ISSN: 2664-9640

GhostNet-based Lightweight YOLOv4 Mask Wear Detection Model

Bo He, Jiaoqiu Shi

School of Computer Science and Engineering, Chongqing University of Technology, Chongqing, 400054, China

Abstract

Aiming at the problems of large and complex network structure, many computational parameters and slow detection speed in the mask wear detection model, Proposing a lightweight GhostNet-based mask wear detection model for YOLOv4 (GN-YOLOv4). Based on the original YOLOv4 model, the original backbone feature extraction network was first replaced with GhostNet to reduce the number of parameters of the network model, reduce the computational complexity and improve the detection speed of the model. Then the PANet structure of the feature fusion part of the YOLOv4 network is improved to further reduce the number of parameters of the network model. Finally, a strategy of fusing the batch normalization layer with the convolutional layer was adopted to further reduce the compu-tational complexity of the model. Experimental results show that the improved YOLOv4 model improves detection efficiency (FPS) by 175%, reduces model parameters by 60% and reduces computational effort by 77% compared to the original model, while ensuring little loss of network accuracy. A lightweight mask-wearing detection model with high detection accuracy and real-time detection rates is implemented to meet the accuracy and real-time requirements of mask-wearing detection tasks.

Keywords

GhostNet; YOLOv4; deep learning; mask wear detection.

1. Introduction

At present, the situation of COVID-19 prevention and control is still very serious. COVID-19 threatens the health of people all over the world, and brings great trouble to people's social production and daily life. Droplet transmission is a common route for the spread of novel coronaviruses. Respiratory droplets produced by a person infected with the virus by talking, coughing or sneezing may cause infection in other people through direct inhalation of gases. In order to reduce the probability of COVID-19 infection and prevent the continuous spread of the epidemic, wearing masks during travel or in public crowded places is currently the mainstream measure adopted by people at home and abroad, as well as the most direct and effective measure. And airports, stations, shopping malls and other places where the flow of people is both dense and continuous, the general way to manually supervise the mask wearing inspection, which relies on human and material support, resulting in huge manpower consumption and a large waste of public resources. This method also has low detection efficiency, which makes people crowded and queuing, prone to cross infection, missing inspection and other problems, and it is difficult to carry out 24-hour inspection.

In recent years, with the rapid development of deep learning, the field of computer vision has also made great progress, and target detection is an important task in computer vision. Deep learning-based target detection algorithms are divided into two major categories, one is the two-stage detection algorithm based on candidate regions represented by R-CNN [1] and Faster R-CNN [2], which are performed in two steps: first, a candidate region that may contain an object is generated, and then the final result is obtained by classifying and predicting the region

ISSN: 2664-9640

through convolutional neural networks. Another class is the one-stage detection algorithm based on border regression represented by YOLO (You Only Look Once) series [3] and

SSD (Single Shot MultiBox Detection) [4], which is based on the regression idea and uses convolutional neural network to directly classify and localize the target object with extremely fast detection speed. The two-stage detection algorithm has a high detection accuracy, but the detection speed is slow and the real-time performance is poor, while the one-stage detection algorithm has a fast detection speed, which can meet the requirements of real-time performance, but there will be some loss of detection accuracy. Researchers have achieved rich results by optimizing and improving on a generic target detection model to suit the mask wearing detection task.

The YOLO algorithm was first proposed by Redmon and based on it YOLOv2 [5], YOLOv3 [6] were developed.

In 2020, researcher Bochkovskiy proposed the YOLOv4 [7] algorithm, which made a series of optimizations based on YOLOv3, and achieved a balance of speed and accuracy. However, the YOLOv4 model suffers from a large number of parameters and redundant model structure, which makes the network difficult to deploy. Considering the training cost and real-time problems, lightweight model research is necessary. Zixun Ye et al. [8] replaced the v4 backbone feature extraction network with MobileNetv3 [9], and then optimized the detection performance using the self-attention mechanism and SiLU activation function. Yujie Luo et al. [10] also applied MobileNet, and adopted adaptive spatial feature fusion methods [11] to enhance the performance. Xin Jin et al. [12] also applied Mobilenet network as the backbone network of v4 and used depth-separable convolution instead of the original 3 × 3 standard convolution of PANet to improve the recognition speed of the model. Pei Ding et al. [13] introduced a lightweight backbone network, Light-CSPDarkNet, and used a lightweight feature enhancement module and a multiscale attention mechanism to improve the detection accuracy and speed of the model. Xiaoxi Cao et al. [14] introduced attention mechanism and reused spatial pyramid pooling technique to improve the mask wearing detection algorithm.

Although the above improved YOLOv4 model achieves good results, the improvement in detection speed is necessary for real-time mask wear detection for dense crowds in complex scenes. Aiming at the problems of large and complex YOLOv4 network structure, many calculation parameters and slow detection speed, this paper proposes a lightweight mask wearing detection model based on GhostNet. This model is based on YOLOv4 and uses GhostNet [15] as the backbone feature extraction network, which reduces the number of network model parameters and decreases the computational complexity, thus improving the detection speed of the model; and the convolutional module of the PANet structure of YOLOv4 is designed as a contextual network structure [16] to increase the perceptual field, which further reduces the number of model parameters; the strategy of fusing the batch normalization layer with the convolutional layer for computation is also adopted to further reduce the computational complexity of the model.

