A High-Accuracy Deformable Model for Human Face Mask Detection

Xinyi Gao, Minh Nguyen, and Wei Qi Yan

Auckland University of Technology, Auckland 1010 New Zealand

Abstract. Human face mask detection leverages computer vision technology to discern whether individuals in images or videos are wearing masks. Ensuring proper mask usage is crucial in settings such as hospital operating rooms and flu clinics. As deep learning advances, mask detection has emerged as a significant research area within the computer vision field. In this paper, we propose a deformable state-of-the-art (DSOTA) model based on Deformable ConvNets v2 (DCNv2) and YOLOv8(i.e., You Only Look Once). We use this new model to improve the accuracy of mask detection. Our experimental results show that the integration of DCNv2 and YOLOv8 significantly improves the accuracy of mask detection. The average highest accuracy rate of the YOLOv8n model is 91.7%, and the average highest accuracy rate of the DSOTAn model is 94.4%. The average highest accuracy rate of the YOLOv8s model is 97.0%, and the average highest accuracy rate of the DSOTAs model is 97.4%. These promising results underscore the potential of our approach for practical applications and further exploration in the computer vision domain.

Keywords: Human Face Mask Detection · Deformable ConvNets · YOLOv8 · Deep Learning.

1 Introduction

Hospitals encounter diverse patients daily, many of whom may carry various viruses. A significant number of these viruses, such as the flu virus, are airborne and can spread quickly. Wearing a mask effectively reduces the risk of contracting infectious diseases. In high-risk environments like operating rooms, medical staff wearing masks can substantially decrease the risk of infection for patients during surgical procedures. Additionally, wearing masks can prevent the cross-infection of pathogens among hospital patients. Consequently, proper mask usage has become mandatory in numerous hospitals and medical institutions. Wearing a mask correctly not only offers self-protection against viruses but also helps mitigate the spread.

In a hospital setting, wearing a mask can significantly reduce the risk of contracting the flu. To ensure patients comply with this requirement, a number of hospitals employ professionals to remind individuals to wear masks at their entrances. However, this approach can lead to the inefficient use and waste of human resources. Deep learning-based face mask detection can assist hospitals in determining whether patients entering the facility are wearing masks

correctly [8] [36], offering a more efficient and cost-effective solution for maintaining public health and safety.

Mask detection, as a detection task, has emerged as a main research direction in the field of computer vision in recent years [12]. With the rapid advancement of deep learning algorithms [9], various neural networks have found widespread application in mask detection tasks. YOLO, a popular algorithm for visual object detection [21], has become a benchmark in single-stage target detection due to its exceptional performance in both recognition accuracy and inference speed.

As a classic single-stage target detection algorithm, YOLO has been extensively utilized in a range of object recognition and detection research endeavours. In recent years, the YOLO series of algorithms have undergone continuous updates and iterations, resulting in significant improvements to the performance of the YOLO model in recognition accuracy and reasoning speed. These advancements have solidified YOLO's position as a leading algorithm for mask detection and other computer vision tasks.

In this paper, we adopt the Face Mask Detection dataset as the training and testing datasets. YOLOv8 [2] is trained on the Face Mask Detection dataset and will eventually be used for human face mask detection tasks. The contribution of this paper is to propose a new model based on Deformable ConvNets v2 [38] and YOLOv8 [32], and apply it to the face mask detection task. By analyzing and comparing the prediction results of the model, it is proved that the improved DSOTA model has achieved better detection accuracy.

2 Related work

2.1 YOLO

YOLO is a state-of-the-art deep learning model for real-time object detection that can detect visual objects from images and videos with high accuracy and speed. Due to its excellent performance, YOLO has been widely employed in various fields.

YOLO model has undergone a number of evolutions since its initial release. The YOLOv1 model took use of GoogLeNet to extract feature maps, with the output by putting the extracted feature map directly on the fully connected layer. This dramatically improves the inference speed of the model. YOLOv2 further enhances the ability to detect small objects by adding Darknet19 as the backbone network [23]. YOLOv3 improved the backbone network and proposed a Darknet53 backbone network framework [24]. It also integrates Feature Pyramid Networks (FPN) into the network. On the basis of YOLOv3 [14], YOLOv4 added Cross Stage Partial (CSP) to the backbone network so that the model can obtain richer gradient information. In addition, YOLOv4 also added modules such as Spatial Pyramid Pooling (SPP) and Path Aggregation Network (PAN) to fuse features and reduce the amount of calculation [3]. YOLOv5 improves SPP and proposes Spatial Pyramid Pooling-Fast (SPPF) [33]. YOLOvX joins SimOTA label assignment strategy and takes advantage of decoupling operations

