Hindawi
Journal of Sensors
Volume 2023, Article ID 7230463, 14 pages
https://doi.org/10.1155/2023/7230463

Research Article

Safety Helmet-Wearing Detection System for Manufacturing Workshop Based on Improved YOLOv7

Xiaowen Chen [1,2] **and Qingsheng Xie** [1]

[1] *Key Laboratory of Advanced Manufacturing Technology, Ministry of Education, Guizhou University, Guiyang, Guizhou 550025, China*
[2] *School of Biology and Engineering, Guizhou Medical University, Guiyang, Guizhou 550025, China*

Correspondence should be addressed to Qingsheng Xie; gs.ly17@gzu.edu.cn

Received 3 January 2023; Revised 17 April 2023; Accepted 6 May 2023; Published 17 May 2023

Academic Editor: Rosalba Liguori

Safety helmets play a vital role in protecting workers' heads. In order to improve the accuracy of the detection model in complex environments, such as complex backgrounds and different lighting and distances, we propose a safety helmet-wearing detection algorithm based on the improved YOLOv7. In the backbone network, 16-channel features are used to replace 3-channel RGB features. Structured pruning is performed in the head network, and the loss function is replaced by SIoU. Experiments on the "helmet-head," "helmet-data," and "helmet" data sets show that the mAP and F1 of YOLOv7_ours improved in this paper are better than Faster RCNN, YOLOv5, and YOLOv7 series models. On image data of different application scenarios, light intensity, and color depth, YOLOv7_ours has better stability and higher accuracy and can detect at 112.4FPS (1000/8.9). Based on the improved YOLOv7_ours, we integrated face recognition technology and text-to-speech (TTS) to realize helmet detection, identity recognition, and automatic voice reminder capabilities and developed a safety helmet-wearing detection prototype system. We verified the feasibility of the helmet detection algorithm and system in the semifinished product manufacturing workshop.

1. Introduction

In the working environment of manufacturing workshop, safety helmet plays a crucial role in protecting the head of the operator [1, 2], and it can effectively protect the head of the operator. The behavior of workers not wearing safety helmets in the work area poses a huge safety risk. However, the manual safety inspection management mode has problems such as high manual management costs and low efficiency. It has become a trend to upgrade traditional equipment into intelligent systems with the help of technologies such as artificial intelligence and the Internet [3]. In recent years, deep learning [4] has achieved good results in object detection [5–8], and it is widely used in the industry [9], transportation [10, 11], and other fields. The target detection method based on machine vision has the advantages of strong real-time performance, wide coverage area, and low cost [2], and the automatic detection of helmet wearing in video surveillance systems has become a current research hotspot [12].

Li et al. proposed a helmet detection algorithm based on Faster RCNN [13], which can identify the wearing status of the helmet. Based on YOLOv4, Zhang et al. designed a hard hat-wearing detection algorithm SCM-YOLO [14] for complex scenes, which effectively alleviated the problem of insufficient feature extraction in complex scenes. To solve the problem of low detection accuracy of small targets and dense targets, Song et al. proposed an intelligent helmet recognition system based on the combination of DeepSort and YOLOv5 detectors [1], which improved the detection speed and accuracy of the model. Deng et al. designed a lightweight helmet detection algorithm based on YOLOv3 [15], which effectively reduces the computational cost of the model. Chen et al. proposed a safety helmet-wearing detection method that improves the YOLOv4 algorithm [16], which uses a lightweight network PP-LCNet as the backbone network to reduce model parameters. In order to overcome the problem of slow hard hat detection for high-resolution images in the construction industry [17], a multichannel attention module is used to improve the breadth of feature capture and the image resolution before the detection module. Song and Wang proposed a novel anchor-free mechanism-based object detection model (RBFPDet) [18], which uses strong semantic feature points to implement the detection task of safety helmets.

Although computer vision-based safety helmet detection algorithms have been applied in construction, industrial workshops, etc., there are still some specific challenges in manufacturing workshops. For complex backgrounds and environments with different lighting and distances, the detection algorithm needs to have stronger target recognition capabilities and intelligent processing. This paper proposes an improved YOLOv7 algorithm. Based on this algorithm, we integrate face recognition technology [19] and speech synthesis [20, 21] to realize helmet detection, identity recognition, automatic voice reminder, etc. and developed a safety helmet-wearing detection prototype system. We make the following contributions:

(1) This paper proposes an improved YOLOv7 safety helmet detection model. On the "helmet-head," "helmet-data," and "helmet" data sets, compared with Faster RCNN, YOLOv5, and YOLOv7 series models, the mAP and F1 of YOLOv7_ours are better than these models. It has good stability and high precision in different application scenarios, light intensity, and color depth data and can perform detection at 112.4FPS

(2) Based on YOLOv7_ours, this paper integrates face recognition technology and TTS technology to realize the identification of violators and automatic voice reminder

(3) A web-based safety helmet-wearing detection prototype system is developed in the manufacturing workshop. The data interaction and sharing between the detection model and the management system are realized through the database, and the feasibility of the improved model and detection system is verified in the semifinished product processing zone scenario

This paper is organized as follows: Section 2 describes the frame structure of the safety helmet detection system in the manufacturing workshop. Section 3 proposes an improved YOLOv7 network model and verifies the performance of the model on a public data set. Section 4 introduces the prototype system and verification experiment of safety helmet-wearing detection in the web-based manufacturing workshop. Section 5 summarizes the work of this paper and proposes subsequent optimization content and research directions.

2. System Framework and Process

2.1. System Architecture. This paper studies the methods of object detection and recognition based on machine vision, proposes an improved algorithm based on YOLOv7, and builds a prototype system of safety helmet detection for manufacturing workshop. The system consists of device layer, processing layer, and application layer, as shown in Figure 1.

In Figure 1, the device layer is mainly composed of multiple high-definition webcams and loudspeakers located around the workshop. The webcam transmits the video data captured in the monitoring area to the system, and when a worker not wearing a safety helmet is detected, a safety warning message is output through the speaker.

The processing layer consists of data preprocessing, target detection, face recognition, TTS, and other modules. The improved YOLOv7 is used as the target detection system in the data processing layer. The face recognition module is used to identify workers who do not wear safety helmets, and the TTS module is used to convert the safety warning information of the application layer into sound information.

The application layer consists of device management, face management, monitoring and warning, and log management. Device management is used to register the name, ID, installation location, and coverage area of each webcam. Face management includes the maintenance of basic face information to provide object identification for subsequent information without safety helmet. Monitoring and warning, according to the video input of multichannel cameras, determine whether the operators in each area wear safety helmets and make corresponding processing according to the detection results. The log management provides the query of the working status and identification results of the camera.

2.2. Detection Process. The video information captured by the webcam is pushed to the safety helmet detection system, which detects and recognizes the target of the video stream [22, 23] and extracts the region of interest (ROI) [24] of the head and face for face recognition, so as to realize the identity recognition of the active object in the video. When the system detects that the personnel without safety helmet enters the operation area, it combines the identified object name to form a text warning message and then outputs the text information through the loudspeaker with the help of the TTS module, so as to realize the automatic detection of the safety helmet and intelligent warning functions. Its flow is shown in Figure 2.

In Figure 2, the video data are processed by data preprocessing, head and helmet detection, and face detection and recognition successively, and then, the results of target detection and recognition are output to the log and database. When a worker not wearing a helmet is detected in the work area, a corresponding voice reminder will be given. The specific algorithm is shown in Algorithm 1.

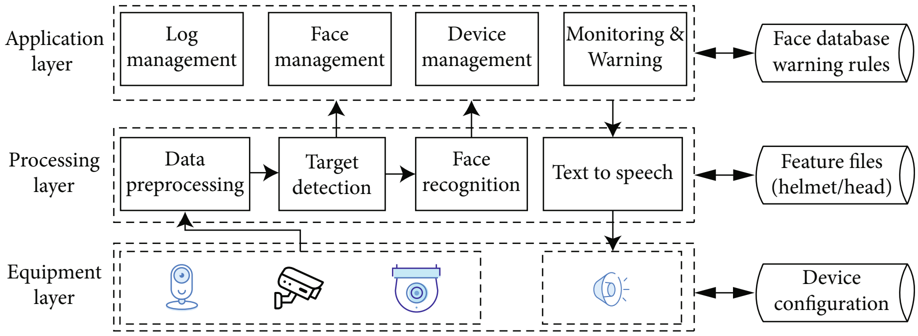


Figure 1: Architecture diagram of safety helmet detection system in manufacturing workshop.

