



International Journal of Sensor Networks

ISSN online: 1748-1287 - ISSN print: 1748-1279

<https://www.inderscience.com/ijsnnet>

A road traffic sign recognition method based on improved YOLOv5

Lu Shi, Haifei Zhang

DOI: [10.1504/IJSNET.2024.10066756](https://doi.org/10.1504/IJSNET.2024.10066756)

Article History:

Received:	09 July 2024
Last revised:	16 July 2024
Accepted:	16 July 2024
Published online:	13 January 2025

A road traffic sign recognition method based on improved YOLOv5

Lu Shi and Haifei Zhang*

School of Computer and Information Engineering,
Nantong Institute of Technology,
Nantong 226602, China
Email: 2562917104@qq.com
Email: zhanghf@ntit.edu.cn
*Corresponding author

Abstract: With the rapid development of artificial intelligence technology, the automatic driving of intelligent vehicles has gradually entered people's lives. The traditional vision will fail in many scenarios, such as snow, lane line wear, occlusion, or haze weather. This study aims to provide an accurate and efficient method for recognising traffic signs in their natural surroundings. The advanced convolutional block attention module attention mechanism is embedded in the infrastructure of you only look once version 5 (YOLOv5), which strengthens the network's ability to capture key features of the image. Then, the transformer module is introduced in the core part of YOLOv5, which uses its self-attention mechanism to effectively enhance the overall context connection, thereby significantly enhancing the model's detection accuracy and achieving an impressive 91% high accuracy performance. According to the most recent experimental data, the enhanced YOLOv5 model performs exceptionally well at recognising traffic signs in various natural settings.

Keywords: deep learning; object detection; traffic sign recognition; YOLOv5.

Reference to this paper should be made as follows: Shi, L. and Zhang, H. (2025) 'A road traffic sign recognition method based on improved YOLOv5', *Int. J. Sensor Networks*, Vol. 47, No. 1, pp.47–59.

Biographical notes: Lu Shi is an undergraduate at Nantong Institute of Technology and will start her Master's degree at Yunnan University in 2024. His research interests include deep learning.

Haifei Zhang is a Professor in the School of Computer and Information Engineering at Nantong Institute of Technology. He received his Master's degree from Soochow University in 2005. His research interests include deep learning, deep reinforcement learning, and geographic information systems.

1 Introduction

Traditional environment sensing algorithms frequently fall short of the ideal accuracy and real-time response capabilities standards when dealing with complicated road traffic settings. Detection technology has seen a significant increase in in-depth research in recent years because of the swift advancement of computer technology. Since its inception in 2006, the concept of deep learning has rapidly taken the broad landscape of computer technology by storm, especially represented by convolutional neural networks (CNN) (LeCun et al., 2015), which have led the trend of technological innovation with their outstanding performance. For a long time, low recognition accuracy and lack of real-time implementation have presented challenges for computer vision researchers. However, since the introduction of convolutional neural networks, an effective solution to these problems seems to be found (Shubhashish and Abhimanyu, 2023). CNNs have constantly been evolving with the times, and network models specifically designed for image classification, such as R-CNN

(Region-CNN) (He et al., 2017), visual geometry group (VGG) (Simonyan and Zisserman, 2014), AlexNet (Krizhevsky et al., 2017), etc. have emerged, followed by advanced network architectures such as solid state drive (SSD) (Liu et al., 2016) and you only look once (YOLO) (Redmon et al., 2016) for more efficient target detection.

As deep learning technology advances, more professionals and academics apply their excellent algorithms to identify traffic signs. In Xu (2021), ResNet50-D was introduced by the researchers as the main feature extraction module of you only look once version 3 (YOLOv3), aiming to enhance the model's efficiency in extracting complex features, hence considerably raising the task's accuracy for identifying traffic sign targets. In Liu et al. (2021a), the YOLOv3 core architecture was selected by the researchers, and they creatively optimised the feature extraction and fusion network to improve the detection accuracy of minor traffic signs greatly. Liu et al. (2021b) innovatively combined the spatial pyramid pooling layer (SPP) with an improved version of the YOLOv3 architecture. This version uses the feature pyramid network

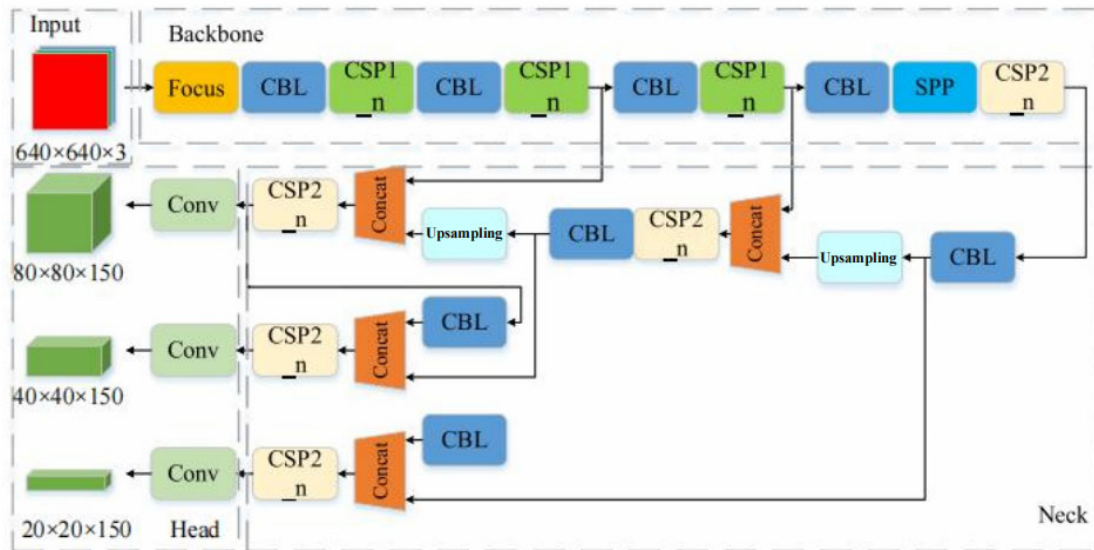
(FPN) to increase target detection accuracy greatly. Zhang et al. (2020) innovatively developed the Cascaded R-CNN algorithm, which demonstrated excellent recognition accuracy of up to 99% when dealing with traffic sign detection in China; however, its relatively slow recognition speed has become an urgent need for optimisation. In Jiang et al. (2020), the authors innovatively incorporated the deep separable convolution technique into the YOLOv3 architecture, which successfully lowers the model's parameter count, thus significantly improving the computational efficiency of detecting traffic signs. In Zuo et al. (2017), the researchers used a modified version of the Faster R-CNN network model dedicated to efficiently and accurately recognising traffic signs. Han (2012) innovates on its base, the well-known you only look once version 2 (YOLOv2) model (Zhang et al., 2017), by introducing pruning techniques to reduce the computational load and storage requirements effectively. It implements gridding on the input images, an improvement that significantly increases the algorithm's ability to recognise tiny traffic indicators. In 2020, the you only look once version 4 (YOLOv4) algorithm (Bochkovskiy et al., 2020) surpassed many other detection algorithms under its significant advantages, both in accuracy and speed of detection. Specifically, the YOLOv4 model selected the cutting-edge cross stage partial Darknet (CSPDarknet53) architecture as its central component during the feature extraction stage. To increase the feature fusion's efficacy, the model discarded the traditional feature pyramid network (FPNet) (Liu et al., 2018) approach and instead introduced the more effective path aggregation network (PANet) (Lin et al., 2017) strategy, a creative design that greatly enhances the accuracy and expressiveness of the model.

