An Improved Method Based on YOLOv7 for Detecting the Safety Helmets of Two-Wheeled Bicycle Riders

Xufei Wang[1], Penghui Wang[2], Zishuo Wang[3], Jeonyoung Song[4], Jinde Song[5]
School of Mechanical Engineering, Shaanxi University of Technology, Hanzhong, China[1, 2]
School of Electrical Information Engineering, Northeast Petroleum University, Daqing, China[3]
Department of Computer Engineering, Pai Chai University, Daejeon, Korea[4]
Jiangsu Zhonggui Heavy Industry Co., Ltd., Yancheng, China[5]

***Abstract*—Convolutional neural networks (CNNs) were widely used in object detection tasks. Usually, CNNs with strong object detection performance were difficult to apply to small, mobile embedded systems with limited computational resources due to the large number of parameters. Aiming at this problem, the lightweight improvement method for the safety helmet object detection task based on YOLOv7 has been studied. The first step was the lightweight improvement of the network. Taking YOLOv7 and YOLOv7-Tiny as the basic networks, respectively, the backbone network was improved using the MobileOne network. YOLOv7-MobileOne (YOLOv7-MO) and YOLOv7-Tiny-MobileOne (YOLOv7-TMO) were obtained. Compared with the original network parameters, the number of parameters decreased by 36.8% and 37.9%, respectively. Verified on the Pascal VOC dataset, the YOLOv7-MO had a 3.7% decrease in mAP @.5 compared to the YOLOv7. The YOLOv7-MO had a 9.8% increase in mAP @.5 compared to the YOLOv7-TMO. The second step was to improve the detection accuracy. The Coordinate Attention (CA) module was integrated at different positions of YOLOv7-MO and YOLOv7-TMO, respectively, to obtain YOLOv7-MO-Coordinate Attention (YOLOv7-MOC) and YOLOv7-TMO-Coordinate Attention (YOLOv7-TMOC). Verified on the Pascal VOC dataset, YOLOv7-MOC improved 1.44% compared to YOLOv7-MO's mAP @.5 and reduced FPS by 5.4Hz. Verified on the self-constructed two-wheeled cyclists helmet dataset (TCHD), YOLOv7-MOC increased by 0.8% compared to YOLOv7-MO's mAP @.5 and reduced FPS by 0.3Hz. YOLOv7-MOC increased by 1.0% compared to YOLOv7's mAP @.5 to 77.1%. The corresponding FPS was 28.7Hz higher, reaching 89.3Hz. Finally, experiments were conducted using the Raspberry Pi 4B embedded development board, based on the Linux system and the Pytorch framework, with the YOLOv7-TMOC network model. The results proved that the improved network model can be applied to the object detection of small embedded systems.**

Keywords*—Object detection; YOLOv7; MobileOne; CA module; TCHD

I. Introduction

Two-wheelers have become one of the common means of transportation. While they bring about convenience in transportation, they also lead to a continuous increase in traffic accidents. Such accidents often cause injuries to the riders of two-wheelers. If the riders do not wear helmets for protection, it will result in more serious head injuries. Safety helmets could effectively reduce the degree of head injury caused by traffic accidents to two-wheeled cyclists [1]. How to supervise and ensure that two-wheeled cyclists wear helmets correctly when riding on the road has become a problem that traffic safety management departments and scholars have paid attention to. With the rapid development of computer vision technology, it was a feasible way to use computer vision technology to help solve the above problems [2-5].

The convolutional neural networks have become a research hotspot in the field of computer vision for object detection. In 2016, Redmon et al. proposed the object detection network YOLO (You Only Look Once) [6]. With this network, the input image could obtain the position of the object and the confidence probability of the category to which the object belonged after only one round of network learning, achieving end-to-end detection and improving real-time performance. Subsequently, YOLOv2 [7], YOLOv3 [8], and YOLOv4 [9] have been proposed to improve the object detection network based on the YOLO network, etc. In 2022, Glenn et al. proposed the YOLOv5 network based on the YOLO series network [10]. YOLOv5 inherited the CSPDarknet53 backbone feature extraction network and introduced PANet [11] to process objects of different scales, effectively improving the detection accuracy of the network model. The improvement of the feature fusion network by the YOLOv5 network resulted in a larger network model. In 2023, Wang et al. proposed the YOLOv7 network model based on module parameterization and a dynamic label assignment strategy [12]. Their proposed multi-branch stacking module could massively reduce the redundant channels in network training, which improved the detection accuracy of the YOLOv7 network but led to a slower detection speed.

When it came to using convolutional neural networks for object detection of safety helmets, Rattapoom et al. used the K-Nearest Neighbor (KNN) classifier to classify the head with or without a helmet [13]. The results showed that the average correct detection rate was 84%, 68% and 74% for near, far, and dual lanes, respectively. Although the detection accuracy was high, the network model was large, and the detection speed was too slow to be suitable for real-time detection. Yu et al. proposed an EV helmet detection method based on the YOLOv3 network

in 2019 [14]. The method achieved higher accuracy in object detection by increasing the diversity of the dataset and optimizing the training parameters. In addition, the researchers further combined the optical flow calculation method to improve the algorithm's robustness and real-time performance. In 2020, Siebert et al. developed an object detection algorithm that collected 91,000 frames of annotated motorcycle helmet videos at observation sites in seven cities in Myanmar [15]. After training, the algorithm was found to detect motorcycle helmets with higher accuracy compared to human observers. However, it was not suitable for real-time detection due to its slow detection speed. Huang et al. used an improved Faster R-CNN network for helmet detection and localization in 2021 [16]. Due to the dataset and algorithm limitations, the method still had specific problems of false and missed detection. Jia et al. proposed a motorcycle helmet detection algorithm based on YOLOv5 [17], which improved the network accuracy by 5.2% using a soft NMS instead of YOLOv5's NMS. However, it had a more extensive network model and higher hardware requirements, which were unsuitable for real-time road detection. Zhao et al. proposed a method to improve the YOLOv7 network by using the GSConv network to replace some of the Conv in 2023 [18]. The improved network parameter counts reached 36.4M, making the network structure complex. Bao et al. proposed a method to improve the object detection of the YOLOv8s backbone network by using the MobileOne network with a parameter count reaching 11.7M in 2024 [19], which did not apply to embedded devices with small calculating power.

