[1] **Bingyan Lin**
[1,*] **Maidi Hou**

Face Mask Detection Based on Improved YOLOv8

Journal of Electrical Systems

Abstract: - The detection of face mask wear is one of the essential measures to prevent the spread of infectious diseases in public places. In order to balance the is-sues of inference speed and performance of target detection models on em-bedded devices, this paper proposes a face mask detection based on the im-proved YOLOv8 algorithm, YOLOv8n-SLIM-DYHEAD. By improving the YOLOv8n algorithm, the balance between detection time and accuracy is-sues is achieved. The Mosaic data augmentation method is used to increase the detection targets of various sizes, enrich the sample dataset of masks of various scales. On the neck network, the Slim-neck structure is used to fuse features of different sizes extracted by the leading network, reducing the complexity of the model while maintaining accuracy. In the detection layer, DyHead is used to integrate better feature diversity caused by target scale differences and target shape position differences. Experimental results show that the improved algorithm YOLOv8n-SLIM-DYHEAD has increased the mAP @0.5 and mAP @0.5:0.95 of the original YOLOv8n algorithm by 2.1 and 5.5 percentage points, respectively. In addition, the complexity and pa-rameters of the model have remained relatively high, and it can accurately detect the wearing of masks in real-time.

Keywords: Face mask detection; YOLOv8 algorithm; Mosaic data augmentation; Slim-neck, DyHead, YOLOv8n-SLIM-DYHEAD.

I. INTRODUCTION

Face masks is an important measure to prevent the spread of infectious diseases and reduce the harm of harmful particles generated during factory production, thereby enhancing life safety and improving hygiene. Monitoring the wearing of masks in public places has become a core activity. However, the manual supervision of mask-wearing by personnel is time-consuming and complex and carries certain safety risks for the inspectors due to close contact. Therefore, establishing a mask-wearing monitoring system to detect individuals' use of masks holds long-term practical significance and importance for safety control automation.

In the past two decades, mask detection has been broadly divided into two categories: traditional mask detection and deep learning-based mask detection. With the advancement of GPUs and big data, deep learning has gradually demonstrated its superiority. Traditional methods usually require multiple stages to complete tasks, whereas deep learning can directly process many facial mask images through end-to-end training [1–3].

Thanks to the rapid development of deep learning, the academic community has witnessed an array of deep learning-based detection models. Most classic object detection algorithms rely on convolutional networks for feature extraction. Notably, foundational networks such as VGGNet [4], GoogLeNet [5], and ResNet [6] have shown outstanding results in feature extraction. Deep learning-based mask detection techniques are divided into two categories: those based on candidate region two-stage (two-stage) detection algorithms and those based on regression one-stage (one-stage) detection algorithms. Due to their generation of numerous candidate boxes, two-stage detection algorithms have slow detection speeds that do not align with the real-time requirements of mask detection. On the other hand, one-stage algorithms perform object classification and position prediction through a single feature extraction. The YOLO series of algorithms has been extensively utilized in this category, showcasing exceptional accuracy and speed. Many scholars have made improvements to the YOLO algorithm to enhance detection accuracy. [7] proposed the ETL-YOLO v4 network, which modified and improved feature extraction and prediction networks for the micro YOLO v4 model. The algorithm improved the micro YOLO v4's central architecture by adding a modified dense SPP network, two more detection layers with modified and optimized CNN layers (which helped with accurate prediction), and Mish as the activation function. This made mask detection more accurate. Wu et al. [8] introduced a novel mask detection framework, FMD-Yolo, for monitoring individuals in public places for correct mask-wearing. The feature extractor used Im-Res2Net-101. It combines Res2Net modules with a deep residual network that uses hierarchical convolutional structures, deformable convolutions, and non-local mechanisms to get input data. Furthermore, an enhanced path aggregation network (En-PAN) was applied for feature fusion, fully integrating high-level semantic information

[1] Fujian Polytechnic of Information Technology, Fuzhou 350003, P. R. China
*Corresponding author: Maidi Hou

and low-level details, strengthening the model's robustness and generalization ability. [9] proposed a novel mask detection algorithm based on the YOLO-GBC network. In the backbone network, a global attention mechanism (GAM) was integrated to enhance the extraction of crucial information. An improved feature pyramid structure was implemented through cross-layer cascading to achieve effective bi-directional, multi-scale connections and weighted feature fusion. The sampling method of content-aware feature rearrangement (CARAFE) was added to the feature pyramid network to keep all of the semantic information and global features of the feature map. [10] investigated various pruning and quantization techniques for improving compressed models' frames per second (FPS) while maintaining detection accuracy. [11] utilized YOLO to combine high-level semantic information with various feature maps and machine-learning modules to concurrently recognize masks and social distancing. Zhao et al. [12] introduced position-insensitive loss and semi-deformable convolutional network methods into the YOLOv5s, improving accuracy. [13] proposed an improved algorithm based on the YOLO-v4 algorithm and incorporated attention mechanism modules at appropriate network layers to strengthen key features of faces wearing masks while suppressing useless information.