2. YOLOv4 Algorithm

The YOLOv4 network structure is an improved version of YOLOv3 and incorporates some training techniques to improve the network performance based on YOLOv3. The YOLOv4 algorithm contains a backbone feature extraction network, SPP and PANet network. YOLOv4 network structure is shown in Fig. 1. The backbone feature extraction network of YOLOv4 is a CSPDarkNet53 structure composed by incorporating CSPNet [17] on the DarkNet network, the backbone network contains five residual structures, which in turn contain 1, 2, 8, 8, and 4 residual units respectively, and the input image After feature extraction by CSPDarkNet53, 3 effective feature layers will be obtained; where the last effective feature layer enters into the

ISSN: 2664-9640

SPP network, the SPP structure is processed using the maximum pooling of four different scales 13×13, 9×9, 5×5, 1×1 respectively, which can increase the perceptual field and extract better contextual features ; then the same as the first two enter the PANet respectively for feature fusion, output 3 different scales of feature maps, and finally processed by YOLO Head to get the prediction results.

YOLOv4 uses the Mosaic data enhancement method on the input side to do further processing on the dataset, which reads four images at a time, flips, scales, and stitches the four images separately, and then combines them to get a new image, by which the size of the dataset can be expanded, and also the background of the detected objects can be enriched, and the generalization ability of the model can be improved, which can be used with limited GPU resource conditions to get better results.

YOLOv4 uses the CIOU loss in the location regression loss function. the CIOU takes into account the overlap rate between the predicted frame and the real frame, the center distance of the two frames, and the aspect ratio of the two frames, making the frame regression more sta-ble. the CIOU formula is shown in (1).

$$CIOU = IOU - \frac{\rho^2(b, b^{gt})}{c^2} - \alpha v \tag{1}$$

where $\rho^2(b, b^{gt})$ represents the Euclidean distance between the prediction frame and the centroid of the real frame, respectively, and c represents the diagonal distance of the smallest closed region that can contain both the prediction frame, in Eq. (1) α and v are calculated as shown

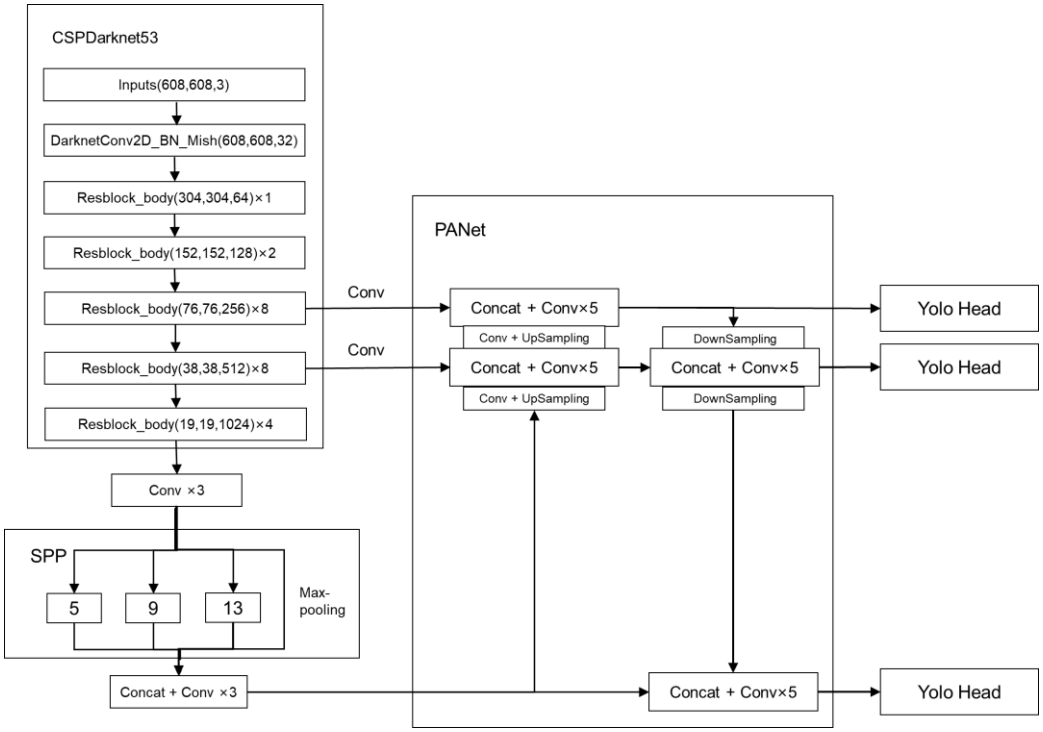


Fig. 1 YOLOv4 network structure

in Eq. (2), (3).

$$\alpha = \frac{v}{1 - IoU + v} \tag{2}$$

ISSN: 2664-9640

$$v = \frac{4}{\pi}(\alpha rctan\frac{w^{gt}}{h^{gt}} - arctan\frac{w}{h})^2 \quad (3)$$

The full expression for CIOU as a loss function is shown in Equation (4).

$$CIoU_{Loss} = 1 - IoU + \frac{\rho^2(b, b^{gt})}{c^2} + \alpha v \quad (4)$$

3. GhostNet-based Lightweight YOLOv4 Mask Wear Detection Model (GN-YOLOv4 Model)

In this paper, we make a lightweight improvement based on the YOLOv4 model, and the proposed GN-YOLOv4 model mainly contains a backbone network, a neck network and a head network. The backbone network, i.e. the backbone feature extraction network, uses GhostNet as the backbone extraction network; the neck network is composed of SPP and the modified PANet network, which designs part of the convolutional blocks of PANet into a contextual network structure, and uses the fusion technique of batch normalization and convolutional layers; the head network is used to detect targets of different sizes and output the final prediction results .