In summary, the current convolutional neural network-based two-wheeled cyclists helmet detection methods must be revised. Most of them used YOLO and SDD networks to improve accuracy. When embedded devices are used, the application is often limited due to minor hardware calculation power. Based on the objective conditions of practical application, drawing on the research experience of other scholars, a lightweight and high-precision two-wheeled cyclists helmet convolutional neural network model was obtained, from two aspects of the research, the establishment of the two-wheeled cyclists helmet dataset and the improvement of the YOLOv7 convolutional neural network.

The contributions of this study include the following three points:

- Collected images of two-wheeled cyclists helmets by field photography and expanded the dataset using seven data enhancement methods, such as rotation, mirroring, and panning transformation to obtain a two-wheeled cyclists helmet dataset (TCHD) containing 5957 images.
- The convolution and feature extraction layers of YOLOv7 and YOLOv7-Tiny networks were modified using the MobileOne network. Then, the CA module and K-means++ clustering algorithm were introduced, respectively, to improve the detection accuracy of the network, thereby obtaining YOLOv7-MOC and YOLOv7-TMOC networks. The detection results on the Pascal VOC dataset and TCHD showed that the lightweight network model had good detection performance.
- Using the Raspberry Pi 4B embedded development board, based on the Linux system and the Pytorch framework, and equipped with the YOLOv7-C-MO network model, experimental tests were conducted. The results proved that the improved YOLOv7 network model can be applied to the object detection of small embedded systems.

The rest of the study was organized as follows: Section II briefly describes the works related to the YOLOv7 network, the TCHD establishment, and the evaluation metrics. Section III focused on the research on lightweighting and accuracy improvement based on the YOLOv7 network, as well as the development of the testing system and experimental verification. Section IV is the conclusion of this study.

II. Related Work

Based on the analysis of the current research status of object detection networks presented in the introduction, the YOLO series of networks exhibited exceptionally high performance in object detection. However, for object detection tasks with constrained computational resources, there remains potential for optimization within the YOLO network architecture.

A. YOLOv7 Network

Through the comparison of the YOLO series of networks, the YOLOv7 network was chosen for research [20]. The network structure of YOLOv7 is shown in Fig. 1.

B. Create a Two-Wheeled Cyclists Helmet Dataset

The two-wheeled cyclists helmet dataset (TCHD) used in our study was jointly established by members of the research group. They took random images from the side of the road, including cars, motorcycles, bicycles, and pedestrians, and these images included images of multiple two-wheeled cyclists overlapping on the road and other complex scenes. There were 851 images in the entire original TCHD. Fig. 2 shows 16 samples of these images.

To enhance the training performance of the network, mitigate overfitting, and improve generalization capabilities, the TCHD was augmented using seven distinct image data augmentation techniques. An image taken from the TCHD was shown in Fig. 3(a) as an example. The transformed images were shown in Fig. 3(b) to Fig. 3(h).

- Hue-Saturation-Value (HSV) transformation: In OpenCV, set three hyperparameters as h_gain (Hue) as 0.5, s_gain (Saturation) as 0.5, and v_gain (Value) as 0.5, respectively. Fig. 3(a), after transformation using the HSV function, is shown in Fig. 3(b).
- Brightness: Used the brightness enhancement function in OpenCV and set the brightness value to 1.5. Transformed Fig. 3(a) to obtain Fig. 3(c).
- Contrast: Used the contrast enhancement function in OpenCV and set the contrast value to 1.5. Transformed Fig. 3(a) to obtain Fig. 3(d).
- Mirroring: It was implemented using the image mirroring function in OpenCV. Fig. 3(a) was mirrored horizontally to obtain Fig. 3(e).

- Shear variation: Fig. 3(a) was sheared by 30° along the horizontal direction to obtain Fig. 3(f).
- Affine transform: The image expansion was carried out by rotating and scaling the original image. Taking the center of Fig. 3(a) as the rotation center, the rotation angle was set to 30°, and the scaling factor was set to 0.5 for transformation to obtain Fig. 3(g).
- Panning variation: Using the translation function in OpenCV, the movement distance in Fig. 3(a) was set to 5 in the horizontal direction and 30 in the vertical direction, and the translation result was shown in Fig. 3(h).

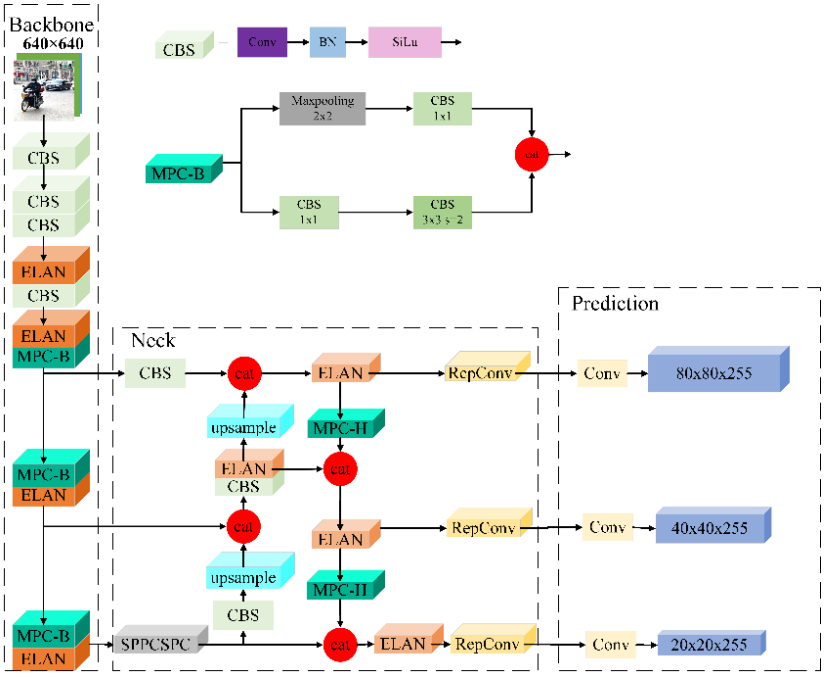


Fig. 1. The architecture of YOLOv7 network.

Fig. 2. Sixteen image samples in the TCHD.