In contrast to the YOLOv3 algorithm (Redmon and Farhadi, 2018), although our new method achieves a notable increase in detection precision, it requires a correspondingly higher configuration of computer hardware. Although the YOLOv4 algorithm excels in target detection performance,

it is limited by its high demand for hardware resources and its operational efficiency, making it unsuitable for traffic sign detection, which has a stringent need for real-time performance. You only look once version 5 (YOLOv5), one of the current outstanding object detection models, can be comparable to the previous YOLOv4. Although it is not inferior in performance, it is relatively more approachable regarding computational device requirements, especially for applications with high real-time performance demands. YOLOv5 demonstrates significant advantages, which has made it highly favoured in the market (Zeng et al., 2022).

Aiming at the problem that the current vehicle detection algorithm has poor perception ability, lacks petiteness, and is prone to false detection and missing detection for small targets on the traffic road, this paper proposes an improvement based on the YOLOv5 algorithm. The deep neural network architecture of YOLOv5 and its optimisation principles are thoroughly analysed. Then, the impact of two different attention mechanisms on model performance is highlighted and compared. An enhanced convolutional block attention module (CBAM) attention module is incorporated into the core layer of the YOLOv5 deep neural network architecture to boost the model's capacity to focus on the important aspects of the input image, ultimately leading to an increase in recognition accuracy. To tackle the challenging issue of small target recognition, we creatively integrate the transformer module into the foundational architecture of YOLOv5. Through its self-attention mechanism, this module effectively improves the model's overall scene understanding, leading to a notable increase in recognition accuracy and precision. The enhanced intersection over union loss function soft intersection over union (SIoU) is used to replace the complete intersection over union (CIoU), which improves the positioning accuracy of the bounding box and improves the feature extraction ability of the network.

Figure 1 The model of signal mixing and separation (see online version for colours)



The contributions of this paper are summarised as follows: compared with the original YOLOv5 algorithm, the precision rate is increased by 17.7%, the recall rate is increased by 28.9%, and the map is increased by 7.8%. The experimental results show that the proposed method has good detection performance. It has a good improvement in the detection accuracy of small road targets.

The rest of the paper is organised as follows: in the first section, the background and motivation of the research are elaborated, and the work and contributions of this paper are summarised. The second part deeply discusses the theoretical basis, and the related technology is introduced in detail. In the third part, the performance of the YOLOv5 algorithm in traffic sign detection is optimised and improved. In the fourth part, the detailed experimental construction and the in-depth analysis of the results are carried out, and in the fifth section, we conclude the paper.

2 Related work

2.1 YOLOv5 network structure

Figure 1 depicts the network topology of the YOLOv5 algorithm. The following will explain each of the four components:

1 Input

To enhance the identification of tiny targets, the YOLOv5 algorithm will screen the dataset and data enhancement operations before the training data into the network. YOLOv5 uses Mosaic for data enhancement.

2 Feature extraction backbone network

YOLOv5's core network comprises downsampled focus, convolution module cross stage partial network (CSPNet), and downsampled convolutional block layer (CBL) convolution. In deep learning, the focus downsampling layer serves as a key component whose core function is to perform preliminary low-resolution processing of the input image, which is essentially a chunking or slicing operation that refines the main features of the image by reducing the pixel density.

3 Feature enhanced fusion network

The YOLOv5 neck module combines the functionality of the path aggregation network and the feature pyramid network (Liu et al., 2018). FPN achieves the effective integration of high-level semantic information and low-level specific features at the PANet and adopts a bottom-up strategy to effectively reduce the distance between the shallow and deep networks, thus enhancing the semantic understanding of the higher layers in the feature map with the underlying edge and colour details, colour features, and spatial pyramid pooling (SPP) structure.

4 Detection head

The head layer of YOLOv5 is a key component specifically responsible for obtaining three feature maps from the feature enhancement fusion network at various spatial sizes and performing efficient target detection and accurate category recognition.

2.2 YOLOv5 loss function

During training, the loss function is a crucial tool for determining how similar the predicted and expected values are. In YOLOv5, the loss function comprises three parts: the target confidence loss L_{ij}^{conf} , the classification loss L_{ij}^{cls} , and the target coordinate frame regression loss L_{ij}^{reg} , and its total loss is weighted and summed by these three components, with the expression shown in equation (1):

$$Loss = \lambda_1 L_{ij}^{conf} + \lambda_2 L_{ij}^{cls} + \lambda_3 L_{ij}^{reg} \quad (1)$$

Binary cross entropy is used to determine L^{conf} and L^{cls} , where the weights of target confidence loss, target classification loss, and target regression loss are λ_1 , λ_2 and λ_3 , respectively. CIOU is used to calculate L^{cls} (Zheng et al., 2020). The bounding box coordinates regression loss computation technique, or CIOU is demonstrated in (2).

$$L_{CIOU} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + av \quad (2)$$

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (3)$$

$$a = \frac{v}{(1 - IoU) + v} \quad (4)$$

$$v = \frac{4}{\pi^2} \left(\arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2 \quad (5)$$

CIOU considers several variables, such as the area, centre distance, and width-to-height ratio of the real and predicted frames, to more accurately describe their positional relationship. In equation (2), the real centre is denoted by b^{gt} , and the predicted centre is denoted by b , c represents the minimum outer rectangle's diagonal distance, while ρ indicates the separation between b^{gt} and b . In equation (3), the intended frame is represented by B , and the true frame by A . The intersection over union (IoU) calculates the ratio of the two frames' intersection and concatenation. The width and height of the expected frame are represented by w and h in equation (5), whereas the desired frame's width and height are represented by w^{gt} and h^{gt} .