While these studies have often achieved remarkable performance, they have not optimized the models for lightweight deployment, resulting in high computational complexity that makes them challenging to apply on resource-limited embedded devices. MobileNet [14] added more effective depthwise separable convolutions, which sped up network speeds and expanded the use of convolutional neural networks on mobile devices. By reducing computational complexity, MobileNet achieved significant improvements in accuracy. However, there is still room for further reduction in computational complexity in theory. ShuffleNet [15] effectively reduced the computational burden of point convolutions by leveraging group convolutions and channel shuffling, achieving superior performance. With the continuous advancement of mobile devices and the diversification of application scenarios, lightweight networks have demonstrated excellent engineering value.

In recent years, the YOLO algorithm has undergone continuous optimization. The Ultralytics team introduced the YOLOv8 version in 2023. This algorithm meets real-time requirements and demonstrates high detection accuracy and a relatively lightweight network structure, making it suitable for mask detection [16–17]. There is currently limited research on mask-wearing detection using YOLOv8, and our study aims to fill this gap. Based on the YOLOv8 algorithm, we optimized the model. We proposed an enhanced YOLOv8n mask-wearing detection algorithm, named YOLOv8n-SLIM-DYHEAD, to increase accuracy and make the model more suitable for deployment on embedded devices. In comparison to existing research, our main contributions are as follows:

(1) We employed Mosaic data augmentation technology to address the issue of limited mask detection datasets. This approach involves merging regions from different images to generate diverse targets, enriching various scaled mask sample datasets, and effectively resolving the issue of dataset scarcity. After the data augmentation, we expanded the dataset, providing the algorithm with more training samples. An enriched dataset can significantly enhance the algorithm's detection performance for various scaled targets, allowing for more accurate detection of different mask targets.

(2) To maintain accuracy while reducing model complexity, we introduced the slim-neck structure. This structure effectively integrates differently scaled feature maps extracted by the backbone network, optimizing the network structure to maintain model accuracy while lowering model complexity and enhancing the practicality and efficiency of the algorithm.

(3) To better integrate the feature diversity caused by differences in target scale and position, we introduced the DyHead block in the neck. This structure effectively integrates features of different scales and shapes, enhancing the model's learning capability, enabling the algorithm to comprehensively learn and capture features of different target sizes, and improving the robustness and accuracy of the mask detection algorithm.

The remaining sections of the document are arranged as follows: In Section 2, the YOLOv8 algorithm is explained. We talk about the architecture of the model in Section 3. The results and discussion of the experimental analysis are presented in Section 4. Section 5 finally provides a summary and conclusion of our work.

II. YOLOv8 ALGORITHM

YOLO (You Only Look Once) is a high-performance, versatile object detection model. YOLOv1 [18] ingeniously achieved the tasks of classification and target localization through a single-stage structure. Subsequently, YOLOv2 [19] and YOLOv3 [20] improved speed and accuracy, further advancing object detection applications in the industrial sector. YOLOv4 [21] allows training on common GPUs, such as the 1080Ti. YOLOv5 [23], more flexible than YOLOv4, offers four different-sized versions: YOLOv5s, YOLOv5m,

YOLOv5l, and YOLOv5x. These versions gradually increase in model size and accuracy by adjusting the number of bottlenecks. In addition, factors similar to EfficientNet that control channels and layers are added to make it easier to switch between versions. This lets you choose the right model size for each application scenario.

YOLOv8, released by Ultralytics on January 10, 2023, is a significant update to YOLOv5. It supports image classification, object detection, and instance segmentation tasks. As depicted in Figure1, YOLOv8 introduces new features, including a novel backbone network, an anchor-free detection head, and a new loss function designed to run across various hardware platforms from CPU to GPU. YOLOv8 efficiently and flexibly supports multiple export formats and operates on both CPU and GPU. Within the YOLOv8 model, five models per category are employed for detection, segmentation, and classification. Compared to the YOLOv5 network, substantial improvements are made, replacing YOLOv5's C3 structure in the backbone with the more gradient-rich C2f structure, adjusting channel numbers for different scale models, and significantly enhancing model performance. Two significant improvements over YOLOv5 on the head side include: 1) adopting the prevalent decoupled head structure, separating classification and detection heads, and 2) transitioning from anchor-based to anchor-free. YOLOv8 does away with the old IOU matching or unilateral proportional allocation methods regarding the loss function. Instead, it uses a Task-Aligned Assigner for positive and negative sample matching and adds Distribution Focal Loss (DFL).

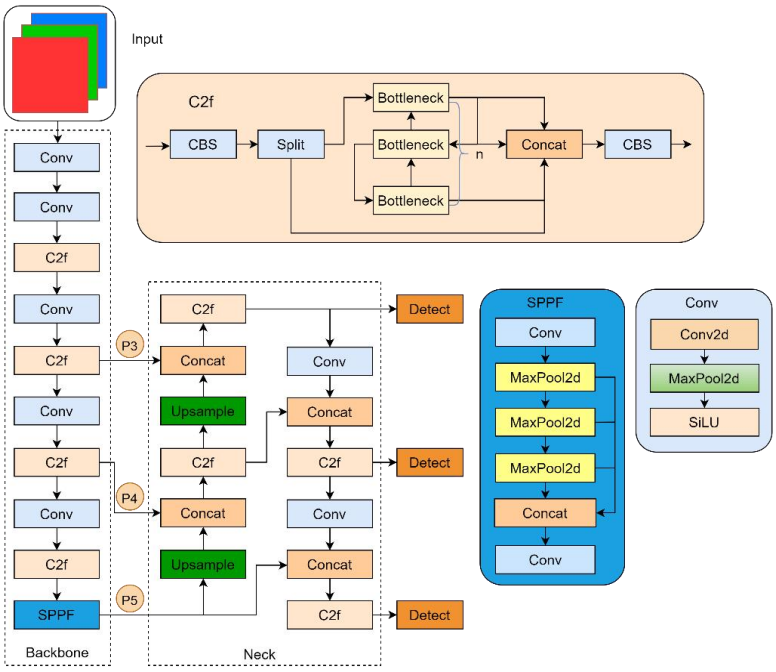


Figure 1. YOLOv8n network architecture [17].