in the head. YOLOv6 makes use of the EfficientRep backbone. Rep-PAN was also designed additionally [15]. The concept of an Extended Efficient Layer Aggregation Network (E-ELAN) [26] was proposed in YOLOv7 model [31]. YOLOv8 has been improved on the basis of YOLOv5. In the backbone of the network, the C3 module is replaced with C2f. In the Head, decoupling operations are also used. As the latest YOLO model, YOLOv8 outperforms other YOLO models. At the same time, YOLOv8 is also the basic model in this paper, the specific structure will be explained in the methodology [13]. In summary, YOLO is a state-of-the-art object detection model, which remains the most popular choice for object detection tasks.

2.2 Deformable ConvNets

Traditional convolution does not work very well while facing visual objects with complex deformations. In response to this, Deformable ConvNets (DCN) were proposed [4]. DCN is an improvement over traditional convolution operations. By including a learnable offset in the receptive field, the receptive field is more flexible. The improved receptive field is closer to the actual shape of the object and has better performance on the object detection task. After using DCN, the characteristics of the network will be more easily affected by irrelevant image content, which in turn will affect the performance.

To improve this problem Deformable ConvNets v2 (DCNv2) is proposed. Improving the shortcomings of DCN requires the new DCN to have stronger training and modeling capabilities [28]. Therefore, DCNv2 introduces deformable convolution structure with more layers of the feature extraction network. The modulation is also introduced in deformable convolution. Modulation is simply weight and more accurate feature extraction are achieved by assigning different weights. This makes the feature extraction process more focused on the effective information area. These improvements effectively improve the performance of DCNv2, making DCNv2 perform better than DCN in object detection.

2.3 Mask Detection

In recent years, YOLO was employed in the field of mask detection. Liu et al. adopted the YOLO model to replace the Mask R-CNN with ResNet as the backbone [19] [22]. By using the YOLO model to reduce the computational cost of the automatic mask detection with RNN, the processing speed is increased without reducing the accuracy. In addition, they also utilized simple CNAPs to improve classification performance. Yu et al. achieved a mAP of 98.3% by using a YOLOv4-based model for mask detection on a dataset of 10,855 images [37]. Abbasi et al. adopted YOLOv4+CNN to perform mask detection tasks [1]. Among them, YOLOv4 is employed as a target detector for mask detection, and a fast and efficient CNN model for classification. An accuracy rate of 99.5% was obtained through this model.

Wu et al. proposed a FMD-YOLO by using Im-Res2Net-101 as the feature extractor of the network [30]. In the end, FMD-YOLO got mAP50 values of 92.0%

and 88.4% on the dataset. Wang et al. created a PP-YOLO-Mask model using PP-YOLO based on YOLOv3 [10]. The final experimental results show that the model has obtained a mAP value of 86.69% and has faster accuracy and detection speed than YOLOv3. Degadwala et al. took use of a pre-programmed YOLOv4 model to detect medical mask models [5]. After extensive testing, the model achieved an accuracy of 98.90%. The YOLO model used in most of the work is based on YOLOv3 or YOLOv4, and there are few tasks for mask detection using YOLOv8.

3 Methodology

In this paper, YOLOv8 is employed for mask target detection, a DSOTA model based on the YOLOv8 for object detection is proposed. The YOLOv8 model is the latest object detection algorithm based on YOLOv5. The specific structure of the model includes: Backbone, Neck and Head. We describe each structure of YOLOv8 in the following sections. We also detail the difference between our proposed model and YOLOv8.

3.1 Backbone

Backbone is the backbone network of YOLOv8 for extracting image features. The network structure of Backbone still uses the Cross Stage Partial (CSP) DarkNet structure [27]. A simple model diagram of the network is shown in Fig. 1.

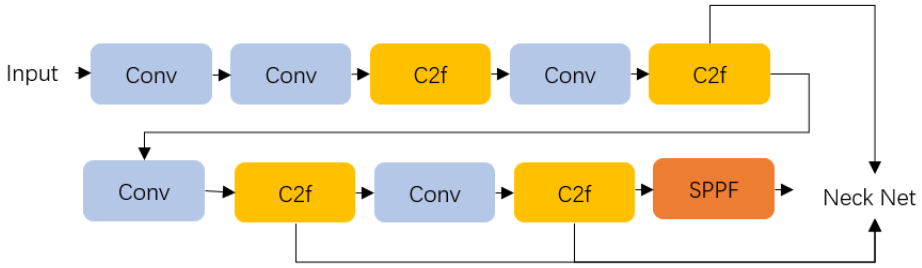

Fig. 1. Structure of Backbone network of YOLOv8

We see that the network is composed of Conv+BN+SiLU (CBS) layer, C2f layer and SPPF layer in terms of specific structure. Compared with YOLOv5, YOLOv8 changes C3 in the backbone to C2f. In terms of design, the design of the C2f module refers to the C3 module of YOLOv5 [16] and the E-ELAN module of YOLOv7 [26]. This design not only ensures the lightweight of YOLOv8, but also helps the model to obtain richer gradient flow information. The C2f module consists of two convolutional layers and Bottlenecks. Bottleneck is a residual