3.1. Backbone Network

GhostNet is a lightweight neural network proposed by Huawei Noah's Ark Laboratory. One important feature of the CNN model is the existence of redundancy in the feature map, which leads to an increase in the number of parameters and computational effort. The core idea of GhostNet is to use some less computationally intensive operations to generate redundant feature maps to reduce the number of parameters and increase the detection speed of the model while ensuring good detection results.

GhostNet uses the Ghost Module, which functions as a replacement for ordinary convolution. As shown in Fig. 2. Ghost Module divides the ordinary convolution into two parts, firstly, the ordinary 1×1 convolution is used to compress the number of channels of the input image to generate the feature concentration of the input feature layer; then the depth-separable convolution is performed to get more feature maps, and this depth-separable convolution is a layer-by-layer convolution, and the different feature maps are Concat together and combined into a new output.

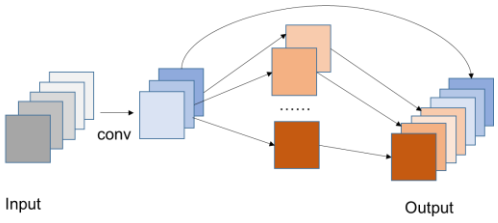


Fig. 2 Ghost Module

ISSN: 2664-9640

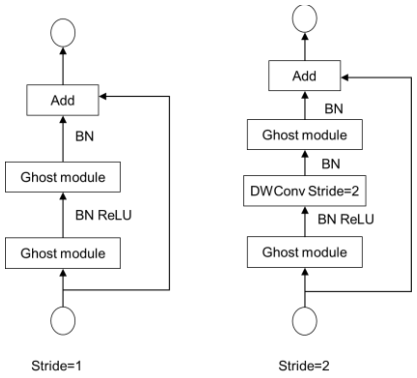


Fig. 3 Ghost bottleneck structure

Ghost BottleNeck is a bottleneck structure composed of Ghost Modules, which is similar to the basic residual block in ResNet [18], where multiple convolutional layers and shortcut are integrated. Ghost BottleNeck contains 2 types of Ghost modules, as shown in Fig. 3. Ghost bottleneck structure. In the left figure, the step size (stride) is 1. The first Ghost module is used to expand the number of channels, and the second Ghost module decreases the number of channels to match the shortcut path, and then the shortcut is used to connect the inputs and outputs of these two Ghost modules. The right figure shows the case with a step size (stride) of 2. A Depthwise Convolution with a stride of 2 is added between the two Ghost modules, which is used to compress the size of the feature layers and reduce the parameter size.

In this paper, the GN-YOLOv4 model uses the GhostNet network as the backbone feature extraction network of the model, and GhostNet is mainly composed of Ghost bottleneck, which is built with the Ghost module as the basis. An image of size 608×608 is input to the GN-YOLOv4, and the backbone network GhostNet first performs a standard convolutional block of 16 channels, and then the channels are gradually increased through a series of stacking of bottleneck structures. The backbone network structure of the GN-YOLOv4 model is shown in Table 1.

Table 1 GN-YOLOv4 backbone network structure

Input	Operator	Output channels	Stride	SE
608×608×3	Conv2d	16	2	—
304×304×16	G-bneck	16	1	—
304×304×16	G-bneck	24	2	—
152×152×24	G-bneck	24	1	—
152×152×24	G-bneck	40	2	1
76×76×40	G-bneck	40	1	1
76×76×40	G-bneck	80	2	—
38×38×80	G-bneck	80	1	—
38×38×80	G-bneck	80	1	—
38×38×80	G-bneck	80	1	—
38×38×80	G-bneck	112	1	1
38×38×112	G-bneck	112	1	1
38×38×112	G-bneck	160	2	1
19×19×160	G-bneck	160	1	—
19×19×160	G-bneck	160	1	1

ISSN: 2664-9640

19×19×160	G-bneck	160	1	—
19×19×160	G-bneck	160	1	1

3.2. Neck Network

The neck network is also known as the enhanced feature extraction network, and GN-YOLOv4 uses SPP and a modified PANet network as the neck network. The SPP structure is shown in Fig. 4. This network performs the maximum pooling operation of 1×1, 5×5, 9×9, 13×13 on the input feature maps respectively, the size of the feature maps before and after pooling is constant, the feature maps before pooling and the feature maps after three pooling processes are summed by channel to obtain the output feature maps, this operation can increase the sensory field and reduce the overfitting to a certain extent.

