Machine Vision and Applications (2025) 36:94
https://doi.org/10.1007/s00138-025-01709-5

RESEARCH

Towards safer roads: benchmarking object detection models in complex weather scenarios

Ba-Thinh Tran-Le[1] · Vatsa Patel[1,2] · Viet-Tham Huynh[3,4] · Mai-Khiem Tran[3,4] · Kunal Agrawal[1] · Minh-Triet Tran[3,4] · Tam V. Nguyen[1]

Received: 8 February 2025 / Revised: 13 May 2025 / Accepted: 15 May 2025

Abstract
The performance of object detection models in adverse weather conditions remains a critical challenge for intelligent transportation systems. Since advancements in autonomous driving rely heavily on extensive datasets, which help autonomous driving systems be reliable in complex driving environments, this study provides a comprehensive dataset under diverse weather scenarios like rain, haze, nighttime, or sun flares and systematically evaluates the robustness of state-of-the-art deep learning-based object detection frameworks. Our Adverse Driving Conditions Dataset features eight single weather effects and four challenging mixed weather effects, with a curated collection of 50,000 traffic images for each weather effect. State-of-the-art object detection models are evaluated using standard metrics, including precision, recall, and IoU. Our findings reveal significant performance degradation under adverse conditions compared to clear weather, highlighting common issues such as misclassification and false positives. For example, scenarios like haze combined with rain cause frequent detection failures, highlighting the limitations of current algorithms. Through comprehensive performance analysis, we provide critical insights into model vulnerabilities and propose directions for developing weather-resilient object detection systems. This work contributes to advancing robust computer vision technologies for safer and more reliable transportation in unpredictable real-world environments.

Keywords Weather augmentation · Synthetic dataset · Generative AI · Computer vision · Autonomous driving

1 Introduction

The rapid evolution of computer vision algorithms, particularly in object detection, has transformed modern traffic surveillance and autonomous vehicle systems. However, despite significant progress in identifying and tracking objects in complex traffic scenarios, real-world deployment remains challenging due to the unpredictability of environmental conditions. For instance, vehicles exhibit distinct characteristics during the day compared to nighttime, and

Ba-Thinh Tran-Le and Vatsa Patel contributed equally to this work.

✉ Tam V. Nguyen
tamnguyen@udayton.edu

Ba-Thinh Tran-Le
tranl5@udayton.edu

Vatsa Patel
patelv20@udayton.edu

Viet-Tham Huynh
hvtham@selab.hcmus.edu.vn

Mai-Khiem Tran
tmkhiem@selab.hcmus.edu.vn

Kunal Agrawal
agrawalk2@udayton.edu

Minh-Triet Tran
tmtriet@fit.hcmus.edu.vn

[1] Department of Computer Science, University of Dayton, 300 College Park, Dayton, OH 45469, USA

[2] Space Power Systems Research Group, University of Dayton Research Institute, 1700 S. Patterson Blvd , Dayton 45469, USA

[3] University of Science, VNU-HCM, 227 Nguyen Van Cu Street, Ho Chi Minh City 70000, Vietnam

[4] Vietnam National University, Ho Chi Minh City 70000, Vietnam

Published online: 16 June 2025

adverse weather conditions, such as rain, haze, or sun flare, introduce additional complexity that often undermines the accuracy of object detection models. These conditions can obscure objects, reduce visibility, and introduce distortions like glare and reflections, leading to detection failure. For example, in Fig. 1, the DETER model [1] misclassified objects under flare conditions, such as mistaking a car for a bus, or fail to detect objects altogether under nighttime conditions, generating false negatives. Such failures can pose serious safety risks in critical areas like autonomous driving.

