ISSN 2277-2685
IJESR/April-June. 2025/ Vol-15/Issue-2s/1177-1184
B. Naresh *et. al.*, / International Journal of Engineering & Science Research

Road Object Detection In Foggy Complex Scenes Based On Improved Yolov10

T. Rama Krishna[1], B. Naresh[2], K. Dheeraj Kumar[3], K.Sai Charan Reddy[4]
[1]Associate professor, Department of IT, Guru Nanak institutions technical campus (AUTONOMOUS), India.
[2,3,4]B.Tech Students, Department of CSE, Guru Nanak institutions technical campus (AUTONOMOUS), India.

ABSTRACT

Foggy weather presents substantial challenges for vehicle detection systems due to reduced visibility and the obscured appearance of objects. To overcome these challenges, a novel vehicle and Humans detection algorithm based on an improved lightweight YOLOv10 model is introduced. The proposed algorithm leverages advanced preprocessing techniques, including data transformations, Dehaze Formers, and dark channel methods, to improve image quality and visibility. These preprocessing steps effectively reduce the impact of haze and low contrast, enabling the model to focus on meaningful features. An enhanced attention module is incorporated into the architecture to improve feature prioritization by capturing long-range dependencies and contextual information. This ensures that the model emphasizes relevant spatial and channel features, crucial for detecting small or partially visible vehicles in foggy scenes. Furthermore, the feature extraction process has been optimized, integrating an advanced lightweight module that improves the balance between computational efficiency and detection performance. This research addresses critical issues in adverse weather conditions, providing a robust framework for foggy weather vehicle and Humans detection.

1-INTRODUCTION

Foggy weather significantly hinders the performance of vehicle detection systems, as fog reduces visibility, obscures object details, and creates low-contrast conditions. Accurate detection of vehicles and pedestrians in such weather conditions is critical for improving the safety and reliability of autonomous driving systems. Traditional detection methods often fail to deliver satisfactory results under foggy conditions due to the blurring and reduction in contrast that fog causes. In this paper, we propose a novel vehicle and human detection algorithm based on an improved lightweight YOLOv10 model designed to overcome these challenges. By incorporating advanced preprocessing techniques such as Dehaze Formers and dark channel methods, we aim to enhance image quality and restore visibility, enabling the model to focus on critical features despite the fog. Furthermore, an enhanced attention module is introduced to prioritize important spatial and channel features, ensuring better performance in detecting small or partially visible objects, which are common in foggy scenes. Our approach leverages the power of YOLOv10, optimized for both computational efficiency and high detection accuracy, to deliver reliable results even in the most challenging conditions.

EXISTING SYSTEM:

- YOLOv8 is an evolution of the YOLO object detection family, continuing its core design of a single-stage architecture that simultaneously predicts both object classes and bounding boxes. It is an end-to-end deep learning model that uses convolutional layers for feature extraction, followed by a series of dense layers for classification and localization.

- YOLOv8 builds on the improvements from previous versions by optimizing the backbone architecture, enhancing the detection head, and introducing various techniques to reduce computational costs while improving detection accuracy. The model has become widely used in real-time applications due to its efficiency and ability to handle high-speed processing tasks.

PROPOSED SYSTEM

- YOLOv10 is the latest advancement in the YOLO series, with improvements that make it better suited for real-time detection in various environments, including low-visibility or adverse weather conditions. It introduces several innovations in architecture and feature extraction to handle small, occluded, or distorted objects more effectively. YOLOv10 also integrates enhancements that reduce computational cost, making it more efficient without compromising on accuracy.
- The model's architecture allows it to process images faster, enabling its use in time-sensitive applications where rapid decision-making is critical. YOLOv10's increased robustness, especially in complex and challenging environments, makes it a superior choice for vehicle detection, surveillance, and other computer vision tasks**.**

