A Two-Step Real-Time Complex Environmental Vehicle Detection Model

Zhihui Huo[1], Yiqian Liang*[2], Xingju Wang[3]
Hebei Jingshi Expressway, Development Co., Hebei, China 050800[1],
Shijiazhuang Tiedao University, Hebei, China 050043[2,3]

***Abstract*—In recent years, as a critical pillar supporting the national economy and daily life, the safe and efficient operation of road traffic has highly relied on precise environmental perception capabilities. To address this, this study proposes a two-stage "denoising-detection" framework: the first stage restores clear images using an improved Uformer algorithm, which fundamentally merges a probabilistic sparse self-attention mechanism. In contrast, the second stage leverages YOLOv11 for real-time object detection. This framework is introduced in the field for the first time and enhances the accuracy and robustness of vehicle detection in traffic images under complex weather scenarios, providing technical support for intelligent driving systems and traffic monitoring applications. Our experimental validation on our own flexible weather car detection database demonstrated the superior performance of the proposed model: CM-YOLO achieved 0.95 precision and 0.91 mAP50, which promoted 0.2 than the YOLOv11.**

***Keywords*—Object detection; vehicle detection; image denoising**

I. Introduction

In recent years, highway transportation has become an important support for the national economy and daily life, and its safe and efficient operation highly relies on accurate environmental perception capabilities. However, severe weather can significantly reduce the quality of traffic images, seriously affecting the performance of object detection algorithms and increasing the risk of traffic accidents. According to statistics, the incidence of traffic accidents under complex weather conditions is 3-5 times higher than that under normal weather conditions[1]. Traditional object detection algorithms face two core problems in foggy scenes. On the one hand, complex weather scenes lead to blurry image features and reduced contrast between the target and background. On the other hand, the integration between existing denoising algorithms and detection models is insufficient, making it difficult to balance denoising effectiveness and detection efficiency. However, in recent years, transformer-based computer vision algorithms have provided new directions for image denoising and object detection by leveraging the advantages of the self-attention mechanism in global information capture. Therefore, this study proposes a two-stage framework of "denoising detection", in which the first stage restores clear images through an improved Uformer algorithm, and the second stage uses YOLOv11 to achieve real-time object detection, aiming to improve the accuracy and robustness of traffic vehicle detection in complex weather scenes and provide technical support for traffic monitoring systems.

*Corresponding author.

Image denoising algorithms can be divided into traditional methods and deep learning methods. Traditional methods, represented by dark channel prior (DCP) [2], estimate transmittance and atmospheric light through physical models, but edge distortion is prone to occur in complex scenes. In deep learning based methods, convolutional neural network (CNN) models such as AOD Net [3] and DehazeNet [4] achieve denoising through end-to-end learning, but are limited by local receptive fields and difficult to capture global complex noise distribution patterns. In recent years, transformer-based denoising algorithms have become a research hotspot. The Uformer model[5] integrates Transformer and U-Net structures, captures long-range dependencies through self self-attention mechanism, and performs well in denoising and other tasks. However, the traditional Uformer's window self-attention mechanism has the problem of high computational complexity, which limits its application in real-time scenes.

Object detection algorithms can be divided into two stages such as Faster R-CNN and single-stage, such as YOLO method [6]. YOLOv11, as the latest single-stage algorithm, achieves high-precision detection while maintaining real-time performance through anchor-free design and a feature fusion mechanism. However, YOLOv8 lacks the ability to recognize blurry targets in low-quality images and needs to be combined with dehazing preprocessing to improve performance. Moreover, existing joint frameworks often adopt a "denoising+detection" serial structure, and there are still shortcomings in the collaborative optimization of dehazing and detection modules. Based on this, this study improves the attention mechanism of Uformer and deeply combines it with YOLOv11 to achieve efficient object detection in complex sky scenes.