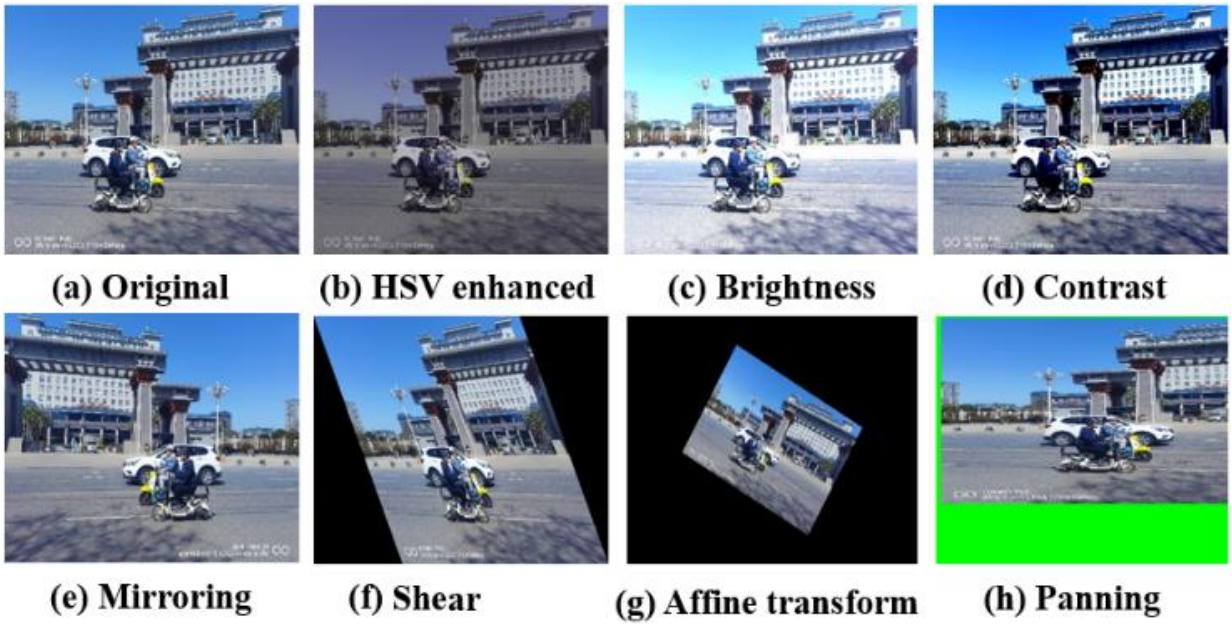


Fig. 3. The results obtained through 7 image enhancement methods.

Through the expansion of image data by the above methods, the number of TCHD images has increased from 851 to 5,957. Then, the objects in the images were classified. According to the different characteristics of two-wheeled cyclists wearing safety helmets and those not wearing safety helmets, the objects were divided into two categories: those wearing safety helmets and those not wearing safety helmets. Then, the labeling tool named Labelimg was used to label all the objects in the images.

The newly established TCHD was different from the COCO dataset in many aspects, such as the number of classes [21]. When the new TCHD was used to train the YOLOv7 network, the size of the anchor boxes would also need to be modified. So, the K-Means++ [22] algorithm was used to cluster the TCHD dataset, and the anchor boxes size was obtained as (7,7), (19,23), (33,36), (51,57), (70,74), (99,109), (258,240), (392,340), (469,430).

C. Network Model Evaluation Metrics

This study evaluated the network model performance in terms of detection accuracy and detection speed, and mean average precision (mAP) and frames per second (FPS) were chosen to measure the network model performance, respectively. The mAP was the mean average accuracy of all classes in the dataset and was used as an evaluation index for the detection accuracy of the network model [23]. The higher the value of the mAP, the better the performance of the network model. The FPS represented the number of images that the network model could detect per second and was used as an evaluation index for the detection speed of the network model. The larger the value of this evaluation index was, the faster the object detection task was carried out. Usually, the number of parameters, computation amount, and network model size affected the FPS.

III. Network Model Improvement

A. Network Model Lightweight

1) Feature extraction network lightweight: The MobileOne [24] was a lightweight model that has been optimized based on MobileNetV1 [25]. This model primarily consisted of a depthwise convolutional layer, two pointwise convolutional layers, a branched convolutional layer, and an SE module [26]. The characteristics of the MobileOne network were high real-time performance and low computational complexity. The MobileOne was used to replace the ELAN network in the Backbone part of the YOLOv7 network. The modified network was named YOLOv7-MobileOne (YOLOv7-MO), and its network structure is shown in Fig. 4.

The detailed structure of the YOLOv7-MO feature extraction network is shown in Table I.

Table I shows the parameter information of the YOLOv7-MO, including sequence, convolutional module, and channel number of input and output parameters. Compared with the YOLOv7 feature extraction network, the number of layers in the YOLOv7-MO network was reduced from 57 layers to 27 layers.

A MobileNetV1 was one of the MobileOne series of networks. The idea and method of replacing ELAN in YOLOv7 with MobileNetV1 were the same as those of MobileOne. After the replacement, the YOLOv7-Mobilenetv1 (YOLOv7-MV1) network was obtained.

In order to further reduce the size of the network, the small version of the YOLOv7 network, the YOLOv7-Tiny network, was selected for improvement. The MobileOne module was used to establish the YOLOv7-Tiny feature extraction network by using the same idea and method as the improved YOLOv7 network. The structure of the YOLOv7-Tiny-MobileOne (YOLOv7-TMO) feature extraction network was obtained, as shown in Fig. 5.