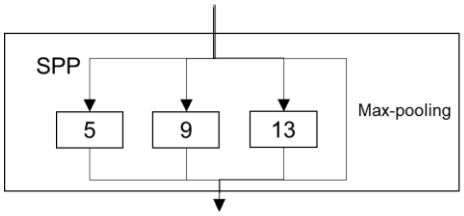


Fig. 4 SPP network structure

To introduce more contextual information, this paper refers to the contextual network of [16] and designs the Conv×5 in the enhanced feature extraction network PANet structure as a contextual network structure. In a two-stage detector, the context is usually merged by expanding the window around the candidate solution, and the context network module simulates this strategy by a simple convolutional layer. The modified PANet network structure is shown in Fig. 5. In order to further reduce the number of parameters in the model, Conv×5 is first passed through a 1×1 convolutional layer, then a 3×3 convolutional layer and two 3×3 convolutional layers respectively, and finally Concat gets the final output. Modeling the context in this way increases the perceptual field of the corresponding layer and also increases the target scale in the module, which also reduces the amount of model computation.

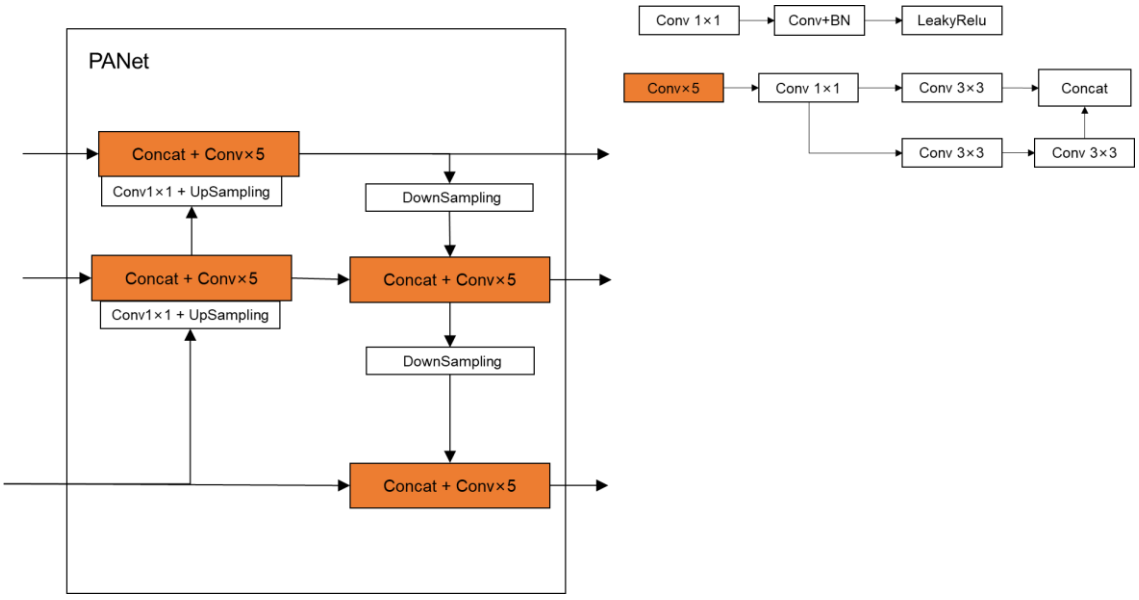


Fig. 5 The modified PANet network structure

Batch Normalization layer [19] can improve the model training speed and improve the network generalization performance. YOLOv4 introduces the batch normalization layer, which can

ISSN: 2664-9640

speed up the network convergence, thus avoiding overfitting and solving problems such as gradient disappearance and gradient explosion, and is generally placed after the convolutional layer. However, the batch normalization layer increases the amount of computation and makes the network more complex, which affects the model performance to a certain extent and occupies more space. Therefore, in order to further reduce the amount of parameter computation and improve the model inference speed, a strategy of fusing the batch normalization layer with the convolutional layer is proposed.

If the model training batch has a total of n samples, whose features are $x_1, x_2, \cdots, x_n$, for the ith sample, the convolution layer is calculated as shown in Equation (5).

$$x_{conv} = x_i * \omega_i + b \tag{5}$$

The arithmetic process for the BN layer is shown in equation (6).

$$y_i = \gamma \frac{x_{conv} - \mu}{\sqrt{\sigma^2 + \in}} + \beta = \frac{\gamma}{\sqrt{\sigma^2 + \epsilon}} x_{conv} + \left(\beta - \frac{\gamma\mu}{\sqrt{\sigma^2 + \epsilon}}\right) \tag{6}$$

where μ and σ^2 are the mean and variance of x_i within a batch, and $\in$ is a very small constant set to avoid division errors, e.g. 0.000001. γ is a scaling factor， β is the bias amount. The mean and variance are calculated as shown in equation (7)(8).

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{7}$$

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - \mu)^2 \tag{8}$$

Combining the convolution and BN layers yields equations (9) and (10).

$$y_i = \frac{\gamma}{\sqrt{\sigma^2 + \epsilon}} (x_i * \omega_i + b) + \left(\beta - \frac{\gamma\mu}{\sqrt{\sigma^2 + \epsilon}}\right) \tag{9}$$

$$y_i = \frac{\gamma\omega_i}{\sqrt{\sigma^2 + \epsilon}} x_i + \left(\frac{\gamma(b - \mu)}{\sqrt{\sigma^2 + \epsilon}} + \beta\right) \tag{10}$$

The merged weight parameter is set to $\hat{\omega}$ and the bias parameter is set to $\hat{\beta}$ and the equations are shown in (11) and (12).

$$\hat{\omega} = \frac{\gamma\omega_i}{\sqrt{\sigma^2 + \epsilon}} \tag{11}$$

$$\hat{\beta} = \frac{\gamma(b - \mu)}{\sqrt{\sigma^2 + \epsilon}} + \beta \tag{12}$$

Then the combined calculation is as shown in equation (13).

$$y_i = \hat{\omega} * x_i + \hat{\beta} \tag{13}$$

ISSN: 2664-9640

It can be seen that new weights are generated in the operation of the convolutional layer after the fusion of the convolutional layer with the BN layer is completed, and a new bias is also introduced to function as a batch normalization layer.