YOLOv8 requires further optimization on different datasets to achieve higher detection accuracy. To address the balance between detection speed and accuracy in mask and face detection algorithms on small embedded devices, this paper proposes an improved YOLOv8 detection algorithm named YOLOv8n-SLIM-DYHEAD.

III. YOLOv8n-SLIM-DYHEAD Algorithm

As a universal object detection algorithm, YOLOv8 lacks mask detection effectiveness across various scales and shapes. Figure 2 illustrates the YOLOv8n-SLIM-DYHEAD network structure. To better integrate feature diversity caused by differences in target scale and position and reduce model size, the following improvements are made:

(1) adopting Mosaic data augmentation to increase the dataset size across various hierarchical levels

(2) introducing DyHead to enhance the detection effectiveness of targets at various scales

(3) utilizing Slim-neck in the neck network to decrease model complexity while maintaining accuracy

A. Mosaic Data Augmentation

Mosaic data augmentation is a ubiquitous technique in image data enhancement, primarily employed in computer vision and image processing [24]. Mosaic is predominantly utilized to enhance image data for training purposes, thereby augmenting the generalization capability of deep learning models. Given this study's relatively

modest scale of the mask dataset, data augmentation becomes imperative to bolster the dataset's volume. The essence of Mosaic data augmentation lies in amalgamating multiple distinct images into a singular large image, subsequently employed for model training.

To elaborate, Mosaic data augmentation typically encompasses the following steps:

(1) Source Image Selection: Randomly choose 4 or 9 distinct mask images awaiting detection from the training dataset.

(2) Random Cropping and Merging: Randomly crop and merge these images in proportion to form a large composite image, usually joining them horizontally and vertically. This enables the model to learn diverse scenes and features during training simultaneously.

(3) Label Processing: Corresponding labels necessitate analogous combinations and adjustments involving coordinate transformations and normalization.

(4) Model Training: Employ the synthesized large image and corresponding labels to train the deep learning model, enabling the model to better adapt to various scenes and data distributions.

The application of Mosaic data augmentation technology facilitates the model in adeptly capturing information from different scenes, mitigating overfitting, and enhancing the model's generalization prowess. The Mosaic data augmentation approach, as illustrated in Figure 3, involves concatenating 4 or 9 images. Post-augmentation, various targets of varying sizes are generated, enriching the mask detection sample dataset and significantly elevating the algorithm's performance in detecting masks of various scales.

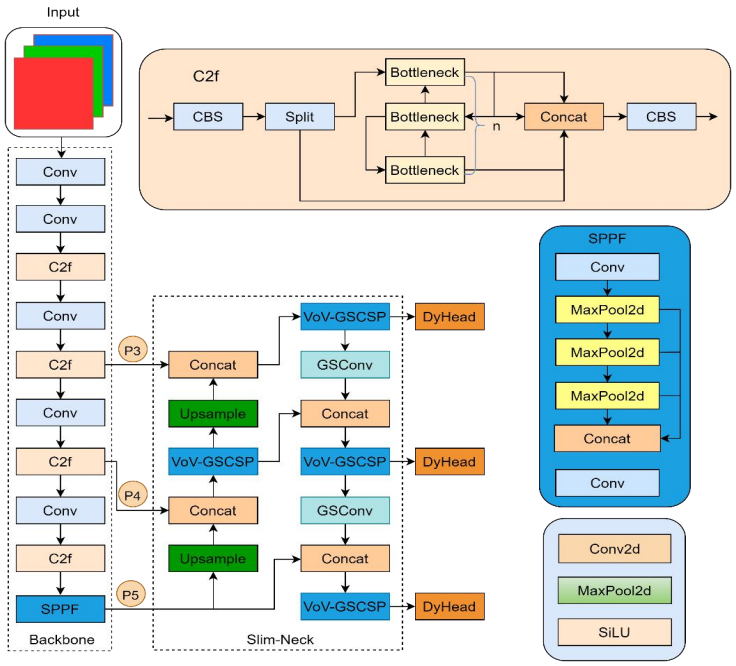
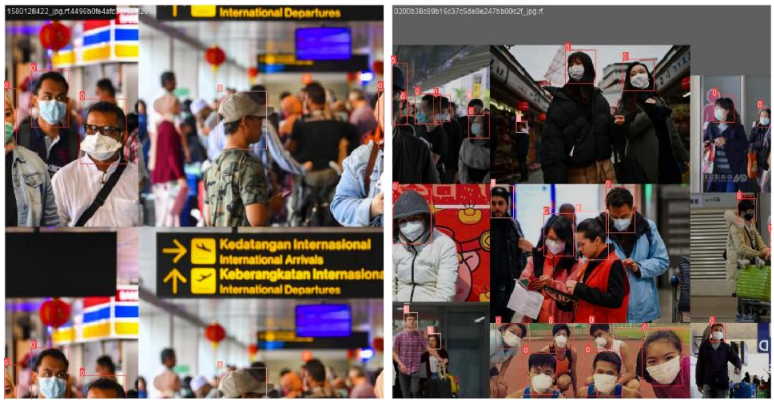


Figure 2. YOLOv8n-SLIM-DYHEAD network structure.