2-LITERATURE SURVEY

Run, don't walk: Chasing higher FLOPS for faster neural networks, **J. Chen, S.-H. Kao, H. He, W. Zhuo, S. Wen, C.-H. Lee, and S.-H.-G. Chan,(2023)**

To design fast neural networks, many works have been focusing on reducing the number of floating-point operations (FLOPs). We observe that such reduction in FLOPs, however, does not necessarily lead to a similar level of reduction in latency. This mainly stems from inefficiently low floating-point operations per second (FLOPS). To achieve faster networks, we revisit popular operators and demonstrate that such low FLOPS is mainly due to frequent memory access of the operators, especially the depthwise convolution. We hence propose a novel partial convolution (PConv) that extracts spatial features more efficiently, by cutting down redundant computation and memory access simultaneously. Building upon our PConv, we further propose FasterNet, a new family of neural networks, which attains substantially higher running speed than others on a wide range of devices, without compromising on accuracy for various vision tasks. For example, on ImageNet1k, our tiny FasterNet-T0 is 2.8×, 3.3×, and 2.4× faster than MobileViT-XXS on GPU, CPU, and ARM processors, respectively, while being 2.9% more accurate. Our large FasterNet-L achieves impressive 83.5% top-1 accuracy, on par with the emerging Swin-B, while having 36% higher inference throughput on GPU, as well as saving 37% compute time on CPU.

PConv: Simple yet effective convolutional layer for generative adversarial network., **: S. Park, Y.-J. Yeo, and Y.-G. Shin, 2022.**

This paper presents a novel convolutional layer, called perturbed convolution (PConv), which performs not only a convolution operation but also a dropout one. The PConv focuses on achieving two goals simultaneously: improving the generative adversarial network (GAN) performance and alleviating the memorization problem in which the discriminator memorizes all images from a given dataset as training progresses. In PConv, perturbed features are generated by randomly disturbing an input tensor before performing the convolution operation. This approach is simple but surprisingly effective. First, to produce a similar output even with the perturbed tensor, each layer in the discriminator should learn robust features having a small local Lipschitz value. Second, since the input tensor is randomly perturbed during the training procedure like the dropout in neural networks, the memorization problem could be alleviated. To show the generalization ability of the proposed method, we conducted extensive experiments with various loss functions and datasets including CIFAR-10, CelebA, CelebA-HQ, LSUN, and tiny-ImageNet. The quantitative evaluations demonstrate that PConv effectively boosts the performance of GAN and conditional GAN in terms of Frechet inception distance (FID).

S2-MLP: Spatial-shift MLP architecture for vision, **T. Yu, X. Li, Y. Cai, M. Sun, and P. Li, 2022.**

Recently, visual Transformer (ViT) and its following works abandon the convolution and exploit the self-attention operation, attaining a comparable or even higher accuracy than CNN. More recently, MLP-mixer abandons both the convolution and the self-attention operation, proposing an architecture containing only MLP layers. To achieve crosspatch communications, it devises an additional token-mixing MLP besides the channel-mixing MLP. It achieves promising results when training on an extremely large-scale datasetsuch as JFT-300M. But it cannot achieve as outstanding performance as its CNN and ViT counterparts when training on medium-scale datasets such as ImageNet-1K. The performance drop of MLP-mixer motivates us to rethink the token-mixing MLP. We discover that token-mixing operation in MLP-mixer is a variant of depthwise convolution with a global reception field and spatial-specific configuration. In this paper, we propose a novel pure MLP architecture,spatial-shift MLP (S2 -MLP). Different from MLP-mixer, our S2-MLP only contains channel-mixing MLP. We devise a spatial-shift operation for achieving the communication between patches. It has a local reception field and is spatialagnostic. Meanwhile, it is parameter-free and efficient for computation. The proposed S2 -MLP attains higher recognition accuracy than MLP-mixer when training on ImageNet1K dataset. Meanwhile, S2-MLP accomplishes as excellent performance as ViT on ImageNet-1K dataset with considerably simpler architecture and fewer FLOPs and parameters.