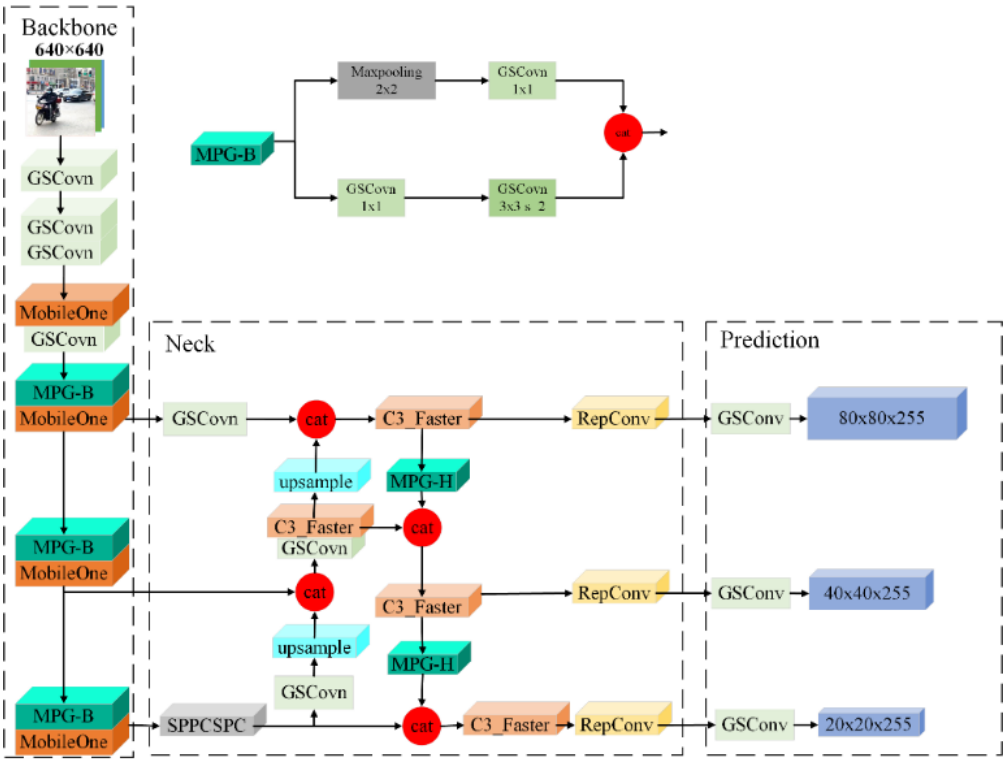


Fig. 4. YOLOv7-MO network architecture.

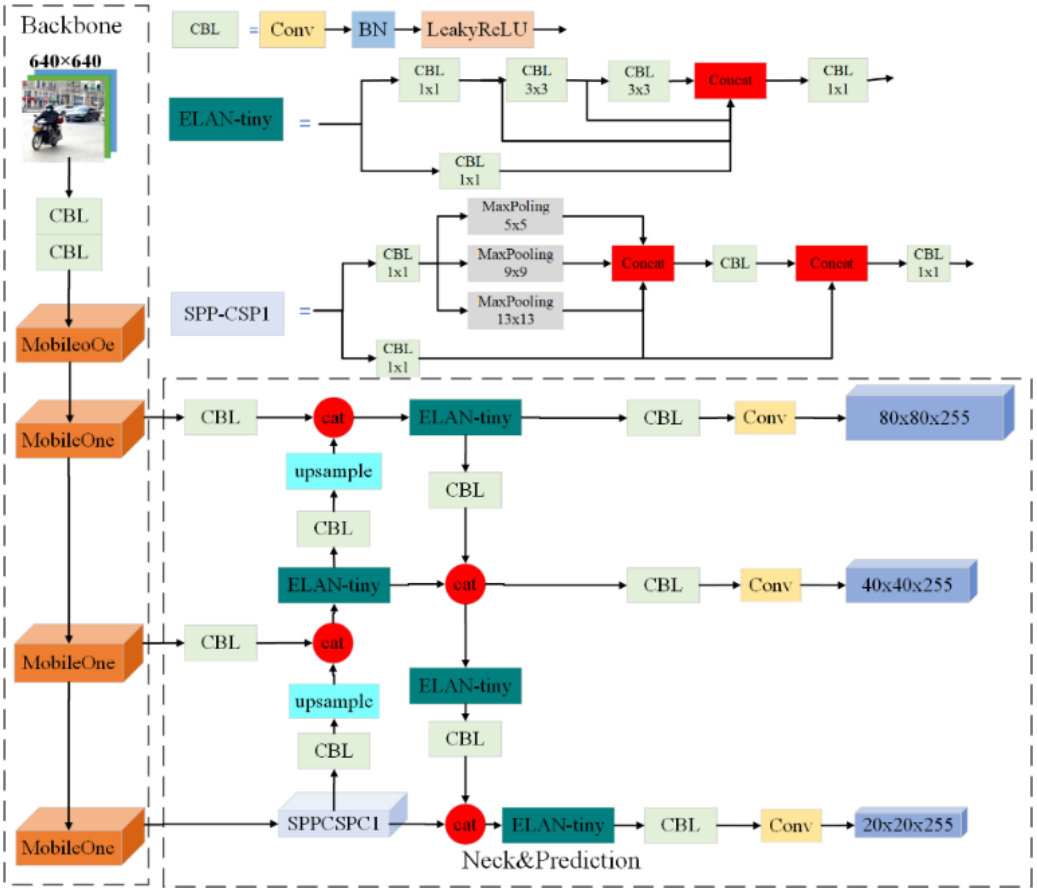


Fig. 5. YOLOv7-TMO network architecture.

TABLE I. THE STRUCTURE OF YOLOV7-MO NETWORK

Stage	Number	Model	Input size	Output size
0	1	GSConv	3	32
1	1	GSConv	32	64
2	1	GSConv	64	64
3	1	GSConv	64	128
4	1	MobileOne	128	128
5	1	GSConv	128	256
6	1	MP	-	-
7	1	GSConv	256	128
8	1	GSConv	256	128
9	1	GSConv	128	128
10	1	Concat	-	-
11	1	MobileOne	256	256
12	1	GSConv	256	512
13	1	MP	-	-
14	1	GSConv	512	256
15	1	GSConv	512	256
16	1	GSConv	256	256
17	1	Concat	-	-
18	1	MobileOne	512	512
19	1	GSConv	512	1024
20	1	MP	-	-
21	1	GSConv	1024	512
22	1	GSConv	1024	512
23	1	GSConv	512	512
24	1	Concat	-	-
25	1	GSConv	1024	256
26	1	MobileOne	1024	1024
27	1	GSConv	1024	256

2) Experiment and analysis: The main experimental conditions included Inter(R)Core(TM) i9-10900XCPU@2 as the computer core, NVIDIA GeForcePTX3080 (10GB) as the graphics processing unit, and 64GB RAM as the computer memory. The deep learning framework was PyTorch, the acceleration environment was CUDA11.3, and the programming language was Python3.7. The main parameter settings include: the input image size was Img_size=(640,640), the number of samples in each batch of iterative training was Batch_size=8, the iteration training times epoch=200, the initial learning rate was 0.01, and the weight attenuation factor was 0.0005 [27].

The public dataset PASCAL VOC was selected for the experimental dataset [28], and the training set sharing was set to 90%, resulting in 18,277 images for training and 3,226 images for verification [29].

