9	7.8	5.53

Table 3. Results of ablation with IYA.

Algorithm	K	mAP(%)
IYA	1	87.9
	2	88.8
	3	91.2
	4	91.8
	5	88.6

Table 4. Results of k-fold cross-validation.

rate and avoid getting trapped in a local optimum compared to the strategy of linear decay.The Weight decay inherits from the Darknet framework's default Settings, and it forms a regularization complementary to SGD momentum. Finally, the formula for the Final learning rate is as follows:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{initial} - \eta_{\min})\left(1 + \cos\left(\frac{T_{current}}{T_{\max}}\pi\right)\right) \tag{18}$$

Where $\eta_{initial}$,$\eta_{\min}$,$T_{current}$,$T_{\max}$denote the initial learning rate, the minimum learning rate, the last cycle and the total training period, respectively.The values of them are 0.01,0.0001, 100 and 100,respectively.According to the above illustration, the final result is 0.0001.

Ablation experiments

To evaluate the effectiveness of the improved algorithm, ablation experiments are designed under the same training environment. The DSConv convolution, Dysample sampling mode, and WIoU loss function are mixed with the original YOLOv10 algorithm which is also the baseline. The results of the ablation experiments with IYA are represented in Table 3. Here, "√" stands for that the corresponding module is leveraged in the training process, and "×" otherwise.

It can be found the use of DSConv convolution improves the accuracy and the mAP value by 1%,, and 0.1%, respectively, and the floating-point computation is reduced by 7.1%, however. Likewise, the model size is increased by 4% compared to the baseline algorithm. In comparison with the original Upsample sampling method, the accuracy by leveraging the Dysample upsampling method and mAP increased by 0.2%, and 0.2%, respectively. The floating-point computation load is not reduced, and the model size is increased by 0.5%. The WIoU loss function makes the mAP increase by 0.2%, and the floating-point computation load and the model size remain unchanged. At the same time, with the DSConv convolution and Dysample sampling, the accuracy is increased by 1%. Furthermore, through the combined use of DSConv convolution, Dysample sampling, and WIoU replacing the corresponding modules, the precision and the mAP value are increased by 5.7%, and 0.5%, respectively. The floating-point computation load, and the model size are reduced by 7.1%, and 1.4%, respectively. Ablation experiments demonstrate that there are obvious enhancements with the simultaneous applications of the three improved modules, and validate the efficiency and effectiveness of IYA.

In order to validate the generalization ability of the model more reliably and to reduce the evaluation bias due to the different ways of data segmentation, we adopted the k-fold cross-validation method to evaluate the IYA model, and the evaluation results are shown in Table 4.

After five cross-validations, the mAP values of the IYA are all stable above 87%, which indicates that the IYA algorithm has a strong model generalization ability.

Algorithm performance comparison

To further assess the proposed algorithm, we carry out a series of experiments. The batch size and the learning rate are 16, and 0.01, respectively. A Stochastic Gradient Descent (SGD) optimizer is utilized. The model is trained on the validation set 100 times iteratively, and the results are shown in Table 5. We can find that the precision of IYA is enhanced and the metric of mAP is increased by 0.5%. Meanwhile, the floating-point computation load and the model size are reduced by 7.1%, and 1.4%, respectively. The precision of IYA is reduced by 0.1% in the scenario of wearing helmets for workers. On the contrary, the precision is improved by 0.2%. It is also noticed that IYA not only enhances the detection accuracy but also reduces GFLOPs and model size, meanwhile improving efficiency.

Algorithm	*P* (%)		mAP/(%)	Floating point calculations / GFLOPs	Model Size/MB
	With safety Helmet	Without safety helmet			
YOLOv10	89.5	91.7	93.4	8.4	5.61
IYA	89.4	91.9	93.9	7.8	5.53

Table 5. Algorithm comparison.