- DINO: DETR with improved de noising anchor boxes for end-to-end object detection, **H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, 2022**

e present DINO (DETR with Improved deNoising anchOr boxes), a strong end-to-end object detector. DINO improves over previous DETR-like models in performance and efficiency by using a contrastive way for denoising training, a look forward twice scheme for box prediction, and a mixed query selection method for anchor initialization. DINO achieves 49.4AP in 12 epochs and 51.3AP in 24 epochs on COCO with a ResNet-50 backbone and multi-scale features, yielding a significant improvement of +6.0AP and +2.7AP, respectively, compared to DN-DETR, the previous best DETR-like model. DINO scales well in both model size and data size. Without bells and whistles, after pre-training on the Objects365 dataset with a SwinL backbone, DINO obtains the best results on both COCO val2017 (63.2AP) and test-dev (63.3AP) with model size under 1 billion parameters. Compared to other models on the leaderboard, DINO significantly reduces its model size and pre-training data size while achieving better results. The code will be available.

- Low-illumination object detection method based on dark-YOLO., **] J. Zetao, X. Yun, and Z. Shaoqin,2023.**
  Low-light object detection is an important research area in computer vision, but it is also a difficult issue. This research offers a low-light target detection network, NLE-YOLO, based on YOLOV5, to address the issues of insufficient illumination and noise interference experienced by target detection tasks in low-light environments. The network initially preprocesses the input image with an improvement technique before suppressing high-frequency noise and enhancing essential information with C2fLEFEM, a unique feature extraction module. We also created a multi-scale feature extraction module, AMC2fLEFEM, and an attention mechanism receptive field module, AMRFB, which are utilized to extract features of multiple scales and enhance the receptive field. The C2fLEFEM module, in particular, merges the LEF and FEM modules on top of the C2f module. The LEF module employs a low-frequency filter to remove high-frequency noise; the FEM module employs dual inputs to fuse low-frequency enhanced and original features; and the C2f module employs a gradient retention method to minimize information loss. The AMC2fLEFEM module combines the SimAM attention mechanism and uses the pixel relationship to obtain features of different receptive fields, adapt to brightness changes, capture the difference between the target and the background, improve the network's feature extraction capability, and effectively reduce the impact of noise. The AMRFB module employs atrous convolution to enlarge the receptive field, maintain global information, and adjust to targets of various scales. Finally, for low-light settings, we replaced the original YOLOv5 detection head with a decoupled head. The Exdark dataset experiments show that our method outperforms previous methods in terms of detection accuracy and performance.

3-PROJECT DESCRIPTION

This project proposes a robust detection framework that enhances vehicle and human detection in foggy conditions using an improved YOLOv10 model. By leveraging advanced image preprocessing techniques such as Dehaze Formers and dark channel methods, the model improves image visibility, reducing the impact of haze and low contrast. An advanced attention module is integrated into the model to ensure that it captures long-range dependencies and contextual information, prioritizing relevant features and improving detection performance, particularly for small or occluded objects. Additionally, the feature extraction process has been optimized to create a lightweight yet efficient detection system that strikes a balance between performance and computational cost. The proposed method is evaluated on challenging foggy road scenes, demonstrating its effectiveness in accurately detecting vehicles and pedestrians, thereby improving safety and reliability for autonomous systems in adverse weather conditions.

METHODOLOGIES

1. Data Collection

This module involves gathering a comprehensive dataset of road scenes captured under foggy weather conditions. The dataset should include images of vehicles, pedestrians, and other objects typically present on roads. Data collection can be done using cameras in real-world environments or sourced from publicly available datasets like Foggy Cityscapes. Ensuring diversity in fog density, object types, and scene layouts is crucial for developing a robust detection system.

2. Data Labels Analysis

Data labeling involves categorizing the objects in the dataset, such as vehicles, humans, and other road elements. Analysis of these labels ensures balanced representation of all categories and helps identify potential issues like class imbalance or mislabeling. Accurate and well-distributed labels are essential for training a model that generalizes well across different scenarios.