In addition, in the field of image restoration, many related studies have provided ideas for enhancing complex weather scene images, such as the application of enhanced deep residual networks in super-resolution tasks [7], the hierarchical visual Transformer design of Swin Transformer [8], and the exploration of spatial attention mechanisms [9] and model-driven deep neural networks [10] in complex weather denoising tasks. At the same time, the solution to specific image restoration tasks, such as removing patterns [11], also provides a reference for processing complex foggy scenes. In terms of model optimization, techniques such as decoupling weight decay regularization [12] and dual residual networks [13] provide effective means to improve the performance and stability of denoising and detection models.

To address vehicle detection in complex traffic scene images, this paper proposes a system framework named CM-YOLO, which autonomously denoises and analyzes complex

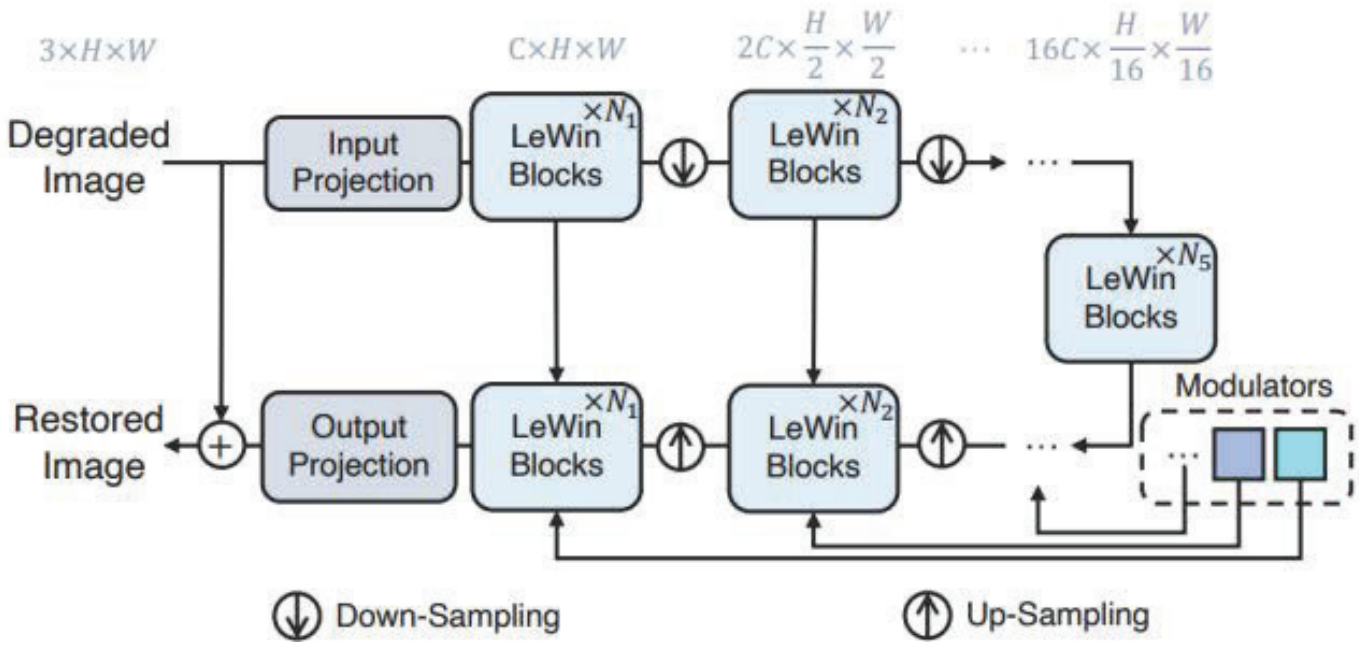

Fig. 1. Schematic of the Uformer.

scene information within images. Owing to the dual requirements of accuracy and real-time performance imposed by dispatch centers for traffic flow regulation, the target detection algorithm must maintain high detection accuracy while completing tasks with high efficiency. In this framework, all module is trained separately. The image will first be denoised by the improved Uformer algorithm to simplify image scenes, and then use the YOLOv11 algorithm as the recognition module to perform target detection on the simplified images. This cascaded architecture effectively achieves target detection in complex traffic scene images.

