RESEARCH ON A MULTI-WEATHER CHALLENGED PEDESTRIAN AND VEHICLE DETECTION METHOD BASED ON IMPROVED YOLOv8s

Jiahuai JIANG[1], Zhigui DONG[2*]

In order to achieve rapid and accurate detection of pedestrians and vehicles under diverse weather conditions, this paper proposes a pedestrian and vehicle detection method based on the improved YOLOv8s. This method aims to flexibly adapt to the identification and detection of target features in different weather conditions. Building upon the YOLOv8s model, we introduce Deformable Convolutional Networks version 2 (DCNv2) deformable convolution and Deformable Attention Transformer (DAT) module (YOLOv8+DCNv2+DAT, referred to as YOLOv8-Def). The model dynamically adjusts the shape, position, and attention weights of convolution kernels and combines methods such as sample splitting and Mosaic data augmentation to enhance detection accuracy. At an Intersection over Union (IoU) of 0.3, the size of the YOLOv8-Def model is 48 MB, with a detection accuracy of 83.4%, a recall rate of 74.8%, and a detection speed of 76 frames per second. The mean Average Precision (mAP@0.5) reaches 82.6%. Compared to the standard YOLOv8s model, the absolute mAP improvement is 5.9%. When compared to Faster-RCNN, the YOLOv8-Def model shows a 3.7% increase in absolute mAP, with a frame per second (FPS) higher than 51. Compared to the RT-DETR model, the YOLOv8-Def model demonstrates a 3.5% increase in absolute mAP, with FPS higher than 25. Experimental results indicate that the YOLOv8-Def model significantly improves detection accuracy while minimizing parameter increase. Moreover, compared to the original YOLOv8s, the detection speed of this model is only reduced by 10 frames per second, demonstrating outstanding practicality. Therefore, this method provides a theoretical foundation for the algorithm research and application of pedestrian and vehicle detection under various adverse weather conditions.

Keywords: improved YOLOv8, multi-weather, object detection, deformable convolution, attention mechanism

1. Introduction

In today's society, adverse weather phenomena are becoming increasingly frequent. The deterioration of climate has resulted in widespread and undeniable

[1] * School of Electronic and Information Engineering, Liaoning Institute of Science and Technology, China, e-mail: 3336545349@qq.com
[2] Prof., School of Electronic and Information Engineering, Liaoning Institute of Science and Technology, China, * Corresponding author, e-mail: dongzhigui@163.com

global social problems, such as haze, heavy rain, and snowstorms. The adverse weather conditions have had broad and profound impacts on human society, affecting daily life, production, and the environment. Apart from the direct effects on routine activities like transportation, agriculture, and energy, extreme weather poses challenges to modern technology and technical applications, particularly in the realm of visual recognition and object detection systems. In diverse meteorological conditions, adverse weather hinders the clear identification and effective detection of target objects. This difficulty makes accurate detection of pedestrians and vehicles exceptionally challenging, significantly impacting the application, promotion, and popularization of visual recognition and object detection systems. Therefore, there is an urgent and crucial need to research and improve visual recognition and object detection algorithms under adverse weather conditions, enhancing the robustness and accuracy of object detection systems.

Traditional object detection algorithms rely on the design of complex feature extractors and classifiers, such as Haar features, HOG, and SIFT. With the maturity and development of deep learning, Convolutional Neural Networks (CNN)[1], especially algorithms like Mask R-CNN [2], Fast R-CNN[3], and Faster R-CNN [4], have significantly improved detection accuracy and speed by introducing mechanisms such as region proposals and Region Proposal Networks (RPN). The emergence of single-stage detection algorithms like YOLO [5] (You Only Look Once) and SSD [6] (Single Shot MultiBox Detector) further simplified the process by treating object detection as a regression problem. These algorithms streamline the workflow, enabling end-to-end detection and training. Not only do they simplify the algorithmic process, but real-time detection is also enhanced, improving the efficiency and accuracy of the detection process.

Currently, algorithms for target detection under adverse weather conditions can be broadly categorized into two types: one involves performing dehazing on the images before conducting target detection [7]. This method can reduce or eliminate the blurring effect in images, improving image quality and clarity, and making targets more easily detectable. However, the dehazing process may lead to the loss of some information in the image, particularly in extreme weather conditions, where useful details or information may be sacrificed. Additionally, performing dehazing incurs additional computational costs, potentially increasing the overall time cost of the detection process. The other type of algorithm involves joint optimization, where dehazing algorithms are jointly optimized with target detection algorithms, as exemplified by IA-YOLO [8]. Joint optimization algorithms take into account the mutual influence between dehazing and target detection tasks, resulting in a more comprehensive model that performs better in addressing target detection challenges under adverse weather conditions. However, joint optimization may be more suitable for specific scenes or particular adverse weather conditions and may lack universality for a broad range of adverse weather

conditions.