3. Annotations

Annotations define the exact location and category of objects in the dataset by marking bounding boxes around vehicles and pedestrians. This module uses tools like LabelImg or VIA (VGG Image Annotator) to manually or semi-automatically annotate images. High-quality annotations are critical for training object detection models, as they directly influence the accuracy of predictions.

4. Data Preprocessing

Preprocessing prepares the dataset for model training by improving the quality of images and ensuring consistency. Techniques include:

Dehazing: Using Dehaze Formers or dark channel methods to remove fog and improve visibility.

Normalization: Scaling pixel values to a uniform range to optimize model performance.

Augmentation: Enhancing dataset diversity by applying transformations such as rotation, flipping, and scaling to simulate various real-world scenarios.

5. Model Apply

In this module, the improved YOLOv10 model is applied for detecting vehicles and humans in foggy scenes. The model is trained on the preprocessed and annotated dataset, with the inclusion of an enhanced attention module to prioritize relevant features. Training involves optimizing hyperparameters and using techniques like transfer learning to achieve high detection accuracy even in challenging conditions.

4-REQUIREMENTS ENGINEERING

We can see from the results that on each database, the error rates are very low due to the discriminatory power of features and the regression capabilities of classifiers. Comparing the highest accuracies (corresponding to the lowest error rates) to those of previous works, our results are very competitive.

HARDWARE REQUIREMENTS

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It should what the system do and not how it should be implemented.

- PROCESSOR : DUAL CORE 2 DUOS.
- RAM : 4GB DD RAM
- HARD DISK : 250 GB

SOFTWARE REQUIREMENTS

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team's progress throughout the development activity.

- Operating System : Windows 7/8/10

- Platform : Spyder3
- Programming Language : Python
- Front End : Spyder3

FUNCTIONAL REQUIREMENTS

A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behavior, Firstly, the system is the first that achieves the standard notion of semantic security for data confidentiality in attribute-based deduplication systems by resorting to the hybrid cloud architecture.

NON-FUNCTIONAL REQUIREMENTS

The major non-functional Requirements of the system are as follows

Usability

The system is designed with completely automated process hence there is no or less user intervention.

Reliability

The system is more reliable because of the qualities that are inherited from the chosen platform python. The code built by using python is more reliable.

5-DESIGN ENGINEERING

Design Engineering deals with the various UML [Unified Modelling language] diagrams for the implementation of project. Design is a meaningful engineering representation of a thing that is to be built. Software design is a process through which the requirements are translated into representation of the software. Design is the place where quality is rendered in software engineering.

SYSTEM ARCHITECTURE:

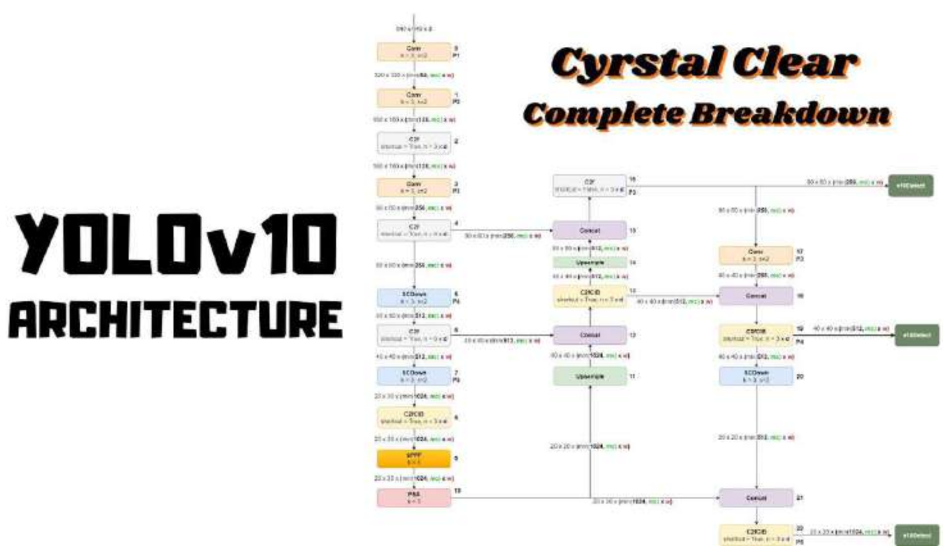

Fig 4.11: System Architecture

6-CONCLUSION

In this study, we proposed an efficient and lightweight YOLOv10 network model tailored for traffic target detection in foggy weather. By integrating advanced modules such as DCN, Involution, and FasterNex, we successfully reduced model parameters and size while enhancing detection accuracy and performance. A novel S5attention module was introduced to improve feature fusion, and an additional small target detection layer significantly boosted the accuracy of detecting small objects and refined boundary box regression. The optimized YOLOv10 model effectively addresses the challenges posed by foggy road conditions, offering improved detection performance and computational efficiency for real-world applications.

REFERENCES

1] C. Li, C. Guo, J. Guo, P. Han, H. Fu, and R. Cong, ''PDR-Net: Perception-inspired single image dehazing network with refinement,'' IEEE Trans. Multimedia, vol. 22, no. 3, pp. 704–716, Mar. 2020, doi: 10.1109/TMM.2019.2933334.

[2] X. Xiaomin and L. Wei, ''Two stages end-to-end generative network for single image defogging,'' J. Comput.-Aided Des. Comput. Graph., vol. 32, no. 1, pp. 164–172, 2020, doi: 10.3724/SP.J.1089.2020.17856.

[3] J. Wu, Z. Kuang, L. Wang, W. Zhang, and G. Wu, ''Context-aware RCNN: A baseline for action detection in videos,'' 2020, arXiv:2007.09861.

[4] L. Jiang, J. Chen, H. Todo, Z. Tang, S. Liu, and Y. Li, ''Application of a fast RCNN based on upper and lower layers in face recognition,'' Comput. Intell. Neurosci., vol. 2021, pp. 1–12, Sep. 2021.

[5] R. Girshick, ''Fast R-CNN,'' in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Santiago, Chile, Dec. 2015, pp. 1440–1448.

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick, ''Mask R-CNN,'' 2017, arXiv:1703.06870.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, ''You only look once: Unified, real-time object detection,'' 2015, arXiv:1506.02640.

[8] J. Redmon and A. Farhadi, ''YOLO9000: Better, faster, stronger,'' 2016, arXiv:1612.08242.

[9] J. Redmon and A. Farhadi, ''YOLOv3: An incremental improvement,'' 2018, arXiv:1804.02767.

[10] A. Bochkovskiy, C.-Y. Wang, and H.-Y. Mark Liao, ''YOLOv4: Optimal speed and accuracy of object detection,'' 2020, arXiv:2004.10934.

[11] M. Wang, W. Yang, L. Wang, D. Chen, F. Wei, H. KeZiErBieKe, and Y. Liao, ''FE-YOLOv5: Feature enhancement network based on YOLOv5 for small object detection,'' J. Vis. Commun. Image Represent., vol. 90, Feb. 2023, Art. no. 103752, doi: 10.1016/j.jvcir.2023.103752.

[12] C.-Y. Wang, A. Bochkovskiy, and H.-Y.-M. Liao, ''YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Vancouver, BC, Canada, Jun. 2023, pp. 7464–7475.

[13] L. Wei, A. Dragomir, E. Dumitru, S. Christian, R. Scott, F. ChengYang, B. Alexander, 'SSD: Single shot MultiBox detector,'' in Computer Vision—ECCV 2016, vol. 9905. Cham, Switzerland: Springer, 2016.