3.3. Yolo Head

The head network, i.e. the prediction network, Yolo Head uses the multi-scale features obtained from the PANet structure for regression and classification prediction. Yolo head network structure is shown in Fig. 6. PANet provides three feature layers with dimensions 76 × 76 × 40, 38 × 38 × 112, and 19 × 19 × 160 corresponding to the middle layer, lower middle layer, and bottom prediction frame. Yolo Head obtains prediction results at three different scales after convolving these three feature layers with 3 × 3 and 1 × 1 respectively.

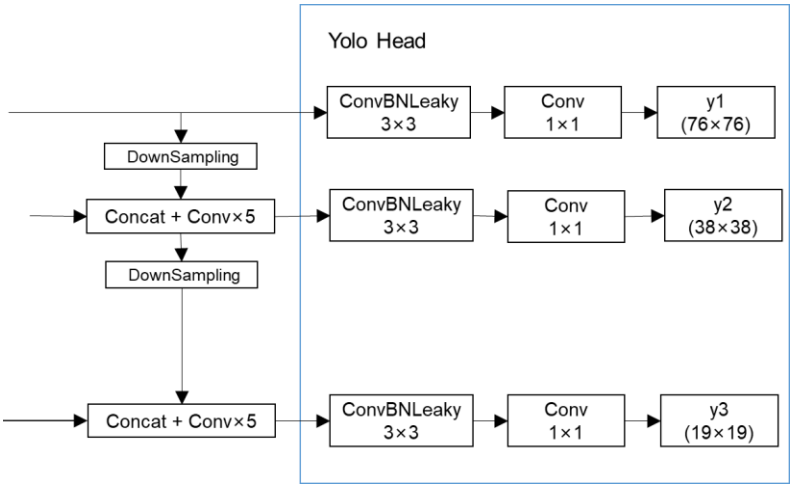


Fig. 6 Yolo head network structure

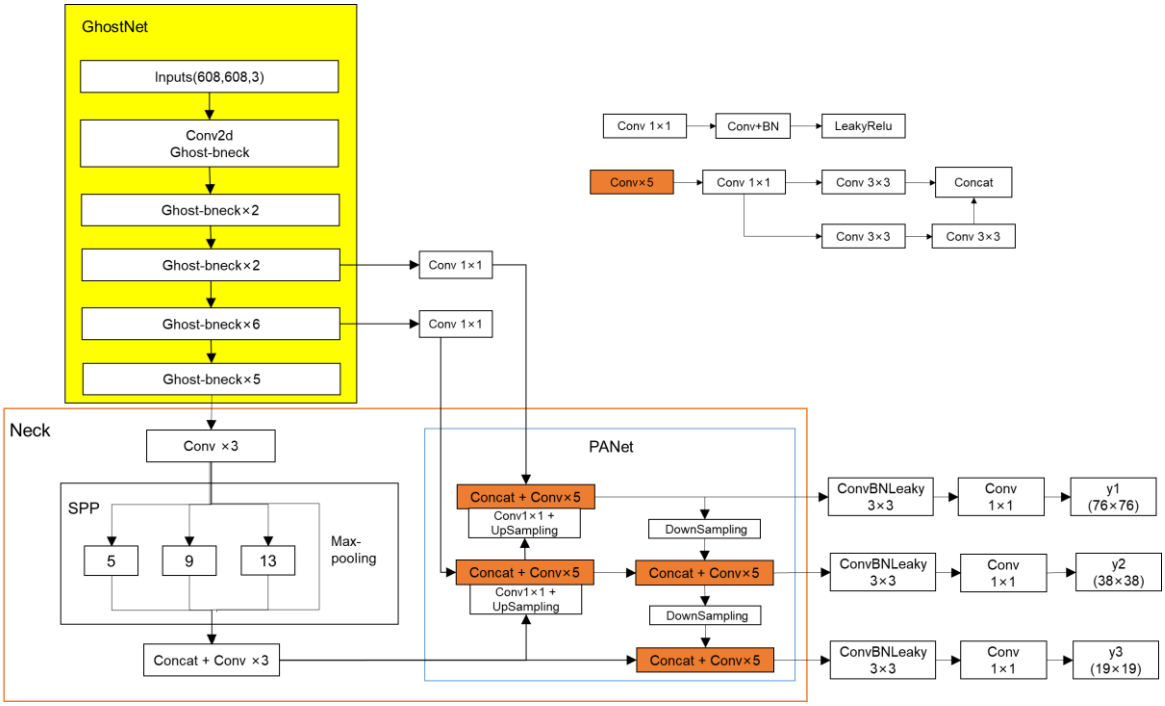


Fig. 7 Network structure of GN-YOLOv4 model

3.4. Network Structure of GN-YOLOv4 Model

The network structure of the proposed lightweight GN-YOLOv4 model in this paper is shown in Fig. 7. The improvements based on the original YOLOv4 network are mainly as follows: first,

ISSN: 2664-9640

in the feature extraction stage, GhostNet is used as the backbone extraction network of the network to reduce the number of network model parameters and reduce the computational complexity; second, the convolutional module in the PANet structure is designed as a contextual network structure by combining the idea of contextual network, so that the number of model parameters is further reduced, and the a strategy of fusing the batch normalization layer with the convolutional layer for computation to further reduce the amount of model operations.