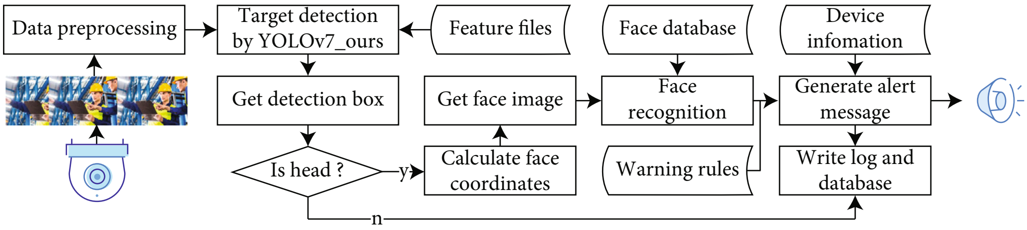


Figure 2: Safety helmet detection process.

```
Input: Input video monitored from a camera
Output: Result =Hemlt / Head (Alarm sounds)
Begin
  for each clip in video do
  Result='Hemlt' # set default value
  objects=getDetectionBox(clip) # get getDetectionBox by YOLOv7_ours
  for xywh,conf, cls in objects # conf is the confidence of bounding box
      objectname =“
      warningtext=“
      if cls='Head' and conf >0.7 then
          face =getFaceLocation(xywh) # Calculate face coordinates
          objectname= faceRecognition(face)
          warningtext=createWarningText(objectname) #generate alert message
          textToSpeech(warningtext) # broadcast warning
          Result='Head'
      end if
      writeLogAndDatabase(clip , xywh , conf , cls , objectname , warningtext)
    end for
  end for each
End
```

Algorithm 1: Algorithm of helmet-wearing detection.

In this paper, YOLOv7_ours is used to detect the wearing status of safety helmets of workers in the manufacturing workshop. It is an optimized model based on YOLOv7. In the model preparation stage, the public data set of safety helmet was obtained from the official website of Kaggle. LabelIMG tool was used to label all the data, and the human head in the image was divided into two types: wearing safety helmet and not wearing safety helmet. We use some labeled data to train YOLOv7_ours and improve the accuracy and accuracy of the models by optimizing the network structure or parameters.

Face recognition is a biometric identification technology based on human facial feature information [19]. The face detection system is composed of four parts: face image acquisition, face image preprocessing, face image feature extraction, and matching and recognition. Firstly, OpenCv is used to extract the face region from the image to form the face region data. Secondly, the feature points are used to correct the posture of the side face to the front face. Then, the facial features in the image are calculated, and the 128-dimensional face feature values are generated [25, 26]. Finally, the formed feature values are matched with the data in the face feature database in the system, so as to obtain the identity information of the person corresponding to the face. Face recognition library in Python is used for face image preprocessing, feature coding, and matching in the system.

In the monitoring and warning module, YOLO is used to capture human objects in videos or images, and the border coordinates of each individual in the images are recorded. Then, the helmet-wearing status detection and face recognition are carried out in the border area, and the test results are written into the log and database. When the current individual is detected not wearing a safety helmet, the region information of the current camera and face recognition results are extracted to generate warning text information. Finally, the warning text is converted into sound information through the TTS module and output through the loudspeaker.

3. Improved YOLOv7 Network

3.1. YOLOv7 Network. YOLOv7 network is the latest product from YOLO. It is characterized by high detection accuracy, fast detection speed, and lightweight, which surpasses all known object detectors in both speed and accuracy [8]. YOLOv7-tiny, YOLOv7, and YOLOv7-w6 are designed, respectively, for edge GPU, normal GPU, and cloud GPU. Based on the above three basic models, the depth and width of the network are adjusted for different application scenarios to form YOLOv7-x, YOLOv7-e6, YOLOv7-d6, YOLOv7-e6e, and other models. Except YOLOv7-tiny, which uses leaky ReLU activation function, all other models adopt sigmoid linear unit (SiLU) as activation function.

The YOLOv7 network structure includes Input, backbone network, and head, as shown in Table 1. The default image size of Input terminal is 640*640. Input terminal adopts data enhancement, adaptive anchor, and adaptive image scaling module, while Mosaic and Mixup are used for data enhancement. The backbone network is a convolutional neural network composed of CBS, ELAN, and MP modules. The backbone network inputs three different fine-grained image features to the Neck network. Head network is a series of hybrid image feature aggregation layer, mainly composed of CBS, SPPCSPC, ELAN-W, MP, Upsample, and REP_CBS modules.

3.2. Improvement of YOLOv7 Network. Helmet safety detection in manufacturing workshop is very important, and the accuracy and response speed of detection model directly affect the safety of operators. The model based on edge GPU is difficult to meet the requirements of high accuracy and precision, while the model based on cloud GPU has a huge amount of computation, which requires higher computing power and makes it difficult for ordinary computing devices to respond immediately. Based on the above considerations, we modified the YOLOv7 model based on normal GPU type and obtained a safety helmet-wearing state detection model suitable for manufacturing workshop through network pruning, module adjustment, parameter configuration, and other aspects. The network architecture of our improved YOLOv7 model is shown in Figure 3.

Table 1: YOLOv7 network structure.

YOLOv7	Function/module
Input	Data augmentation, adaptive anchor, and adaptive image scaling
Backbone	CBS, ELAN, and MP
Head	CBS, SPPCSPC, ELAN-W, MP, Upsample, and REP_CONV
Loss	CIoU

As can be seen in Figure 3, the improved network structure is still a three-layer structure. The first node on the backbone network is CBS_6, as shown in Figure 4. The feature extraction of the input image is carried out by 6 CBS operations. CBS is an important component unit of YOLO in the whole network structure, which is composed of convolutional layer, BN layer, and SiLU activation layer. CBS_X represents a module consisting of X CBS in series. In CBS, k represents the size of the convolution kernel, s represents the step size, and t represents the number of channels of the output feature, where $(k, s) \in \{(1, 1), (3, 1), (3, 2)\}$ and $t \in \{8,16,32,64,128,256,512,1024\}$. S_i represents the size of the input, and S_o represents the size of the output, where $S_i/S_o \in \{1, 2\}$.

Adjusting the size of the input image, the number of model layers and the number of channels are common methods of model shortening. Controlling the shortest and longest gradient path is helpful to the learning and convergence of the deep network. ELAN is located in the second layer of the backbone network. ELAN uses multidimensional shallow features and deep features of the same size for feature reconstruction, and its structure is shown in Figure 5. ELAN structure can be regarded as a stack of multiple residual components of the same level. It first divides the input feature map into two parts and then extracts the feature through multiple levels of CBS. Then, the features of different levels and dimensions are reversely connected and merged according to the dimensions of the features. The final operation takes place in the CBS module. The ELAN module outputs features twice the size of the input features.

The third layer in the backbone network is the MP_Y module, as shown in Figure 6. The MP_Y module is mainly composed of maxpool, CBS, and connection modules. Y represents the ratio between the output features and the input features. When Y is 1, the feature dimension of the output and input of the MP module is the same.

In the backbone layer, each MP_Y+ELAN combination outputs a set of eigenvalues to the head network. The eigenvalues of the third combination are input to the SPPCSPC module, the first node in the head network, whose structure is shown in Figure 7. The SPPCSPC module is composed of one CBS_3, three maxpools, two CBS_1, and one CBS_2. The internal structure of the module can be seen as a module composed of a residual component and a CBS component.