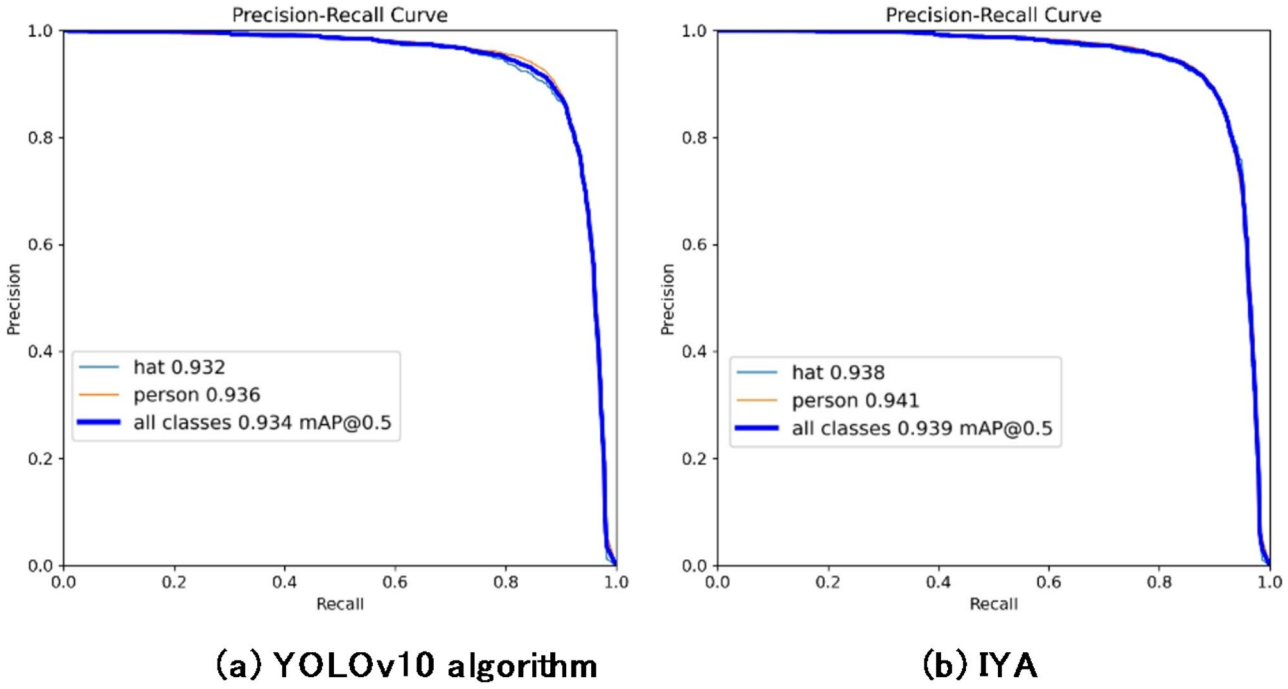

Fig. 7. P-R curve.

There are two kinds of classes in the training model: one is a "hat" signified correctly wearing a safety helmet, and the other one is a "person" represented incorrectly or without wearing a safety helmet. The comparison of P-R curves with the original YOLOv10 algorithm and the IYA on the validation set are present in Fig. 7(a) and Fig. 7(b), respectively. It describes the variation of the pair of precision and recall. The area enclosed by this curve and the X-axis denotes higher accuracy and lower incorrect precision. We can find that the area located between the P-R curve of IYA and the X-axis is significantly larger than that of the original YOLOv10. It demonstrates that the precision of IYA is superior to the YOLOv10. This visual and quantitative comparison demonstrates the enhanced performance of IYA in terms of achieving more accurate detection and reducing the rate of false positives.

Figures 8 and 9 show the results of the original YOLOv10 model and the IYA model, respectively. The two models are trained in the same training environment with a batch size of 16, a learning rate of 0.01, and 100 iterations on the validation set using the Stochastic Gradient Descent (SGD) optimizer. The comparison results demonstrate that the IYA model significantly outperforms the original YOLOv10 model on all metrics both in the training and validation phases, that is, achieving significant improvements in recall, precision, mAP50, and mAP50-95.

In order to better demonstrate the performance of the IYA algorithm, we validate it using a custom-collected factory image dataset, and the validation results are shown in Fig. 10:

As can be seen from Fig. 10, the mAP value of IYA algorithm in the factory image dataset of the customized cell phone is 91.2%, which indicates that IYA algorithm performs well.