There is a growing need for specialized datasets to address these challenges and improve the performance of object detection systems. Although various public datasets exist, creating a new one is often essential to meet specific domain requirements. However, the effort typically requires sophisticated experimental setups, which environmental factors, accessibility, and logistical limitations can constrain. It also necessitates expensive hardware (vehicles, LiDARs, radar, IMUs, etc.) along with extensive labeling efforts, making the process both costly and time-intensive. As a result, researchers suggest that developing synthetic datasets is among the most effective strategies to overcome the limitations of real-world data [2, 3]. Johnson-Roberson et al. (2017) [4] also demonstrated that for the task of vehicle detection, training a CNN model solely on synthetic images can outperform the same model trained on real-world datasets like Cityscapes [5].

Existing datasets offer robust annotations and content; however, they often do not capture the diverse spectrum of edge-driving scenarios. This gap is especially evident when assessing the resilience of object detection models when exposed to challenging weather conditions. Addressing this critical need, our research aims to contribute to advancing computer vision frameworks tailored to real-world applications. Our contributions are threefold:

- *A synthetic dataset with twelve weather conditions*. Our Adverse Driving Conditions Dataset (ADCD) is an extension of the Urban Weather Diversity Dataset (UWDD) [6]. The ADCD comprises a curated collection of 50,000 traffic images sourced from well-established sources, including Udacity Self-Driving Car Dataset, ApolloScape [7], Indian Driving Dataset (IDD) [8], Audi Autonomous Driving Dataset (A2D2) [9], and the newly developed Dayton Driving Dataset (DDD). We aim to provide a dataset that captures a broad spectrum of weather conditions, including haze, rain, snow, night, sunset, and additional extreme scenarios like a cracked windshield, sun flare, and raindrops on the windshield. We also introduce four complex combinations that emulate practical cases, as illustrated in Fig. 2. This design results in a total of 600,000 augmented samples, making the ADCD a comprehensive benchmark for evaluating object detection models and advancing adaptive perception systems.
- *Weather synthesis methodology*. Our approach focuses on preserving the exact positions of all objects within the images while altering only the weather effects. This is because one of the key objectives of a synthetic dataset is to eliminate annotation costs by providing automatically generated, accurate ground truth data for tasks such as object detection and tracking. Cloning images from popular datasets, we leverage a combination of deep-learning models and traditional methods to introduce diverse environmental conditions while preserving the integrity of the original scene.
- *Benchmark evaluation*: Our evaluation focuses on widely adopted object detection models, including YOLO (v5 onwards) [10–16], DETR [1], R-CNN [17], Faster R-CNN [18], RetinaNet [19], and SSD [20]. As discussed, the performance of these models may be challenged by adverse conditions compared to clear weather scenarios. By identifying such challenges, we provide critical insights into the limitations of current approaches and suggest pathways for developing more resilient algorithms. Our findings emphasize the need for adaptive and weather-aware models to ensure the safety and reliability of intelligent transportation systems.

Fig. 1 Object detection model, i.e., DETR [1] fails to detect objects accurately under weather challenge (left: original images with ground truth, middle: original images with DETR detections, right: weather synthetic images with DETR detections)