Compared with the structure of YOLOv7, the following three aspects are adjusted:

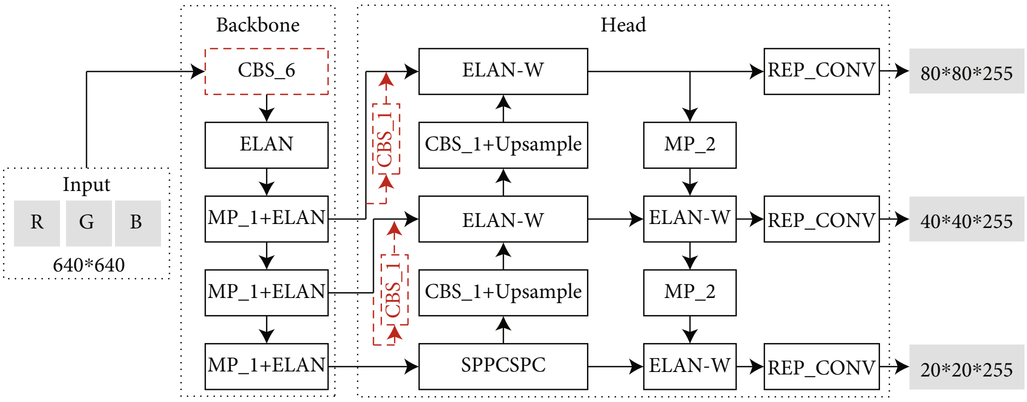


FIGURE 3: Structure of improved YOLOv7 network.

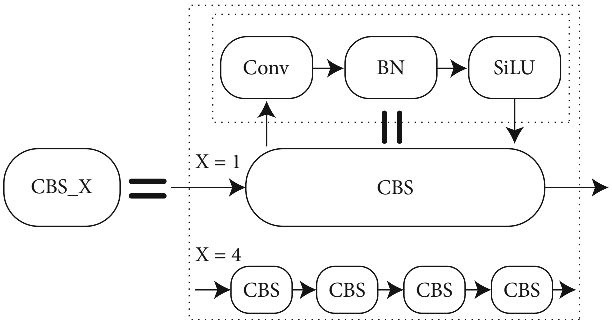


FIGURE 4: Structure of CBS_X module.

(1) The depth of the network is increased in the backbone network, and the CBS_2 module is added between the input and the CBS_4 module, as shown in Figure 8. CBS_2 consists of two CBSs with a convolution kernel size of 3 and a stride of 1. CBS_2 converts the three-channel features into 640*640*16 features and serves as the input of the original CBS_4 module. Replacing the 3-channel RGB features of the original image with 16-channel features enables the main backbone network to learn more complex and abstract features, which helps to improve the model's expressive ability, representation ability, and generalization ability

(2) We perform structural pruning on the head network and delete two CBS_1 modules, as shown in Figure 9. The input of the ELAN-W (1) module is adjusted from the original two 80*80*128 features to 80*80*512 and 80*80*128. The input of the ELAN-W (2) module is adjusted from the original two 40*40*256 features to 40*40*1024 and 40*40*256. By broadening the width of the input features of ELAN-W (1) and ELAN-W (2) modules, the ELAN-W module can learn more abundant and complex features, which helps to improve the expressive ability of the model and speed up the convergence speed

(3) We replace the loss function with SIoU. In the target detection algorithm, Intersection over Union (IoU) is an important indicator of object detection accuracy in target detection algorithms [27], which is mainly used to measure the overlap between target boxes and label boxes, as shown in

$$\mathrm{IoU} = \frac{\mathrm{area}\left(b \cap b^{gt}\right)}{\mathrm{area}\left(b \cup b^{gt}\right)}. \tag{1}$$

In Equation (1), b and b^{gt} represent prediction box and label box, respectively. The CIoU loss function adopted by YOLOv7 comprehensively measures parameters such as overlapping area, center distance, and aspect ratio, which improves the accuracy of model regression. The orientation issue where the ground truth box does not match the predicted box is not considered in the CIoU function, which may lead to slower convergence. In the improved model, we replace the loss function with the SIoU function. In the SIoU function, the angle between the regression vector and the expected regression vector is considered, which helps to improve the training speed of the model and the accuracy of inference [28]. The SIoU function is shown in

$$L_{\mathrm{box}} = 1 - \mathrm{IoU} + \frac{\Omega + \Delta}{2}. \tag{2}$$

In Equation (2), Ω represents the shape cost, as shown in

$$\begin{cases} \Omega = \sum\limits_{z=w,h} \left(1 - e^{-\omega_z}\right)^{\theta}, \\ \omega_w = \dfrac{\left|w - w^{gt}\right|}{\max\left(w, w^{gt}\right)}, \\ \omega_h = \dfrac{\left|h - h^{gt}\right|}{\max\left(h, h^{gt}\right)}. \end{cases} \tag{3}$$

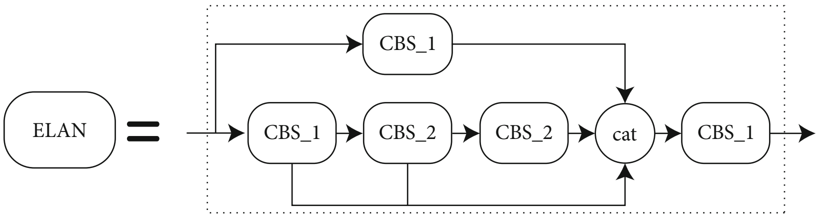


Figure 5: Structure of ELAN module.

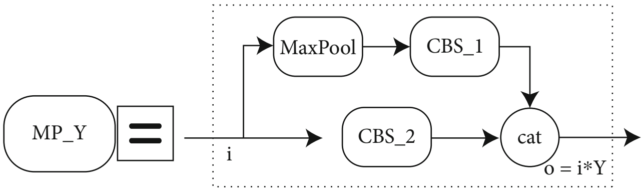


Figure 6: Structure of MP_Y module.

Δ represents the distance cost function, as shown in

$$\begin{cases} \Delta = \sum_{z=x,y} \left(1 - e^{-\gamma\rho_z}\right), \\ \rho_x = \left(\dfrac{b_{c_x}^{gt} - b_{c_x}}{c_w}\right)^2, \\ \rho_y = \left(\dfrac{b_{c_y}^{gt} - b_{c_y}}{c_h}\right)^2, \\ \gamma = 2 - \wedge. \end{cases} \quad (4)$$

$\wedge$ represents the angle cost, as shown in

$$\wedge = 1 - 2 * \sin^2\left(\arcsin (x) - \frac{\pi}{4}\right). \quad (5)$$

3.3. Performance of YOLOv7_ours

3.3.1. Evaluation Indicators. In order to verify the performance of the improved YOLOv7_ours model, we compared it with the two-stage Faster RCNN and one-stage YOLOv7 and YOLOv5 series. This paper makes a comprehensive evaluation from two aspects of inference speed and detection performance. The reasoning speed is measured by FPS, and the larger the value is, the faster the reasoning speed is. In terms of detection performance, we mainly use mAP0.5 and F1 for evaluation. The larger the value, the better the detection performance of the model. The calculation rules of mAP and F1 are shown in

$$\begin{cases} \text{mAP} = \dfrac{1}{c}\sum_{k=i}^{n} P(k) \bullet \Delta R(k), \\ F1 = \dfrac{2 \bullet P \bullet R}{P + R}, \\ P = \dfrac{\text{TP}}{\text{TP} + \text{FP}}, \\ R = \dfrac{\text{TP}}{\text{TP} + \text{FN}}. \end{cases} \quad (6)$$

In Equation (6), c represents the number of object categories, n represents the number threshold of IoU, k is the IoU threshold, $P(k)$ is the precision, and $R(k)$ is the recall rate.

3.3.2. Experimental Environment and Data Set. During the experiments in this study, the training and testing of the model used a system with the following specifications: CPU is AMD EPYC™ 7002 Series; memory is 64 G; GPU model is NVIDIA® GeForce® RTX 3090 with 24G video memory; and OS is Centos7.9 with Python 3.8 and PyTorch 1.11.

During the validation process, 3 public data sets are adopted by us. “helmet-head,” “helmet-data,” and “helmet” come from Kaggle’s official website, and each image contains several “Head” and “Helmet” tags. “helmet-head” has accumulated 20528 images, including 15887 training images and 4641 validation images. It covers images taken during the day and at night in different working scenes such as manufacturing workshops, high-altitude operations, and construction sites. The data in “helmet-data” mainly comes from construction sites, including 10,780 image data. There are 8895 images in the training set and 1885 images in the validation set. The “helmet” data set is a color image collected from different angles in a strong light environment, with a total of 2250 images, including 2000 images in the training set and 250 images in the verification set.