A. Denoising Module

To address the challenges of vehicle detection in complex traffic image scenarios with variable weather conditions on highways, this paper employs the Uformer model as the denoising module of CM-YOLO. This algorithm, as shown in Fig. 1, rooted in a Transformer-based architecture, integrates the classic U-Net structure, enabling high-quality execution of diverse image denoising tasks such as noise reduction, deblurring, and defogging. Notably, Uformer excels at capturing long-range dependencies within images, making it particularly well-suited for processing high-resolution images. By balancing restoration quality with computational efficiency, it effectively reduces computational complexity while maintaining high-quality image recovery, thus addressing the dual requirements of accuracy and real-time performance in traffic flow regulation scenarios.

Meanwhile, given the characteristics of terminal devices with limited memory and slow computing speeds, it is imperative to reduce the complexity of Uformer. To address this, this paper replaces the window-based multi-head self-attention mechanism used in traditional Uformer with a probabilistic sparse self-attention mechanism.

In the probabilistic sparse self-attention mechanism, each self-attention operation defines attention through a query sparsity measurement based on

$$q_i(K,V) = \sum_{j=1} \frac{k(q_i,k_j)}{\sum_{l=1} k(q_i,k_l)} v_j = E_p(k_j|q_i)[V_j] \quad (1)$$

In the probabilistic sparse self-attention mechanism[14], the attention of the i-th feature over all values is defined as a probability $p(k_j|q_i)$, and the output is generated by k combining this probability with the corresponding value v. If $p(k_j|q_i)$ approaches a uniform distribution

$$q(k_j,q_i) = \frac{1}{L_K} \quad (2)$$

Self-attention then degenerates into the summation of value V, becoming repetitive with other fixed inputs. Therefore, the similarity between two distributions p and q can be used to distinguish the importance of queries. The probabilistic sparse self-attention mechanism measures the similarity between these two distributions using the Kullback-Leibler divergence.

$$KL(q||p) = ln \sum_{l=1}^{L_K} e^{\frac{q_i k_l^t}{\sqrt{d}}} - \frac{1}{L_K} \sum_{j=1}^{L_K} \frac{q_i k_l^t}{\sqrt{d}} \quad (3)$$

Here, the first term represents the average of the log-sum-exp over all features, and the second term denotes their arithmetic mean. If the i-th query obtains a larger $M(q_i,K)$ its attention probability becomes more diverse, increasing the likelihood of including key head-field pairs in the head-tail self-attention distribution that determines the output.

Based on the proposed measurement method, the probabilistic sparse self-attention mechanism ensures each feature