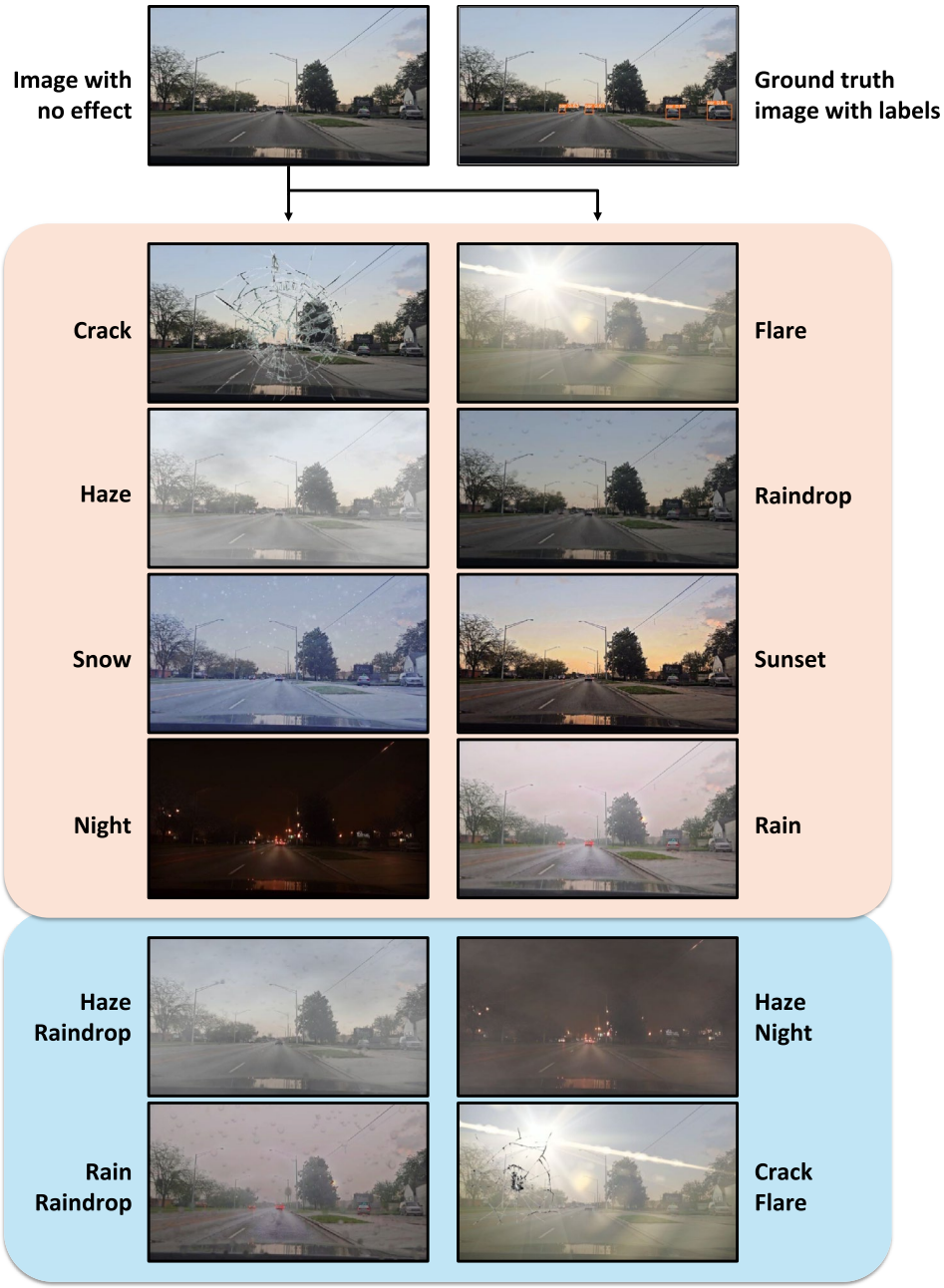


Fig. 2 Illustration of the Adverse Driving Conditions Dataset (ADCD) showcasing base image, ground truth labeled image, 8 single methods (brown), and 4 mixed weather synthesis (blue)

2 Related work

2.1 Real-world datasets

Many datasets have become widely recognized as benchmarks for training and testing, offering a foundation for evaluating model performance. Janai et al. [21] has provided an overview of datasets spanning both computer vision and autonomous driving research. Within the field of computer vision, specific datasets have been developed to address distinct problems, playing a crucial role in advancing the state of the art. For instance, ImageNet [22], Pascal VOC [23], and Microsoft COCO [24] have become benchmarks for object recognition. MOTChallenge benchmark [25, 26] focuses on object tracking, while the Middlebury stereo benchmark [27–29] and the DTU MVS dataset [30] have contributed significantly to stereo and 3D reconstruction tasks. Meanwhile, in the context of autonomous driving, the release of groundbreaking datasets such as KITTI [31] and Cityscapes [5] has set the standard for various tasks. Following these foundational datasets, others [7, 32–35] have introduced rigorous benchmarks focused on the (temporal coherence of) approaches related to semantic segmentation, motion estimation, recognition, tracking, and more.