(a) (b)

Figure 3. Mosaic data augmentation. (a) Mosaic data augmentation was applied to four images. (b) Mosaic data augmentation was applied to nine images.

B. Slim-neck

The YOLOv8n algorithm integrated numerous standard convolutions and C2F modules to enhance accuracy. However, this led to reduced operational speed and increased model parameters. To alleviate model complexity while preserving accuracy, the neck network adopted the Slim-neck structure to merge different-sized feature maps extracted by the leading network.

To ease the network burden, lightweight network architectures like Xception [25] and ShuffleNet [26] introduced depth-wise separable convolutions, effectively addressing the computational overhead of standard convolutions. However, these lightweight methods sacrificed model detection accuracy while enhancing computational efficiency. Within the slim-neck structure, to lighten the network, we opted for GSConv [27] to replace the standard convolutions in the neck layer. Additionally, we introduced the VoV-GSCSP module [27], successfully reducing computational and structural complexity while maintaining adequate precision. As depicted in Figure 4, in the neck design, VoV-GSCSP replaced the traditional CSP, embedding the Slim-neck module into the neck network of YOLOv8. The Slim-neck module was crafted as a feature-fusion module specifically tailored for object detection tasks. Its design aimed to enhance model speed and efficiency by diminishing network parameters and computation volumes. From Table 1, adopting the Slim-neck in the YOLOv8n algorithm led to a 9.76% reduction in FLOPs compared to the original YOLOv8n algorithm, a 6.98% reduction in parameters, and a 9.52% speed improvement.

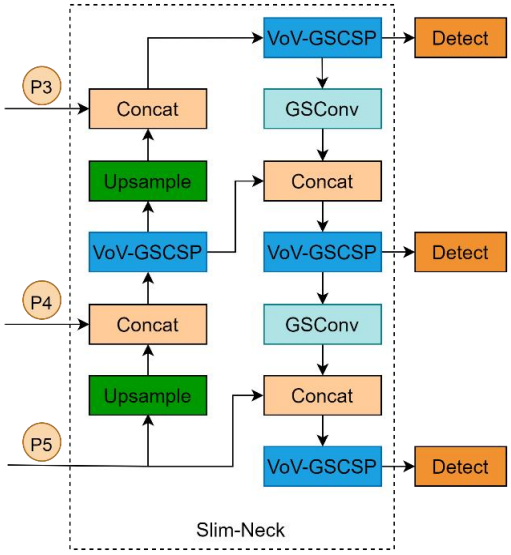


Figure 4. Illustration of the slim-neck embedded in the YOLOv8n structure.

Table 1. Performance comparison of YOLOv8n using the slim-neck structure.

Model	FLOPs (GB)	Parameters (MB)	Speed (ms)
YOLOv8n	8.2	3.01	2.1
YOLOv8n+Slim-neck	7.4	2.801	1.9

C. DyHead

This paper introduces a novel dynamic head framework, DyHead [28], which unifies different object detection heads using attention mechanisms. Attention mechanisms between feature hierarchies aid in scale perception, between spatial positions for spatial perception, and within output channels for task perception. This method significantly enhances the expression capability of the model's object detection heads without increasing the computational load.

To better amalgamate feature diversity arising from differences in target scale and shape positions, this paper introduced the DyHead block [28] in the neck. As Eq. (1) shows, it transforms three types of attention into a sequential arrangement, with each sequential attention focusing on a module constructed as nested attention functions.

$$W(F) = \pi_C(\pi_S(\pi_L(F) \cdot F) \cdot F) \cdot F \quad (1)$$

Where W represents the attention function, F is a $L \times S \times C$ three-dimensional feature tensor, where L denotes the feature map's level, S denotes the width-height product of the feature map, C denotes the number of channels

in the feature map, and π_L, π_S, π_C respectively represent the scale-awareness attention module, spatial-awareness attention module, and task-awareness attention module. Each attention module operates at different feature map levels, width-height products, and channel numbers. The structure of a single DyHead block is depicted in Figure 5. Any backbone network can extract feature pyramids, resizing them into a 3D tensor of the same scale as input for the dynamic head. Subsequently, several DyHead blocks containing scale, spatial, and task awareness are stacked; finally, the DyHead output is used to model mask detection task centers and boundaries.

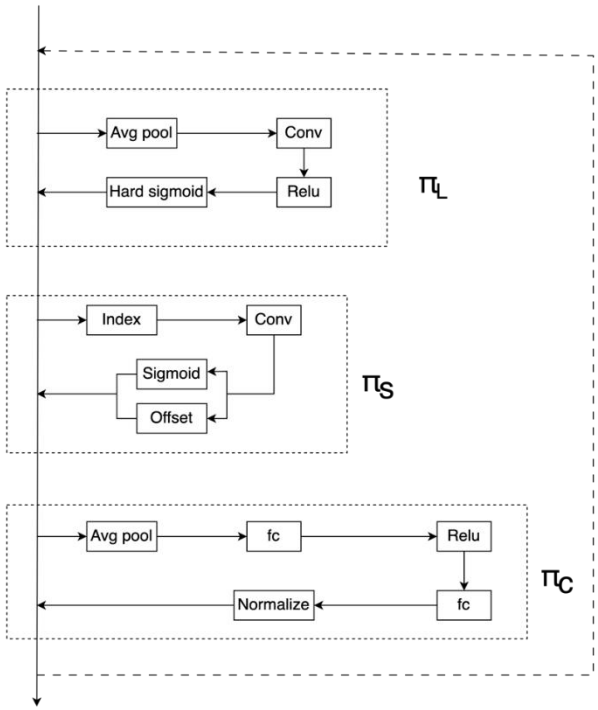


Figure 5. Structure of DyHead.