The YOLOv7 and YOLOv7-Tiny, as well as the improved YOLOv7-MO, YOLOv7-TMO and YOLOv7-MV1 were respectively trained on PASCAL VOC dataset. The number of model parameters obtained after training was listed in Table II.

As shown in Table II, compared with the YOLOv7, the number of parameters in the YOLOv7-MO and YOLOv7-MV1 was reduced to 23.65 M and 25.19 M, respectively. Compared with the YOLOv7-Tiny, the number of parameters in the YOLOv7-TMO was reduced by 37.9% to 3.76 M.

TABLE II. THE PARAMETERS' NUMBERS OF 5 NETWORKS

Network	Parameter' number /M
YOLOv7-Tiny	6.06
YOLOv7-TMO	3.76
YOLOv7	37.43
YOLOv7-MV1	25.19
YOLOv7-MO	23.65

The loss curves of the 5 networks are compared in Fig. 6.

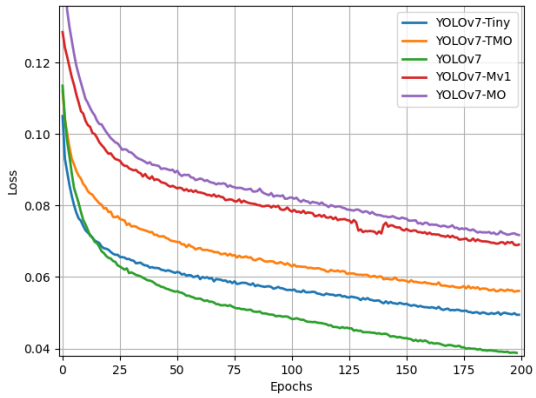


Fig. 6. Loss curves of the 5 networks.

As shown in Fig. 6, the value of loss decreases with the increase of iterative calculation for all five kinds of networks, but the convergence speed of the improved network was significantly slower than that of the pre-improved network. At the 200 step, the network corresponding to loss from large to small was YOLOv7-MO, YOLOv7-MV1, YOLOv7-TMO, YOLOv7-Tiny, and YOLOv7. The maximum value of loss was close to 0.072, and the minimum value was close to 0.02. It takes longer for lightweight networks to converge.

Further comparing the object detection performance of the five networks on the PASCAL VOC dataset, the values of precision (P), recall (R), mAP@.5, FPS and model size were listed in Table III.

TABLE III. THE DETECTION RESULTS OF 5 NETWORKS ON THE PASCAL VOC

Network	P/%	R/%	mAP@.5/%	FPS/Hz	Model size/MB
YOLOv7-Tiny	74.8	762	77.5	93.5	11.7
YOLOv7-TMO	74.3	71.4	76.9	97.4	7.7
YOLOv7	85.3	87.7	90.4	45	79.4
YOLOv7-MV1	80.5	82.0	81.6	72	57.3
YOLOv7-MO	83.4	87.3	86.7	94.2	49.6

As shown in Table III, the YOLOv7-MO network had a 1.9% decrease in precision and a 3.7% decrease in mAP @.5 compared to the YOLOv7 network, but the model size decreased by 37.5% and the FPS increased 49.2 Hz. Compared to the

YOLOv7-MV1 network, the precision improved by 2.9%, the mAP@.5 improved by 5.1%, the model size of the network decreased by 13.4%, and the FPS improved by 23.5%. YOLOv7-TMO decreased mAP@.5 by 0.6% compared to YOLOv7-Tiny, but the model size decreased by 34.1%, and the FPS increased 3.9 Hz. So, the YOLOv7-MO had comprehensive advantages in mAP@.5 and FPS compared with other networks.

B. Integration of Attention Mechanisms

An Attention Mechanism (AM) is a method to improve the accuracy of network models for object detection tasks. A common AM module was the Coordinate Attention (CA) [30]. The CA module embedded the object's position information in the image into the channel attention so that the detection model could better localize and detect the object.

The CA modules were added and combined at different positions in the Neck of the YOLOv7-MO, and three YOLOv7-MO network structures with CA modules were designed for comparative study, as shown in Fig. 7.