Data sets includes two types of pictures of workers wearing helmets and not wearing helmets in different resolutions, different environments, different colors of helmets, and different construction sites. Part of the data set samples this time are shown in Figure 10.

In the data set, each image contains several “Head” and “Helmet” labels, and the labels are stored in YOLO format. For comparison with other algorithms, we convert labels into PASCAL VOC format.

3.3.3. Ablation Experiments. To verify the impact of each improvement on model performance, we performed ablation experiments based on YOLOv7. Table 2 provides the results of the ablation experiments.

3.3.4. Results and Analysis. Based on the above experimental environment and data set, this paper trains and verifies YOLOv7_ours with Faster RCNN, YOLOv5, and YOLOv7 series. The test results of multiple models on three data sets are shown in Table 3.

It can be seen from Table 3 that the precision, recall, mAP, and F1 of the improved YOLOv7_ours in this paper on the “helmet-head” data set are 94.61%, 94.76%, 97.54%, and 94.68%, respectively. On the “helmet-head” data set,

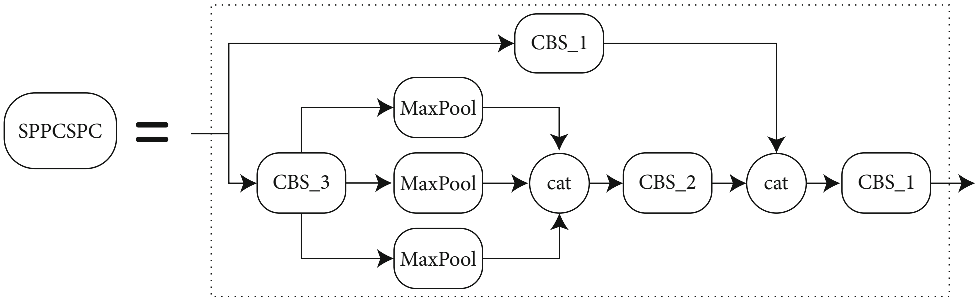


Figure 7: Structure of SPPCSPC module.

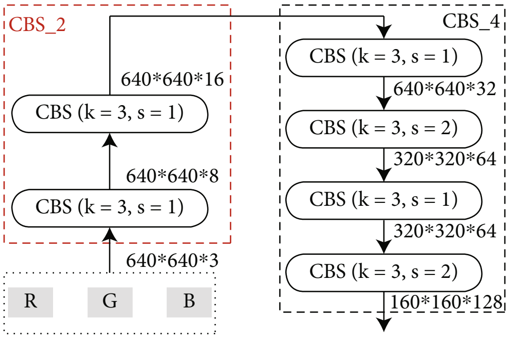


Figure 8: Structure of CBS_2 and CBS_4.

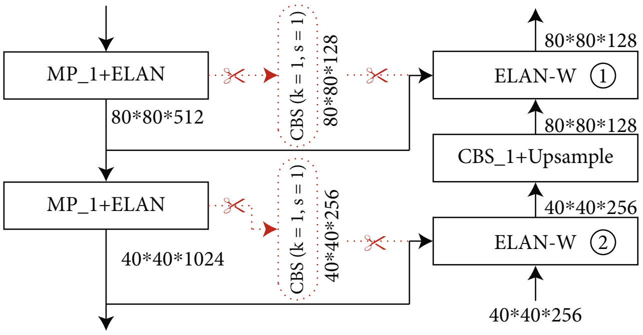


Figure 9: Structure after pruning.

compared with other models, the improvement of mAP and F1 of YOLOv7_ours is shown in Figure 11.

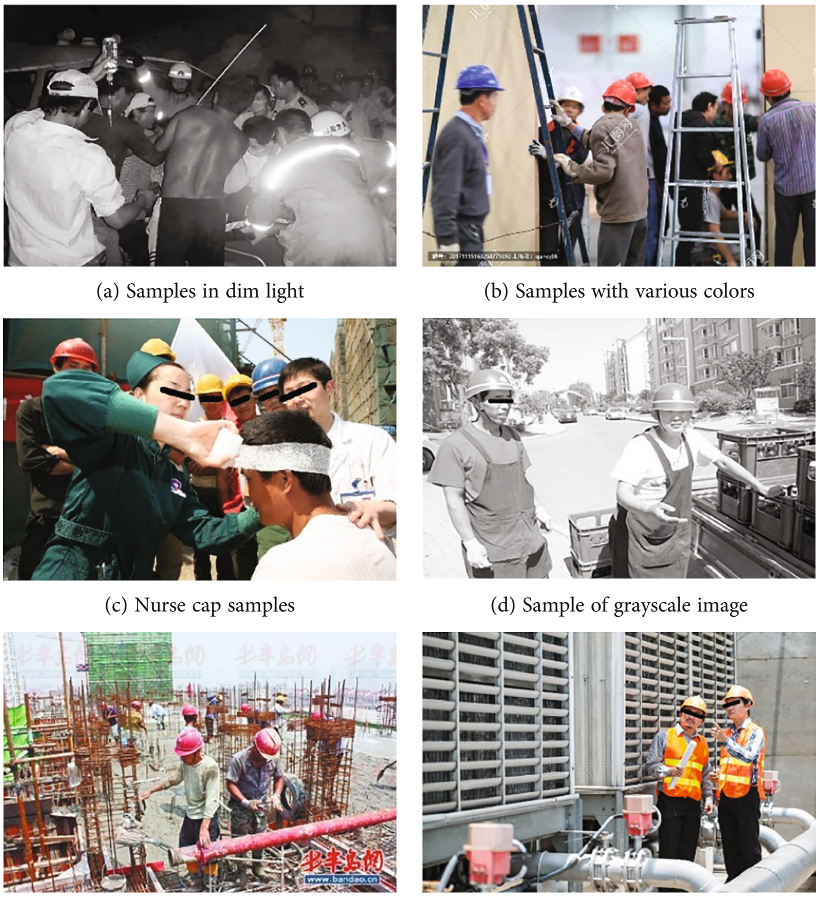
It can be seen from Figure 11 that (1) compared with Faster RCNN, the mAP and F1 of the improved model in this paper have increased by 3.5 and 2.84 percentage points, respectively. (2) Compared with 5 l, 5 m, 5 s, and 5x of the YOLOv5 series, the mAP values of the algorithm in this paper have increased by 1.03, 1.30, 1.75, and 0.92 percentage points, respectively, and the F1 has increased by 0.40, 0.83, 1.48, and 0.24 percentage points. (3) Compared with YOLOv7-tiny on the edge GPU, the mAP and F1 of YOLOv7_ours have increased by 2.45 and 2.80 percentage points, respectively. (4) Compared with the cloud GPU-based model, the mAP of the improved model in this paper is 3.82, 1.46, 2.46, and 3.23 percentage points higher than YOLOv7-d6, YOLOv7-e6, YOLOv7-e6e, and YOLOv7-w6, respectively, and the F1 values are also increased by 4.52, 1.53, 2.73, and 3.80 percentage points, respectively. (5) Compared with YOLOv7 and YOLOv7-x, the mAP of YOLOv7_ours increased by 0.39 and 0.28 percentage points, respectively, and F1 also increased by 0.32 and 0.23 percentage points, respectively.

It can be seen from Table 3 that on the "helmet-data" data set, the precision, recall, mAP, and F1 of YOLOv7_ours are 92.73%, 90.47%, 94.76%, and 91.59%, respectively. Compared with other models, the improvement of mAP and F1 of YOLOv7_ours is shown in Figure 12.

It can be seen from Figure 12 that (1) compared with Faster RCNN, the mAP and F1 of the improved model in this paper have increased by 4.57 and 4.48 percentage points, respectively. (2) Compared with the 5 l, 5 m, 5 s, and 5x of the YOLOv5 series, the mAP of the improved model in this paper has increased by 0.45, 2.88, 3.00, and 2.47 percentage points, respectively. The F1 in this paper has increased by 1.73, 1.97, 2.37, and 1.80 percentage points, respectively. (3) Compared with YOLOv7-tiny, the mAP and F1 of the model in this paper have increased by 1.82 and 1.92 percentage points, respectively. (4) Compared with the YOLOv7-d6, YOLOv7-e6, YOLOv7-e6e, and YOLOv7-w6 models, the improved model in this paper has increased by 2.55, 3.03, 1.99, and 2.79 percentage points in mAP, respectively. The F1 indicators have increased by 2.40, 2.85, 1.93, and 2.90 percentage points, respectively. (5) Compared with YOLOv7, the mAP and F1 of the algorithm in this paper have increased by 0.70 and 0.45 percentage points, respectively. Compared with YOLOv7-x, the mAP and F1 of the algorithm in this paper have increased by 0.40 and 0.11 percentage points, respectively.