In diverse adverse weather conditions, deep semantic features of objects can to some extent alleviate the impact of blurring on image features. By dynamically adjusting the shape and position of convolutional kernels, attention weights, etc., it is helpful to enhance the robustness of the model to images affected by various adverse weather conditions, such as haze. To improve the speed and accuracy of target detection under diverse adverse weather conditions, this paper utilizes the concept of deformability in the spatial dimension for dynamic adjustments and adaptation to process images. We propose an improved method to flexibly capture the non-uniformity, blurriness, and changes in target shapes in images under adverse weather conditions, particularly haze. The YOLOv8-Def model, based on the YOLOv8 framework, incorporates the second-generation deformable convolution (DCNv2) [9] to enable dynamic adjustments of convolutional kernels in the spatial dimension. This adaptation allows for better accommodation of complex target shapes and scene structures. Additionally, a Deformable Attention Transformer (DAT) [10] is introduced to provide greater flexibility for challenging image conditions, such as haze or occlusion, allowing the model to better adapt to changes or deformations in target shapes, further enhancing the model's generalization capability.

2. Based on the improved YOLOv8s object detection model

2.1 Advantages of YOLOv8 algorithm

YOLO (You Only Look Once) is an end-to-end object detection model that can rapidly and efficiently detect multiple objects in an image, predicting their categories and bounding boxes. Currently, YOLOv8 stands as the latest outstanding model (State-of-the-Art, SOTA) in the field of object detection, garnering widespread attention due to its remarkable performance [11]. On the COCO dataset, YOLOv8 demonstrates faster detection speed and higher accuracy when compared to YOLOv7 [12] and YOLOv5 [13] as shown in Fig 1.

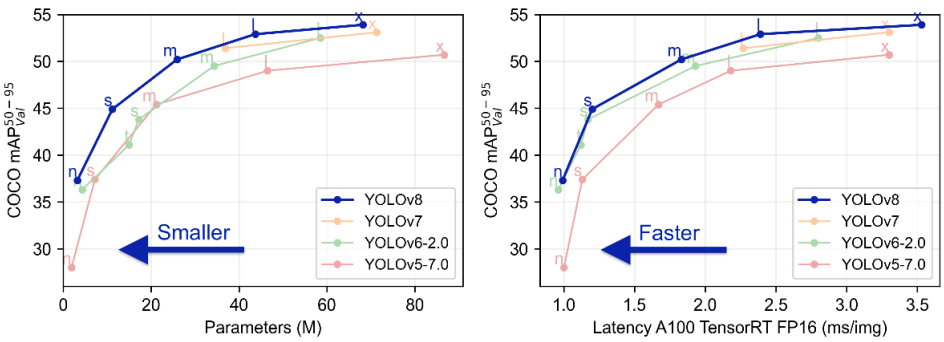

Fig 1. YOLOv5-v8 data comparison

While YOLOv8 retains the CSPDarkNet-53 [14] network as its backbone, it introduces the C2f module, which has a richer gradient flow, to replace the C3 module, achieving model lightweighting. Additionally, YOLOv8 abandons the Anchor-Based [15] approach, opting for Anchor-Free [16] methodology to determine positive and negative samples without anchor boxes. This not only simplifies the process but also surpasses the detection accuracy and speed of models using a two-stage Anchor-Based approach. Furthermore, the head layer of YOLOv8 transitions from a coupled head to the currently prevalent decoupled head structure (as illustrated in Fig.2), separating the classification and detection heads in a more modular fashion.

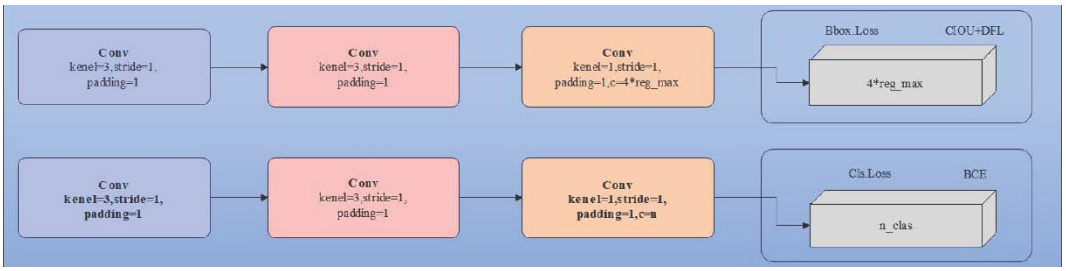

Fig. 2. YOLOv8 decoupled header structure

Another notable advantage of YOLOv8 is the application of the PANet (Path Aggregation Network) [17]. PANet, serving as a feature fusion module, combines features from different levels of the model's backbone network, significantly enhancing the model's ability to detect small objects and objects under challenging conditions.