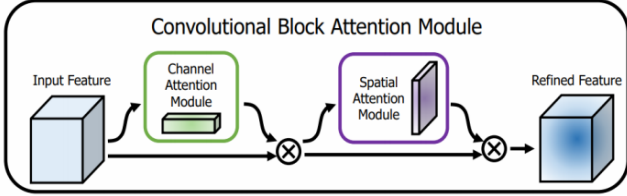
3 YOLOv5 network improvements

3.1 Introduction of the CBAM attention mechanism

CBAM attention mechanism is a lightweight module that can be inserted into the YOLOv5 network. It comprises two modules: the channel attention module (CAM) and the

spatial attention module (SAM), which reinforce the desired features on the channel and spatial dimensions, respectively. Then, the input feature maps are multiplied by the weights that the attention mechanism determined to produce a feature reinforcement (Woo et al., 2018). Figure 2 depicts the CBAM attention module; Figure 3 shows the channel attention module's structure; and Figure 4 shows the spatial attention module's structure.

Figure 2 CBAM attention module (see online version for colours)



Looking at Figure 2 in detail, the channel attention mechanism fine-tunes the input feature map to fuse the information through weighted multiplication, and through the spatial attention mechanism, this processed feature map further interacts with the M_c feature map, again using weighted multiplication, to enhance its representation in the spatial dimension. Ultimately, this dual-view fusion strategy ensures that the generated feature maps are deep in channel-level understanding and rich in spatial relational insights. To compute the enhanced feature map, the channel attention method is first applied to every channel in the feature map M_c input, as shown in Equation (6), and by summing M_c with the weights of the initial input feature map, a new feature map is produced; this new feature map is sent to the spatial attention channel for processing, and further transformed into the feature map M_s , as shown in equation (7). Ultimately, the final feature map F'' is created by fusing and weighting M_s with the prior feature map F' .

$$F' = M_c(F) \otimes F \quad (6)$$

$$F'' = M_c(F') \otimes F' \quad (7)$$

Figure 3 Structure of channel attention module (see online version for colours)

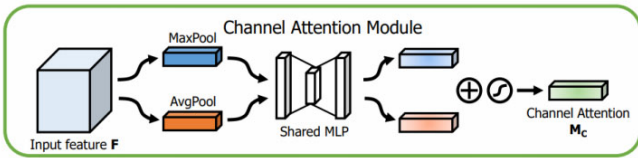


Figure 5 Structure of the backbone network after the addition of CBAM to YOLOv5



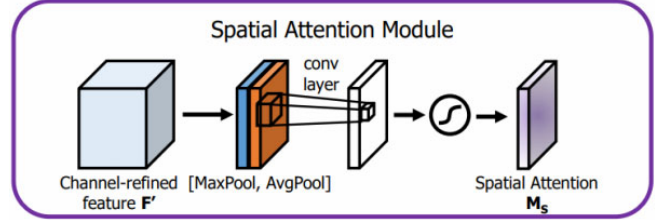
The details of the construction of the channel attention module are shown in detail in Figure 3: a careful feature extraction is implemented on the input feature map F by carrying out operations for average and maximum pooling, accordingly. The results of these two processes are fed into a multilayer perceptron (MLP) convolutional neural network for further processing, resulting in two unique feature maps. After their summation operation, the final channel attention is produced by introducing a sigmoid activation function, i.e., M_c , computed as shown in equation (8), which is extremely critical and insightful.

$$M_c(F) = \sigma(MLP(AvgPool(F)) + MLP(MaxPool(F))) \quad (8)$$

$$= \sigma(W_1(W_0(F_{avg}^c)) + W_1(W_0(F_{max}^c)))$$

where W_0 and W_1 represent the weight information in the MLP, and F_{avg}^c and F_{max}^c stand for the average pooling and maximum pooling operations, appropriately.

Figure 4 Structure diagram of the spatial attention module (see online version for colours)



The spatial attention module (SAM) development is readily apparent, as Figure 4 illustrates. Taking the channel attention mechanism (CAM) output feature map F' as input, the process is similar to that of the CAM module, which processes the feature maps using average and maximum pooling, each transformed into $H \times W \times 1$ dimensions, and the concat function merges these two kinds of feature maps in channel dimensions. The feature maps are compressed to a single-channel output through a 7×7 convolutional layer processing. Ultimately, the feature maps M_s are obtained by applying the sigmoid activation function processing; equation (9) displays their mathematical expression.

$$M_s(F) = \sigma(f^{7 \times 7}([AvgPool(F); MaxPool(F)])) \quad (9)$$

$$= \sigma(f^{7 \times 7}([F_{avg}^s; F_{max}^s]))$$

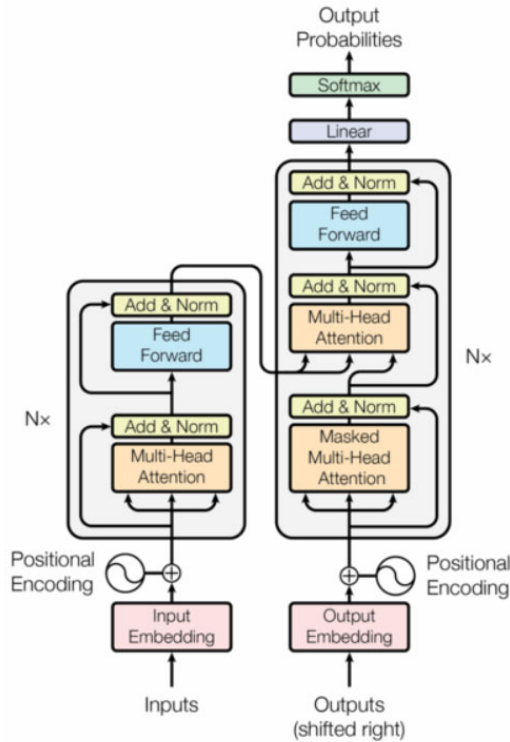
where f is the 7×7 convolutional kernel, where σ is the sigmoid activation function, and F_{avg}^s and F_{max}^s represent the maximum and average pooling operations, respectively.

Implementing CBAM modules in different parts of the backbone network brings about varying impacts. Experimental data shows that incorporating CBAM modules into this core structure can greatly improve the model's feature mining performance, enhancing overall prediction accuracy (Li et al., 2022). After integrating the sophisticated CBAM module into the YOLOv5 network architecture, its backbone network architecture is presented in Figure 5, demonstrating the enhanced features.

3.2 Introducing the transformer module

Transformer is a cutting-edge deep learning model with a special attention mechanism that greatly raises the model's training effectiveness (Asiri et al., 2024). In essence, the transformer model is a deep learning architecture that only uses a self-attention system and is uniquely designed to enable efficient parallelisation of processing. Thanks to its complex intrinsic structure, transformer significantly outperforms the previously widely used RNN recurrent neural networks regarding precision and effectiveness of execution, which makes it stand out in a wide range of tasks. Figure 6 shows an exhaustive schematic of the complete structure of the transformer.

Figure 6 Overall architecture of the transformer (see online version for colours)



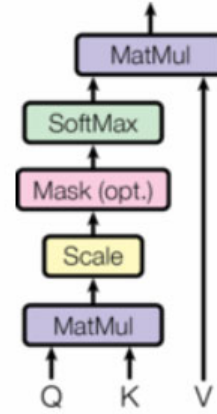
It consists of two primary parts: the encoder and the decoder.