Finally, we compare the model performance of the IYA and YOLOv10 algorithms shown in Fig. 11, including precision, recall, mAP@0.5, and mAP@0.5:0.95. Here, the yellow curve denotes the IYA variation tendency, and the blue curve represents the original YOLOv10 algorithm. We observe that the precision of IYA generally approximates to YOLOv10 algorithm shown in Fig. 11(a). It is known that the higher the precision, the greater the number of correctly detected small positive targets. Meanwhile, the higher the recall is, the fewer the missed detection. It also can be seen that the recall of IYA is larger than that of the YOLOv10 algorithm shown in Fig. 11(b). Similar to the trend of recall, the metric of mAP@0.5 with IYA on the SHWD dataset is superior to the YOLOv10 algorithm shown in Fig. 11(c). The larger mAP@0.5:0.95 is, the more accurate the prediction box. Figure 11(d) shows that the mAP@0.5 curve of IYA approximates the original YOLOv10.

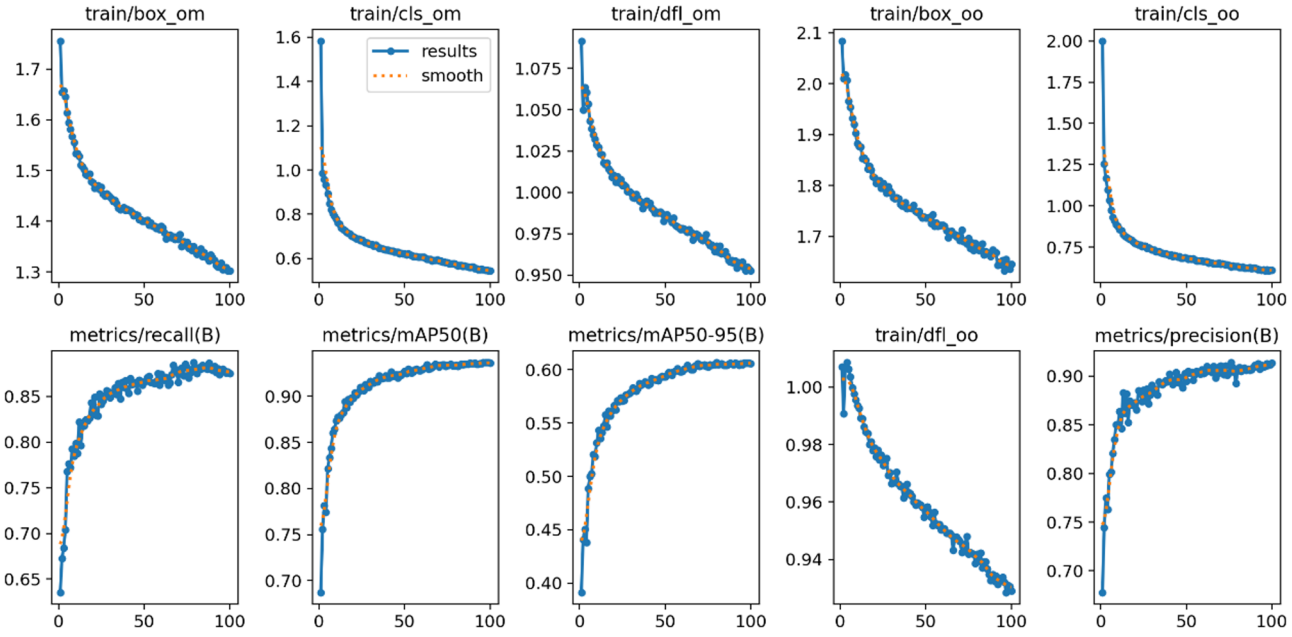

Fig. 8. The models were performed during training using the original YOLOv10.