module that also consists of two convolutions. The C2f module divides multiple tensors of the channel dimension through Split [25]. The C2f module stitches together multiple tensors of the channel dimension through Concat.

The SPPF layer improved by SPP is applied to improve the detection speed of the model. There is one CBS convolutional layer and three maxpooling layers in SPPF [34]. It concatenates the feature map without max pooling and the feature map obtained after each increase of max pooling to achieve feature fusion. As shown in Fig. 2.

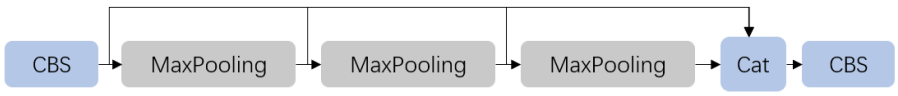

Fig. 2. SPPF module

3.2 Neck

Neck is placed between the backbone and the head in the overall network structure. Neck structurally takes use of the PAN framework and the FPN framework for multi-scale feature fusion. As the depth of the network gradually deepens, there are more and more convolution operations. The increase in convolution operations may cause information loss. Using multi-scale feature fusion can help the network reduce this information loss and make better use of the features extracted by the backbone. The structure of Neck is shown in Figure 3. In YOLOv8, the C2f module is widely used in backbone and neck to extract and fuse multi-scale features and build a more refined target detection model.

3.3 Head

The role of Head layer is to output prediction results, predicting the location and category of objects. In the Head layer, YOLOv8 uses Decoupled-Head. It also separates the classification head and detection head in the Head layer. In addition, Anchor has also been changed from Anchor-Based to Anchor-Free. Anchor-Free means that the anchor is no longer present, but is positioned directly through the center of the object and detected or identified.

3.4 C2fDCNv2

To improve the detection accuracy, we propose the DSOTA model. We combine DCNv2 with C2f in the network and implement a new module called C2fDCNv2. The structure diagram of this module is shown in Fig. 4.