The encoder module in the classic transformer structure contains six encoder blocks, each of which has two sub-modules: the feedforward fully connected layer and the multi-head self-attention layer. Using a scaled dot-product attention computation approach, the multi-head

self-attention layer can extract features from many heads at a higher level of detail than a single head, as demonstrated by the experimental findings. The feed-forward fully connected layer consists of two fully connected layers with the addition of a Relu activation function in the middle of the linear transformation. The specific dimension follows a 4-fold relationship, meaning that the transformation within the layer has a dimension of $d_{ff} = 2,048$, and the multi-head self-attention model's d_{model} is 512.

Firstly, the computation rule of self-attention is shown in Figure 7.

Figure 7 Calculation rules for self-attention (see online version for colours)



Regarding the attentions above, they can be conceptualised as producing an output computed as a weighted sum of values from a set of query and key-value pairings, with each value's weight determined by the query's matched key and its similarity function. Scaled dot-product attention is the term for this kind of attention, and equation (10) below is the related mathematical expression.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (10)$$

Q , K and V are first mapped using the parameter matrix in the so-called multi-head self-attention layer, after which self-attention is applied, and the outcome is then spliced into a completely linked layer as in Figure 8.

The above-mentioned multiple self-attention corresponds to a mathematical formula in equation (11) below.

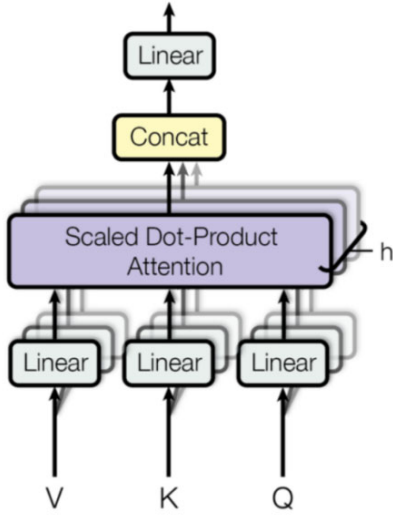
$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W^o \quad (11)$$

where $head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$

$$W_i^Q \in R^{d_{model} \times d_k}, W_i^K \in R^{d_{model} \times d_k},$$

$$W_i^V \in R^{d_{model} \times d_v}, W_i^O \in R^{d_{model} \times h d_v}$$

The role of the multi-head self-attention layer: experimental results show that better feature extraction is possible with multiple heads since they can extract the features of each head more thoroughly while requiring the same amount of processing power overall as a single head.

Figure 8 Calculation rules for multi-head self-attention (see online version for colours)

The two linear transformations that make up the feed-forward fully connected layer module are divided by an activation function of Relu; the corresponding mathematical formulas are the following [equation (12)].

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (12)$$

The input and output dimensions ($d_{model} = 512$) and the connection dimension ($d_{ff} = 2,048$) within the layer employ a 4-fold size relation. The feed-forward fully connected layer's function: the fully linked layer is introduced to the network to increase its capabilities because the basic multi-attention technique is insufficient to extract the required features.

The decoder module in the classic transformer architecture consists of six decoder blocks, and each decoder block contains three sub-blocks, which are the multihead self-attention layer, encoder-decoder attention layer, and feedforward fully connected layer. The encoder-decoder attention layer adopts the same scaled dot-product attention computation as the encoder module. The biggest difference is that it needs to add a look-ahead mask, i.e., 'masking' future information'. The primary distinction between the encoder-decoder attention layer and its predecessor is the encoder-decoder attention layer. The main difference between the encoder-decoder attention layer and the previous multi-head self-attention layer is that $Q \neq K = V$, and the matrix Q is derived from the previous decoder block's output. In contrast, K and V are obtained from the encoder side's output. The encoder's feedforward fully linked layer is identical to this one.

The add and norm modules are attached to each encoder block and each sub-layer of the decoder block. Each encoder block has add and norm after two sub-layers; for each decoder block, there are add and norm after three sub-layers. Add stands for redundant connection, which improves the model's fit by transferring information deeper without any loss. Norm means LayerNorm, which is a numerical normalisation of the layer level. Norm stands for LayerNorm, a numerical standardisation operation at the

layer level, which prevents the learning process from being abnormal and the model from converging too slowly due to too large or too small parameters.

Transformer positional encoding: the positional information is directly encoded using the sine and cosine functions in equations (13) and (14).

$$PE_{(pos, 2i)} = \sin(pos/10,000^{2i/d_{model}}) \quad (13)$$

$$PE_{(pos, 2i+1)} = \cos(pos/10,000^{2i/d_{model}}) \quad (14)$$

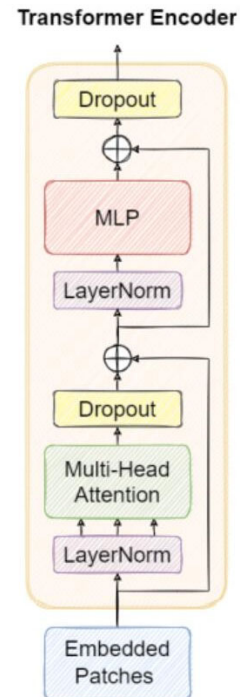
An important advantage of the trigonometric functions applied here is that for any $PE(pos + k)$, the expression for it is a linear function of $PE(pos)$, which greatly facilitates the computation. Moreover, the sequence length does not limit the periodic function, it may also improve the model's capacity for generalisation. The following equations (15) and (16).

$$\sin(\alpha + \beta) = \sin(\alpha)\cos(\beta) + \cos(\alpha)\sin(\beta) \quad (15)$$

$$\cos(\alpha + \beta) = \cos(\alpha)\cos(\beta) - \sin(\alpha)\sin(\beta) \quad (16)$$

Applying transformer prediction head (TPH) on low resolution feature maps can reduce the computational and storage expenses because the feature maps at the network's end are low resolution. The TPH and backbone end are created using the Transformer encoder block based on YOLOv5. In addition, some TPH blocks from earlier layers can be optionally removed when scaling up the resolution of the input image to make the training process usable.

Certain convolutional and cross stage partial (CSP) structures are replaced with transformer encoders employed in vision. The transformer offers a superior detection effect and a distinctive attention mechanism.