The image size of 608×608 is input to GN-YOLOv4 network, and after feature extraction by the backbone network GhostNet, three effective feature layers 76×76×40, 38×38×112, and 19×19×160 are obtained, where the last effective feature layer enters into the SPP structure after 3 times convolution, and SPP uses the maximum pooling kernel of different sizes to feature layers for maximum pooling, the output results are stacked and then convolved 3 times, the output feature layer with 38×38 and 76×76 size feature layers enter into PANet for feature fusion after UpSampling, DownSampling and contextual network structure to get 3 effective features of different scales and then the prediction results are output by YOLO Head.

4. Analysis of Experiments and Results

4.1. DataSets and Evaluation standard

The experimental dataset was selected from RMFD [20] and AIZOO datasets respectively, and combined with a portion of images collected from the Internet to build a mask detection dataset containing 5001 images of multiple scenes and different people, and the images were labeled by the Labeling tool, containing a total of two types of targets: wearing a mask (with_mask) and not wearing a mask (without_mask). A randomly selected 20% of the data in the dataset was used as the test set, and the training and validation sets were divided in the ratio of 9:1.

This experiment is a self-constructed dataset, and the experimental dataset needs to be clustered using the K-means clustering algorithm to obtain the appropriate prior frame sizes. The prior frame sizes and assignment results obtained from clustering are shown in Table 2, where large scale feature maps are assigned three small size prior frames to predict small targets, medium scale feature maps are assigned three medium size prior frames, and small scale feature maps are assigned three large size prior frames to predict large targets. And Mosaic data enhancement is used to increase the diversity of training samples and improve the generalization ability of the model.

Table 2 A priori frame size assignment

Testing scale	Priori frame size		
19×19	(99,87)	(98,152)	(206,226)
38×38	(36,65)	(61,52)	(54,97)
76×76	(12,21）	(19,34)	(27,45)

In order to verify the effectiveness of the improved network model, Precision (P), Recall (R), Average Accuracy (AP), Mean Average Accuracy (mAP), and Detection Speed (FPS) are generally selected as evaluation standard. P represents the proportion of true positive cases to the total number of all data predicted as positive samples, and R represents the proportion of correctly predicted positive cases to the total number of samples of the total number of true positive cases in the sample, and the calculation formulae are shown in (14) and (15).

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

ISSN: 2664-9640

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

where TP indicates the number of positive cases predicted as positive, FN indicates the number of positive cases predicted as negative, and FP indicates the number of negative cases predicted as positive.

The relationship between precision and recall can be demonstrated by a P-R plot, which is plotted with the find rate P as the vertical axis and the completion rate R as the horizontal axis, giving the finding rate - completion rate curve, referred to as the P-R curve. The area under the PR curve is defined as AP, while mAP is the average of the AP values for all categories in the dataset, and the calculation formulae are shown in (16) and (17).

$$AP = \int_0^1 P(R)\, dR \tag{16}$$

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

FPS, as an evaluation metric for model detection speed, indicates how many images can be processed by the model in one second, i.e. the time required to process a single image. In many practical applications, there is a high demand for real-time detection, and it is also important to improve the model detection speed while ensuring a certain detection accuracy of the model.

4.2. Experimental Results and Analysis

The precision values P and recall R of the GN-YOLOv4 model and YOLOv4 detection results in this paper are shown in Table 3.

Table 3 Comparison of P and R results

		Precision	Recall
YOLOv4	with_mask	88.18%	88.83%
	without_ mask	89.81%	85.84%
GN-YOLOv4	with_mask	88.86%	86.69%
	without_ mask	89.71%	79.74%

As can be seen from the table, the GN-YOLOv4 model maintains a certain level of detection accuracy with the original model, and the detection accuracy without the mask is basically the same as the original model, while the result with the mask is still a little higher compared to the original model. When the threshold value is set to 0.5, the recall rate of GN-YOLOv4 model decrease somewhat compared with YOLOv4 model, which may be due to the fact that the amount of parameters becomes smaller after the lightweighting process, which will lose part of the feature extraction ability, resulting in a certain weakening of the recall ability of the samples.

A comparison of the actual detection effects is shown in Fig. 8, where Fig. 8. (a) shows the GN-YOLOv4 detection effect and Fig. 8. (b) shows the YOLOv4 detection effect. In complex dense crowd scenes, YOLOv4 suffers from missed detection and poor results in the case of occlusion detection and long-range detection, while the detection effect of GN-YOLOv4 mitigates the

ISSN: 2664-9640

target missed detection, while the model improves the detection of side-face targets and occluded targets.