[14] B. Gao, Z. Jiang, and J. Zhang, ''Traffic sign detection based on SSD,'' in Proc. 4th Int. Conf. Autom., Control Robot. Eng., Jul. 2019, p. 16, doi: 10.1145/3351917.3351988.

[15] L. Xuan, L. Jing, and W. Haiyan, ''Study on traffice scene object detection algorithm under complex meteorological conditions,'' Comput. Simul., vol. 38, no. 2, pp. 87–90, 2021.

[16] W. Qi-Ming, Z. He, D. Zhang, and Z. Mao, ''Research on pedestrian and vehicle detection method based on YOLOv3 in foggy scene,'' Control Eng. China, vol. 1, pp. 1–8, Sep. 2023.

[17] J. Zetao, X. Yun, and Z. Shaoqin, ''Low-illumination object detection method based on dark-YOLO,'' J. Comput.-Aided Des. Comput. Graph., vol. 35, no. 3, pp. 441–451, 2023.

[18] J. Yin, S. Qu, Z. Yao, X. Hu, X. Qin, and P. Hua, ''Traffic sign recognition model in haze weather based on YOLOv5,'' J. Comput. Appl., vol. 42, no. 9, pp. 2876–2884, 2022.

[19] K. Li, Y. Wang, and Z. Hu, ''Improved YOLOv7 for small object detection algorithm based on attention and dynamic convolution,'' Appl. Sci., vol. 13, no. 16, p. 9316, Aug. 2023, doi: 10.3390/app13169316.

[20] C. Bhagya and A. Shyna, ''An overview of deep learning based object detection techniques,'' in Proc. 1st Int. Conf. Innov. Inf. Commun. Technol. (ICIICT), Chennai, India, Apr. 2019, pp. 1–6.

[21] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, ''Deformable DETR: Deformable transformers for end-to-end object detection,'' 2020, arXiv:2010.04159.

[22] B. Li, W. Ren, D. Fu, D. Tao, D. Feng, W. Zeng, and Z. Wang, ''Benchmarking single-image dehazing and beyond,'' IEEE Trans. Image Process., vol. 28, no. 1, pp. 492–505, Jan. 2019.

[23] K. He, X. Zhang, S. Ren, and J. Sun, ''Spatial pyramid pooling in deep convolutional networks for visual recognition,'' IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[24] S. Ioffe and C. Szegedy, ''Batch normalization: Accelerating deep network training by reducing internal covariate shift,'' 2015, arXiv:1502.03167.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, ''Attention is all you need,'' in Proc. NIPS, 2017, pp. 1–11.

[26] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, ''Deformable convolutional networks,'' 2017, arXiv:1703.06211.

[27] X. Zhu, H. Hu, S. Lin, and J. Dai, ''Deformable ConvNets v2: More deformable, better results,'' 2018, arXiv:1811.11168.

[28] Y. Guo, Y. Li, R. Feris, L. Wang, and T. Rosing, ''Depthwise convolution is all you need for learning multiple visual domains,'' 2019, arXiv:1902.00927.

[29] J. Chen, S.-H. Kao, H. He, W. Zhuo, S. Wen, C.-H. Lee, and S.-H.-G. Chan, ''Run, don't walk: Chasing higher FLOPS for faster neural networks,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2023, pp. 12021–12031.

[30] S. Park, Y.-J. Yeo, and Y.-G. Shin, ''PConv: Simple yet effective convolutional layer for generative adversarial network,'' Neural Comput. Appl., vol. 34, no. 9, pp. 7113–7124, May 2022, doi: 10.1007/s00521-021- 06846-2.

[31] T. Yu, X. Li, Y. Cai, M. Sun, and P. Li, ''S2-MLP: Spatial-shift MLP architecture for vision,'' in Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV) Jan. 2022, pp. 3615–3624.

[32] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, ''DINO: DETR with improved de noising anchor boxes for end-to-end object detection,'' 2022, arXiv:2203.03605.