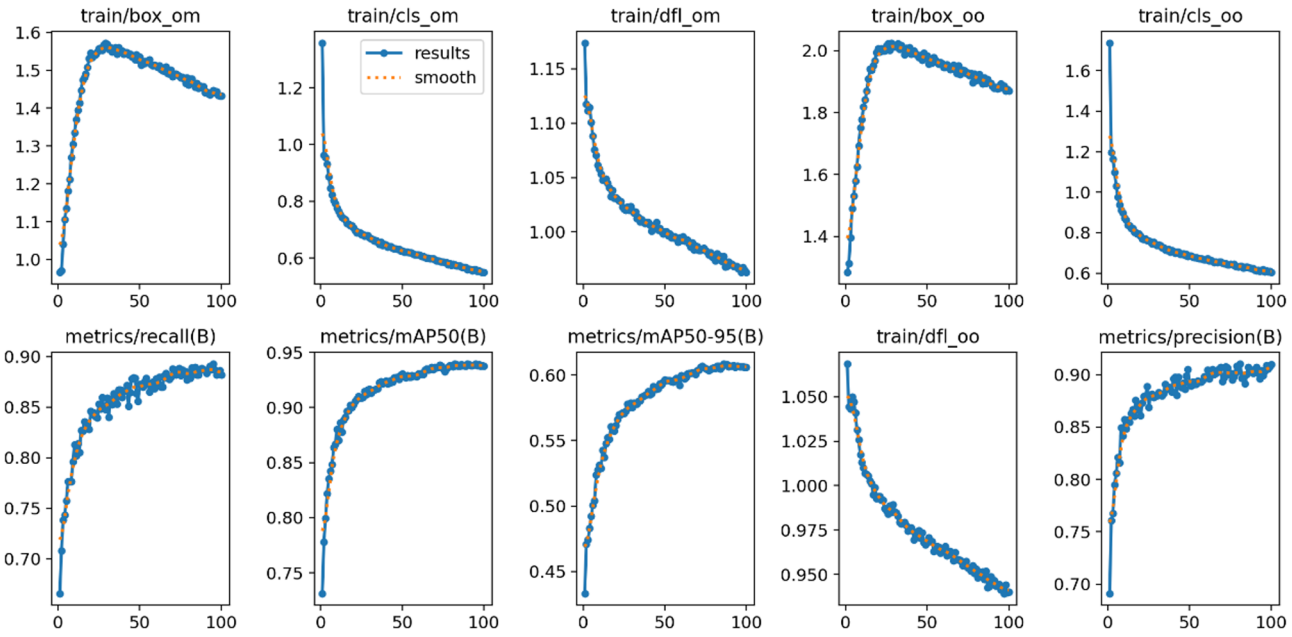

Fig. 9. The models were performed during training using the IYA.

To assess the efficacy of the optimized model, we compared it with the typical object recognition algorithm, including SSD, Faster RCNN, YOLOv3, YOLOv8, YOLOv9-tiny, YOLOv10, YOLOv10-DSConv, YOLOv10-DynamicConv[22], YOLOv10-Dysample, YOLOv10-CARAFE[23], YOLOv10-SIoU, YOLOv10-EIoU and YOLOv10-WIoU.They are trained under the same conditions and dataset partitioning methods. The relevant results are presented in Table 6. The mAP@0.5 of IYA is up to 93.9%, which is superior to other algorithms. The model size of IYA is the smallest, however. In general, IYA demonstrates better performance in contrast with other YOLOv10-based algorithms. As to the metrics of the GFLOPs and the model size, IYA is notably the smallest of all of the algorithms, which shows the effectiveness of the combined model.

Case analysis

The detection effect of the YOLOv10 algorithm and IYA on the dataset of SHWD is illustrated in Fig. 12. It is noticed that both the YOLOv10 algorithm and IYA have the phenomenon of loss detection. The latter has