It can also be seen from Table 3 that on the "helmet" data set, the precision, recall, mAP, and F1 of the improved YOLOv7_ours in this paper are 87.54%, 80.40%, 85.98%, and 83.82%, respectively. Compared with other models, the improvement of mAP and F1 of YOLOv7_ours is shown in Figure 13.

It can be seen from Figure 13 that (1) compared with Faster RCNN, the mAP and F1 of the improved model in this paper have increased by 5.39 and 6.47 percentage points, respectively. (2) The mAP value of YOLOv7_ours is 6.57 to 8.47 percentage points higher than that of the YOLOv5 series model, while the F1 is also 4.54 to 6.71 percentage points higher. (3) Compared with YOLOv7-tiny, the mAP and F1 of YOLOv7_ours increased by 6.55 and 5.75 percentage points, respectively. (4) Compared with the YOLOv7-d6, YOLOv7-e6, YOLOv7-e6e, and YOLOv7-w6 models, the



(a) Samples in dim light

(b) Samples with various colors

(c) Nurse cap samples

(d) Sample of grayscale image

(e) Samples with complex backgrounds

(f) Normal samples

Figure 10: Safety helmet samples.

Table 2: Results of the ablation experiments on "helmet".

No.	Methods	mAP (%) (*,*)	F1 (%) (*,*)
1	YOLOv7 baseline	85.19	82.51
2	+Add CBS_2 in backbone	85.67 (+0.48,+0.48)	82.63 (+0.12,+0.12)
3	+Structure after pruning	85.78 (+0.59,+0.11)	82.93 (+0.42,+0.30)
4	+Replace loss function with SIoU	85.98 (+0.79,+0.20)	83.82 (+1.31,+0.89)

1→2: the CBS_2 module is added between the backbone networks to replace the 3-channel RGB features of the original image with 16-channel features, so that the backbone network can learn deeper features. Compared with the baseline model, mAP and F1 are improved by 0.48% and 0.12%, respectively. 2→3: in order to reduce the depth of the network, we perform structured pruning on the head network. The depth of the head network is reduced by 2 layers, and at the same time, the width of the input features of the ELAN-W module is widened, which enables ELAN-W to learn more features. Compared with the baseline model, mAP and F1 are improved by 0.59% and 0.42%, respectively. 3→4: the loss function was replaced. In addition to IoU, the SIoU function also includes angle cost, distance cost, and shape cost. Compared with model 3, mAP and F1 increased by %0.2 and 0.89%, respectively. Compared with the baseline model, mAP and F1 are improved by %0.79 and 1.31%, respectively.

improved model in this paper has increased by 0.47, 0.98, 1.84, and 2.81 percentage points in mAP, respectively. The F1 indicators have increased by 1.35, 1.33, 1.66, and 1.63 percentage points, respectively. (5) Compared with YOLOv7, the mAP and F1 of the algorithm in this paper have increased by 0.79 and 1.31 percentage points, respectively. Compared with YOLOv7-x, the mAP and F1 of the algorithm in this paper have increased by 2.01 and 2.83%, respectively.

The inference speed of each algorithm is shown in Figure 14. The detection speed of YOLOv7_ours is 112.4FPS. Except for the edge cloud-based YOLOv7-tiny, the reasoning speed of the model is higher than other models. (1) The reasoning speed of YOLOv7_ours is 92.4FPS faster than Faster RCNN. (2) Compared with the YOLOv5 series, the reasoning speed of the model is 45.2 FPS, 34.2 FPS, 6.0 FPS, and 51.4 FPS faster than YOLOv5l, YOLOv5m, YOLOv5s, and YOLOv5x, respectively. (3)

Table 3: Performance of different algorithms on three data sets.

Data set	Methods	P (%)	R (%)	mAP (%)	F1 (%)
Helmet-head	Faster RCNN	92.91	90.81	94.04	91.85
	YOLOv5l	94.83	93.75	96.51	94.29
	YOLOv5m	94.64	93.07	96.24	93.85
	YOLOv5s	94.41	92.03	95.79	93.20
	YOLOv5x	94.74	94.15	96.62	94.44
	YOLOv7-tiny	92.94	90.86	95.09	91.89
	YOLOv7-d6	92.40	88.03	93.72	90.16
	YOLOv7-e6	94.11	92.22	96.08	93.16
	YOLOv7-e6e	93.22	90.73	95.08	91.96
	YOLOv7-w6	92.48	89.34	94.31	90.88
	YOLOv7	94.71	94.02	97.15	94.36
	YOLOv7-x	94.38	94.52	97.26	94.45
	YOLOv7_ours	94.61	94.76	97.54	94.68
Helmet-data	Faster RCNN	88.65	0.8562	90.19	87.11
	YOLOv5l	91.64	88.14	94.31	89.86
	YOLOv5m	93.05	86.42	91.89	89.61
	YOLOv5s	92.11	86.51	91.76	89.22
	YOLOv5x	92.02	87.66	92.29	89.79
	YOLOv7-tiny	90.83	88.53	92.94	89.67
	YOLOv7-d6	91.55	86.95	92.21	89.19
	YOLOv7-e6	92.38	85.36	91.73	88.73
	YOLOv7-e6e	91.66	87.73	92.77	89.65
	YOLOv7-w6	90.22	87.21	91.97	88.69
	YOLOv7	92.73	89.59	94.06	91.13
	YOLOv7-x	93.19	89.82	94.36	91.47
	YOLOv7_ours	92.73	90.47	94.76	91.59
Helmet	Faster RCNN	81.35	0.7373	80.59	77.35
	YOLOv5l	85.24	72.75	79.01	78.50
	YOLOv5m	89.72	69.14	77.69	78.10
	YOLOv5s	86.32	69.67	77.51	77.11
	YOLOv5x	85.36	74.01	79.41	79.28
	YOLOv7-tiny	85.66	71.71	79.43	78.07
	YOLOv7-d6	83.76	81.22	85.51	82.47
	YOLOv7-e6	87.71	77.85	85.00	82.49
	YOLOv7-e6e	86.62	78.13	84.14	82.16
	YOLOv7-w6	85.33	79.27	83.17	82.19
	YOLOv7	87.72	77.88	85.19	82.51
	YOLOv7-x	86.11	76.45	83.97	80.99
	YOLOv7_ours	87.54	80.40	85.98	83.82

Compared with the YOLOv7 series, the inference speed of the model is 49.1 FPS, 40.4 FPS, 67.1 FPS, and 17.1 FPS faster than cloud GPU-based YOLOv7-d6, YOLOv7-e6, YOLOv7-e6e, and YOLOv7-w6, respectively. The inference speed of the model is 2.5 FPS and 10.3 FPS faster than YOLOv7 and YOLOv7-x based on ordinary GPUs, respectively.

Part of the test results are shown in Figure 15. The red detection frame is a worker wearing a hard hat, while the green detection frame is a worker without a hard hat. The first column is the result of Faster RCNN, the second is the result of YOLOv5, the third is the result of YOLOv7, and the fourth is the result of YOLOv7_ours. In Figure 15(a), Faster RCNN detected a total of 5 workers wearing hard hats, but one of them mistakenly detected the light as a hard hat. Both YOLOv5 and YOLOv7 detect 3 workers wearing hardhats, while YOLOv7_ours detects 4. In Figure 15(b), YOLOv5 missed 2 construction workers wearing hard hats, YOLOv7 missed 1 construction worker wearing a helmet, and Faster RCNN and YOLOv7_ours detected 4 workers

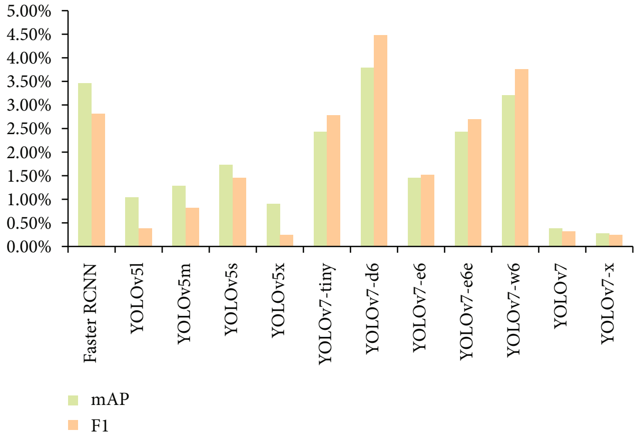


Figure 11: Improved performance of YOLOv7_ours on "helmet-head."