In this paper, the morphology and behavioral characteristics of whether individuals wear masks remain relatively consistent, allowing the model to integrate easily with Feature Pyramid Network (FPN) structures. The model further extracts multi-scale features by employing the DyHead block to fuse extracted base feature maps.

Figure 6 shows a comparison of heatmaps before and after introducing the DyHead into the original YOLOv8n algorithm. After replacing the DyHead, we observe that the heatmap exhibits more precise coverage, a stronger focus on the targets, and weakened interference from complex backgrounds.

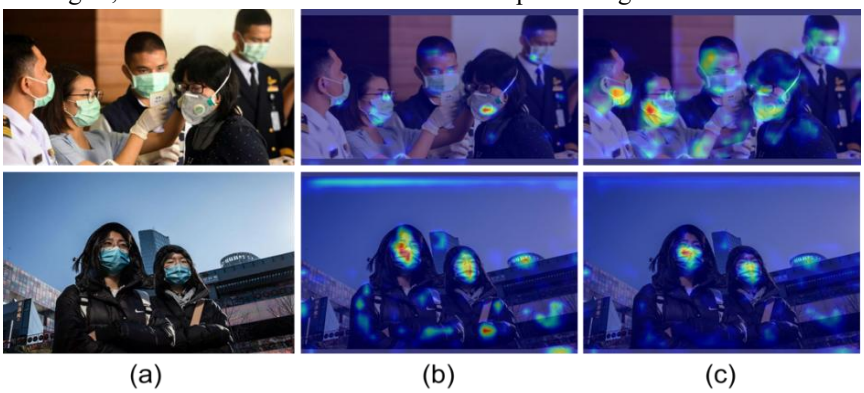


Figure 6. Comparison of heatmaps before and after introducing DyHead. (a) Original image. (b) Heatmap of YOLOv8n's original detection head. (c) Heatmap after YOLOv8n introduces the DyHead detection head.

IV. Experimentation and Results Analysis

A. Experimental Environment and Data

The operating system used for this study was Ubuntu 20.04 LTS, running on an Intel® Core™ i7-9700K CPU and a GeForce RTX3080Ti GPU with 12GB of VRAM and 32GB of RAM. The model was trained based

on Baidu Paddle version 2.4, utilizing CUDA 11.6.

A custom-built dataset of facial mask images, primarily obtained from the internet, comprising 8000 images was employed. The mask data was converted into YOLO format, bifurcated into two categories (category 0: mask-wearing, category 1: without mask), and accompanied by bounding box coordinates denoting target positions. The labeled information of the training dataset is illustrated in Figure 7. Analyzing the dataset and generating visual representations, as depicted in Figure 7, revealed insights. In Figure 7(c), 'x' and 'y' signify the central point position of the target box, where darker colors denote a more concentrated distribution of target box central points. In Figure 7(d), 'width' and 'height' denote the objects' width and height within the images. Figure 7(c) and Figure 7(d) analysis indicate a relatively uniform distribution of objects within the dataset, aligning with typical real-world scenarios. However, Figure 7(a) highlights an imbalance between masked and unmasked data within the dataset, which will be alleviated through mosaic data augmentation.

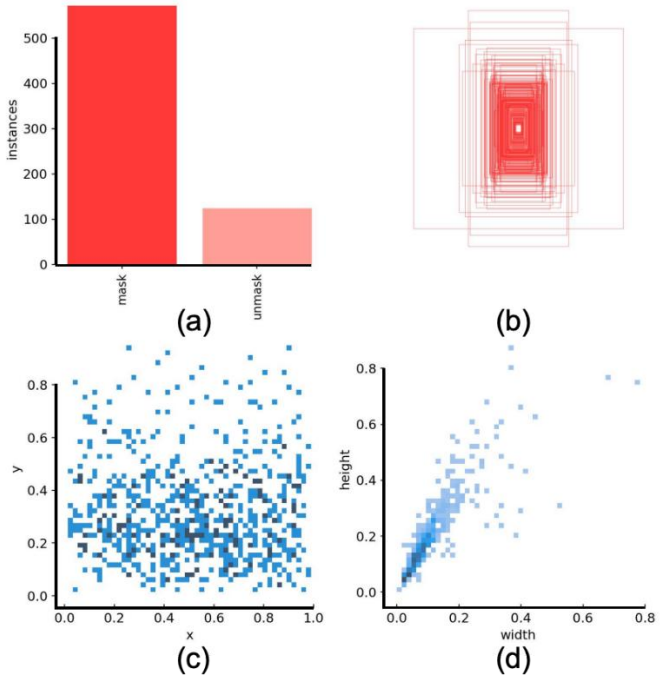


Figure 7. Dataset analysis. (a) Distribution of object categories within the dataset. (b) Dataset bounding boxes. (c) Distribution of object central point positions within the dataset. (d) Distribution of object sizes within the dataset.