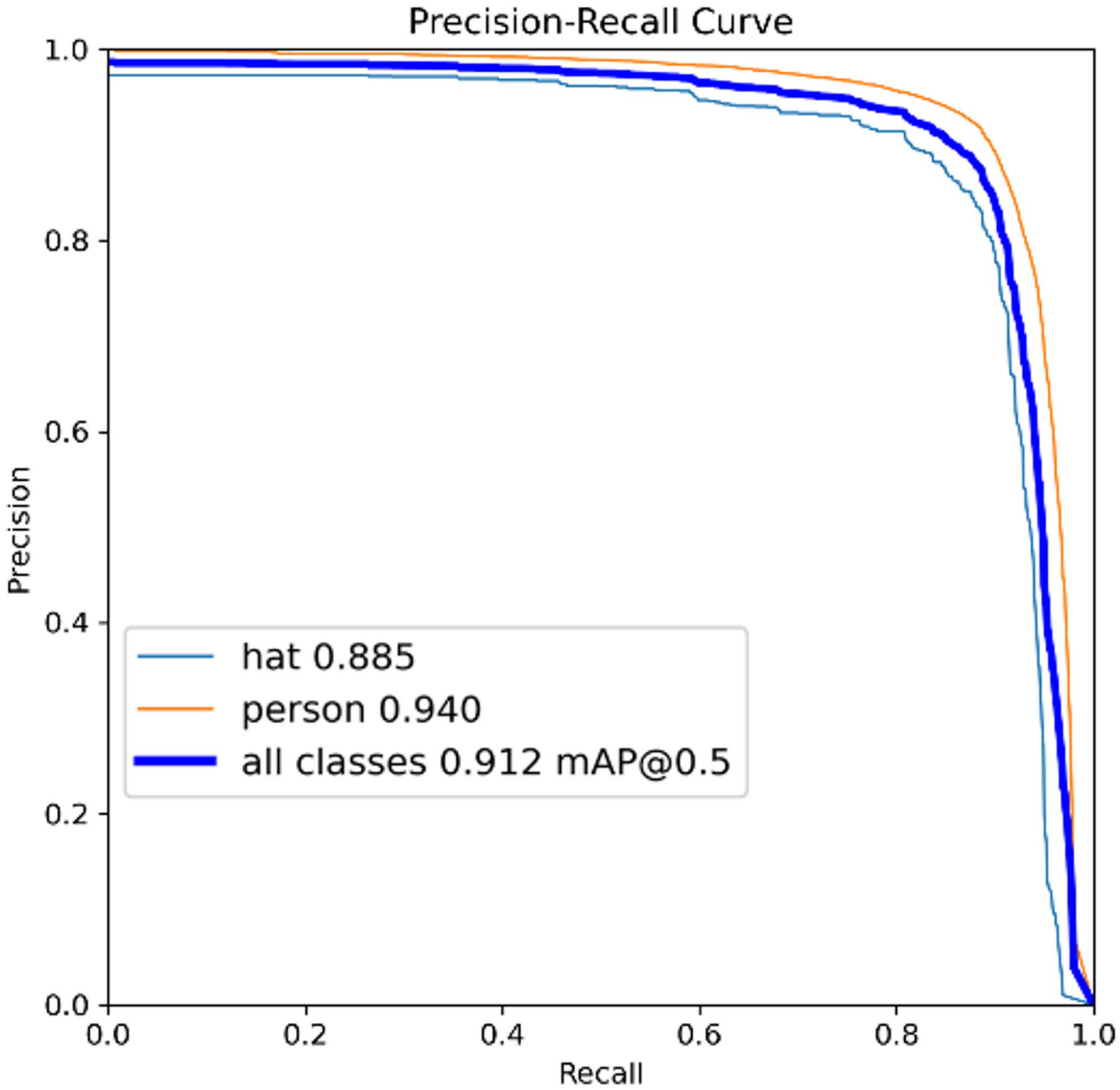

Fig. 10. Real factory dataset validation results.

a lower loss-detecting rate, however. There is missed detection with the YOLOv10 algorithm while IYA can correctly detect all images. At the same time, the small object detection rate of the IYA algorithm is higher compared with YOLOv10.

Additionally, the improve algorithm has been deployed on the edge devices to further test its effectiveness. The results demonstrated that the YOLOv10 algorithm has an FLOPs of 0.7 on Atlas 200I DK A2, and the IYA algorithm has an FPS of 0.9. The inference speedup of about 28.6%.

Conclusions

A safety helmet detection algorithm, IYA based on YOLOv10 is proposed in this research. Comprehensive experiments demonstrated that it has the advantages of higher accuracy, faster detection speed, and less computational resources. At the same time, IYA has higher accuracy, lower computation complexity of floating-point, and a smaller model size compared with SSD, Faster RCNN, and YOLOv9-tiny. Since we focus on the improvement of detection accuracy and the reduction of computational load, we expect the continuous improvement can reach the optimal performance in the field of small object detection in the future. Additionally, we will adopt the strategies of image reprocessing and image enhancement to further enhance the detection accuracy, and take a lightweight approach to enhance the model and reduce the model size without affecting