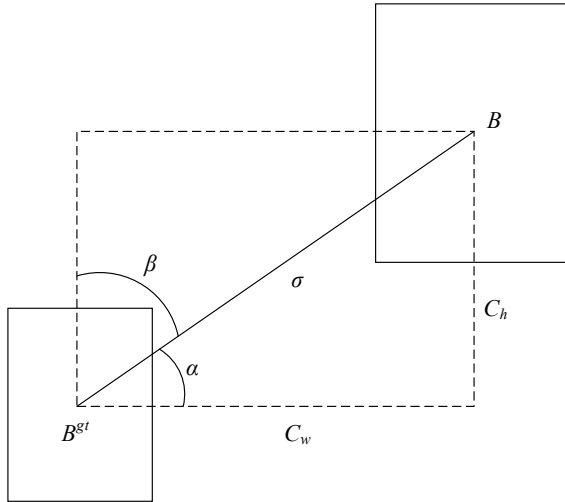
Figure 9 Module structure diagram of YOLOv5 after adding transformer (see online version for colours)

3.3 Loss function optimisation

The YOLOv5 model was originally based on the CIoU algorithm for accurate regression of the anchor frames of the target objects; however, in this paper, SIOU (Albawi et al., 2017) (minimum enveloping IoU) is innovatively introduced as an alternative to optimising the regression loss computation process, which particularly enhances the sensitivity and accuracy of the identification of tiny targets. The CIoU loss function is a novel optimisation technique that considers the area of overlap between the anticipated anchor frame and the real target frame, as well as the aspect ratio matching and centroid accuracy. In target identification tasks, CIoU demonstrates notable improvements over the typical IoU loss, particularly in big target recognition. However, for small targets, the relatively small change in aspect ratio may lead to slower convergence. To address this problem, the researchers introduced SIOU, which is a composite loss function that includes four components, namely angle (Anglecost), distance (Distancecost), shape (Shapecost), and IoU, which guarantees the accuracy as well as accelerates the optimisation process of the small target prediction frame.

The fundamental concept of the SIOU loss function is angle loss (Chen et al., 2024), which is to achieve the part of the prediction frame that can be in the same direction as the real frame faster, and the angle loss calculation scheme is displayed in Figure 10.

Figure 10 Schematic diagram of the calculation scheme of the angular loss function



As seen in Figure 10, the angles of the centre points of the real frame and predicted frames' centre points are α and β , respectively. The goal of the angular loss is to move the anticipated frame's centre to the x or y axis in the coordinate system where it is placed that is closest to the real frame's centre point and to reduce the angular direction of the real frame and the predicted frame by constant iterative computation. The convergence process tries to minimise α if $\alpha \leq \pi/4$, otherwise minimise $\beta = \pi/2 - \alpha$. The angular loss function is shown in equation (17).

$$LF = 1 - 2 * \sin^2 \left(\arcsin(x) - \frac{\pi}{4} \right) \quad (17)$$

$$x = \frac{C_h}{\sigma} = \sin(\alpha) \quad (18)$$

$$\sigma = \sqrt{(b_{cx}^{gt} - b_{cx})^2 + (b_{cy}^{gt} - b_{cy})^2} \quad (19)$$

$$c_h = \max(b_{cy}^{gt}, b_{cy}) - \min(b_{cy}^{gt}, b_{cy}) \quad (20)$$

where x is the sine of angle α as in equation (18). The centre distance, or σ , between the actual and forecasted frames, is determined using equation (19). Equation (20) can compute the perpendicular distance, or c_h , between the real frame and the anticipated frame's centre point. After combining the angular loss function, equation (21) estimates the distance loss between the two frames:

$$\Delta = \sum_{t=x,y} (1 - e^{-\gamma \rho_t}) \quad (21)$$

where $\gamma = 2 - LF$, as a measure of the distance function between the true and anticipated frames, ρ_t is the ratio of the width and height of the smallest outer rectangle in the true and predicted frames to the horizontal and vertical distance from the centre point. From this, it can be concluded that as α tends to 0, the loss increases with angle, and the distance loss's contribution to the overall loss is significantly diminished. Equation (22) defines the form loss.

$$\Omega = \sum_{t=w,h} (1 - e^{-\omega_t})^\theta \quad (22)$$

where ω_h is the ratio of the total amount that differs between the height h^{gt} of the true frame and the height h of the predicted frame to $\max(h^{gt}, h)$, and ω_w is the ratio of the total amount that differs between the width w^{gt} of the true frame and the width w of the predicted frame to $\max(w^{gt}, w)$. θ , on the other hand, is the cost of the shape loss, a value used for adjusting how much attention is given to the shape's cost, usually defined as 2–6. The SIOU loss function combines the four parts of the loss calculation and is defined as equation (23) below:

$$L = 1 - IoU + \frac{\Delta + \Omega}{2} \quad (23)$$

4 Experimental results and analysis

4.1 Introduction and processing of the dataset

This paper's experimental section uses the latest CCTSDB 2021 dataset, composed of multiple parts, including a training set, test set, classification test set, and negative samples. In the training and testing part, the training and testing sets contain 17,856 images and annotation information in total. Among them, some of the training set's images have changed the lateral size, added salt and pepper noise, adjusted brightness, etc., on the original image, and the images at the bottom of the sequence number are

obtained by extracting frames from driving videos, and the scene repeatability is large. The labelling information is divided into three categories: prohibitory, warning, and mandatory.

Almost 40,000 traffic signs are included in the almost 20,000 sample photographs of traffic signs. At the same time, it contains image datasets classified by size and scene, which contain rich road background information. The source of the image is collected from a normal driving road in China, and the shooting angle is mostly from street photography and the driving recorder in the car, which is basically from the perspective of the driver and co-driver positions in the car. The dataset's image summary is shown in Figure 11.

Figure 11 CCTSDB traffic dataset (see online version for colours)

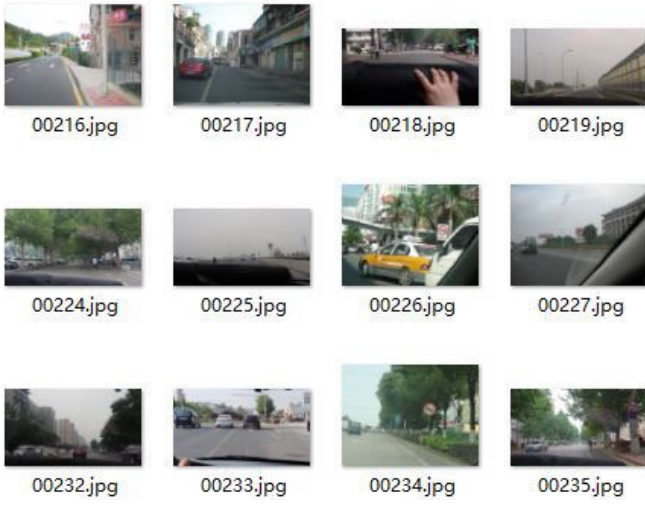
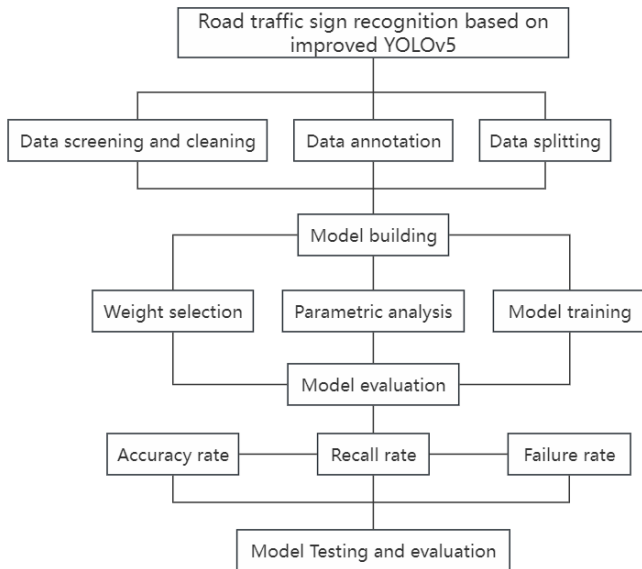