2.2 Deformable Convolutional Networks v2(DCNv2)

DCNv2 (Deformable Convolutional Networks version 2) is an evolved structure in convolutional neural networks. By introducing the deformable convolution mechanism, DCNv2 allows the convolutional kernel to dynamically adjust in the spatial dimension, providing better adaptation to complex target shapes and scene structures. This enhancement improves the model's perception of subtle features and irregular objects, resulting in significant performance improvements in tasks such as object detection and image segmentation.

The standard convolution calculation involves sampling a set of pixels, denoted as R, from the input feature map. Subsequently, convolutional operations are applied to compute the sampled data, resulting in an output of

$$Y(\mathrm{p}_0)=\sum_{p_n\in R} w(p_n)*X(p_0+p_n) \tag{1}$$

In this context, p_0 represents a specific location in the output feature map, p_n denotes the position offset of the convolutional kernel (the offset relative to the

center within the receptive field), $X(p_0 + p_n)$ signifies the pixel value at the corresponding position in the input feature map, $w(p_n)$ represents the weight parameters within the convolutional kernel.

Deformable convolution, in contrast, involves modifying the sampled data and does not directly alter the shape of the convolutional kernel. In this study, we employ Δp_n to expand the p_n point on the feature map, where $\Delta p_n \,|\, n = 1, 2, 3 \cdots, N$ represents the convolutional kernel offset values predicted through convolution operations. At this juncture, the result of the deformable convolution computation is as follows:

$$Y(p_0) = \sum_{p_n \in R} w(p_n) * X(p_0 + p_n + \Delta p_n) \tag{2}$$

However, the offset values predicted through convolution operations are typically fractional. DCNv1[18] utilized bilinear interpolation during sampling, wherein the pixel value at the current sampling point is dependent on the four integer neighbors surrounding the floating-point position after offset. Nevertheless, a major drawback exposed by DCNv1 is that the new position of the sampling point after offset may exceed the ideal sampling position. This results in convolution points for certain deformable convolutions possibly encompassing areas unrelated to the object content, as illustrated in Fig.3.

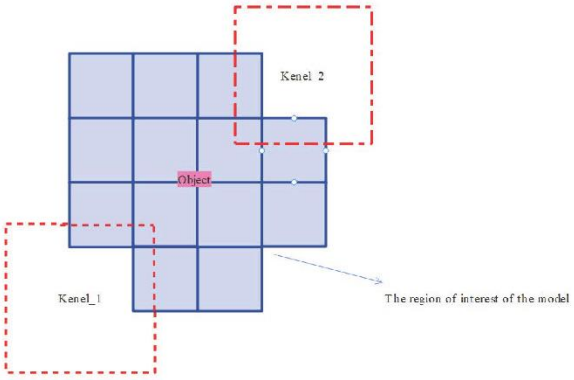

Fig. 3. Sampling position of DCNv1

In comparison to DCNv1, DCNv2 possesses enhanced modeling capabilities for learning deformable convolutions. The augmentation of this modeling capability takes two complementary forms: firstly, equipping more convolutional layers with offset learning capabilities, allowing DCNv2 to control sampling across a broader range of feature levels; secondly, each sample undergoes not only learned offset adjustments but also amplitude modulation through learned features. DCNv2 introduces an additional weighting coefficient $\Delta m, (0 < \Delta m < 1)$ for each sampling point. Introducing Δm to control the magnitude of the offset, the

deformable convolution results in DCNv2 are as follows:

$$Y(p_0)=\sum_{p_n \in R} w(p_n) * X(p_0 + p_n + \Delta p_n) * \Delta m_n \quad (3)$$

Its sampled positions compared to standard ordinary convolution are illustrated in Fig.4.

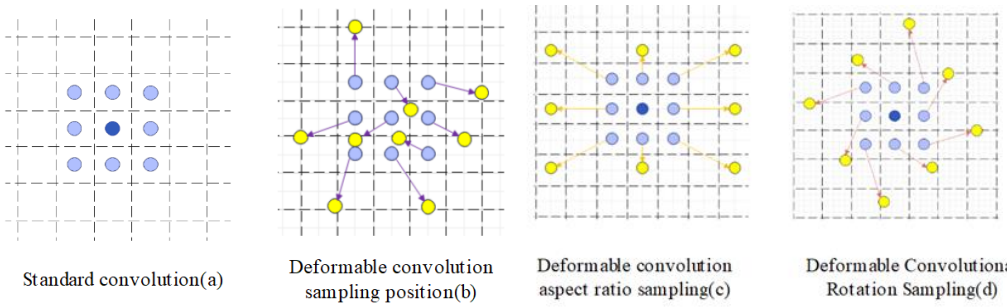

Fig. 4. Comparison of sampling locations

Fig.4(a) illustrates the sampling grid for standard convolution (blue dots), Fig.4(b) depicts the deformed sampling positions based on offset values in deformable convolution (purple arrows), and Fig.4(c) and Fig.4(d) showcase two specific scenarios of deformed sampling positions