These datasets have played a crucial role in bridging the gap between controlled laboratory settings and the complex challenges of real-world environments.

2.2 Synthetic datasets

Traditional computer vision methods, primarily through physics-based rendering techniques, have contributed to visibility restoration problems, including dehazing, deraining, and desnowing. For example, the Koschmieder model has been applied to generate pioneering foggy datasets like FRIDA and FRIDA2 [36, 37], or Foggy Cityscapes [38], developed by Sakaridis et al. in 2018, utilized an optical model of fog implemented on the MATLAB platform. Meanwhile, streak-based models are employed for rain simulation, considering factors such as velocity and wind direction, as demonstrated in [39, 40]. For snow simulation, researchers rely on image overlay methods to mimic snowflakes, with DesnowNet [41] as an example.

Another non-deep learning approach involves using the 3D world of game engines, which have been used to create notable datasets such as Virtual KITTI [42], SYNTHIA [43], and those based on GTA V [44], followed by VIPER [45]. These datasets offer diverse environmental conditions: SYNTHIA simulates variations in daylight, Virtual KITTI covers four weather conditions (clear, cloudy, foggy, and heavy rain), and VIPER introduces additional conditions, including sunset, rain, snow, and night. Although this method offers customization and precise control over variables like weather, lighting, or camera angles, it often comes with a trade-off in the realism of the generated images compared to real-world data.

The advent of neural networks, particularly generative models, has revolutionized synthetic dataset creation. The introduction of GANs by Goodfellow et al. [46] marked a breakthrough in generating high-quality images, inspiring subsequent innovations [47–50]. In the context of autonomous driving, GAN-based methods like CycleGAN [51] and Pix2Pix [52] have been applied to transform more complex weather conditions in driving scenes, such as day-to-night or summer-to-winter. However, GANs face challenges in adapting to new domains. Diffusion models, which iteratively reconstruct data [53], surpassed GANs in quality for some benchmarks, as shown by Dhariwal and Nichol [54]. Notable diffusion-based methods, such as InstructPix2Pix [55] and CycleGAN-Turbo [56], enhance the ability to create diverse driving scenarios. Recent advances in LLMs [57–59] further complement these efforts, enabling integration of image synthesis with context to produce data that balances realism and domain relevance for autonomous driving systems.

Table 1 Summary of selected subsets with location, release year, and the number of images compiled in our Adverse Driving Conditions Dataset

Dataset	Location	Year	Images
Udacity	Mountain View, United States	2016	24,007
ApolloScape	China	2018	7040
IDD	India	2019	5713
A2D2	Germany	2020	12,469
DDD	Dayton, United States	2024	755