B. Evaluation Metrics

Precision (P), recall (R), and mean average precision (mAP) serve as the pertinent metrics for assessing model performance [29]. The formulae for these performance indicators are depicted in Eq. (2)–(6):

1). The Intersection over Union (IOU), denoted as IOU, measures the overlap between the model's predicted and actual bounding boxes. The IOU value is calculated as the intersection area between the real and predicted bounding boxes divided by their union area. This metric is commonly used to evaluate the accuracy and performance of object detection models.

$$\mathrm{IOU} = \frac{\mathrm{A} \cap \mathrm{B}}{\mathrm{A} \cup \mathrm{B}} \tag{2}$$

The expressions A and B, respectively, denote the predicted bounding box and the actual bounding box.

2). Precision (P), or accuracy, refers to the percentage of correctly predicted positive samples out of all detected samples. It pertains to the predictive results:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{3}$$

Where TP represents the number of samples where the detected mask-wearing category matches the actual mask-wearing category, and FP represents the number of samples where the detected mask-wearing category

does not align with the actual mask-wearing category.

3). Recall, denoting sensitivity, signifies the proportion of correctly predicted positive samples out of all actual positive samples:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

FN represents the number of samples where the true target remains undetected.

4). Average Precision (AP): The area under the precision-recall curve represents the AP value for a particular category.

$$A_P = \int_0^1 P\,(r)dr \tag{5}$$

Where N denotes the number of detected categories, set to 2 in this context, corresponding to wearing and not wearing masks.

5). Mean Average Precision (mAP): Averaging each category's AP.

$$\text{mAP} = \frac{\sum_{i=1}^{N} A_{P,i}}{N} \tag{6}$$

6). The model's complexity is assessed using parameters, floating point operations per second (FLOPs), and comparing the detection speed through the speed metric.

C. Model Training

We conducted training on both the training and validation datasets for 200 epochs. The final loss curve is depicted in Figure 8. The three computed losses by the model are:

Box_loss: Measures the disparity between the predicted and actual box positions within the object detection model. A lower box loss indicates an accurate prediction of the target's location.

Cls_loss: Measures the accuracy of the object detection model in classifying different categories. A lower class loss indicates higher precision in distinguishing between various categories.

Dfl_loss: This is an improved loss function designed to better address the issue of sample imbalance in object detection tasks. This loss function achieves this by dynamically adjusting the weights of each predicted bounding box to handle samples of different difficulty levels.

The YOLOv8n-SLIM-DYHEAD algorithm's validation dataset loss reaches its minimum after approximately 150 iterations, indicating the model's convergence. However, beyond 200 iterations, the loss increases, suggesting overfitting of the model.

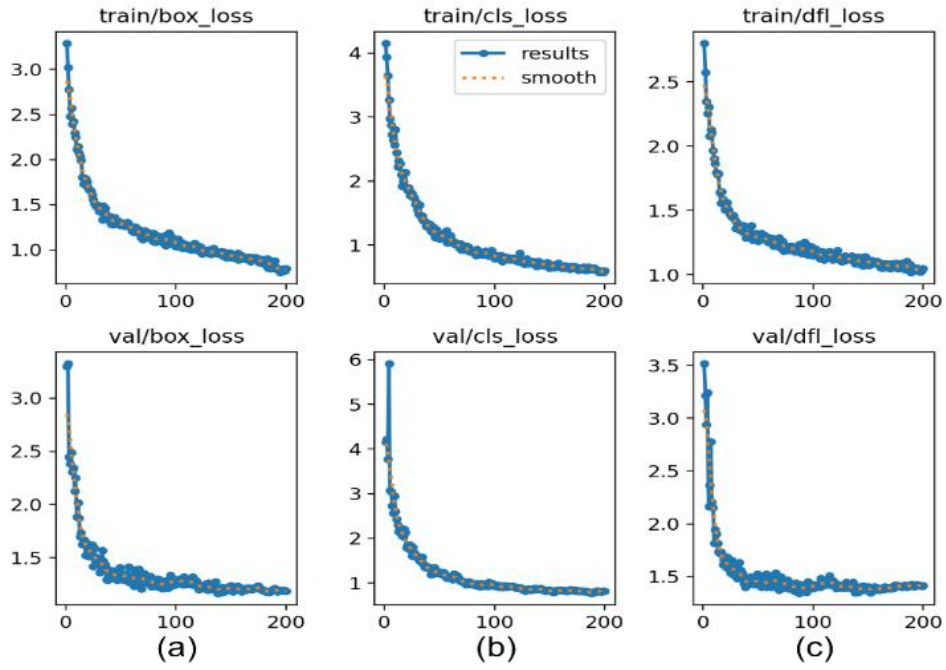


Figure 8. Illustrates the model's losses on the training and validation datasets. (a) Bounding box regression loss. (b) Class loss. (c) Dynamic focal loss.

D. Ablation Experiment

We conducted ablation experiments to assess the impact of the modular integration on the improved algorithm. Table 2 facilitates a comparative analysis by training six distinct sets of data. The original YOLOv8n algorithm serves as the baseline, while the symbol' + denotes the integration of modular enhancements.