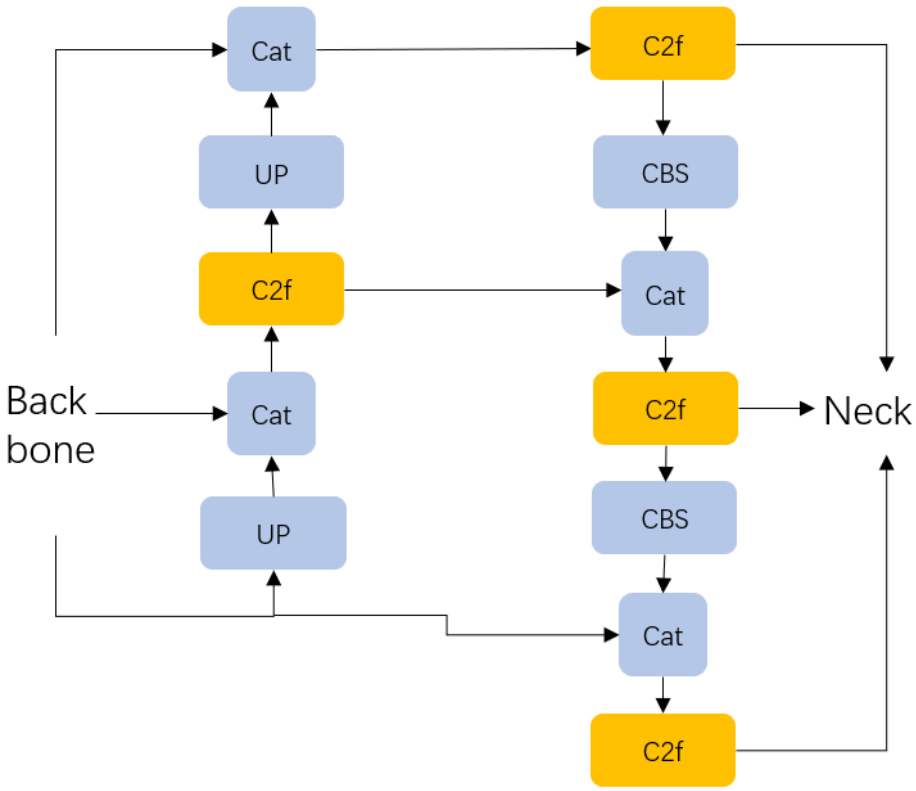

Fig. 3. Structural of the Neck part

The C2fDCNv2 module includes two convolutional layers, a series of BottleneckDCNv2 modules for extracting and fusing features of different scales. In the first half of the C2fDCNv2 module, the input features are extracted through the first convolutional layer to extract high-level features, and then divided into two parts. The two parts of features are sent to multiple BottleneckDCNv2 modules for further processing of different scales and feature fusion. In the second half of the C2fDCNv2 module, the processed features are stitched together by concat, and then sent to the cv2 convolutional layer for final feature extraction and output. The BottleneckDCNv2 module is improved based on the standard bottleneck module. The standard bottleneck module consists of two convolutional layers. We replace these two convolutional layers with DCNv2 to implement the deformable convolution function. This module is harnessed to perform depth expansion and feature transformation on the input features to extract richer feature information. We propose a DSOTA model based on the C2fDCNv2 module. The structure of the backbone part of the DSOTA model is shown in Fig 5. The Neck part and Head part of the model are the same as those introduced before. Fur-

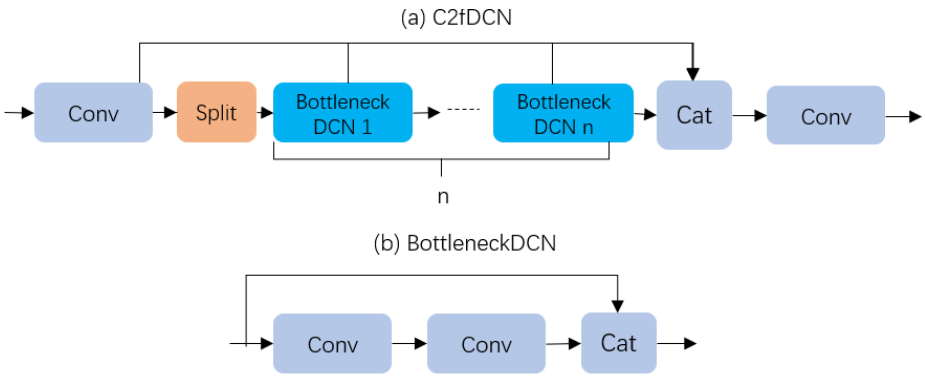

Fig. 4. C2fDCN module

thermore, we doubled the width of the DSOTA model, and kept the number of channels and network layers unchanged. The improved model is called DSOTAs model. We refer to the base model as DSOTAn.

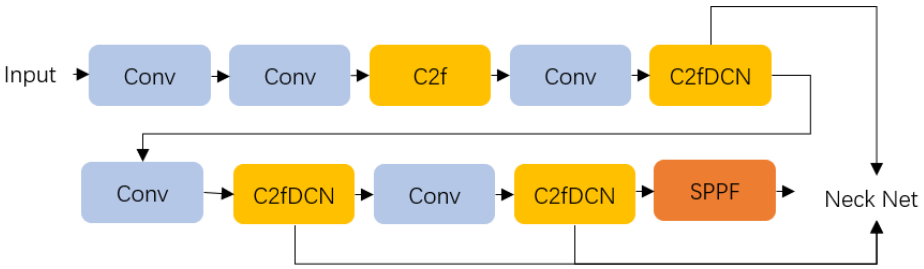

Fig. 5. The Backbone structure of DSOTA model

3.5 Loss Function

Before calculating the loss function, positive and negative samples need to be determined. We use Task Aligned Assigner as the positive and negative sample assignment strategy for the model. Task Aligned Assigner is a sample assigner in the Training Objectives for Object Detection (TOOD) framework [6]. The purpose is to assign each anchor to the closest ground-truth object in order to better match the features between the object and the anchor. Specifically, Task Aligned Assigner will match each anchor point with the corresponding real target [18]. Anchor points whose matching degree is higher than a certain threshold is marked as positive samples and other anchor points will be marked as negative samples. The specific formula is as follows,

$$t = s^{\alpha} \times u^{\beta} \tag{1}$$

where s are classification scores, u is the IoU value, α and β are weight hyperparameters, t implements Task Alignment Assigner by optimizing s and u. By using this allocation strategy, our model can better learn the key features in the object detection task, focusing on high-quality anchors.