2.3 Deformable Attention Transformer (DAT) mould

DAT (Deformable Attention Transformer) is a variant of attention mechanisms commonly employed in neural networks for tasks involving non-uniform shapes, complex spatial relationships, or requiring a more refined distribution of attention. In traditional self-attention mechanisms, attention weights are computed based on the similarity between Query and Key. However, this mechanism may have limitations when dealing with data of non-uniform shapes or complex structures. The deformable attention mechanism introduces learnable offset values, allowing the model to dynamically adjust the correlation at different positions during attention computation, better adapting to the specific shapes or spatial distributions of input data.

Adverse environmental conditions such as haze, rainy weather, and snowfall can result in issues such as degraded image quality, blurred targets, and lost details, posing significant challenges for target detection and recognition. DAT (Deformable Attention Transformer) has the capability to dynamically adjust attention weights based on features in different regions, thereby flexibly adapting to changes in target characteristics under various weather conditions. This contributes to enhancing the model's perceptual ability across different regions. Additionally, targets may exhibit non-uniform shapes due to weather-related occlusion or deformation, presenting challenges for traditional attention mechanisms. DAT, by learning positional adjustment information, can more

accurately capture the non-uniform shapes and spatial relationships of targets, thus improving the model's capability to detect deformed targets. As shown in Fig.5.

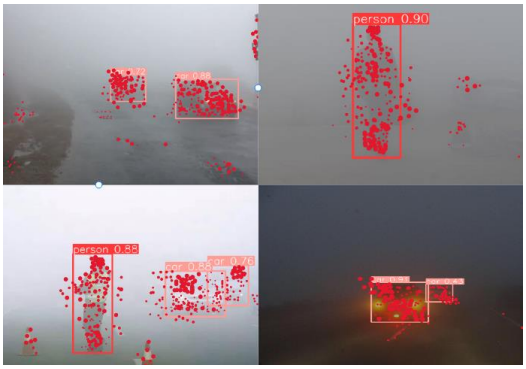

Fig. 5 Visualization of the most important keys

It can be seen from Fig.5, the red circles in the image represent keys that have been moved and have higher Attention Scores. The size of the circle indicates the accumulated attention score, with larger circles corresponding to higher scores. This figure illustrates that DAT has learned to focus on important regions in the input image.

Traditional attention mechanisms process input data (typically feature representations obtained from a certain layer of a neural network) through a linear transformation, such as a fully connected layer or convolutional layer, resulting in three vectors: the Query vector $\overline{Q}$, the Key vector $\overline{K}$, and the Value vector $\overline{V}$. Each Query vector is attentively matched with all Keys, and the relationship between them can be expressed as:

$$\text{Attention}\left(Q,K,V\right)=Soft\max\left(\frac{QK^{T}}{\sqrt{d_{k}}}\right)\times V \tag{4}$$

It is important to note that in DAT, the Query vectors are not computed for attention weights with every position of the global Key vectors. Instead, for each Query, attention is calculated only for sampled positions globally. Moreover, the values are sampled and interpolated based on these positions. Finally, the attention weights obtained for these local positions are added to the corresponding values, as illustrated in Fig.6.