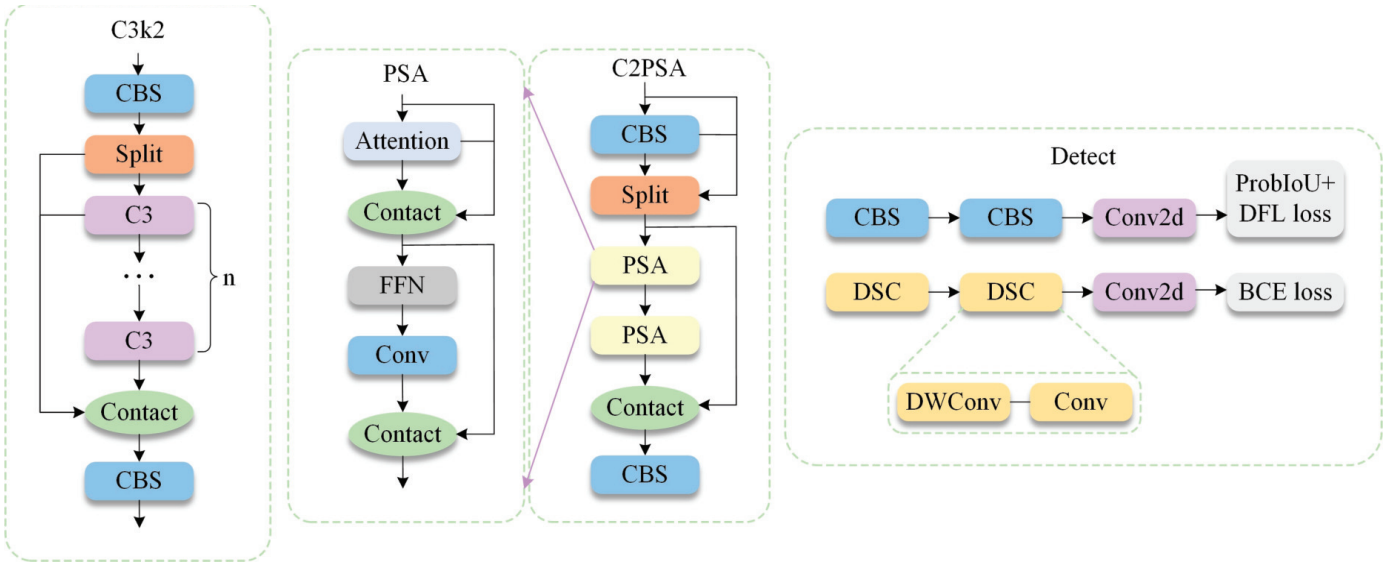

Fig. 2. The detection module structure.

only attends to u key queries:

$$A(Q,K,V) = Softmax(\frac{\overline{Q}K^T}{\sqrt{d}})V \tag{4}$$

Here, Q denotes a sparse matrix of the same size as q, containing only the $M(q,K)$ most critical queries under the sparsity measure u.

By utilizing this strategy, the space and time complexity ultimately approximates to $O(LlnL)$. and the Table I shows the superiority of this self-attention mechanism.

TABLE I. COMPARISON OF ROAD DAMAGE DETECTION RESULTS FOR VARIOUS MODEL SETS

method	time complexity	space complexity	runtime	memory
window-based self-attention	$O(L_K L_Q)$	$O(L_K L_Q)$	3.01ms	15.60GB
probabilistic sparse self-attention	$O(L_K lnL_Q)$	$O(L_K lnL_Q)$	2.57ms	11.35GB

B. Detection Module

To ensure efficient processing of simplified images, the model proposed in this paper employs the YOLOv11 algorithm as shown in Fig. 2 for vehicle recognition. Compared to traditional object detection algorithms such as R-CNN and its variants (e.g., Fast R-CNN, Faster R-CNN), the most significant advantage of YOLOv11 lies in its processing speed. The R-CNN series first generates region proposals and then classifies each region individually. Although this two-stage process achieves superior accuracy, its slower speed renders it unsuitable for real-time applications. In contrast, the holistic architectural design of YOLOv11 enables it to execute detection tasks at extremely high speeds, thereby meeting the rapid detection requirements of dispatch centers in this vehicle recognition task.

II. RESULTS

A. Datasets

In this study, the I-HAZE[15] and NH-Haze[16][17] were used to train the Denoising module and test on SPAD[18] for denoising. The train image samples were shown as Fig. 3. The BITVehicle dataset cite dongVehicleTypeClassification2015 was used to train the detection module. For efficient validation of our model, we built our own flexible weather car detection database, which has 6789 pictures with 17355 annotations. The Denoising module and the Detection module were trained separately, but detected sequentially; the undetected image will first utilize the improved Uformer module to denoise and use YOLOv11 to detect.

B. Evaluation Metrics

To rigorously validate the performance of CM-YOLO in multi-scale pavement crack detection, we adopted the following metrics aligned with real-time detection challenges:

The peak signal-to-noise ratio (PSNR) is an objective metric that quantifies the quality of a reconstructed image by measuring the ratio of the maximum possible pixel intensity to the noise power. It is widely used in image processing and compression tasks.

$$MSE = \frac{1}{MN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[I(i,j) - K(i,j)]^2 \tag{5}$$

where I represents the original image, K represents the processed image, M and N are the dimensions of the image.

$$PSNR = 10log_{10}\frac{MAX_I^2}{MSE} \tag{6}$$

Fig. 3. Samples of I-HAZE and NH-Haze database.

Fig. 4. Samples of BITVehicle database.

where MAX is the maximum pixel value.