Intersection over Union (IoU) is a measure of the degree of overlap between two collections [11]. In computer vision, IoU is often used to evaluate the performance of object detection algorithms. Specifically, the intersection ratio measures the degree of overlap of two sets by calculating the ratio between their intersection and union. Let the intersection of sets A and B be C, and their union be D, then the intersection-union ratio can be expressed as,

$$IoU(A, B) = \frac{|C|}{|D|} \tag{2}$$

where $|C|$ represents the number of elements in set C, and $|D|$ represents the number of elements in set D. If the IoU value is larger, it means that the degree of overlap between the two sets is higher, and the performance of the target detection algorithm is better.

On the loss function, YOLOv8 divides the loss function into classification loss and regression loss and discards the objectness loss. The classification loss uses Binary Cross-Entropy Loss (BCE Loss) [29]. The BCE loss measures the specific performance of the model by calculating the cross-entropy between the output of the model and the real label.

$$BCE_{loss} = -wylogp + (1 - y)log(1 - p) \tag{3}$$

where y is the labeled true value, and p is the predicted value of the model output and w is the weight. The regression loss is composed of two parts: CIoU loss [7] and Distribute Focal Loss (DFL) [17]. CIoU loss can more accurately evaluate the distance between the predicted box and the real object. This is because the CIoU loss takes more account of the distance between the center of the predicted frame and the center of the real object, the difference between the aspect ratio and the influence of the overlapping area. The specific formula is as follows,

$$\mathrm{CIoU}_{loss} = 1 - [IoU - \frac{d^2}{c^2} - \alpha \frac{v}{(1 - IoU) + v}] \tag{4}$$

where d^2 represents the Euclidean distance between the center points, c^2 represents the diagonal length of the predicted frame and the real target, v represents the aspect ratio difference between the predicted frame and the real target, α is an adjustable parameter, the general value is 0.5.

DFL is a loss function used to alleviate the class imbalance problem and sample imbalance problem in object detection tasks. It can make the model pay more attention to rare class samples. The DFL loss function Helps the

model to better learn the distribution of the data set. This can help improve the performance of object detection. The specific formula is as follows,

$$DFL(S_i, S_{i+1}) = -((y_{i+1})log(S_i) + (y - y_i)log(S_{i+1})) \tag{5}$$

where S_i is the sigmoid output of the network, y_i and y_{i+1} are the predicted probabilities, and y is the label value.

In the next experimental part, we will make use of the model to conduct experiments and analyze the obtained experimental results.

4 Experimental Results and Analysis

4.1 Dataset and Environment

In this paper, we utilize the Face Mask Detection dataset [20] as our primary data source. The dataset comes from Kaggle and contains a total of 853 images. The dataset is divided into three different categories: mask wearing, mask not wearing, and mask wearing incorrectly.

We randomly partition the Face Mask Detection dataset into training, validation, and test sets. The training set accounts for 80% of the total dataset, the validation set for 5%, and the test set for 15%.

Throughout our experiments, we employ a Tesla T4 GPU for training, validation, and testing of the models [35]. During the training process for all models used in this paper, we set the number of epochs to 100 and the batch size to 8. Ultimately, we obtain all the necessary experimental results for our analysis.

4.2 Experimental results and analysis

Experimental results In order to better evaluate the performance of the model, we sequentially trained YOLOV8n, DSOTAn, YOLOv8s and DSOTAs on the dataset. YOLOv8n and YOLOv8s are two models of different sizes based on YOLOv8. The two models have the same number of convolutional layers and channels, and the network width of YOLOv8s is twice that of YOLOv8n. The DSOTAs model is also obtained by doubling the network width on the basis of DSOTAn. The parameters of the four groups of models are shown in Table 1.

Table 1. The parameters of YOLOV8n, DSOTAn, YOLOv8s and DSOTAs

Model	Layers	Parameters	FLOPs	Gradients
YOLOv8n	225	3011433	8.2	3011417
DSOTAn	225	3167863	7.7	3167847
YOLOv8s	225	11136761	28.7	11136745
DSOTAs	225	11449351	25.2	11449335

According to the above picture, we see that after 100 epochs of training, all the models get a smooth curve. This means that the model has finally converged.

In addition, after observing the precision graphs of the four models, we found that each model had different degrees of overfitting in the initial stage of training. This may be because the model learned too much noise early in the training phase. The overfitting phenomenon of the model without DCNv2 is more obvious than that of the model with DCNv2. This is because the model using DCNv2 is better able to extract the really useful features in the data than the unused model.