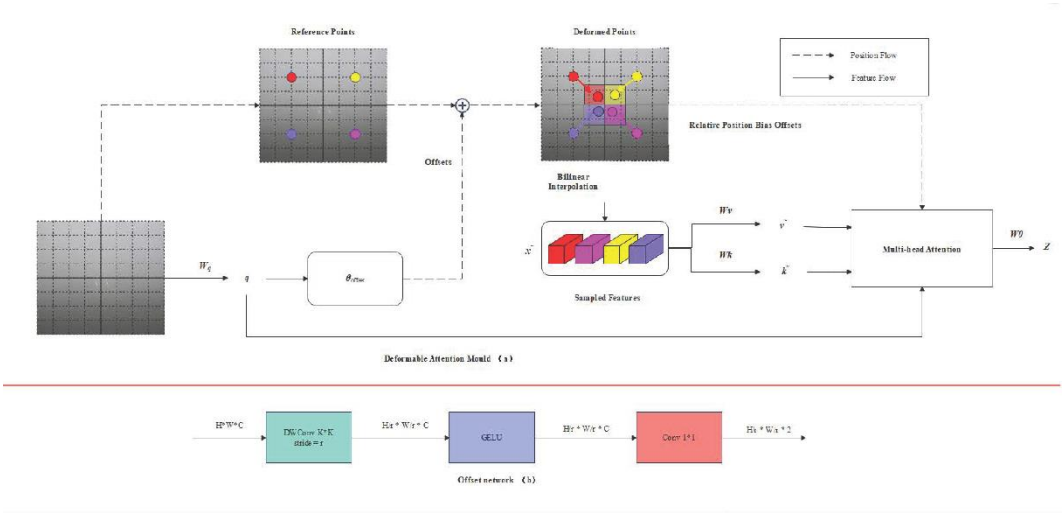

Fig.6 Deformable attention module

Fig.6(a) illustrates the information propagation mechanism of the deformable attention. On the feature map of the image, a set of reference points is uniformly placed, and their offsets are learned from the Query through an offset network. Subsequently, based on these deformed points, we project their deformed Keys and Values from the sampled features. This approach, which utilizes deformed points to calculate relative positional deviations, contributes to strengthening the multi-head attention mechanism, thereby enhancing the performance of the output transformed features. Fig.6(b) depicts the detailed structure of the offset generation network, with the size labeled according to the feature map.

This can effectively reduce computational complexity, enhance the model's flexibility and adaptability, increase attention to local information, and simultaneously reduce the model's parameter count.

2.4 Improved YOLOv8 algorithm

This study adopts the YOLOv8 object detection model as the baseline and replaces a portion of the C2f modules in its backbone with the second-generation deformable convolution, enhancing the model's ability to detect targets of different scales, shapes, and positions. This modification improves the model's feature extraction and adaptability in complex environments. Additionally, the deformable attention mechanism (DAT) is incorporated before the SPPF feature fusion to enhance the model's flexibility in adapting to target shapes. The network structure is illustrated in Fig.7.

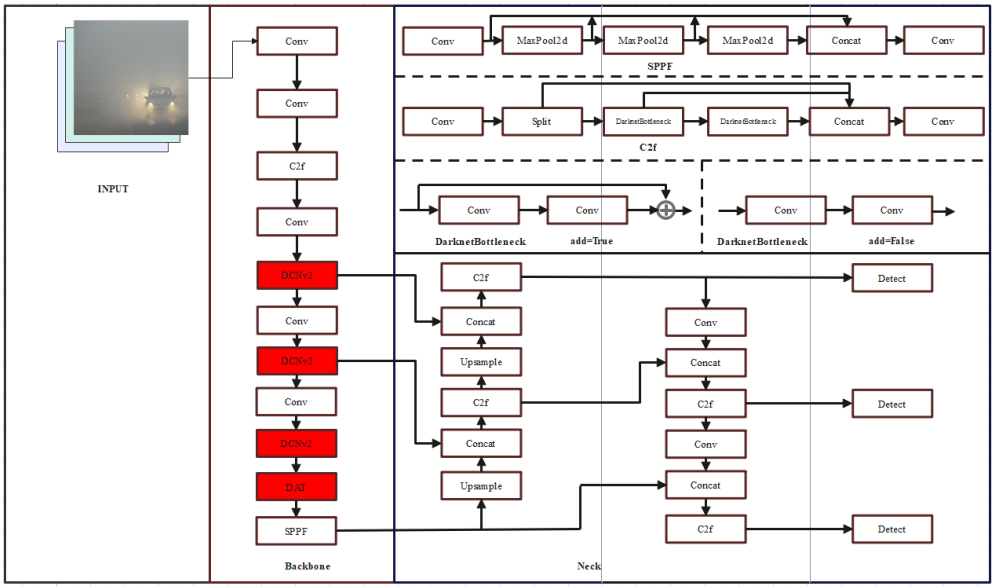

Fig. 7 Network structure of improved YOLOv8 Model

3. Experimental data and methods

3.1 Experimental data and preprocessing

This paper utilizes the Real-world Task-Driven Testing Set (RTTS) dataset. RTTS is a comprehensive real-world dataset encompassing adverse weather conditions such as foggy, snowy, and rainy days. The dataset comprises a total of 4332 images captured under different weather conditions, covering five object categories: cars, pedestrians, buses, bicycles, and motorcycles. Refer to Fig.8 for illustration.

Fig. 8 Example of an RTTS dataset

To facilitate the learning of target features under adverse weather conditions, this paper divides the dataset into training, validation, and test sets in a ratio of 8:1:1. Specifically, the training set comprises 3449 images, while both the validation and test sets consist of 432 images each.

During the training process, the Mosaic data augmentation method embedded in YOLOv8 is employed to enhance data quality. The implementation of this method involves the following steps: Firstly, four images are randomly selected from the dataset. Subsequently, various operations such as translation, flipping, and scaling are applied to these four images. Afterward, the processed images are concatenated based on their positions. Finally, using a matrix approach, fixed regions from the four images are extracted and combined to create a new image containing candidate boxes and other content.