SSIM evaluates the similarity between two images by analyzing structural, luminance, and contrast information, mimicking human visual perception.

Precision (P) measures the proportion of correct crack predictions among all detected regions and is critical for reducing false alarms in crack detection.

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

where TP (true positive) denotes predicted boxes with $IoU \geq 0.5$ and correct class labels, and FP (false positive) indicates spurious detection.

Recall(R) evaluates the model's ability to capture all genuine cracks, which is essential for avoiding missed defects in safety-critical scenarios.

$$Recall = \frac{TP}{TP + FP} \tag{8}$$

where FN (false negatives) represent undetected ground-truth cracks ($IoU \leq 0.5$)

The average precision (AP) quantifies the model's category-specific detection consistency by computing the mean of the maximum precision values achieved across varying recall rates. The mean average precision (mAP) extends this evaluation by averaging AP scores across all object classes, thereby serving as a comprehensive metric for the model's cross-category detection performance. The mathematical formulation is

$$AP = \int_0^1 P(r)\, dr \tag{9}$$

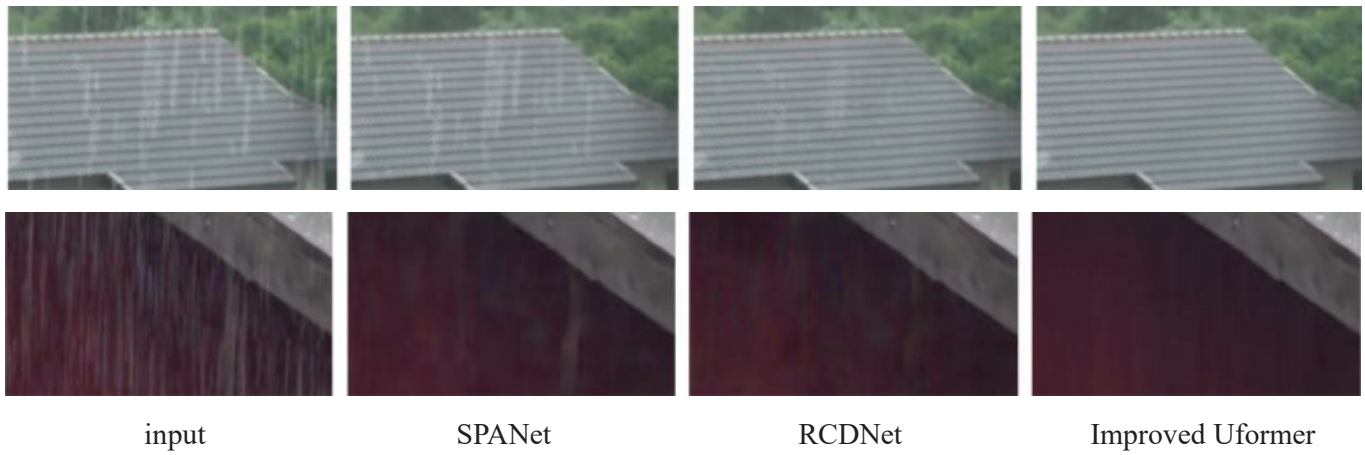

Fig. 5. Comparison of denoising results in SPAD database.

$$mAP = \frac{\sum_{i=1}^{C} AP\,(i)}{C} \tag{10}$$

Here, C is the number of crack subtypes.

C. Experimental Environment

The experimental framework in this study was implemented on a high-performance computing system running Ubuntu 11.04 as the operating system. The hardware setup included an Intel® Xeon® Platinum 8474C processor and an NVIDIA RTX 4090D graphics card with 24 GB of video memory, ensuring efficient computation for deep-learning tasks. The software environment was built using PyTorch 2.0.1, Python 3.10.8, and CUDA 11.8, which provided a robust foundation for model training and evaluation. The key hyperparameters were carefully tuned to optimize performance: The input images were resized to a resolution of 640 × 640 pixels, the initial learning rate was set to 0.0001, and the momentum for the learning rate was configured at 0.9. This combination of hardware, software, and Adaptive Moments with Weight Decay (AdamW) optimizer was used for training. These parameter settings ensured a stable and efficient training process for the proposed model.