Figure 12 Data processing flowchart



In this paper, icons of the CCTSDB2021 dataset are used to filter and sort out, and finally, various traffic sign data in the data are obtained. Data cleaning and labelling work is then performed to split the dataset into training, validation, and test sets. Finally, the framework of YOLOv5 is used for

deep learning training to generate the best weight file, and the test set is used for inference tests. The steps are as follows:

- *Data preprocessing*: the CCTSDB2021 dataset is screened and sorted, various traffic sign data are selected as the research object, and the dataset is divided into training set, validation set, and test set.
- *Data annotation*: a labelling annotation tool was used to annotate the dataset.
- *Model training*: the framework of YOLOv5 is used for deep learning training, the appropriate pre-training weights are selected, the corresponding parameters are set, and the best weight file is generated.
- *Model testing*: use the test set for inference testing. You can test and validate the model by opening the camera, video, and pictures.

4.2 Experimental environment

The deep learning framework, the Windows 10 operating system, and a few hyperparameters put on the network during the training phase are displayed in Table 1 as the experimental environment in this research.

Table 1 Some hyperparameter settings

Training parameters	Parameter value
Momentum	0.937
Weight decay	0.0005
Initial learning rate	0.01
Loop learning rate	0.2
Batch size	16
Training algebra	300

4.3 Assessment of indicators

This experiment uses the accuracy, recall, and mean average precision (mAP) evaluation indices to evaluate this network model's performance unbiasedly. Among these, the average of all categories' prediction precision is known as mean average precision. Recall is the percentage of all real targets that were accurately forecasted, and precision is the percentage of correct predictions among the targets predicted by the model. Equations (24)–(26) give the precise calculation formula.

$$Precision = \frac{TP}{TP + FP} \quad (24)$$

$$Recall = \frac{TP}{TP + FN} \quad (25)$$

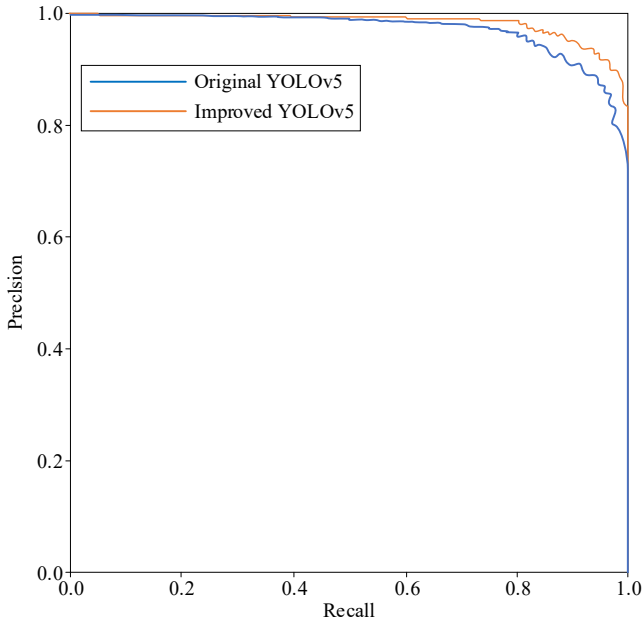
$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (26)$$

where AP is computed by splitting the sum of all the accuracies of a particular category in the set by the number of images containing targets of that category, TP stands for the number of positive samples that the model anticipated with accuracy, FN is the number of negatively predicted samples by the model, and FP denotes the number of positively predicted samples by the model.

4.4 Comparison experiments with the YOLOv5 model

This section compares the YOLOv5 model with the enhanced YOLOv5 model using the CCTSDB 2021 dataset. It compares the experimental findings with this paper's 300 batches of training data. Figure 12 displays the comparison of PR curves before and after YOLOv5 improvement.

Figure 13 Comparison of PR curves before and after YOLOv5 improvement (see online version for colours)



The PR curve is also a parameter index to judge the model's goodness. Its horizontal coordinate is recall, its vertical coordinate is precision, and the region's size determines the AP value; the greater the area, the stronger the model's shown performance. Figure 12 compares the PR curve before and following YOLOv5's improvement, which shows that the PR curve of the improved YOLOv5 is more to the right, and the AP value is bigger, which can be concluded that the suggested enhancement strategy works. We can conclude that the performance of the upgraded YOLOv5 is superior to that of YOLOv5 and that the improvement strategy suggested in this paper is effective.

This work proposes experiments to compare the detection results in various traffic circumstances to more intuitively reflect the enhanced model's recognition ability for traffic signs. Figure 13–15 displays the detection results. The improved model is on the right, and the YOLOv5 model is on the left.

4.5 Ablation experiments

We implemented a series of meticulous ablation experiments to investigate the precise effects of different enhancement techniques on the model recognition performance. Firstly, we added the CBAM attention mechanism to the base model YOLOv5. It helps the network concentrate more on the picture's salient characteristics and improves its localisation capacity. Secondly, we replace the original CIoU loss function with an optimised SIoU loss function, increasing tiny targets' detection efficiency. Finally, we introduce the transformer module, which effectively enhances the model's understanding of global contextual information under its self-attention mechanism, thereby significantly improving the overall detection accuracy. These targeted improvement strategies reveal the multifaceted paths to enhance the model's performance.