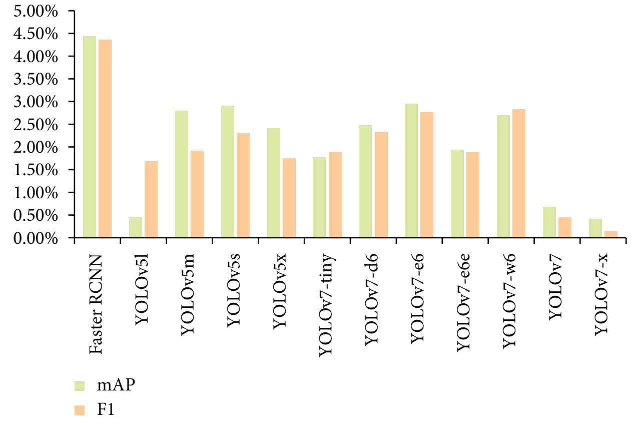


Figure 12: Improved performance of YOLOv7_ours on "helmet-data."

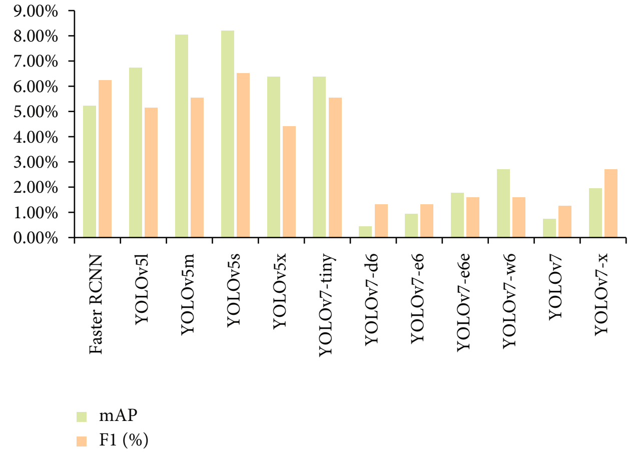


Figure 13: Improved performance of YOLOv7_ours on "helmet."

wearing hard hats at the same time. In Figure 15(c), Faster RCNN missed a worker wearing a helmet, both YOLOv5 and YOLOv7 missed a worker not wearing a helmet, and YOLOv7_ours correctly detected 7 targets.

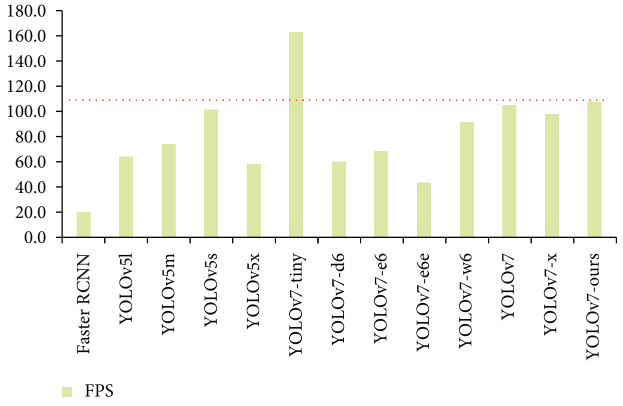


Figure 14: The inference speed of each algorithm.

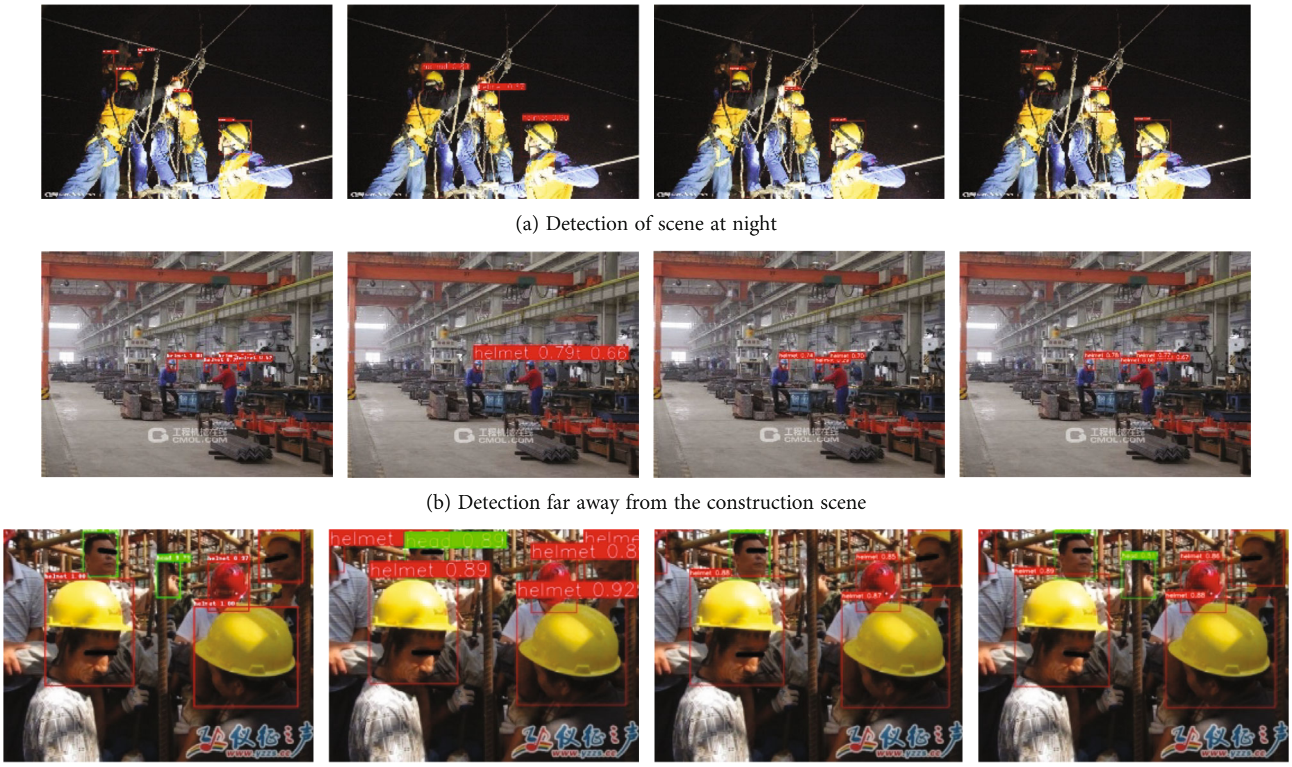
As can be seen from Table 2 and Figures 11–13 and 15, mAP and F1 of the improved YOLOv7_ours model in this paper are superior to other algorithms on the three data sets. It can be seen from Figure 14 that the detection speed of the improved YOLOv7_ours in this paper can reach 112.4FPS, and its detection speed is higher than other models, except for the edge cloud-based YOLOv7-tiny model. Although YOLOv7-tiny has the fastest detection speed, compared with YOLOv7_ours, the mAP of YOLOv7-tiny has dropped by 1.82~6.55 percentage points, and the F1 has also dropped by 1.92~5.75 percentage points. In general, the overall performance of the improved YOLOv7_ours model in this paper is superior to other models. It has better stability and higher accuracy in different application scenarios, light intensity, and color depth data and can run at 112.4FPS (1000/8.9) for detection, which can meet the requirements of strong real-time performance and high accuracy in the manufacturing workshop.