D. Experimental Implementation and Results

1) Comparison of denosing results: This paper employs the improved Uformer algorithm for training on the I-HAZE data set and testing on the SPAD dataset. By analysing the PSNR and SSIM metrics of the improved Uformer algorithm, SPANet and RCDNet models, as Table II, which were tested by SPAD, and Fig. 5 shows that the improved Uformer algorithm can effectively restore images while retaining image features, with less noise, good denoising performance and small memory usage.

2) Ablation experiment results: To comprehensively assess the superiority of the denoising module, we primarily use three evaluation metrics: mAP50, recall, and precision. These metrics are compared against the baseline to validate the performance of the denoising module.

TABLE II. COMPARISON OF DENOSING RESULTS IN SPAD DATABASE

method	PSNR	SSIM	runtime(per image)	memory
SPANet	40.24	0.9811	-	-
RCDNet	41.47	0.9834	-	-
Uformer	47.92	0.9928	3.01ms	15.60GB
Ours	47.84	0.9925	2.57ms	11.35GB

TABLE III. COMPARISON OF DETECTION RESULTS FOR VARIOUS MODEL SETS IN OUR DATABASE

method	Denoising Model	P	R	mPA50
YOLOv11		0.93	0.86	0.89
	✓	0.95	0.88	0.91

As Table III , the results demonstrate that the introduced denoising model has positively affected model performance.

3) Comparison of results: To comprehensively assess the performance of the CM-YOLO model, we primarily use three evaluation metrics: mAP50, recall, and precision. These metrics are compared against YOLOv11 and YOLOv8 models to validate the effectiveness of our multi-modal fusion approach.

TABLE IV. COMPARISON OF DETECTION RESULTS IN OUR DATABASE

method	P	R	mAP50
YOLOv8n	0.90	0.84	0.88
RT-DETR(resnet18)	0.92	0.88	0.86
YOLOv11n	0.93	0.86	0.89
Ours	0.95	0.88	0.91

As the Table IV and Fig. 6 show that the proposed CM-YOLO algorithm demonstrates superior detection performance, achieving $P = 0.95$ and $mAP50 = 0.91$, outperforming YOLOv8n, RT-DETR(resnet18), and YOLOv11n. This advancement enables precise detection of vehicle models in complex weather scenarios, including rain, snow, and fog, where traditional models struggle due to noise interference and reduced visibility.

Fig. 6. Comparison of detection figures in our database.

III. Conclusion

CM-YOLO demonstrates strong adaptability to the complex scenario requirements of real-world traffic conditions, enabling effective image object detection tasks in challenging weather-related traffic environments. By fully leveraging the respective strengths of the Transformer model and the YOLOv11n model, it enhances the performance and efficacy of vehicle model detection in complex scenarios. However, practical applications of this algorithm still face certain limitations. Specifically, its object detection workflow requires parallel execution of two algorithms, imposing higher computational performance demands compared to standalone algorithms. In real-world deployments, its performance is constrained by the processing capabilities of vehicle systems.

For future research on such vehicle detection algorithms, the following directions warrant further exploration:

(1) Model optimization to accelerate detection speed: Current Uformer algorithms-even the version used in this study with probabilistic sparse self-attention mechanisms-exhibit larger parameter counts compared to previous de-fogging models, despite superior de-fogging performance. Continued optimization of model parameters is necessary.

(2) Algorithm/training strategy refinement for stability and accuracy: While the algorithm achieves competitive accuracy on normal scene images, there remains room for improvement. Iterative upgrades to the algorithm are needed. Additionally, scenario-specific data and training strategies should be prioritized. For example, pre-training the model on a dedicated dataset, adjusting pre-trained weights, and fine-tuning on downstream tasks could further enhance performance.

(3) Joint training of detection and de-noising algorithms: Collecting paired normal and complex weather traffic images under the same scenarios, followed by annotation, would enable pre-trained detection and de-noising models to be fine-tuned on this dataset.