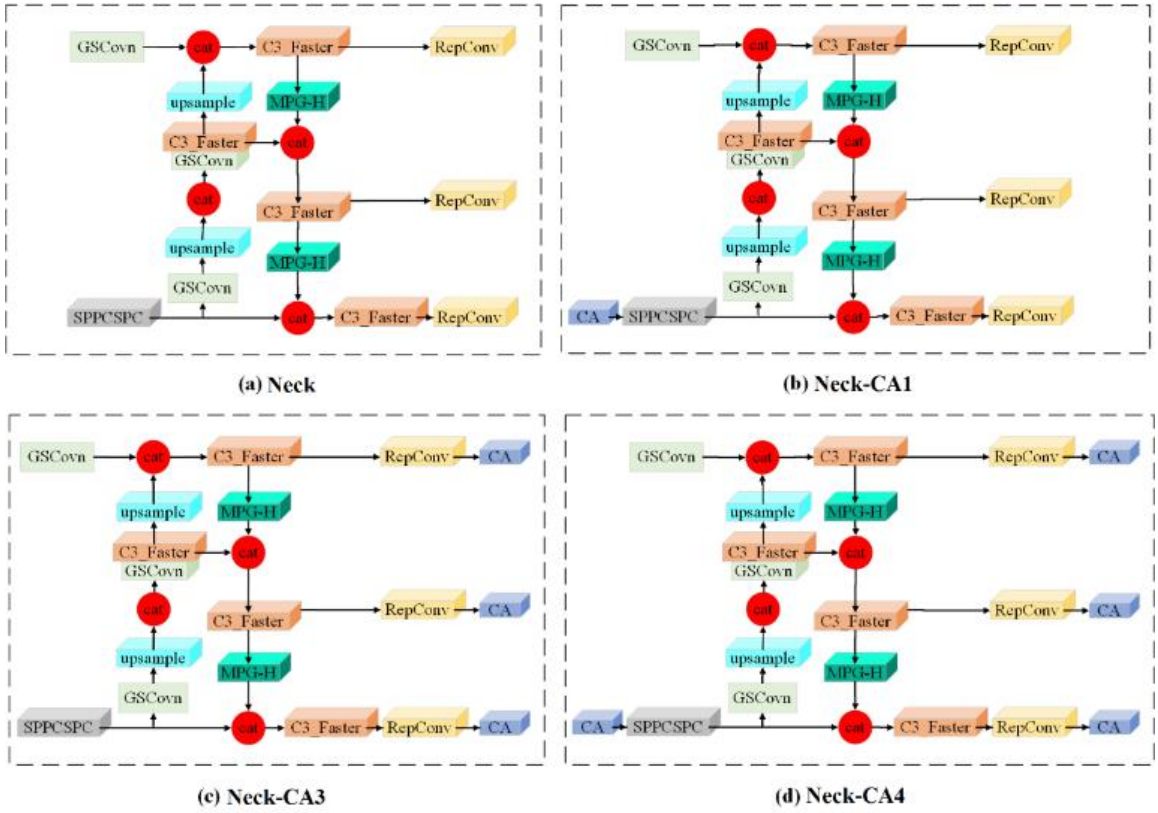


Fig. 7. Three different networks by adding CA in the YOLOv7-MO.

The Neck-CA1 was shown in Fig. 7(b), where 1 CA module was added at the connection between the last layer of the backbone network and the feature fusion network. The Neck-CA3 was shown in Fig. 7(c), where 3 CA modules were added before each of the three scale feature layers enters the detection layer. The Neck-CA4 was shown in Fig. 7(d), where 4 CA modules were added at the connection between the last layer of the backbone network and the feature fusion network.

The 3 networks by adding CA in the YOLOv7-MO were trained and tested on PASCAL VOC dataset with Epoch=100. The test results were shown in Table IV.

As shown in Table IV, compared with YOLOv7-MO, the mAP@.5 of Neck-CA1, Neck-CA3 and Neck-CA4 increased by 0.4%, 0.8% and 1.44%, respectively. Compared with YOLOv7-MO, the FPS of Neck-CA1, Neck-CA3 and Neck-CA4 decreased with the increase of CA number, and Neck-CA4 decreased by 4.9%.

TABLE IV. THE TEST RESULTS OF 4 NETWORKS ON THE PASCAL VOC

Network	CA's number	FPS/Hz	mAP@.5/%
YOLOv7-MO	0	94.2	86.7
Neck-CA1	1	94.3	87.1
Neck-CA3	3	92.5	87.5
Neck-CA4	4	89.6	88.14

The YOLOv7-MO network was improved to obtain YOLOv7-MO-CA (YOLOv7-MOC) by selecting the Neck-CA4 with the highest accuracy. A structure of the YOLOv7-MOC network was shown in Fig. 8.

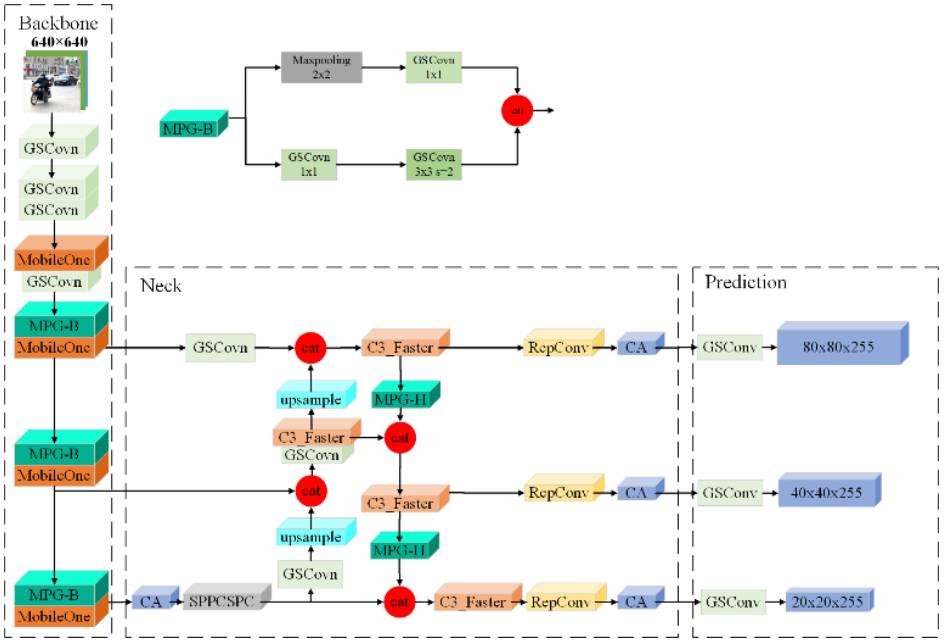


Fig. 8. A structure of the YOLOv7-MOC.

The same method was used to improve the YOLOv7-TMO network, and YOLOv7-TMO-CA (YOLOv7-TMOC) was obtained.

C. Experiments and Analysis

This section used the same experimental platform and parameters as Section III-A(2) to compare and analyze the performance of the improved networks. All the analyzed networks included five such as YOLOv7-Tiny, YOLOv7-TMOC, YOLOv7, YOLOv7-MO, and YOLOv7-MOC. The TCHD was used in the dataset, and the dataset was divided according to the training set accounting for 80% of the dataset, resulting in 4766 images in the training set and 1191 images in the verification set. After training, the loss curves of the five networks were shown in Fig. 9.

As shown in Fig. 9, the loss value of the five networks decreased with the increase of iterative calculation. Among them, the convergence speed of YOLOv7-TMOC network was the fastest. When iteration reached 200 steps, the loss value of YOLOv7-TMOC network was the smallest, close to 0.041, compared with other four networks. The change of loss curves of YOLOv7-MO and YOLOv7-MOC networks was basically the same. When iterating to 200 steps, the loss was close to 0.056. The loss curve of YOLOv7 network and YOLOv7-Tiny network had a similar change trend. When iterating to 200 steps, the loss was close to 0.046.

Further compared with the object detection performance of the five networks on the TCHD, and the values of precision, recall, mAP@.5, FPS and model size were listed in Table V, respectively.

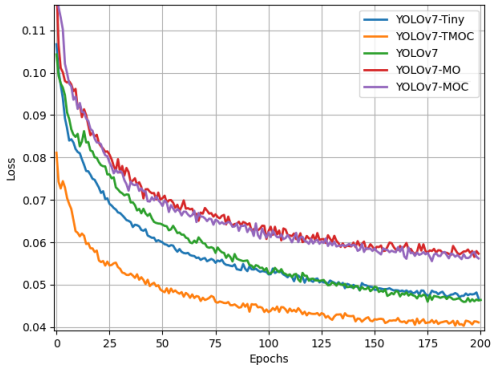


Fig. 9. The loss curves of the 5 networks.