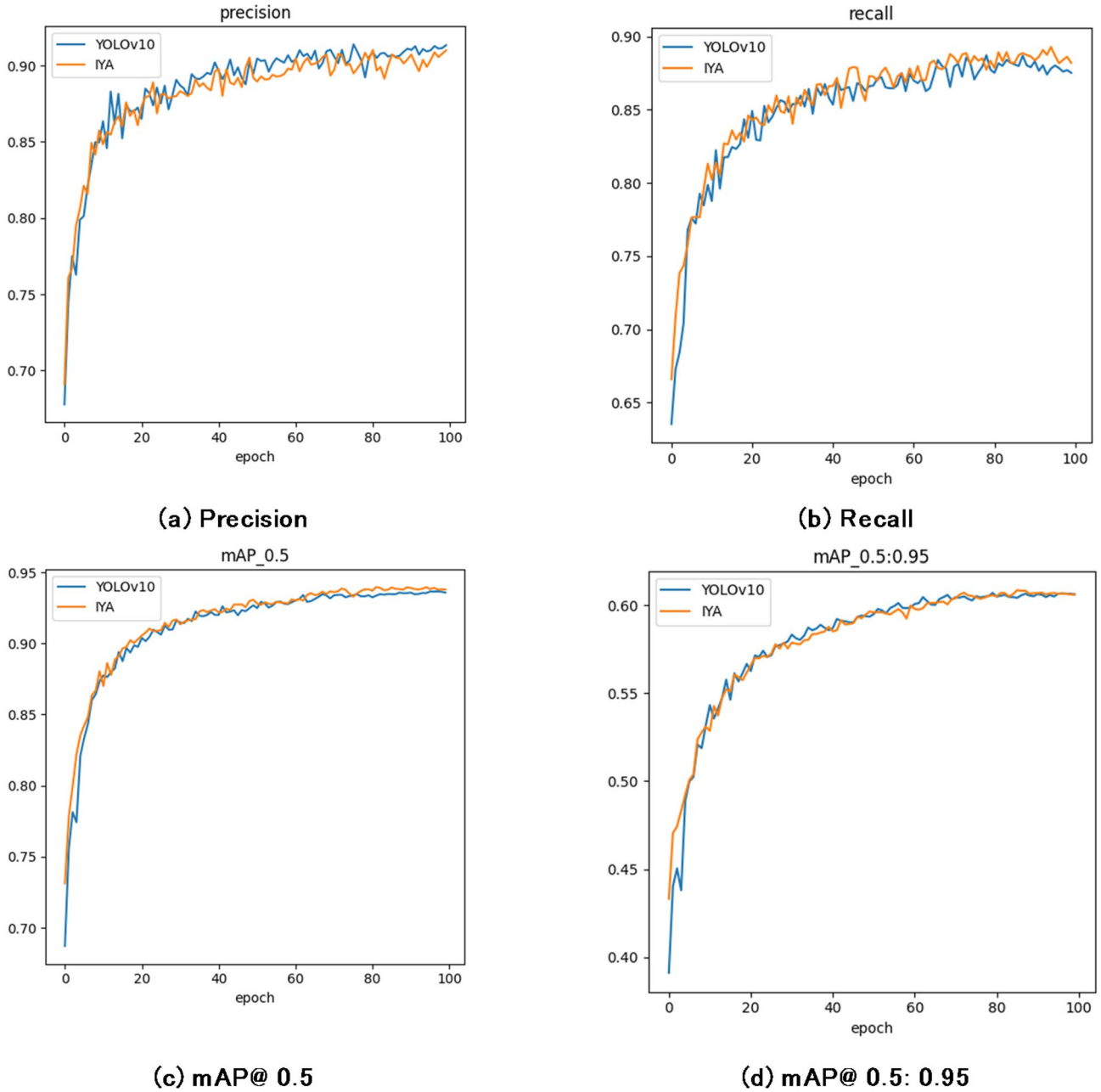

Fig. 11. Comparison curves of accuracy, recall, mAP@0.5, mAP@0.5:0.95.

accuracy, so that it facilitates work on embedded devices in the scenario of limited memory to improve its universality.