In summary, the Transformer-based foggy weather traffic video object detection algorithm still has significant room for development. Ongoing research is essential to continuously improve detection speed, accuracy, and robustness in real-world applications.

Declarations

- Competing interests
  The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
- Ethics approval and consent to participate
  We promise that this manuscript will not be submitted to multiple journals or conferences simultaneously. We promise to abide by ethical standards, respect the autonomy of participants in the use of data, and ensure the legal, transparent, and secure use of data.
- Data availability
  Data will be made available on request.
- Author contribution
  Xingju Wang and Yiqian Liang designed the new method, Yiqian Liang and Zhihui Huo provided suggestions and analyzed the results. Yiqian Liang and Zhihui Huo wrote the manuscript.Yiqian Liang and Xingju Wang reviewed and edited this manuscript. All authors contributed to this work and approved the submitted version.

References

[1] M. Won, T. Park, and S. H. Son, "Toward Mitigating Phantom Jam Using Vehicle-to-Vehicle Communication," *IEEE TRANSACTIONS ON*

INTELLIGENT TRANSPORTATION SYSTEMS, vol. 18, no. 5, pp. 1313–1324, May 2017.

[2] A. Willig, "A Sensitivity Analysis of the DCP/Vardis Protocol for Coordinating Drone Swarms," in *2025 IEEE Vehicular Networking Conference (VNC)*.

[3] B. Li, X. Peng, Z. Wang, J. Xu, and D. Feng, "An All-in-One Network for Dehazing and Beyond," Jul. 2017.

[4] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "DehazeNet: An End-to-End System for Single Image Haze Removal," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 25, no. 11, pp. 5187–5198, Nov. 2016.

[5] Z. Wang, X. Cun, J. Bao, W. Zhou, J. Liu, and H. Li, "Uformer: A General U-Shaped Transformer for Image Restoration," in *2022 IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR 2022)*. Los Alamitos: IEEE Computer Soc, 2022, pp. 17 662–17 672.

[6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," May 2016.

[7] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced Deep Residual Networks for Single Image Super-Resolution," *IEEE*, 2017.

[8] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in *2021 IEEE/CVF INTERNATIONAL CONFERENCE ON COMPUTER VISION (ICCV 2021)*. New York: IEEE, 2021, pp. 9992–10 002.

[9] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. H. Lau, "Spatial Attentive Single-Image Deraining with a High Quality Real Rain Dataset," in *2019 IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR 2019)*. New York: IEEE, 2019, pp. 12 262–12 271.

[10] H. Wang, Q. Xie, Q. Zhao, and D. Meng, "A Model-driven Deep Neural Network for Single Image Rain Removal," *IEEE*, 2020.

[11] Y. Sun, Y. Yu, and W. Wang, "Moire Photo Restoration Using Multiresolution Convolutional Neural Networks," *IEEE TRANSACTIONS ON IMAGE PROCESSING*, vol. 27, no. 8, pp. 4160–4172, Aug. 2018.

[12] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," 2017.

[13] X. Liu, M. Suganuma, Z. Sun, and T. Okatani, "Dual Residual Networks Leveraging the Potential of Paired Operations for Image Restoration," in *2019 IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR 2019)*. New York: IEEE, 2019, pp. 7000–7009.

[14] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting," Mar. 2021.

[15] C. O. Ancuti, C. Ancuti, R. Timofte, and C. De Vleeschouwer, "I-HAZE: A dehazing benchmark with real hazy and haze-free indoor images," 2018.

[16] C. O. Ancuti, C. Ancuti, and R. Timofte, "NH-HAZE: an image dehazing benchmark with non-homogeneous hazy and haze-free images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, ser. IEEE CVPR 2020, 2020.

[17] C. O. Ancuti, C. Ancuti, F.-A. Vasluianu, R. Timofte *et al.*, "NTIRE 2020 challenge on nonhomogeneous dehazing," 2020.

[18] K. He, J. Sun, and X. Tang, "Single Image Haze Removal Using Dark Channel Prior," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.