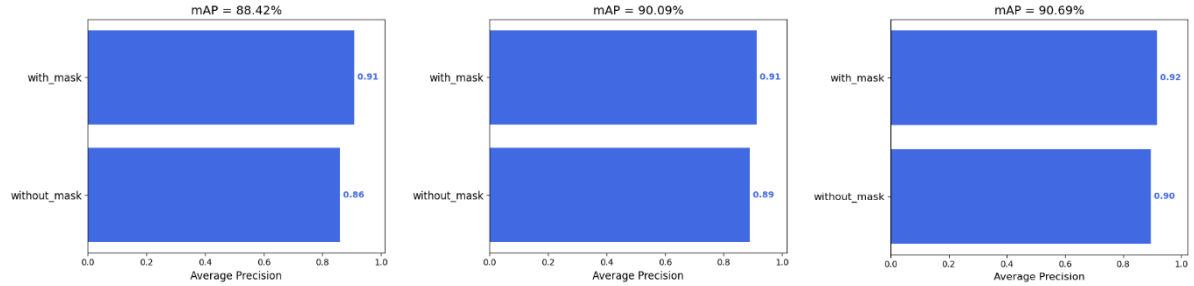
The AP and mAP values calculated after the training of the GN-YOLOv4, YOLOv3 and YOLOv4-tiny models are shown in Fig. 9. AP and mAP indicators. As can be seen from the figure, the mAP values of the GN-YOLOv4 model proposed in this paper are higher than those of YOLOv3 and YOLOv4-tiny, and the AP values with and without masks are also higher than those of both, and both AP and mAP values reach more than 90%, which can meet the requirements for accuracy values in the target detection task.

The model size and detection speed metric FPS are shown in Fig. 10. As can be seen from the figure, both the YOLOv4 and YOLOv3 models are larger than the model proposed in this paper, the model in this pa-per uses GhostNet lightweight network as the backbone feature extraction network, and makes improvements to the convolutional block of PANet, as well as uses the strategy of fusing batch normalization layer with convo-lutional layer, which reduces the number of parameters and computation of the network model, the size of the model in this paper is 39% of YOLOv4, which achieves the lightweight of the model. In terms of detection speed, the model in this paper achieves 93.2FPS, which is 175% higher than YOLOv4, and the model detection speed is substantially improved to achieve the effect of fast model detection, which also proves the effectiveness of the improved model.

(a) GN-YOLOv4 detection effect

(b) YOLOv4 detection effect

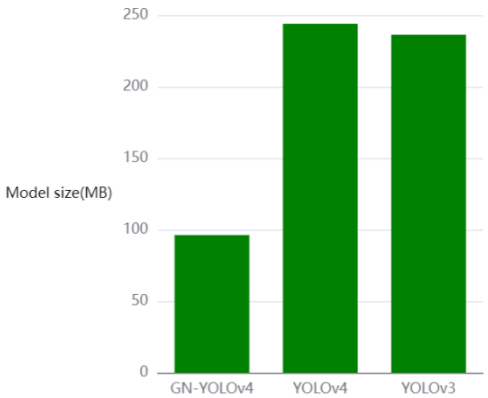
Fig. 8 Comparison of model detection effects



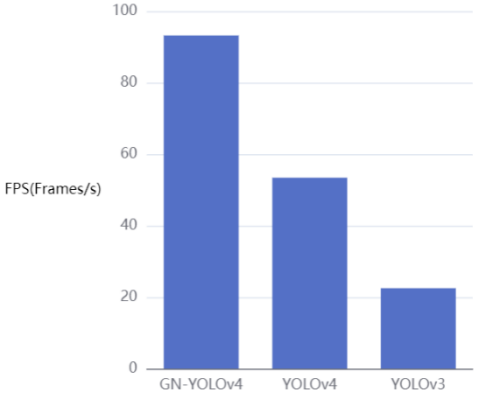
(a) YOLOv3 (b) YOLOv4-tiny (c) GN-YOLOv4

Fig. 9 AP and mAP indicators

ISSN: 2664-9640

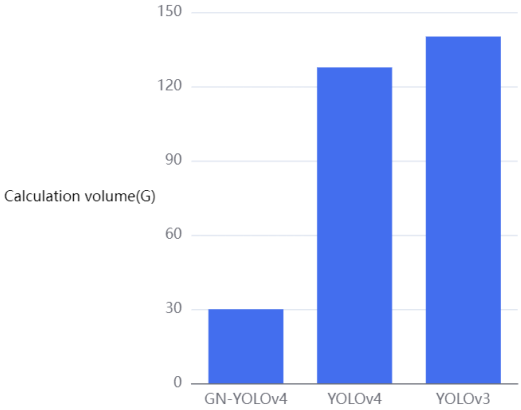


(a) Model size comparison

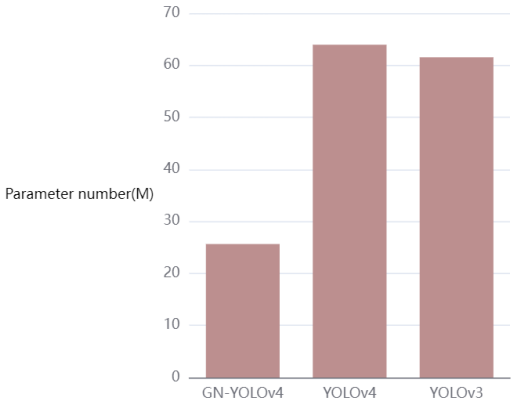


(b) Speed indicator FPS comparison

Fig. 10 Model size and FPS



(a) Comparison of model computations



(b) Comparison of model parameters

Fig. 11 Model computations and parameters size

The number of parameters and computation of the GN-YOLOv4 lightweight model are substantially reduced, as shown in Fig. 11. The computation and number of parameters of GN-YOLOv4 are lower than those of YOLOv3 and YOLOv4, and the number of parameters of the model is reduced by 60% and the computation is reduced by 77% compared with the original model YOLOv4, which significantly reduces the computational complexity of the model.