Figure 14 Comparison of model checking results under low-light conditions, (a) YOLOv5 detection map (b) detection map of the improved algorithm (see online version for colours)



(a)



(b)

Figure 15 Comparison of model checking results under well-lit conditions, (a) YOLOv5 detection map (b) detection map of the improved algorithm (see online version for colours)**Figure 16** Comparison of model checking results under haze conditions, (a) YOLOv5 detection map (b) detection map of the improved algorithm (see online version for colours)**Table 2** Comparison of ablation experiments

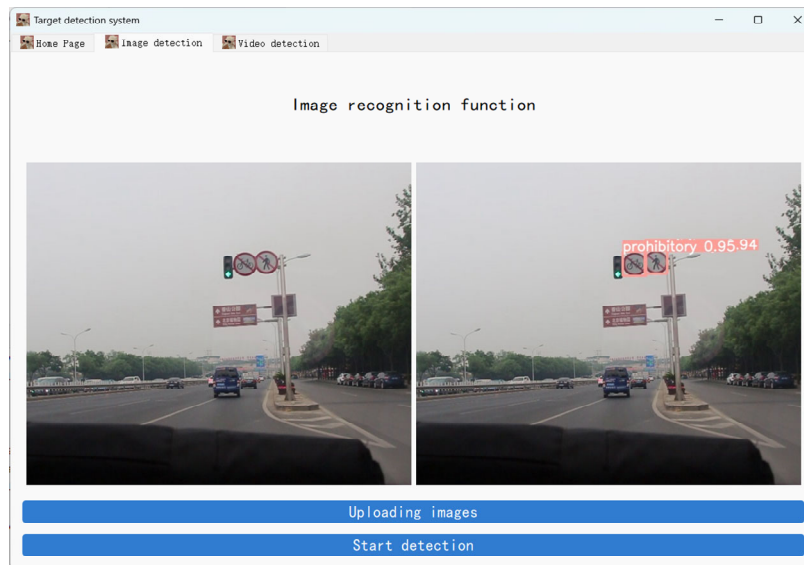
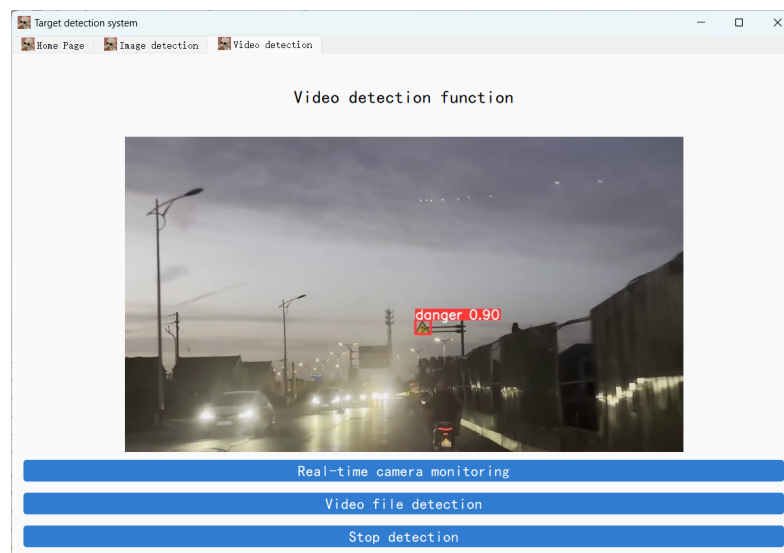
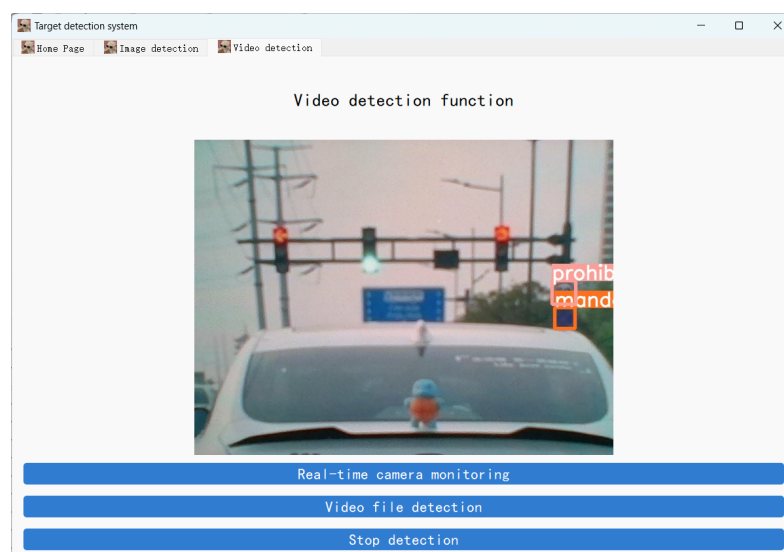
<i>Model</i>	<i>Precision (%)</i>	<i>Recall (%)</i>	<i>mAP (%)</i>
YOLOv5	56.5	38.5	83.3
YOLOv5+CBAM	63.3	49.8	86.1
YOLOv5+SIOU	61.6	48.7	85.2
YOLOv5+transformer	63.9	50.3	86.5
Improved YOLOv5	74.2	67.4	91.1
The performance difference between improved YOLOv5 and YOLOv5	17.7	28.9	7.8

Table 2 shows that the trained model performs effectively. Compared to the original YOLOv5 algorithm, the modified YOLOv5 approach in this paper improves by 17.7% in precision, 28.9% in recall, and 7.8% in map.

4.6 Function test of the system

To evaluate the image detection performance of the traffic sign detection system, the user first clicks on the image detection interactive interface and then uploads a picture. The system will pop up the preset file browsing options, as shown in Figure 16. The user needs to select one of them, and the picture will be uploaded automatically for detection and processing after confirmation. The detailed result output area displays the detection results, showing high accuracy.

To evaluate the video detection performance of the traffic sign detection system, the user first needs to activate the function by clicking the corresponding button. Then, the user must select a target video from the preset video folder and click OK. The system will automatically read and process it. After the detection process is completed, all the detection results will be displayed in the output result display area for users to consult. Please refer to Figure 17 for the visual effects after related operations.

Figure 17 Display of image detection function (see online version for colours)**Figure 18** Display of image detection function (see online version for colours)**Figure 19** Display of image detection function (see online version for colours)

In vehicle driving, the camera detection module plays a key role. Drivers often turn this feature on when they want to see and understand the real-time images captured by the camera. This function is the most frequently used by them while driving because by viewing traffic signs in real-time and feeding relevant information back to them, this system effectively assists drivers in understanding the road conditions, making informed and safe decisions, and ensuring the smooth running of the driving process. When real-time video monitoring is unnecessary, the user can choose the pause detection independently. The visual display of the subsequent operation is clearly shown in Figure 18.

5 Conclusions

To enhance the precision of YOLOv5 in traffic sign recognition and reduce the sensitivity to external factors, we propose the following improvement strategies: improve its capacity for feature extraction by integrating the CBAM attention mechanism into the fundamental network design. This will enable the model to concentrate precisely on important feature regions within the image. YOLOv5's core architecture is enhanced by our creative integration of the transformer module to tackle the intricacy of detecting tiny targets and use its self-attention mechanism to strengthen the global understanding of the overall scene, thus significantly improving the model's accuracy in recognition. Following experimental validation, the optimised algorithm outperforms the original approach in traffic sign recognition accuracy, showing improvements of 17.7% in precision, 28.9% in recall, and 7.8% in map.