3 Proposed work

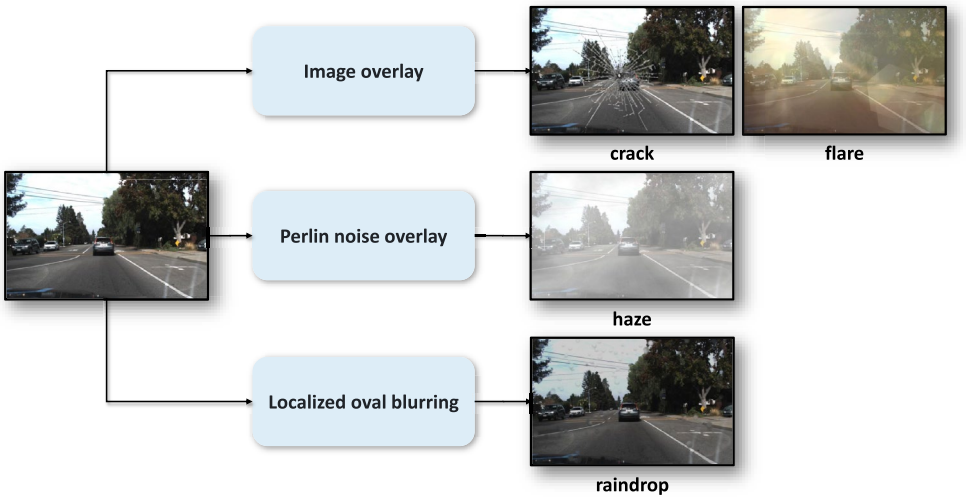
3.1 ADCD dataset collection

Our proposed dataset, the Adverse Driving Conditions Dataset (ADCD), is a combination of five subsets that span a wide range of geographical locations and driving scenarios, making it a valuable resource for creating a diverse dataset. For example, scenes in India capture more people riding motorcycles in dense traffic, whereas other regions feature sparse roads with cars, occasional bikes and pedestrians. Table 1 presents the distribution of selected images across each dataset unit. In ADCD, we intentionally chose to include datasets that, though still popular, are less frequently utilized (except DDD). Since popular datasets such as KITTI and Cityscapes have been widely benchmarked and come with extensive support, serving as foundational resources for many tasks, their widespread usage makes them less ideal for introducing new challenges in adverse driving conditions. In terms of DDD, the aim is to contribute a completely new, high-quality, and well-annotated dataset, as proved via the experiments in Sect. 5.2. Overall, the goal is to provide fresh insights into driving scenarios that have not been as thoroughly explored in existing literature.

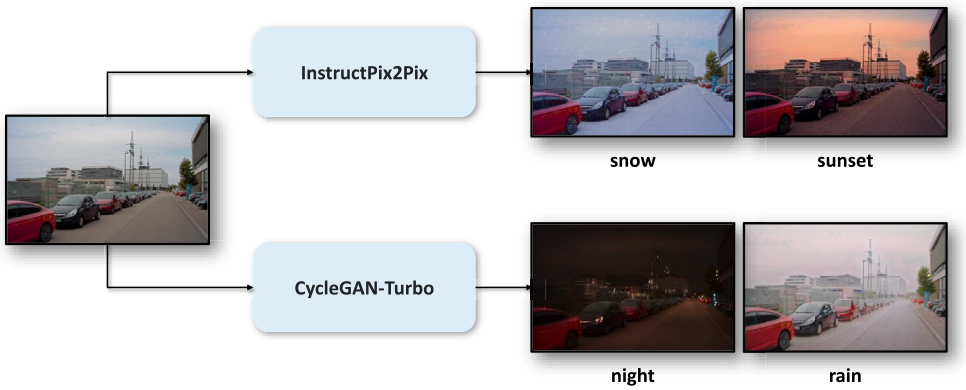
3.2 Weather-centric data augmentation

3.2.1 Single weather effects

Under eight effects, we categorized them into two groups: traditional methods and deep learning models, as shown in Fig. 3. Traditional methods, such as image blending and pixel-level modifications, offer advantages over deep learning models by being computationally efficient, highly controllable, and faster to implement while still providing simple yet realistic effects. However, deep learning models are better suited for more intricate effects requiring semantic understanding or contextual transformation-such as creating water reflections on roads in a rainy scene or identifying streetlight locations to enhance nighttime brightness. These complex requirements are beyond the capabilities of traditional methods.



(a) Traditional methods



(b) Deep-learning models

Fig. 3 Examples of weather effects and augmentation techniques for single effect

- *Traditional methods* are associated with the following effects:
  - *Crack on windshields*. We collected dozens of pure crack images from sources such as Google and FreePik and blended them with the main image. To ensure diversity, we selected cracks with varying shapes and applied transformations such as rotation and flipping. The seamless blending process began by randomly selecting a crack image, resizing it using random scaling factors, and cropping it at random points to fit precisely within the dimensions of the main image. We then adjusted the opacity level of the crack image to integrate it. This approach minimizes the likelihood of two images having identical crack placements, ensuring a highly varied dataset. The synthesized image I_{syn} is generated by blending the crack image I_{effect} with the main image I by adjusting the opacity α(here, α is randomly selected between 0.3 and 0.5).