Table 2. The training results for YOLOv8n, DSOTAn, YOLOv8s, DSOTAs

Model Name	Precision	Recall	mAP50	mAP50-95	Training Time (Hour)
YOLOv8n	0.917	0.823	0.903	0.663	2.341
DSOTAn	0.944	0.824	0.909	0.668	2.478
YOLOv8s	0.970	0.893	0.957	0.732	2.305
DSOTAs	0.974	0.899	0.961	0.773	2.368

In terms of specific training results, the precision of mask detection using YOLOv8n is 91.7%, recall is 0.823, mAP50 is 0.903, mAP50-95 is 0.663, and the training time is 2.341 hours. The precision of mask detection using DSOTAn model is 94.4%, recall is 0.843, mAP50 is 0.909, mAP50-95 is 0.668, and the training time is 2.478 hours. The precision of mask detection using YOLOv8s is 97%, recall is 0.893, mAP50 is 0.957, mAP50-95 is 0.732, and the training time is 2.305 hours. The precision of mask detection using DSOTAs model is 97.4%, recall is 0.899, mAP50 is 0.961, mAP50-95 is 0.773, and the training time is 2.368 hours. We obtained Table 2 according to the training results of each model.

Error Analysis The best model in our evaluation makes an average of 21 mistakes out of 4072 instances. In the test results, "red" means that the mask is not worn, "orange" unfolds that the mask is not worn correctly, and "pink" refers to that the mask is worn correctly. Two-thirds of the 21 errors were model mispredictions that the masks were correctly worn on the faces. For example, the second person at the top of the picture from left to right in Figure 6 just covers her mouth and nose with a scarf and does not wear a mask. However, the test results showed that the mask was worn incorrectly. Other errors were identifying people who were not wearing a mask as wearing one. For example, in the case of the bottom pink box in Figure 6, the person in the picture uses a scarf to cover his face completely, but the detection result we get is that he is wearing a mask.

Overall analysis As shown in Fig. 6, we see that the training speed of the two models based on YOLOv8n is relatively slow. Specifically, the training time of YOLOv8n is 0.036 hours slower than YOLOv8s. The training time of DSOTAn is 0.172 hours than the hours of DSOTAs. The detection precision of the two models based on YOLOv8s is higher. Specifically, the accuracy of YOLOv8n is 5.3% lower than that of YOLOv8s. The precision of DSOTAn is 3% lower than

Fig. 6. An example of a misclassified human face mask image

the precision of DSOTAs. In addition, the model with DCNv2 performs better than the model without DCNv2. Although the training time of the model using DCNv2 is higher than that of the model without DCNv2, the precision of the model using DCNv2 is higher. Specifically, the training time of YOLOv8n is 0.37 hours faster than that of DSOTAn. The training time of YOLOv8s is 0.063 hours faster than DSOTAs. The precision of YOLOv8n is 2.7% lower than that of DSOTAn. The precision of YOLOv8s is lower than that of DSOTAs.

From the value of mAP, the mAP value of the model using DCNv2 is higher than that of the model without DCNv2. Specifically, YOLOv8n's mAP50 is 0.006 lower than DSOTAn, and mAP50-95 is 0.004 lower. The mAP50 of YOLOv8s is 0.004 lower than that of DSOTAs, and the mAP50-95 is 0.041 lower.

To provide a more intuitive assessment of the performance disparities and mAP values among the four models, we generated the PR curves for each model, as depicted in Figure 7. The illustration reveals that the overall curve for the model incorporating DCNv2 encompasses a larger area. These experimental findings demonstrate that the integration of DCNv2 into YOLOv8 has a favourable influence on the model's performance.

From the comprehensive experimental results and data presented, it is evident that the incorporation of the DCNv2 module into the YOLOv8 framework substantially enhances the model's performance. Among the four models evaluated in our study, the DSOTAs model emerged as the most effective, attaining an exceptional 97.7% precision and a 0.961 mAP50 score.