3.2 The software and hardware platform of the experiment

This experiment is based on the PyTorch 1.9.0 deep learning framework, CUDA 11.1, Python 3.8.10, and other deep learning environments for model training and testing. The experiment was conducted on the Ubuntu 18.04 operating system. The CPU used in the experimental platform is a 12-core Intel(R) Xeon(R) Platinum 8352V, and the GPU is an NVIDIA GeForce RTX 4090 with a VRAM capacity of 24GB.

3.3 Experimental parameter setting

The experiment chose yolov8s.pt as the pretraining weight, with the model trained for 200 epochs. To prevent overfitting or a decrease in generalization ability, training was early stopped after 50 epochs without significant performance improvement. The batch size was set to 8, and the optimizer was selected as 'auto.' The initial learning rate was 0.01, confidence threshold set to 0.25, and the intersection over union (IOU) threshold set to 0.3.

3.4 Evaluation index

The experiment employed precision (P), recall (R), mean average precision (mAP), and average frame rate (Frames Per Second, FPS) as evaluation metrics to assess the model's performance. The model parameters were stored in FP32 (4 bytes), and the model size was calculated as the product of the parameter quantity and 4. In the context of size conversions, 1 kilobyte (KB) equals 1024 bytes, and 1 megabyte (MB) equals 1024 kilobytes. The final model size can be expressed as:

$$\mathrm{MB} = \frac{Params \times 4}{1024 \times 1024} \tag{5}$$

If the model has 60 million parameters, the total number of bytes would be 240 million bytes, equivalent to 228 megabytes (MB).

4. Experimental results and analysis

4.1 YOLOv8 version comparison test

YOLOv8 introduces a total of 5 versions, with model sizes ranging from smallest to largest as YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x, denoted as nano, small, medium, large, and x-large (n, s, m, l, x).

YOLOv8n is the smallest model in the YOLOv8 series, but it may not be sufficient for the tasks in this paper. Target detection under certain adverse weather conditions requires a deeper network structure to extract potential information from images, and YOLO8n may produce suboptimal results. YOLOv8x, on the other hand, is the most complex model in YOLOv8. However, this model might be overly complex for the tasks in this paper. Although it can learn well from our dataset, it may lead to overfitting. Therefore, experiments were conducted on these five versions separately on our dataset, and the results are shown in Fig.9.

From Fig.9, it can be observed that YOLOv8s shows the most significant improvement in mAP with a relatively smaller number of parameters, and its fps also reaches around 80 frames. In this study, YOLOv8s is chosen as the base model to ensure that the YOLOv8-Def model achieves higher accuracy and faster detection speed.

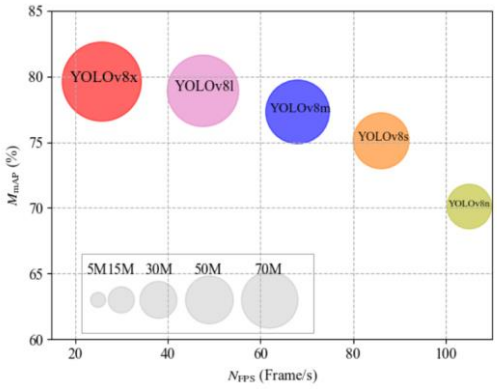

Fig.9. Experimental comparison of different versions of YOLOv8

4.2 Ablation experiment

In order to achieve efficient detection of pedestrians and vehicles in adverse weather conditions, this study builds upon the original YOLOv8s model and proposes a more effective network model. Specifically, we replace some C2f modules in the network backbone with deformable convolutions and introduce DAT deformable attention. Additionally, we compare the results with the CBAM [19] attention mechanism. Six sets of ablation experiments were conducted, and the results are shown in Table 1.

Table 1

Results of ablation experiment

DAT	CBAM	DCNV2	mAP@ 0.5/%	Detectionspeed (frame /s)	Modelsize /MB
-	-	-	**76.7**	**86**	**41**
√	-	-	**79.1**	**79**	**46**
-	-	√	**78.6**	**82**	**44**
-	√	-	**78.3**	**67**	**52**
-	√	√	**80.6**	**63**	**57**
√	-	√	**82.6**	**76**	**48**

[Note] "√" in the table indicates that the module is integrated into the basic model of YOLOv8s, and "-" indicates that the module is not added.

According to Table 1, it can be observed that individually adding DCNv2, DAT, and CBAM to the YOLOv8s model each contributes to a certain degree of performance improvement. It is noteworthy that under the condition of an increase in parameters by only 5M and 3M, DAT and DCNv2 achieved the most substantial mAP improvements, with an increase of 2.4% and 1.9%, respectively. Additionally, compared to the DCNv2+CBAM model, the DCNv2+DAT model reduced the parameter count by 9M, while achieving a 2.0% increase in mAP. These experimental results clearly indicate that the proposed YOLOv8-Def model based on deformable thinking exhibits significant advantages in detecting pedestrians and vehicles in adverse weather conditions.