Algorithm	mAP@0.5 (%)	Floating point calculations/GFLOPs	Model size/MB
SSD	77.24	273.6	93.30
Faster RCNN	69.04	398.9	110.81
YOLOv3	78.0	13.0	17.02
YOLOv8	92.4	8.2	6.09
YOLOv9-tiny	93.5	11.0	17.57
YOLOv10	93.4	8.4	5.61
YOLOv10-DSConv	93.5	7.8	5.85
YOLOv10-DynamicConv	90.1	7.8	5.97
YOLOv10-Dysample	93.6	8.4	5.64
YOLOv10-CARAFE	92.4	8.6	5.76
YOLOv10-WIoU	93.7	7.8	5.61
YOLOv10-SIoU	92.8	8.4	5.61
YOLOv10-EIoU	91.9	8.4	5.61
IYA	93.9	7.8	5.53

Table 6. Comparison of different algorithms.

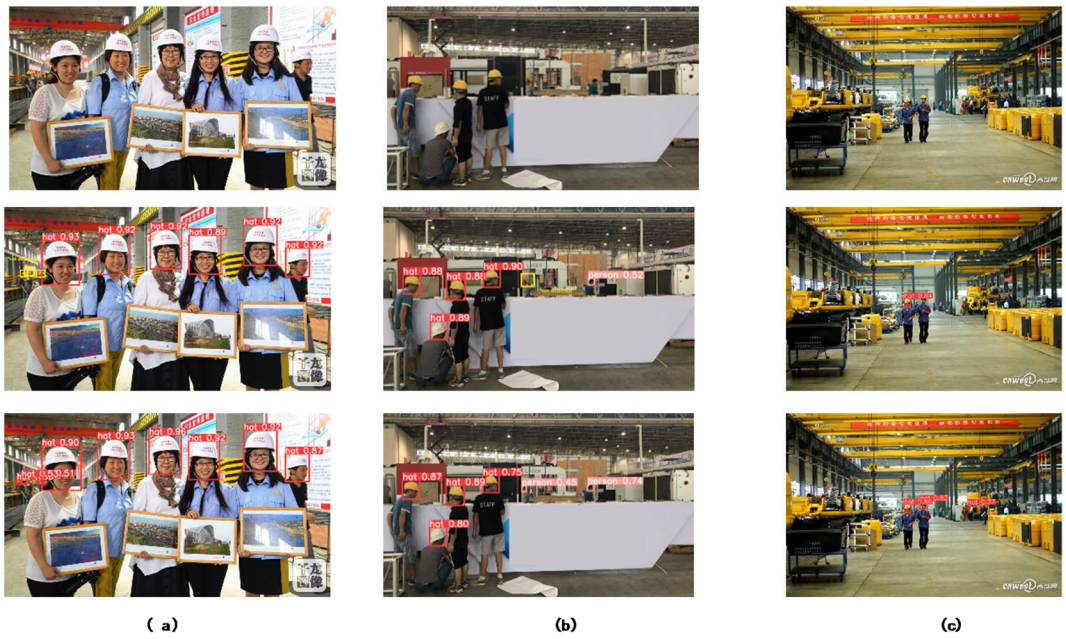

Fig. 12. Comparison of detection results with YOLOv10 algorithm and IYA.The first row signifies the original image, and the second and the third rows represent the detection results of the original YOLOv10 algorithm and IYA, respectively. (**a**) Dense small objects (**b**) Distant small objects (**c**) Small objects with complex backgrounds.

Data availability

All data or codes generated during the study are available from the corresponding authors by request.

Received: 11 February 2025; Accepted: 16 May 2025
Published online: 27 May 2025

References

1. Hnewa, M. & Radha, H. Integrated multiscale domain adaptive YOLO. *IEEE Trans. Image Process.* **32**, 1857–1867 (2023).
2. Liu, M., Chen, Y., Xie, J., He, L. & Zhang, Y. LF-YOLO: A lighter and faster YOLO for weld defect detection of X-Ray image. *IEEE Sens. Journal.* **23** (7), 7430–7439 (2023).
3. Peng, J., Chen, Q., Kang, L., Jie, H. & Han, Y. Autonomous recognition of multiple surgical instruments tips based on arrow OBB-YOLO network. *IEEE Trans. Instrum. Meas.* **71**, 1–13 (2022).
4. Redmon, J. et al. You only look once: unified, Real-Time object detection. *Computer Vis. & Pattern Recognit. IEEE (* (2016).
5. GIRSHICK, R. et al. Rich feature hierarchies for accurate object detection and semantic segmentation [C]//*Proc. of the IEEE conference on computer vision and pattern recognition*. 580–587 (2014).