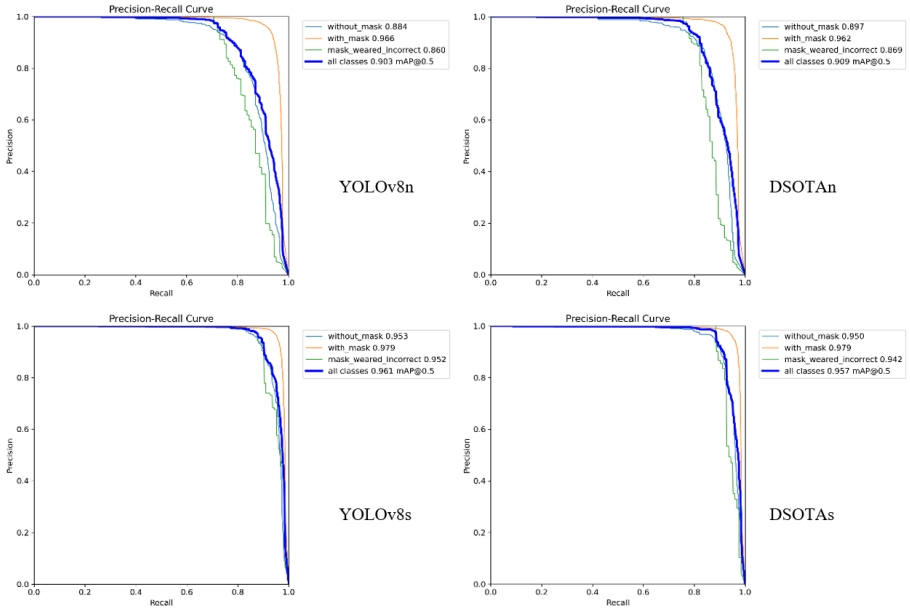

Fig. 7. The PR Curves of the YOLOv8n, DSOTAn, YOLOv8s, DSOTAs

5 Conclusion

In this paper, we highlight the remarkable potential of combining the YOLOv8 model with DCNv2 in human face mask detection. Our systematic comparative analysis and rigorous experimentation prove that integrating the DCNv2 module within YOLOv8 markedly improves mask detection performance. The DSOTAn model delivered an impressive accuracy of 94.4 while the DSOTAs model exceeded expectations, achieving an extraordinary precision of 97.4% in mask detection tasks.

These results have far-reaching real-world implications, particularly in the current global context, where precise and efficient mask detection systems are paramount for public health and safety. The innovations presented in this research have the potential to transform mask detection technology, laying the foundation for more secure and protected environments.

Moving forward, our research efforts should concentrate on developing lightweight models, aiming to strike the optimal balance between a streamlined model and preserving the exceptional accuracy of mask detection demonstrated in this paper. The creation of such lightweight models would enable swifter and more efficient deployment across a range of applications, further underscoring the significance of our discoveries in enhancing public health and safety on a global scale.

References

1. Abbasi, S., Abdi, H., Ahmadi, A.: A face-mask detection approach based on YOLO applied for a new collected dataset. In: International Computer Conference, Computer Society of Iran (CSICC). pp. 1–6. IEEE (2021)
2. Aboah, A., Wang, B., Bagci, U., Adu-Gyamfi, Y.: Real-time multi-class helmet violation detection using few-shot data sampling technique and YOLOv8. arXiv preprint arXiv:2304.08256 (2023)
3. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: YOLOv4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
4. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: IEEE ICCV. pp. 764–773 (2017)
5. Degadwala, S., Vyas, D., Chakraborty, U., Dider, A.R., Biswas, H.: YOLO-v4 deep learning model for medical face mask detection. In: International Conference on Artificial Intelligence and Smart Systems (ICAIS). pp. 209–213. IEEE (2021)
6. Feng, C., Zhong, Y., Gao, Y., Scott, M.R., Huang, W.: TOOD: Task-aligned one-stage object detection. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3490–3499 (2021)
7. Gao, J., Chen, Y., Wei, Y., Li, J.: Detection of specific building in remote sensing images using a novel YOLO-S-CIOU model. Case: Gas station identification. Sensors **21**(4), 1375 (2021)
8. Gao, X., Nguyen, M., Yan, W.Q.: Face image inpainting based on generative adversarial network. In: International Conference on Image and Vision Computing New Zealand (IVCNZ). pp. 1–6. IEEE (2021)
9. Gao, X., Nguyen, M., Yan, W.Q.: A method for face image inpainting based on autoencoder and generative adversarial network. In: Pacific-Rim Symposium on Image and Video Technology. pp. 24–36. Springer (2022)
10. Jian, W., Lang, L.: Face mask detection based on Transfer learning and PP-YOLO. In: IEEE ICBAIE. pp. 106–109. IEEE (2021)
11. Jiang, P., Ergu, D., Liu, F., Cai, Y., Ma, B.: A review of YOLO algorithm developments. Procedia Computer Science **199**, 1066–1073 (2022)
12. Jindal, N., Singh, H., Rana, P.S.: Face mask detection in COVID-19: A strategic review. Multimedia Tools and Applications **81**(28), 40013–40042 (2022)
13. Ju, R.Y., Cai, W.: Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm. arXiv preprint arXiv:2304.05071 (2023)
14. Le, H., Nguyen, M., Yan, W.Q., Nguyen, H.: Augmented reality and machine learning incorporation using YOLOv3 and ARKit. Applied Sciences **11**(13), 6006 (2021)
15. Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., et al.: YOLOv6: A single-stage object detection framework for industrial applications. arXiv preprint arXiv:2209.02976 (2022)
16. Li, J., Liu, C., Lu, X., Wu, B.: CME-YOLOv5: An efficient object detection network for densely spaced fish and small targets. Water **14**(15), 2412 (2022)
17. Li, X., Wang, W., Wu, L., Chen, S., Hu, X., Li, J., Tang, J., Yang, J.: Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. Advances in Neural Information Processing Systems **33**, 21002–21012 (2020)
18. Liu, F., Chen, R., Zhang, J., Xing, K., Liu, H., Qin, J.: R2YOLOX: A lightweight refined anchor-free rotated detector for object detection in aerial images. IEEE Transactions on Geoscience and Remote Sensing **60**, 1–15 (2022)