$$I_{syn} = (1 - \alpha)I + \alpha I_{effect}. \quad (1)$$

  - *Sun flare*. Similar to cracks, we collected flare images from major sources. However, unlike cracks, which can be placed anywhere on the image due to the windshield covering the entire frame in dashcam captures, flares require realistic placement, ideally positioned in the sky area. To achieve this, we utilize Grounding DINO [60], an object detection model guided by prompts, to identify the sky regions. The detected bounding boxes are then processed by the SAM model [61] to generate precise sky segmentations. We accurately position and blend the flare images using these segmented areas as masks. The blending technique applied to flares is similar to the method used for cracks, ensuring the uniqueness of

each flare-blended image. For images where the sky cannot be detected, we use flare images that mimic sunlight reflections on the windshield, creating the effect of reflected flares rather than direct ones, thereby enhancing the dataset's realism.

- *Haze*. Although haze can be categorized into several subtypes, such as fog, mist, smoke, vog, and smog, this approach simplifies the concept by focusing on the visual texture that reduces image clarity, simulating the typical obscuration caused by haze. Perlin noise is generated to create natural, continuous patterns resembling haze. Gaussian blur is then applied to smooth the noise and ensure seamless transitions. Finally, the noise is blended with the main image using alpha transparency, with adjustments to the noise's contrast or opacity to control the haze's density.
- *Raindrop* Unlike rain, the formation of raindrops on a windshield requires simulating a real lens effect. To model this, an elliptical shape, resembling an egg, is created by merging a circle and an oval. A blur algorithm is then applied to soften the edges of the shape, mimicking the optical distortion caused by the refraction of light through the lens-like curvature of the raindrop. This method emulates the visual effects observed in real-world scenarios, where the edges of raindrops appear diffused due to the interaction of light and the windshield's curvature. Similar to other effects, we configure the size and number of raindrops in the generator to keep the dataset's diversity.

- *Deep-learning models* are associated with the following effects:

  - *Snow and Sunset*. We employ InstructPix2Pix, which applies transformation to an input image I using a diffusion-based generative model guided by text instructions.

$$I_{syn} = f(I, T), \tag{2}$$

    where T is the text instruction and f is the neural network function of the InstructPix2Pix model, which learns to modify the image based on T. We guide the generation process with prompts such as "Make it winter" or "Make it sunset". Unlike other diffusion models, InstructPix2Pix is trained from paired data, maintaining the natural integrity of the scene while introducing localized changes to specific objects or areas. This approach aligns with the requirements of our dataset, which prioritizes realistic traffic scenarios. To ensure minimal deviation from the original image, we configure a low number of steps and increase the weighting of the image relative to the text prompts. After the initial generation, we enhance the seasonal effect for winter images by overlaying a snow layer using a similar method for introducing cracks.
  - *Rain and Night*. CycleGAN-Turbo, trained on an extensive dataset focusing on clear-to-rain and day-to-night transitions, outperforms InstructPix2Pix for these specific effects due to its domain-specific training. The transformation of an input image I can be formulated as follows:

$$I_{syn} = G(I), \tag{3}$$

    where $G: X \rightarrow Y$ is the generator function that maps the image from domain X to domain Y. However, CycleGAN-Turbo's night scene transformations often introduce multiple bright spots, resembling moons, in large sky regions. This behavior likely arises from the model learning to replicate cityscapes with abundant artificial lighting, which it then applies indiscriminately to all images. To mitigate this issue and produce more realistic results, the brightness of the sky area, extracted using the SAM model previously applied for flare effects, is reduced. This adjustment guides the model toward recognizing the dark sky characteristic before transformation.

3.2.2 Mixed weather effects

The effect is tailored to reflect practical driving scenarios. Two mixed effects are implemented using only traditional methods: Haze and raindrops simulate conditions where atmospheric humidity is exceptionally high, while flare reflections on cracks are added to heighten visual challenges, making the driving experience more demanding. Two other effects combine deep learning models with traditional methods: While rain and raindrops depict the obvious scene of heavy rainfall, haze and night scenes further demonstrate the impact of high humidity during nighttime conditions.