TABLE V. THE DETECTION RESULTS OF 5 NETWORKS ON THE TCHD

Network	P/%	R/%	mAP@.5/%	FPS/Hz	Model size/MB
YOLOv7-Tiny	74.8	76.2	76.6	94.1	11.6
YOLOv7-TMOC	81.8	67.7	74.2	97.8	6.1
YOLOv7	88.5	72.7	76.1	60.6	71.2
YOLOv7-MO	83.7	75.3	76.3	89.6	49.4
YOLOv7-MOC	85.3	76.2	77.1	89.3	49.6

As shown in Table V, the model sizes of the five networks, arranged from smallest to largest, were YOLOv7-TMOC, YOLOv7-Tiny, YOLOv7-MO, YOLOv7-MOC, and YOLOv7. The mAP@.5 values of the five networks, ranked from highest to lowest, were YOLOv7-MOC, YOLOv7-Tiny, YOLOv7-MO, YOLOv7, and YOLOv7-TMOC. The FPS values of the five networks, ordered from highest to lowest, were YOLOv7-TMOC, YOLOv7-Tiny, YOLOv7-MO, YOLOv7-MOC, and YOLOv7.

The model size of YOLOv7-TMOC was the smallest, which was 47.4% smaller than that of YOLOv7-Tiny and 91.4% smaller than that of YOLOv7. The mAP@.5 of YOLOv7-MOC was the highest, which was 1.0% higher than that of YOLOv7 and 0.5% higher than that of YOLOv7-Tiny. The FPS of YOLOv7-TMOC was the highest, increasing by 3.8% compared to YOLOv7-Tiny and by 38.0% compared to YOLOv7.

Compared with YOLOv7-TMOC, YOLOv7-MOC improved mAP@.5 by 2.9%, reduced FPS by 8.7%, and increased model size by 87.7%.

So, YOLOv7-TMOC had the advantages of a small model size and fast detection speed. YOLOv7-MOC had the advantage of high detection accuracy.

D. Development and Application Testing of the Monitoring System

The improved network was applied to the real-time monitoring system for two-wheeled cyclists' helmets on the road.

The main hardware components of the monitoring system included a Raspberry Pi 4B development board, a camera, a touch screen display, and a power supply, as shown in Fig. 10.

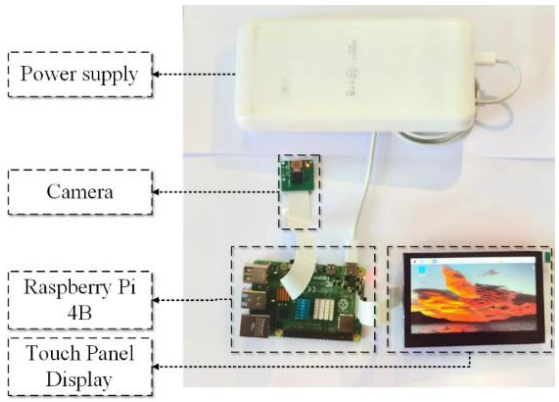


Fig. 10. Hardware of the real-time monitoring system.

The monitoring system software was based on the Linux system, the PyTorch framework, the PyQt5-tools, and was equipped with the YOLOv7-TMOC network model. The main interface of the monitoring system is shown in Fig. 11. The system supported the detection of data such as images and videos.

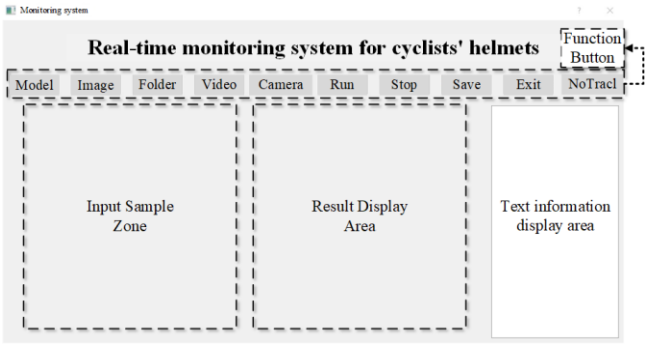
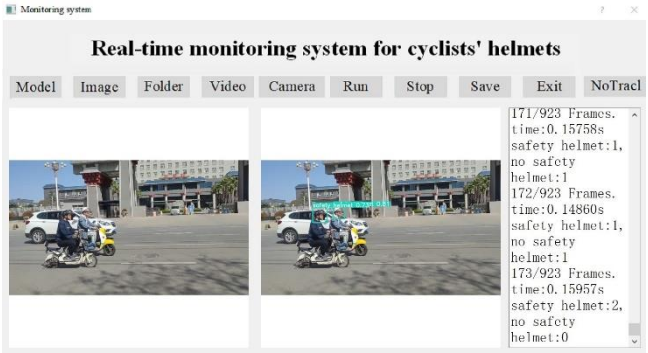


Fig. 11. Main interface of the real-time monitoring system.

The monitoring system was tested at the actual road site, and the results are shown in Fig. 12.

(a) Image detection



(b) Video detection

Fig. 12. Input sample detection.

As shown in Fig. 12(a), the monitoring system input images are captured by the roadside and detected a total of two objects of safety helmets. The system could accurately determine the number of objects as 2, and the precision rates of the two objects from left to right were 73% and 71%, respectively. As shown in Fig. 12(b), the monitoring system could accurately detect the number of objects from the input real-time video as 2, and the precision rates of the two objects from left to right were 73% and 81%, respectively.

Through actual verification, the improved network model and the constructed monitoring system could perform real-time, correct detection of the safety helmets of two-wheeled cyclists.

IV. Conclusion

Regarding the intelligent detection of safety helmets worn by two-wheeled cyclists on the road, this study studied a lightweight detection network that could be used in handheld detection instruments.

Firstly, in order to make the network lightweight, the ELAN structure in the YOLOv7 and YOLOv7-Tiny feature extraction networks was replaced with the lightweight MobileOne network, resulting in YOLOv7-MO and YOLOv7-TMO. To improve the detection accuracy, the feature fusion network of YOLOv7-MO was improved using the CA and C3_Faster modules, resulting in YOLOv7-MOC. In the same way, the YOLOv7-Tiny network was also lightweight modified, resulting in the YOLOv7-TMOC network.