In summary, the model in this paper, while maintaining a high detection accuracy, not only has a significant reduction in the number of parameters, which is only 40% of the original model, the detection speed is improved by 175% compared to before the improvement, and the size of the model is 39% of the original model, while ensuring the detection performance of the model, it also achieves high speed and light weight, which can meet the demand for accuracy and real-time performance of the mask wearing detection task.

5. Conclusion

This paper proposes a lightweight YOLOv4 mask wearing detection model based on GhostNet. The model addresses the problems of high complexity and deployment difficulties of YOLOv4 model. Based on the original YOLOv4 model, adopts the lightweight network GhostNet as the backbone feature extraction network, and designs the convolutional module of the PANet structure of YOLOv4 as a contextual network structure, and finally adopts the strategy of fusing the batch normalization layer with the convolutional layer for calculation, which reduces the model parameters number and reduce the complexity of model computation. The experimental results show that the GN-YOLOv4 model proposed in this paper has good performance, it achieves an mAP value of 90.69% and still maintains high accuracy, the number of parameters

is reduced by 60% compared to YOLOv4, the model size is reduced by 61%, and the detection speed is substantially improved, which achieves the goal of lightweight model and meets the real-time detection in the epidemic prevention and control scenario. In the next step, we will introduce new improvement strategies to further improve the performance of the model while maintaining the lightness of the original model.

Acknowledgements

This research is supported by the Humanities and Social Sciences of Ministry Education of China (No.19XJA910001) and the postgraduate innovation fund project of Chongqing University of Technology (No.gzlcx20223198).

References

[1] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//2014 IEEE Conference on Computer Vision and Pattern Recognition. 2014: 580-587.

[2] REN S, HE K, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[C]//Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems. 2015：91-99.

[3] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection [C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. 2016: 779-788.

[4] LIU W, ANGUELOV D, ERHAN D, et al. SSD：single shot multibox detector[C]//14th European Conference on Computer Vision. Cham: Springer, 2016: 21-37.

[5] REDMON J, FARHADI A. YOLO9000: better, faster, stronger[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. 2017: 6517-6525.

[6] REDMON J, FARHADI A. YOLOv3: an incremental improvement[J]. arXiv: 1804.02767, 2018.

[7] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: optimal speed and accuracy of object detection[J]. arXiv: 2004.10934, 2020.

[8] Zixun Ye, Hongying Zhang. Lightweight improvement of YOLOv4 mask detection algorithm[J]. Computer Engineering and Applications, 2021, 57(17): 157-168.

[9] HOWARD A, SANDLER M, CHU G, et al. Searching for MobileNetv3[C]//2019 IEEE/CVF International Conference on Computer Vision, 2019: 1314-1324.

[10] Yujie Luo, Jian Zhang, Liang Chen, et al. Design of lightweight target detection algorithm based on adaptive spatial feature fusion[J]. Advances in Lasers and Optoelectronics, 2022, 59(4): 310-320.

[11] LIU S, HUANG D, WANG Y. Learning spatial fusion for single-shot object detection[J]. arXiv: 1911.09516, 2019.

[12] Xin Jin, Sike Zeng, Yang Liu, Chuhan Wu. An improved YOLOv4-based algorithm for mask wear detection[J]. Computers and Modernization,2022(01):85-90.

[13] Pei Ding, Korban Alifu, Liting Geng, et al. Real-time face mask detection and normative wear recognition in natural environment[J]. Computer Engineering and Applications, 2021, 57(24): 268-275.

[14] Xiaoxi Cao, Fangyong Cheng, Feizhou Wang, Mingyan Zhang. Mask wearing detection algorithm for dense crowds based on improved yolov4 [J]. Journal of Anhui University of Engineering, 2022,37(03):49-57+69.

[15] HAN K, WANG Y, TIAN Q, et al. GhostNet: more features from cheap operations[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 1580-1589.

[16] Najibi M, Samangouei P, Chellappa R, et al. SSH: Single Stage Headless Face Detector[C]//2017 IEEE International Conference on Computer Vision, 2017: 4875-4884.

ISSN: 2664-9640

[17] WANG C Y, MARK LIAO H Y, WU Y H, et al. CSPNet: a new backbone that can enhance learning capability of CNN[C]// Proceeding of 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 390-391.

[18] HE Kaiming, ZHANG Xiangyu, REN Shaoqing, et al. Deep residual learning for image recognition [C]//Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2016: 770-778.

[19] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift[C]//International conference on machine learning. PMLR, 2015: 448-456.

[20] WANG Z, WANG G, HUANG B, et al. Masked face recognition dataset and application[J]. arXiv: 2003. 09093, 2020.

[21] Wang C Y, Bochkovskiy A, Liao H Y M. Scaled-yolov4: Scaling cross stage partial network [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: 13029-13038.