4. System Implementation

In Python environment, we integrated the improved YOLOv7_ours with modules such as face recognition and TTS to form the helmet-wearing detection. Then, the prototype system of safety helmet-wearing detection is built in the web environment. The development environment of the detection and prototype system is shown in Table 4. Because the development environment and running environment of detection model and prototype system are quite different, they realize data interaction and sharing through database. Key-value and relational databases are used in this paper. Redis is a high-performance key-value database, which is mainly used to store real-time detection results. Sql Server is a relational database management system. It stores configuration information and log information of the system and provides data sources for the information management system.

The prototype detection management system includes device management, face management, monitoring and early warning, and log management modules. The specific functions of each module are shown in Table 5.

The helmet-wearing detection model and safety helmet-wearing detection prototype system were tested in a



(a) Detection of scene at night

(b) Detection far away from the construction scene

(c) Detection occluded by steel bars

Figure 15: Comparison of detection results of four models in different construction scenarios.

Table 4: System development environment and tools.

Model/system	Development platform/ tool	Version
Helmet-wearing detection	Python	3.8
	PyCharm (Python IDE)	2021.2
Detection management system	Visual Studio (Asp.Net)	2015
	.NET Framework	4.5.2
	EasyUI	1.7
	Internet Information Services	10.0
Database	Redis	5.0
	Sql Server	2012

Table 5: List of features of a web-based detection management.

Function name	Description
Device management	This module is the basic information maintenance of webcam. Basic information includes the camera ID, network address or port, name, coverage area, and frame rate.
Face management	This module is the basic information maintenance of face. The basic information contains the number of the face, name, and picture.
Monitoring and early warning	(1) Real-time video display by webcam. (2) If head is detected (without safety helmet), the video will be captured automatically and the warning text will be generated by combining the detected face information. (3) Broadcast warning text information. (4) Generate monitoring logs.
Log management	(1) Query the monitoring logs of the camera. (2) Display of violation information, showing records or screenshots of not wearing safety helmet. (3) Statistical data query.

semifinished product factory. The tester simulated operation in the semifinished product processing area with and without safety helmet. After receiving the webcam video flow, helmet-wearing status of workers is automatically recognized and the detection results are generated and stored to the database. We can get the detection results and corresponding forensic images from the web-based "safety helmet-wearing detection prototype system." The system verification scenario is shown in Figure 16. The interface and verification results of the safety helmet-wearing detection prototype system are shown in Figure 17.

In Figure 17, the detection system shows all the records of not wearing a safety helmet in the form of a list. From the records, you can get the monitoring area, the name of the violator, and the text data of the system warning. In the verification process, helmet-wearing detection can

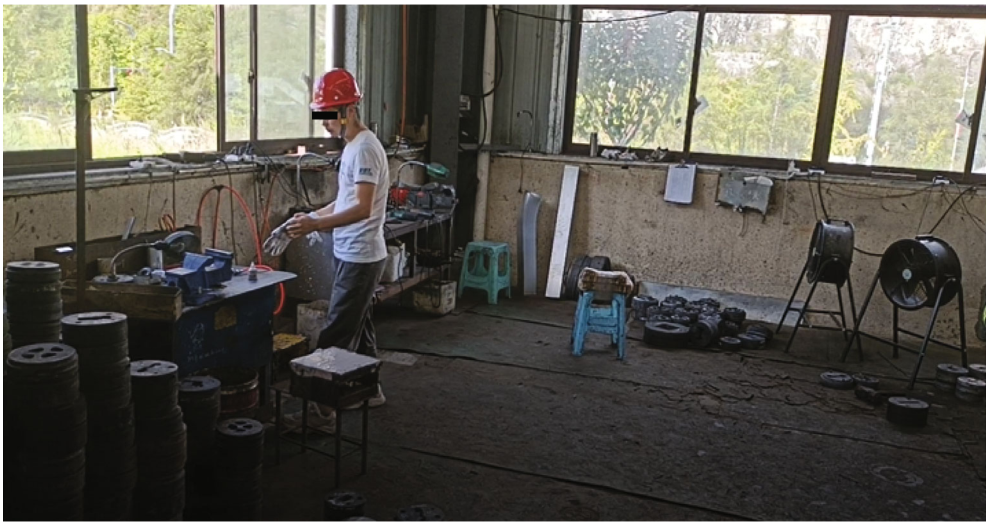
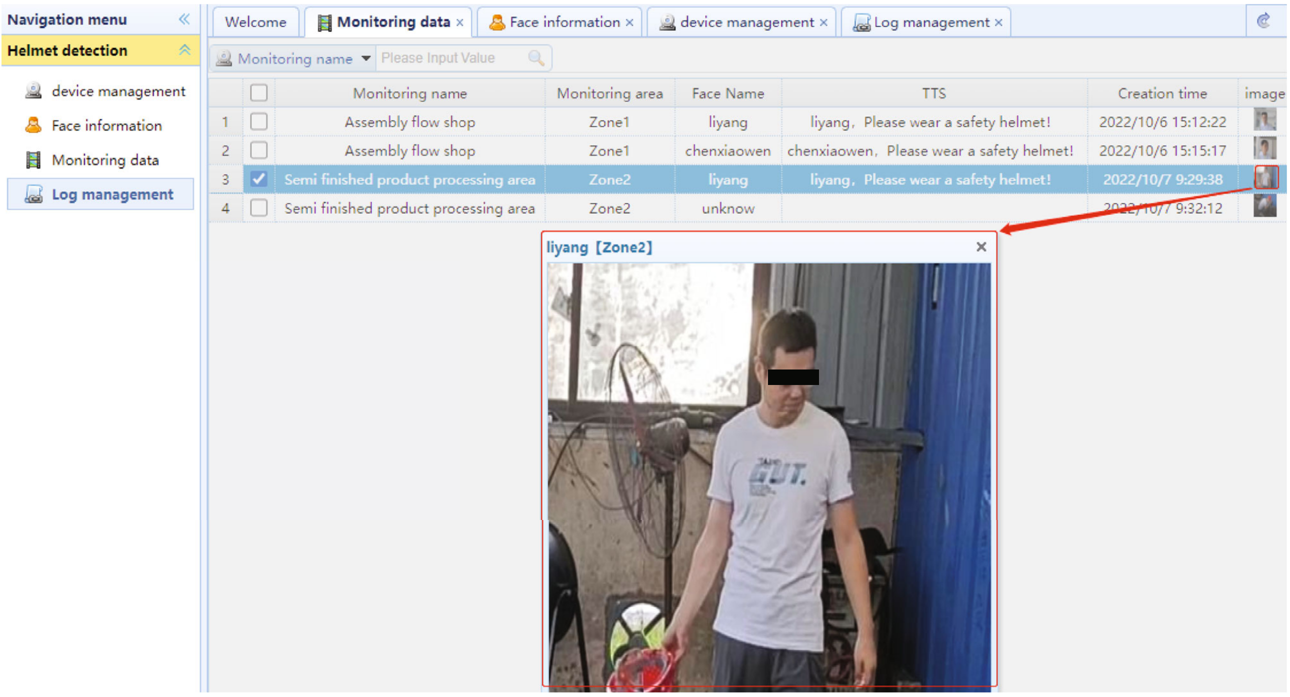


Figure 16: Test environment for prototype system.



	☐	Monitoring name	Monitoring area	Face Name	TTS	Creation time	image
1	☐	Assembly flow shop	Zone1	liyang	liyang，Please wear a safety helmet!	2022/10/6 15:12:22	
2	☐	Assembly flow shop	Zone1	chenxiaowen	chenxiaowen，Please wear a safety helmet!	2022/10/6 15:15:17	
3	☑	Semi finished product processing area	Zone2	liyang	liyang，Please wear a safety helmet!	2022/10/7 9:29:38	
4	☐	Semi finished product processing area	Zone2	unknow		2022/10/7 9:32:12	

liyang［Zone2］

Figure 17: The interface and verification results of the prototype system.

quickly detect the safety helmet-wearing of workshop workers. When helmet is not worn, the reminder message can be played in time, and the detection result and reminder information can be written to the database. The safety helmet-wearing detection prototype system can extract detection data and display it on the page in the form of a list. The verification results show that the detection model and management system can be applied to actual production operation and have great theoretical research and application value.