Secondly, the verification using the self-created dataset TCHD showed that the model sizes of YOLOv7-MOC and YOLOv7-TMOC were respectively 30.3% and 47.4% smaller than those of YOLOv7 and YOLOv7-Tiny. The mAP@.5 of the YOLOv7-MOC network reached 77.1%, and the frame rate reached 89.3 Hz, both higher than the 76.1% mAP@.5 and 60.6 Hz frame rate of YOLOv7. The mAP@.5 of the YOLOv7-TMOC network was 74.2%, slightly lower than that of YOLOv7-Tiny (76.6%), but its frame rate was 97.8 Hz, which exceeded the 94.1 Hz of YOLOv7-Tiny.

Finally, the improved network was applied to the real-time monitoring system for safety helmets worn by two-wheeled cyclists on the road. Through image and video testing of the monitoring system, it was demonstrated that the improved network could be used in portable real-time detection devices.

ACKNOWLEDGMENT

This work was supported by the Research Foundation of Shaanxi University of Technology under Grant SLGRCQD2321 and the Shaanxi Province Key Research and Development Foundation under Grant 2025SF-YBXM-106.

REFERENCES

[1] Singleton, M. D. (2017). Differential protective effects of motorcycle helmets against head injury. Traffic injury prevention, 18(4), pp 387-392.

[2] Boonsirisumpun, N., Puarungroj, W., & Wairotchanaphuttha, P. (2018). Automatic detector for bikers with no helmet using deep learning. In 2018 22nd International Computer Science and Engineering Conference (ICSEC), pp 1-4.

[3] Tang, Z., Wu, Y. & Xu, X. (2024). The study of recognizing ripe strawberries based on the improved YOLOv7-Tiny model. Vis Comput.

[4] H. Mancy, Marwa Elpeltagy, Kamal Eldahshan and Aya Ismail (2025). Hybrid-Optimized Model for Deepfake Detection. International Journal of Advanced Computer Science and Applications(IJACSA), 16(4), pp 148-160.

[5] Xiao Lin, Shuzhou Sun, Wei Huang, Bin Sheng, Ping Li, and David Dagan Feng. (2023). EAPT: Efficient Attention Pyramid Transformer for Image Processing. In: IEEE Transactions on Multimedia, 25, pp 50-61.

[6] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 779-788.

[7] Li, X., Shi, B., Nie, T., Zhang, K., & Wang, W. (2021). Multi-object recognition method based on improved yolov2 model. Information Technology and Control, 50(1), pp 13-27.

[8] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

[9] Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934.

[10] Jocher Glenn. (2022). YOLOv5 release v6.1. https://github. com/ultralytics/yolov5/releases/tag/v6. 1.

[11] Wang, K., Liew, J. H., Zou, Y., Zhou, D., & Feng, J. (2019). Panet: Few-shot image semantic segmentation with prototype alignment. In proceedings of the IEEE/CVF international conference on computer vision, pp 9197-9206.

[12] Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M. (2023). YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7464-7475.

[13] Waranusast, R., Bundon, N., Timtong, V., Tangnoi, C., & Pattanathaburt, P. (2013). Machine vision techniques for motorcycle safety helmet detection. In 2013 28th International conference on image and vision computing New Zealand (IVCNZ), pp 35-40.

[14] Li, Y., Wei, H., Han, Z., Huang, J., & Wang, W. (2020). Deep learning-based safety helmet detection in engineering management based on convolutional neural networks. Advances in Civil Engineering, pp 1-10.

[15] Siebert, F. W., & Lin, H. (2020). Detecting motorcycle helmet use with deep learning. Accident Analysis & Prevention, 134, 105319.

[16] Huang, L., Fu, Q., He, M., Jiang, D., & Hao, Z. (2021). Detection algorithm of safety helmet wearing based on deep learning. Concurrency and Computation: Practice and Experience, 33(13), e6234.

[17] Jia, W., Xu, S., Liang, Z., Zhao, Y., Min, H., Li, S., & Yu, Y. (2021). Real-time automatic helmet detection of motorcyclists in urban traffic using improved YOLOv5 detector. IET Image Processing, 15(14), pp 3623-3637.

[18] Zhao, Z., Guo, G., Zhang, L., & Li, Y. (2023). A new anti-vibration hammer rust detection algorithm based on improved YOLOv7. Energy Reports, 9, pp 345-351.

[19] Di, B., Xiang, L., Daoqing, Y., & Kaimin, P. (2024). MARA-YOLO: An efficient method for multiclass personal protective equipment detection. IEEE Access.

[20] Zhao, L., & Zhu, M. (2023). MS-YOLOv7: YOLOv7 based on multi-scale for object detection on UAV aerial photography. Drones, 7(3), 188.

[21] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, Proceedings, Part V 13, pp 740-755.

[22] Arthur, D., & Vassilvitskii, S. (2007). k-means++: the advantages of careful seeding. ACM-SIAM Symposium on Discrete Algorithms, pp 1027-1035.

[23] Wang, P., Wang, X., Liu, Y., & Song, J. (2024). Research on Road Object Detection Model Based on YOLOv4 of Autonomous Vehicle. IEEE Access.

[24] Vasu, P. K. A., Gabriel, J., Zhu, J., Tuzel, O., & Ranjan, A. (2023). MobileOne: An improved one millisecond mobile backbone. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 7907-7917.

[25] Mijwil, M. M., Doshi, R., Hiran, K. K., Unogwu, O. J., & Bala, I. (2023). MobileNetV1-Based Deep Learning Model for Accurate Brain Tumor Classification. Mesopotamian Journal of Computer Science, pp 32-41.

[26] Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp 7132-7141.

[27] Wang, X., Wang, P., Song, J., Hao, T., & Duan, X. (2023). A TEDE Algorithm Studies the Effect of Dataset Grouping on Supervised Learning Accuracy. Electronics, 12(11), 2546.

[28] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. International journal of computer vision, 88, pp 303-338.

[29] Wang, H., Dong, X., Tian, S., Tie, J., & Xiong, W. (2023). Research on Pedestrian Detection Based on Improved YOLOv7-Tiny Algorithm. In International Conference on Artificial Intelligence in China, pp 373-386.

[30] Hou, Q., Zhou, D., & Feng, J. (2021). Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 13713-13722.