3.3 Evaluating object detection models

In this paper, we benchmark a range of state-of-the-art object detection models:

- *YOLO (You Only Look Once)*. Known for balancing speed and accuracy, YOLO is ideal for real-time tasks

like traffic monitoring. We evaluate versions from YOLOv5 to YOLOv11, each introducing improvements such as improved feature extraction, reduced computational overhead, or better detection of small objects. By comparing these variants, we aim to understand their relative performance.

- *DETR (Detection Transformer)*. DETR leverages attention mechanisms for end-to-end object detection. Unlike traditional methods that rely on region proposals, DETR excels in complex scenes with overlapping objects, making it particularly relevant for the diverse scenarios in ADCD. However, its higher computational demands pose challenges for real-time applications.
- *R-CNN and Faster R-CNN (Region-based Convolutional Neural Networks)*. These two-stage models generate region proposals before classification and regression. While highly accurate, these models are computationally intensive. In addition, Faster R-CNN enhances efficiency by integrating a Region Proposal Network (RPN) into the architecture, reducing inference time.
- *RetinaNet*. RetinaNet utilizes a single-stage object detection architecture augmented by a Focal Loss function, which addresses class imbalance, particularly between foreground and background objects. This model strikes a balance between speed and accuracy, with notable strength in detecting smaller objects.
- *SSD (Single Shot MultiBox Detector)*. SSD is designed for real-time applications, focusing on maintaining speed without compromising accuracy. It employs a multi-scale feature map approach to efficiently detect objects of varying sizes.

4 Experiments

4.1 Evaluation metrics

We validate the performance of object detection models based on two categories: class-level performance and overall performance.

For each class, we calculate the Average Precision (AP). The AP score is the area under the curve (AUC) of the Precision-Recall curve, which is derived from the predictions of the model compared to the ground truth. It is suggested that by interpolating all points, which reduces the impact of the "wiggles" in the curve caused by small variations in the ranking of samples [23], the Average Precision can be interpreted as an approximated AUC of the Precision-Recall curve.

$$\begin{aligned} \mathrm{AP} &= \sum (r_{n+1} - r_n) p_{\mathrm{interp}}(r_{n+1}), \text{ where } p_{\mathrm{interp}}(r_{n+1}) \\ &= \max_{\tilde{r} \geq r_{n+1}} p(\tilde{r}) \end{aligned} \tag{4}$$

When computing Precision/Recall, the IoU between the predicted bounding box and the ground truth bounding box must meet or exceed a threshold $\theta_{IoU} = 0.5$ to be consider a True Positive. The confidence score of the prediction is also at least $\theta_{conf} \geq 0.25$, filtering out predictions that could inflate the False Positive rate. Figure 4 visualizes the detection results of different models across diverse weather scenarios.

The final evaluation metric, mean Average Precision (mAP), is used to evaluate the overall performance of the model. It is computed as the mean of the AP values across all classes:

$$\mathrm{mAP} = \frac{1}{C} \sum_{i=1}^{C} \mathrm{AP}_i, \tag{5}$$

where C is the total number of classes, and AP_i is the AP score for the i-th class.

4.2 Experimental setups

In this work, we focus on transportation-related objects, particularly performing predictions on six classes, namely, car, truck, bicycle, motorcycle, person, and traffic light. The distribution of the base dataset is provided in Table 2. We utilize pretrained state-of-the-art object detection models and proceed with two main experiments.

First, we hypothesize that these models, which are primarily trained under clear-weather data, will experience a performance drop when exposed to adverse effects. They are initially tested on the clear-weather dataset to establish reference performance. Then the models are evaluated on augmented datasets in ADCD, including single and mixed conditions, to assess robustness to environmental challenges.

Second, we investigate whether the proposed ADCD dataset can meaningfully support the future development of object detection models in adverse weather conditions. Specifically, we analyze if restoration techniques can sufficiently recover detection performance to match that of non-affected images.