19. Liu, R., Ren, Z.: Application of YOLO on mask detection task. In: IEEE ICCRD. pp. 130–136. IEEE (2021)
20. Pooja, S., Preeti, S.: Face mask detection using AI. Predictive and Preventive Measures for COVID-19 Pandemic pp. 293–305 (2021)
21. Qi, J., Nguyen, M., Yan, W.Q.: Small visual object detection in smart waste classification using transformers with deep learning. In: International Conference on Image and Vision Computing New Zealand. pp. 301–314. Springer (2022)
22. Qi, J., Nguyen, M., Yan, W.Q.: Waste classification from digital images using ConvNeXt. In: Pacific-Rim Symposium on Image and Video Technology. pp. 1–13. Springer (2022)
23. Redmon, J., Farhadi, A.: YOLO9000: Better, faster, stronger. In: IEEE CVPR. pp. 7263–7271 (2017)
24. Redmon, J., Farhadi, A.: YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
25. Sun, Z., Li, P., Meng, Q., Sun, Y., Bi, Y.: An improved YOLOv5 method to detect tailings ponds from high-resolution remote sensing images. Remote Sensing **15**(7), 1796 (2023)
26. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696 (2022)
27. Wang, C.Y., Liao, H.Y.M., Wu, Y.H., Chen, P.Y., Hsieh, J.W., Yeh, I.H.: CSPNet: A new backbone that can enhance learning capability of CNN. In: IEEE/CVF CVPR workshops. pp. 390–391 (2020)
28. Wang, R., Shivanna, R., Cheng, D., Jain, S., Lin, D., Hong, L., Chi, E.: DCN v2: Improved deep & cross network and practical lessons for web-scale learning to rank systems. In: The Web Conference. pp. 1785–1797 (2021)
29. Wang, Y., Yan, G., Meng, Q., Yao, T., Han, J., Zhang, B.: DSE-YOLO: Detail semantics enhancement YOLO for multi-stage strawberry detection. Computers and Electronics in Agriculture **198**, 107057 (2022)
30. Wu, P., Li, H., Zeng, N., Li, F.: FMD-YOLO: An efficient face mask detection method for COVID-19 prevention and control in public. Image and vision computing **117**, 104341 (2022)
31. Xia, Y., Nguyen, M., Yan, W.Q.: A real-time kiwifruit detection based on improved YOLOv7. In: International Conference on Image and Vision Computing New Zealand. pp. 48–61. Springer (2022)
32. Xiao, B., Nguyen, M., Yan, W.Q.: Fruit ripeness identification using YOLOv8 model. Multimedia Tools and Applications pp. 1–18 (2023)
33. Xue, Z., Lin, H., Wang, F.: A small target forest fire detection model based on YOLOv5 improvement. Forests **13**(8), 1332 (2022)
34. Xue, Z., Xu, R., Bai, D., Lin, H.: YOLO-Tea: A tea disease detection model improved by YOLOv5. Forests **14**(2), 415 (2023)
35. Yan, W.Q.: Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics. Springer (2019)
36. Yan, W.Q.: Computational Methods for Deep Learning: Theory, Algorithms, and Implementations. Springer Nature (2023)
37. Yu, J., Zhang, W.: Face mask wearing detection algorithm based on improved YOLO-v4. Sensors **21**(9), 3263 (2021)
38. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable ConvNets v2: More deformable, better results. In: IEEE/CVF CVPR. pp. 9308–9316 (2019)