4.3 Compared with other models

The performance of the YOLOv8-Def model has been extensively analyzed in the previous sections, further comparing its superiority and feasibility with current mainstream detection algorithms. The comparative experimental results with other models are shown in Table 2.

Table 2

Results of ablation experiment

Model	mAP@0.5 /%	Detectionspeed (frame /s)	Model size /MB
F-RCNN	78.9	25	189
RT-DETR	79.1	51	122
YOLOv5s	69.7	101	21
YOLOv7X	74.6	89	69
YOLOv8s	76.7	86	41
Ours	82.6	76	48

According to the experimental results in Table 2, the YOLOv8-Def model demonstrates a superior mAP, surpassing Faster-RCNN and RT-DETR [20], widely acknowledged for high detection accuracy, by 3.7% and 3.5%, respectively. Moreover, the detection speed (FPS) of YOLOv8-Def is significantly better than

both of these algorithms. Although YOLOv5s and YOLOv7 exhibit higher detection speeds than the YOLOv8-Def model, the mAP values of YOLOv8-Def surpass them by 12.9% and 8.0%, respectively, with minimal differences in model size.

These experimental results indicate that compared to current lightweight improvement algorithms and mainstream object detection algorithms, the proposed YOLOv8-Def model demonstrates a significant performance advantage. This algorithm not only maintains a good FPS with the minimal increase in parameters and a smaller model size but also achieves the highest level of mAP.

To demonstrate the robustness of the YOLOv8-Def model, a random selection of test images from the dataset was taken to perform comparative tests with mainstream algorithm models. The results are presented in Fig.10.

Fig. 10 Test comparison of images of different models

According to Fig.10, in the second and fourth images, where objects are nearly completely occluded under adverse weather conditions, both the YOLOv8-Def model and the RT-DETR model can effectively recognize them. However, YOLOv8 and YOLOv7 exhibit accuracy issues in both recognition and detection. Additionally, in the third image, it is evident that RT-DETR struggles to effectively detect the smaller target cars in the rear. In contrast, the YOLOv8-Def model can rapidly and accurately detect these targets, providing another perspective on the effectiveness of the proposed model in this study.

In this study, about 500 pieces of pedestrian and vehicle data under normal

weather conditions were integrated into the original data to improve the stability and generalization ability of the model through data fusion, and to ensure correct identification under normal weather conditions. The detection and recognition of normal weather conditions as shown in Fig.11.

Fig.11 Detection and recognition of normal weather conditions

5. Conclusions

This study addresses the challenge of detecting vehicles and pedestrians under diverse meteorological conditions by proposing an enhanced YOLOv8s algorithm, denoted as YOLOv8-Def. The main contributions of YOLOv8-Def are as follows:

(1) YOLOv8-Def replaces the C2f module in the Backbone part of YOLOv8 with the second-generation deformable convolution. Additionally, YOLOv8-Def incorporates the Deformable Attention Transformer (DAT) before the SPPF feature fusion module, allowing the model to dynamically adjust its receptive field to better adapt to the features of targets under different adverse weather conditions.

(2) When the Intersection over Union (IoU) is set to 0.3, the YOLOv8-Def model has a size of 48MB, a detection accuracy of 83.4%, a recall rate of 74.8%, a detection speed of 76 frames per second (FPS), and a mean Average Precision (mAP@0.5) of 82.6%.

(3) Compared to the original standard YOLOv8s model, the absolute mAP has improved by 5.9%. Compared to Faster-RCNN, YOLOv8-Def model's absolute mAP value is 3.7% higher, with a higher FPS of over 51 frames/s. In comparison with the RT-DETR model, YOLOv8-Def model's absolute mAP value is 3.5% higher, with an FPS higher than 25 frames/s. The experimental results demonstrate that the YOLOv8-Def model significantly improves detection accuracy while minimizing parameter increase.

(4) It only reduces the detection speed by 10 frames per second, but improves the mAP value by 5.9 percentage points compared to the original

YOLOv8s, exhibiting excellent practicality. This study provides a valuable exploration for addressing the challenges of target detection and recognition under diverse meteorological conditions, offering valuable insights and references for future research and development in this field.

Acknowledgement

This research was supported by the fundamental research project (General Project) of Liaoning provincial department of education 2024 (2024JYTKYTD-02), Liaoning Institute of Science and Technology Doctoral Start-up Fund 2023 (2307B06), Pioneer Research Team of Liaoning Institute of Science and Technology "Technology and Application of Big Data and Intelligent information Processing" (XKT202306). Innovation and Entrepreneurship Training Plan for college students of Liaoning University of Science and Technology in 2024 (202411430013).