Introducing the Mosaic and DyHead modules results in a substantial increase in the model's mAP value. As Tables 1 and 2 indicate, incorporating the slim-neck significantly reduces model parameters and computational complexity with negligible impact on mAP. Considering the cumulative effect, simultaneous application of the three enhancement modules yields optimal results. Compared to the baseline YOLOv8n algorithm, there is a respective improvement of 2.151% and 5.503% in mAP50 and mAP50-95, respectively. This further validates the feasibility of the enhanced algorithm.

Table 2. Comparative analysis of ablation experiments

Model	P(%)	R(%)	mAP @0.5(%)	mAP @0.5:0.95(%)
YOLOv8n	0.934	0.888	0.937	0.595
YOLOv8n+Mosaic	0.930	0.887	0.941	0.621
YOLOv8n+Slim-neck	0.923	0.889	0.938	0.602
YOLOv8n+DyHead	0.941	0.903	0.951	0.630
YOLOv8n+Mosaic+DyHead	0.942	0.905	0.955	0.650
YOLOv8n+Mosaic+Slim-neck+DyHead (YOLOv8n-SLIM-DYHEAD)	0.935	0.894	0.957	0.650

E. Comparison of Detection Algorithms

To comprehensively assess the performance of the improved algorithm in mask detection, we conducted experiments on the same dataset using a consistent experimental platform. The proposed YOLOv8n-SLIM-DYHEAD model was trained alongside existing object detection algorithms such as YOLOv4, YOLOv5l, and YOLOv8n. The performance metrics are presented in Table 3. A comparative analysis reveals that the YOLOv8n-SLIM-DYHEAD model, an enhancement of YOLOv8n, exhibits marginal increases in FLOPs and parameter computation. However, compared to YOLOv8n, it demonstrates significant performance improvements, with a 2.1 percentage point increase in mAP @0.5 and a 5.5 percentage point increase in mAP @0.5:0.95. This indicates a well-balanced trade-off between model lightweight and algorithmic performance, surpassing some standard algorithms.

Table 3. Performance comparison of improved algorithms and existing object detection algorithms

Model	FLOPs (GB)	Parameters (MB)	mAP @0.5(%)	mAP @0.5:0.95(%)	Speed (ms)
YOLOv4	60.3	34.2	0.913	0.617	7.1
YOLOv5n	4.1	1.9	0.915	0.580	1.9
YOLOv5l	107.9	46.0	0.939	0.615	7.3
YOLOv8n	8.2	3.01	0.937	0.595	2.1
YOLOv8n-SLIM-DYHEAD	8.9	3.28	0.957	0.650	2.5

F. Comparison of Detection Results

Figure 9 visually presents the comparative results of the algorithm. The top row displays the detection results of the original YOLOv8n algorithm, while the bottom row showcases the results of the YOLOv8n-SLIM-DYHEAD algorithm. As depicted in Figure 9 (a), YOLOv8n-SLIM-DYHEAD accurately identifies all large targets, whereas YOLOv8n misses one, as highlighted in the yellow box. In Figure 9 (b), the original algorithm has one omission for medium-sized targets, while YOLOv8n-SLIM-DYHEAD accurately recognizes medium-sized targets, with an improvement in confidence scores. Figure 9 (c) illustrates that the original YOLOv8n misclassifies a suitcase as a mask-wearing target for small targets. In Figure 9 (d), across targets of various hierarchical sizes, YOLOv8n-SLIM-DYHEAD successfully identifies all, while the original YOLOv8n algorithm experiences omissions.

The suboptimal performance of the YOLOv8n algorithm in Figure 9 is attributed to the variability in target size, complex backgrounds, and severe occlusions. Similarities between non-target and target morphologies, especially for small targets, contribute to misjudgments. The results indicate the insufficient detection robustness of YOLOv8n across targets of different sizes. Conversely, the YOLOv8n-SLIM-DYHEAD model demonstrates enhanced generalization, effectively reducing omissions and false positives across targets of various sizes.

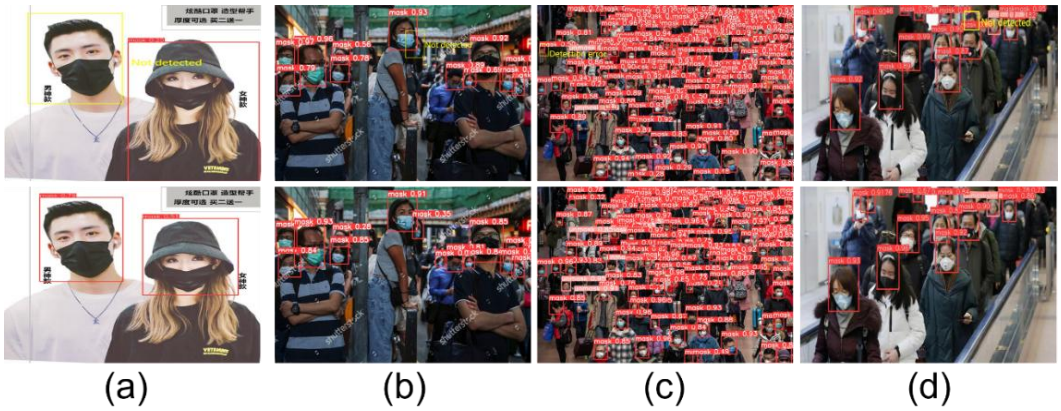


Figure 9. Visual comparison of targets at different scales. (a) Large targets. (b) Medium targets. (c) Small targets. (d) Large, medium, and small targets.