6. Liu, W. et al. SSD: Single shot multibox detector[C]//Computer Vision-ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, 21–37 (2016).
7. Jia, W. et al. Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5detector. *Iet Image Process.* **15**, 3623–3637 (2021).
8. Talaat, F. M. & ZainEldin, H. An improved fire detection approach based on YOLO-v8 for smart cities. *Neural Comput. Applic.* **35**, 20939–20954 (2023).
9. Mochun, L. I. U., Zhenyuan, C. H. U. & Mingshi, C. U. I. Red-ripe strawberry identification and stalk detection based on improved YOLO v8-Pose[J]. *J. Agricultural Mach.* **54** (S02), 244–251 (2023).
10. Sekharamantry, P. K. et al. Deep Learning-Based Apple detection with attention module and improved loss function in YOLO. *Remote Sens.* **15**, 1516 (2023).
11. Chen, J. et al. *Lightweight Helmet Detect. Algorithm Using Improved YOLOv4 Sensors*, **23**, 1256 (2023).
12. Yuanbin, W. A. N. G. et al. A helmet-wearing detection algorithm for underground mine based on improved YOLOv5s[J/OL]. *Coal Sci. Technology*, 1–11 (2024).
13. SONG Cunli. et al. Road small object detection based on improved YOLO v5 algorithm[J/OL]. *Systems Eng. Electronics* :1–11 (2024).
14. Lu, J. et al. Detection and elimination of dynamic feature points based on YOLO and geometric constraints. *Arab J. Sci. Eng* (2024).
15. Xiaomeng, M. I. N., Wenhua, D. U., Nengquan, D. U., Zhiqiang, Z. E. N. G. & Guaner, L. I. U. A machine tool recognition method based on improved YOLOv5[J]. *Tool. Technol.* **58** (3), 156–160 (2024).
16. Wang, S. B. et al. AMEA-YOLO: A lightweight remote sensing vehicle detection algorithm based on attention mechanism and efficient architecture. *Supercomput* **80**, 11241–11260 (2024).
17. Chen, W. et al. YOLO-face: A real-time face detector. *Vis. Comput.* **37**, 805–813 (2021).
18. Han, J. et al. Safety helmet detection based on YOLOv5 driven by Super-Resolution reconstruction. *Sensors* **23**, 1822 (2023).
19. Chen, J., Zhu, J., Li, Z. & Yang, X. YOLOv7-WFD: A novel convolutional neural network model for helmet detection in High-Risk workplaces. *IEEE Access.* **11**, 113580–111392 (2023).
20. Wang, A. et al. YOLOv10: Real-Time End-to-End Object Detection. *arXiv abs/2405.*14458, ().14458, (). (2024).
21. Liu, W., Lu, H., Fu, H. & Cao, Z. Learning to Upsample by Learning to Sample, *IEEE/CVF International Conference on Computer Vision (ICCV)*, Paris, France,6004–6014 (2023).
22. Han, K., Wang, Y., Guo, J., Wu, E. & Recognition, P. ParameterNet: Parameters are All You Need for Large-Scale Visual Pretraining of Mobile Networks, in 2024 IEEE/CVF Conference on Computer Vision and (CVPR), Seattle,15751–15761,(2024).
23. Wang, J. et al. CARAFE: Content-Aware ReAssembly of FEatures, 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 3007–3016(2019). (2019).

Acknowledgements

This research was funded by the Pingdingshan University Doctoral Foundation (Grant No.PXY-BSQD-2022004),Pingdingshan University students Innovation and Entrepreneurship Training plan for 2024 (Grant No.109192025052,109192025068), the Key scientific research projects of colleges and universities in Henan Province of China (25A520040), and the Scientific and Technological Project in Henan Province of China (Grant No.242102210205, 242102210131, 232102220098, 252102210028).

Author contributions

X.Q.: performed Conceptualization, Methodology, J.Q.,Q.S: Writing-Review and Editing. Y.C and B.J.: Writing-Original Draf. Y.F. ,P.Y: and L.S.checked and corrected the manuscript and. All authors have read and agreed to the published version of the manuscript.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to H.Y.