REFERENCES

[1] *K. He, X. Zhang, S. Ren, and J. Sun*, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[2] *K. He, G. Gkioxari, P. Dollár, and R. Girshick,* "Mask R-CNN," in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 2017, pp. 2961-2969, doi: 10.1109/ICCV.2017.322.

[3] R. Girshick, "Fast R-CNN," in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.

[4] *S. Ren, K. He, R. Girshick, and J. Sun,* "Faster R-CNN: Towards real-time object detection with region proposal networks," in Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 39, no. 6, pp. 1137-1149, June 2017, doi: 10.1109/TPAMI.2016.2577031.

[5] *J. Redmon, S. Divvala, R. Girshick, and A. Farhadi*, "You Only Look Once: Unified, Real-Time Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.

[6] *W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg*, "SSD: Single Shot Multibox Detector," in Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 2016, pp. 21-37, doi: 10.1007/978-3-319-46448-0_2.

[7] *Q. Zhu, J. Mai, and L. Shao, "AOD-Net: All-in-OneDehazing Network*," in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, South Korea, 2019, pp. 3087-3095, doi: 10.1109/ICCV.2019.00318.

[8] *Liu, W., Ren, G., Yu, R., Guo, S., Zhu, J., et al.* (2021, December 15). Image-Adaptive YOLO for Object Detection in Adverse Weather Conditions. arXiv preprint arXiv:2112.08088 [cs.CV]. Retrieved from https://arxiv.org/pdf/2112.08088.pdf

[9] *Zhang, G., Xiong, Y., Dai, J., Zhang, L., Lu, H., &Lin,* C. (2018). Deformable ConvNets v2: More Deformable, Better Results. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 9308-9316. DOI: 10.1109/CVPR.2018.00965

[10] *Z. Xia, X. Pan, S. Song, L. E. Li, & G. Huang*, "Vision Transformer with Deformable Attention," in Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern

Recognition (CVPR), New Orleans, LA, USA, 2022, pp. 4784-4793, doi:10.1109/CVPR52688.2022.00475.

[11] *J. Liu, J. Li, S. Pirk, C. Stretcu, and A. C. Bovik*, "Path Aggregation Network for Instance Segmentation," in Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 2018, pp. 8759-8768, doi: 10.1109/CVPR.2018.00913.

[12] *Laranjeira, C., Andrade, D., & dos Santos*, J. A. (2023). YOLOv7 for Mosquito Breeding Grounds Detection and Tracking. arXiv preprint arXiv:2310.10423 [cs.CV]. Retrieved from https://arxiv.org/abs/2310.10423

[13] *Tang, S., Zhang, S., & Fang,* Y. (2023). HIC-YOLOv5: Improved YOLOv5 For Small Object Detection. arXiv preprint arXiv:2309.16393 [cs.CV]. Retrieved from https://arxiv.org/abs/2309.16393

[14] *Sangam, T., Dave, I. R., Sultani, W.*, & Shah, M. (2023). TransVisDrone: Spatio-Temporal Transformer for Vision-based Drone-to-Drone Detection in Aerial Videos. arXiv preprint arXiv:2210.08423 [cs.CV]. Retrieved from https://arxiv.org/abs/2210.08423

[15] *Han, X., Wei, L., Yu, X., Dou, Z., He, X., Wang, K., Han, Z.*, & Tian, Q. (2023). Boosting Segment Anything Model Towards Open-Vocabulary Learning. arXiv preprint arXiv:2312.03628 [cs.CV]. Retrieved from https://arxiv.org/abs/2312.03628

[16] *Yang, X., Song, E., Ma, G., Zhu, Y., Yu, D., Ding, B*., & Wang, X. (2023). YOLO-OB: An improved anchor-free real-time multiscale colon polyp detector in colonoscopy. arXiv preprint arXiv:2312.08628 [cs.CV]. Retrieved from https://arxiv.org/abs/2312.08628

[17] *J. Liu, J. Li, S. Pirk*, C. Stretcu, and A. C. Bovik, "Path Aggregation Network for Instance Segmentation," in Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 2018, pp. 8759-8768, doi: 10.1109/CVPR.2018.00913.

[18] *Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., & Wei, Y*. (2017). Deformable Convolutional Networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 764-773. DOI: 10.1109/ICCV.2017.84

[19] Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3-19. DOI: 10.1007/978-3-030-01231-1_1

[20] *Carion, N., Massa, F., Erhan, D., Kirillov*, A., & Girshick, R. (2020). End-to-End Object Detection with Transformers. In Proceedings of the European Conference on Computer Vision (ECCV), 2020, pp. 213-229. DOI: 10.1007/978-3-030-58452-8_13