V. CONCLUSION

This paper introduces an enhanced mask detection algorithm by incorporating the DyHead structure, unifying attention mechanisms across different target detection heads. Attention mechanisms between feature hierarchies facilitate scale perception, while attention mechanisms between spatial positions enhance spatial perception. Additionally, using the Mosaic method to process the training set improves the model's generalization for practical detection scenarios, making it more suitable for mask-wearing detection scenes. Experimental results demonstrate that, compared to the original YOLOv8n algorithm, the proposed YOLOv8n-SLIM-DYHEAD achieves a 2.1 percentage point increase in mAP @0.5 and a 5.5 percentage point increase in mAP @0.5:0.95, with a detection time of 11.4 ms, meeting real-time detection requirements. However, in complex background scenarios, detection accuracy could be better. Therefore, addressing how to adapt the model to a broader and more complex range of detection scenarios remains an unresolved challenge.

ACKNOWLEDGEMENT

This research was funded by the Research Project of Fujian Province Young and Middle-aged Teacher Education Research (grant number JAT191226), the Baidu Pinecone "Great Country Intelligent Craftsman" Excellent Cooperation Project and the Key Research Project of Fujian Polytechnic of Information Technology (grant number ZK2023-07).

REFERENCES

[1] Gupta S, Sreenivasu S V N, Chouhan K, et al. Novel face mask detection technique using machine learning to control COVID'19 pandemic[J]. Materials Today: Proceedings, 2023, 80: 3714-3718.

[2] Putra R M, Yossy E H, Saputro I P, et al. Face mask detection using convolutional neural network[C]//2023 8th International Conference on Business and Industrial Research (ICBIR). IEEE, 2023: 133-138.

[3] Kumar B A, Bansal M. Face mask detection on photo and real-time video images using Caffe-MobileNetV2 transfer learning[J]. Applied Sciences, 2023, 13(2): 935.

[4] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.

[5] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.

[6] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[7] Kumar A, Kalia A, Kalia A. ETL-YOLO v4: A face mask detection algorithm in era of COVID-19 pandemic[J]. Optik, 2022, 259: 169051.

[8] Wu P, Li H, Zeng N, et al. FMD-Yolo: An efficient face mask detection method for COVID-19 prevention and control in public[J]. Image and vision computing, 2022, 117: 104341.

[9] Wang C, Zhang B, Cao Y, et al. Mask Detection Method Based on YOLO-GBC Network[J]. Electronics, 2023, 12(2): 408.

[10] Liberatori B, Mami C A, Santacatterina G, et al. Yolo-based face mask detection on low-end devices using pruning and quantization[C]//2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO). IEEE, 2022: 900-905.

[11] Kolpe R, Ghogare S, Jawale M A, et al. Identification of face mask and social distancing using YOLO algorithm based on machine learning approach[C]//2022 6th International conference on intelligent computing and control systems (ICICCS).

IEEE, 2022: 1399-1403.
[12] Zhao Z, Liu X, Hao K, et al. PIS-YOLO: Real-Time Detection for Medical Mask Specification in an Edge Device[J]. Computational Intelligence and Neuroscience, 2022, 2022.
[13] Zhao G, Zou S, Wu H. Improved Algorithm for Face Mask Detection Based on YOLO-v4[J]. International Journal of Computational Intelligence Systems, 2023, 16(1): 104.
[14] Howard A G, Zhu M, Chen B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint arXiv:1704.04861, 2017.
[15] Zhang X, Zhou X, Lin M, et al. Shufflenet: An extremely efficient convolutional neural network for mobile devices[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 6848-6856.
[16] Talaat F M, ZainEldin H. An improved fire detection approach based on YOLO-v8 for smart cities[J]. Neural Computing and Applications, 2023, 35(28): 20939-20954.
[17] Lin B. Safety Helmet Detection Based on Improved YOLOv8[J]. IEEE Access, 2024.
[18] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.
[19] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263-7271.
[20] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv:1804.02767, 2018.
[21] Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. Preprint arXiv:2004.10934 (2020).
[22] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2117-2125.
[23] Xu S, Guo Z, Liu Y, et al. An improved lightweight yolov5 model based on attention mechanism for face mask detection[C]//International Conference on Artificial Neural Networks. Cham: Springer Nature Switzerland, 2022: 531-543.
[24] Wu D, Jiang S, Zhao E, et al. Detection of Camellia oleifera fruit in complex scenes by using YOLOv7 and data augmentation[J]. Applied Sciences, 2022, 12(22): 11318.
[25] Chollet F. Xception: Deep learning with depthwise separable convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1251-1258.
[26] Ma N, Zhang X, Zheng H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 116-131.
[27] Li H, Li J, Wei H, et al. Slim-Neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles[J]. arXiv preprint arXiv:2206.02424, 2022.
[28] DAI X, CHEN Y, XIAO B, et al. Dynamic Head: Unifying object detection heads with attentions[C]//Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, June 20-15, 2021. Piscataway: IEEE, 2021: 7373-7382.
[29] Liu Y, Lu B H, Peng J, et al. Research on the use of YOLOv5 object detection algorithm in mask wearing recognition[J]. World Scientific Research Journal, 2020, 6(11): 276-284.