5. Conclusions

The safety helmet plays a vital role in protecting the head of the operator. It can effectively protect the head of the operator and prevent and reduce the damage to the head from external dangerous sources. Failure to wear a safety helmet in the work area poses a huge safety risk to the manufacturing workshop. Compared with the traditional manual inspection, the helmet detection method based on machine vision has the advantages of strong real-time performance, wide coverage area, high degree of intelligence, and low management cost. This paper proposes an improved YOLOv7 safety helmet-wearing detection algorithm, which uses 16-channel features instead of 3-channel RGB features in the backbone network. Structural pruning is performed in the head network, and the loss function is replaced by SIoU. Experiments on the "helmet-head," "helmet-data," and "helmet" data sets show that the mAP and F1 of YOLOv7_ours proposed in this paper are superior to models such as Faster RCNN, YOLOv5, and YOLOv7 series. YOLOv7_ours has good stability and high accuracy in different application scenarios, light intensity, and color depth data and can reason at 112.4FPS (1000/8.9), which is more suitable for real-time performance and high accuracy manufacturing workshop scene.

Based on YOLOv7_ours, we have integrated face recognition technology, TTS, and other technologies; realized helmet detection, identity recognition, automatic voice reminder, and other capabilities; and developed a safety helmet-wearing detection prototype system in the manufacturing workshop. We verified the feasibility of the helmet detection algorithm and system in the semifinished product manufacturing workshop. In the future, we will expand the detection objects to goggles, gloves, tooling, etc., so as to build a safety risk source detection system with strong real-time performance and high accuracy.

Data Availability

The data sets used to support the findings of this study have been deposited in Kaggle official website (https://www.kaggle.com/).

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the following projects: National Natural Science Foundation of China (grants 52065010 and 62163007) and Science and Technology Foundation of Guizhou Province under grants [2019]2814, [2020]6007, and ZK[2021]341.

References

[1] H. Song, X. Zhang, J. Song, and J. Zhao, "Detection and tracking of safety helmet based on DeepSort and YOLOv5," *Multimedia Tools and Applications*, vol. 82, no. 7, pp. 10781–10794, 2022.

[2] M. Sadiq, S. Masood, and O. Pal, "FD-YOLOv5: a fuzzy image enhancement based robust object detection model for safety helmet detection," *International Journal of Fuzzy Systems*, vol. 24, no. 5, pp. 2600–2616, 2022.

[3] X. Chen and G. Yang, "Data sensing and processing tensioning system based on the Internet of Things," *Applied Sciences*, vol. 9, no. 2, pp. 310–327, 2019.

[4] X. Zhang, X. Sun, W. Sun, T. Xu, P. Wang, and S. K. Jha, "Deformation expression of soft tissue based on BP neural network," *Intelligent Automation & Soft Computing*, vol. 32, no. 2, pp. 1041–1053, 2022.

[5] D. Miller, P. Moghadam, M. Cox, M. Wildie, and R. Jurdak, "What's in the black box? The false negative mechanisms inside object detectors," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8510–8517, 2022.

[6] T. Chen, N. Wang, R. Wang, H. Zhao, and G. Zhang, "One-stage CNN detector-based benthonic organisms detection with limited training dataset," *Neural Networks*, vol. 144, no. 1, pp. 247–259, 2021.

[7] V. Arulalan and D. Kumar, "Efficient object detection and classification approach using HTYOLOV4 and M^2RFO-CNN," *Computer Systems Science and Engineering*, vol. 44, no. 2, pp. 1703–1717, 2023.

[8] C. Y. Wang, B. Alexey, and H. Y. M. Liao, "YOLOv7: trainable bag of freebies sets new state of the art for real-time object detectors," 2022, https://arxiv.org/abs/2207.02696.

[9] Q. Hu, K. Hao, B. Wei, and H. Li, "An efficient solder joint defects method for 3D point clouds with double-flow region attention network," *Advanced Engineering Informatics*, vol. 52, no. 1, pp. 101608–101620, 2022.

[10] J. Zhang, Z. Ye, X. Jin, J. Wang, and J. Zhang, "Real-time traffic sign detection based on multiscale attention and spatial information aggregator," *Journal of Real-Time Image Processing*, vol. 19, no. 6, pp. 1155–1167, 2022.

[11] J. M. Zhang, Z. F. Zheng, X. D. Xie, Y. N. Gui, and G. J. Kim, "ReYOLO: a traffic sign detector based on network reparameterization and features adaptive weighting," *Journal of Ambient Intelligence and Smart Environments*, vol. 14, no. 4, pp. 317–334, 2022.

[12] Y. Li, J. Zhang, Y. Hu, Y. Zhao, and Y. Cao, "Real-time safety helmet-wearing detection based on improved YOLOv5," *Computer Systems Science and Engineering*, vol. 43, no. 3, pp. 1219–1230, 2022.

[13] N. Li, X. Lyu, S. Xu, Y. Wang, Y. Wang, and Y. Gu, "Incorporate online hard example mining and multi-part combination into automatic safety helmet wearing detection," *IEEE Access*, vol. 9, no. 1, pp. 139536–139543, 2021.

[14] B. Zhang, C. F. Sun, S. Q. Fang, Y. H. Zhao, and S. Su, "Workshop safety helmet wearing detection model based on SCM-YOLO," *Sensors*, vol. 22, no. 17, pp. 6702–6720, 2022.

[15] L. Deng, H. Li, H. Liu, and J. Gu, "A lightweight YOLOv3 algorithm used for safety helmet detection," *Scientific Reports*, vol. 12, no. 1, pp. 10981–10996, 2022.

[16] J. Chen, S. Deng, P. Wang, X. Huang, and Y. Liu, "Lightweight helmet detection algorithm using an improved YOLOv4," *Sensors*, vol. 23, no. 3, pp. 1256–1274, 2023.

[17] J. Han, Y. Liu, Z. Li, Y. Liu, and B. Zhan, "Safety helmet detection based on YOLOv5 driven by super-resolution reconstruction," *Sensors*, vol. 23, no. 4, pp. 1822–1836, 2023.

[18] R. Song and Z. Wang, "RBFPDet: an anchor-free helmet wearing detection method," *Applied Intelligence*, vol. 53, no. 5, pp. 5013–5028, 2023.

[19] J. Lin, Y. Li, and G. Yang, "FPGAN: face de-identification method with generative adversarial networks for social robots," *Neural Networks*, vol. 133, no. 1, pp. 132–147, 2021.

[20] O. Scharenborg, L. Besacier, A. Black et al., "Speech technology for unwritten languages," *IEEE/ACM Transactions on Audio,*

Speech, and Language Processing, vol. 28, no. 1, pp. 964–975, 2020.

[21] P. Bonifacci, E. Colombini, M. Marzocchi, V. Tobia, and L. Desideri, "Text-to-speech applications to reduce mind wandering in students with dyslexia," *Journal of Computer Assisted Learning*, vol. 38, no. 2, pp. 440–454, 2022.

[22] H. Sne and K. Ajay, "Hyperspectral imaging and target detection algorithms: a review," *Multimedia Tools and Applications*, vol. 81, no. 30, pp. 1573–7721, 2022.

[23] A. Hanif, M. Muaz, A. Hasan, and M. Adeel, "Micro-Doppler based target recognition with radars: a review," *IEEE Sensors Journal*, vol. 22, no. 4, pp. 2948–2961, 2022.

[24] C. Huang, Q. Liu, and S. Yu, "Regions of interest extraction from color image based on visual saliency," *The Journal of Supercomputing*, vol. 58, no. 1, pp. 20–33, 2011.

[25] A. Hamad and Q. Khaled, "I see faces! A review on face perception and attractiveness with a prosthodontic peek at cognitive psychology," *Journal of Prosthodontics*, vol. 31, no. 7, pp. 562–570, 2022.

[26] G. Guo and N. Zhang, "A survey on deep learning based face recognition," *Computer Vision and Image Understanding*, vol. 189, no. 1, pp. 102805–102842, 2019.

[27] J. H. Yu, Y. N. Jiang, Z. Y. Wang, Z. M. Cao, and T. Huang, "UnitBox," in *Proceedings of the Proceedings of the 24th ACM International Conference on Multimedia ACM*, Amsterdam The Netherlands, 2016.

[28] Y. Zhang, H. Li, R. Wang, M. Zhang, and X. Hu, "Constrained-SIoU: a metric for horizontal candidates in multi-oriented object detection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, no. 1, pp. 956–967, 2022.