Acknowledgements

This work was supported by:

- 1 Training Project of Top Scientific Research Talents of Nantong Institute of Technology (XBJRC2021005)
- 2 the Universities Natural Science Research Projects of Jiangsu Province (21KJD210004, 22KJB520032, 22KJD520007, 22KJD520008, 22KJD520009, 23KJD520010 and 23KJD520011)
- 3 Natural Science Foundation of Nantong City (JC2023075, JC2023023)
- 4 Nantong Key Laboratory of Virtual Reality and Cloud Computing (CP2021001)
- 5 Construction Project of Electronic Information Master's Program at Nantong Institute of Technology (879002)
- 6 Key Discipline Construction Project of Software Engineering at Nantong Institute of Technology (879005).

References

- Albawi, S., Mohammed, T.A. and Al-Zawi, S. (2017) 'Understanding of a convolutional neural network', *2017 International Conference on Engineering and Technology (ICET)*, pp.1–6, DOI: 10.1109/ICEngTechnol.2017.8308186.
- Asiri, S, Xiao, Y. and Li, T. (2024) 'PhishTransformer: a novel approach to detect phishing attacks using URL collection and transformer', *Electronics*, Vol. 13, No. 1, p.30, 2024, DOI: 10.3390/electronics13010030.
- Bochkovskiy, A., Wang, C.Y. and Liao, H.Y.M. (2020) 'YOLOv4: optimal speed and accuracy of object detection', *Computer Science*, Vol. 9, No. 2, pp.148–159 [online] <https://arxiv.org/abs/2004.10934> (accessed 31 December 2023).
- Chen, W. et al. (2024) 'A rotatable battery recognition method based on improved YOLOv5', *International Journal of Sensor Networks*, Vol. 45, No. 2, pp.101–114, DOI: 10.1504/ICENET.2024.138913.
- Han, Y.L. (2012) *Research on Road Traffic Signs Recognition Based on Support Vector Machines*, Master thesis, pp.43–47, Dalian University of Technology, China, DOI: CNKI:CDMD:2.1012.394758.
- He, K. et al. (2017) 'Mask R-CNN', *Proceedings of the IEEE International Conference on Computer Vision*, pp.2961–2969, DOI: 10.1109/ICCV.2017.322.
- Jiang, J.H. et al. (2020) 'Improved traffic sign recognition algorithm based on YOLO v3 algorithm', *Journal of Computer Applications (Chinese Journal)*, Vol. 40, No. 8, pp.2472–2478, DOI: 10.11772/j.issn.1001-9081.2020010062.
- Krizhevsky, A., Sutskever, I., Hinton, G.E. (2017) 'ImageNet classification with deep convolutional neural networks', *Communications of the ACM*, Vol. 60, No. 6, pp.84–90, DOI: 10.1145/3065386.
- LeCun, Y., Bengio, Y., Hinton, G. (2015) 'Deep learning', *Nature*, Vol. 521, No 7553, pp.436–444, DOI: 10.1038/nature14539.
- Li, Y.Q. et al. (2022) 'Traffic sign recognition based on attention mechanism', *Electronic Measurement Technology (Chinese Journal)*, Vol. 45, No 8, pp.116–120, DOI: 10.19651/j.cnki.emt.2108743.
- Lin, T. et al. (2017) 'Feature pyramid networks for object detection', *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.578–593, DOI: 10.1109/CVPR.2017.106.
- Liu, S. et al. (2018) 'Path aggregation network for instance segmentation', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.8759–8768, DOI: 10.1109/CVPR.2018.00913.
- Liu, S. et al. (2021a) 'Traffic sign detection based on multi-scale feature fusion network', *Computer Applications and Software (Chinese Journal)*, Vol. 38, No. 2, pp.158–164, 249, DOI: 10.3969/j.issn.1000-386x.2021.02.027.
- Liu, W. et al. (2016) 'SSD: single shot multibox detector', *European Conference on Computer Vision*, pp.21–37, <https://doi.org/10.48550/arXiv.1512.02325>.
- Liu, Z.Y. et al. (2021b) 'YOLOv3 traffic sign detection based on SPP and Improved FPN', *Computer Engineering and Applications (Chinese Journal)*, Vol. 57, No. 7, pp.164–170, DOI: 10.3778/j.issn.1002-8331.2007-0117.
- Redmon, J. and Farhadi, A. (2018) 'YOLOv3: an incremental improvement', *International Journal of Computer Vision*, pp.1804–1815, <https://doi.org/10.48550/arXiv.1804.02767>.

- Redmon, J. et al. (2016) 'You only look once: unified, real-time object detection', *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.779–788, DOI: 10.1109/CVPR.2016.91.
- Shubhashish, G. and Abhimanyu, K. (2023) 'A transformative deep learning framework for traffic modelling using sensors-based multi-resolution traffic data', *International Journal of Sensor Networks*, Vol. 42, No. 3, pp.145–155, DOI: 10.1504/ICENET.2023.132541.
- Simonyan, I.K. and Zisserman, A. (2014) 'Very deep convolutional networks for large-scale image recognition', *Computer Science*, Vol. 78, No. 7, pp.1747–1763. <https://doi.org/10.48550/arXiv.1409.1556>.
- Woo, S. et al. (2018) 'CBAM: convolutional block attention module', *Proceedings of the European Conference on Computer Vision*, pp.146–157, DOI: 10.1007/978-3-030-01234-2_1.
- Xu, Y.C. (2021) 'Improved traffic sign recognition algorithm based on YOLOv3 algorithm', *Digital Technology & Application (Chinese Journal)*, Vol. 39, No. 1, pp.108–111, 116, DOI: 10.19695/j.cnki.cn12-1369.2021.01.34.
- Zeng, Y.Y. et al. (2022) 'Research on small target recognition model based on improved YOLOv5', *Journal of Research in Science and Engineering*, Vol. 4, No. 11, pp.85–94, DOI: 10.53469/jrse.2022.04(11).06.
- Zhang, J. et al. (2017) 'A real-time chinese traffic sign detection algorithm based on modified YOLOv2', *Algorithms*, Vol. 10, No. 4, pp.127–140, DOI: 10.3390/a10040127.
- Zhang, J.M. et al. (2020) 'A cascaded R-CNN with multiscale attention and imbalanced samples for traffic sign detection', *IEEE Access*, Vol. 8, pp.29742–29754, DOI: 10.1109/access.2020.2972338.
- Zheng, Z. et al. (2020) 'Distance-IoU loss: faster and better learning for bounding box regression', *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34, No. 7, pp.12993–13000. <https://doi.org/10.1609/aaai.v34i07.6999>.
- Zuo, Z., Yu, K., Zhou, Q. et al. (2017) 'Traffic signs detection based on Faster R-CNN', *IEEE 37th International Conference on Distributed Computing Systems Workshops*, pp.286–288, DOI: 10.1109/ICDCSW.2017.34.