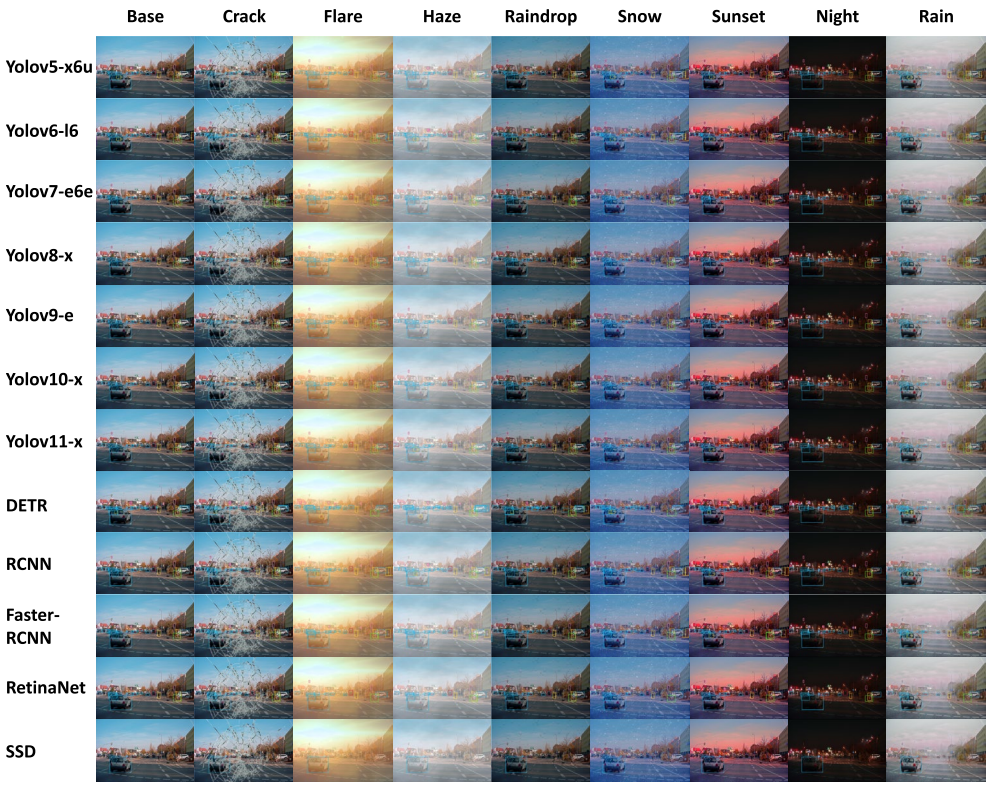


Fig. 4 Visualization of the object detection results in different weather conditions. Different objects are highlighted using different colors: cars (blue), trucks (red), persons (green), bicycles (purple), motorcycles (yellow), and traffic lights (pink). (For a better viewing experience, please zoom to at least 400%)

Table 2 Distribution of transportation-related classes in our dataset

Class	Count
Car	199,840
Truck	34,145
Bicycle	5167
Motorcycle	32,050
Person	93,319
Traffic light	23,736

5 Results

5.1 Overall performance under weather effects

Table 3 summarizes the mAP score of each model across all weather conditions, and the accompanying heatmap visualization in Table 4 shows the percentage decrease in mAP scores compared to the baseline. In general, all models experience a performance decrease when weather effects are applied, especially under mixed-weather datasets.

YOLO variants perform competitively with minimal differences among them. YOLOv5-x6 and Yolov6-l6, in particular, are the most robust and stable, consistently outperforming others with both high mAP scores and low drop rates. RCNN-based models also perform well and closely to one another, though less robust than the YOLOs. RetinaNet follows a similar pattern but delivers below-average performance. In contrast, SDD struggles in all scenarios with scores below usable levels, while DETR show significant mAP degradation under mixed conditions, with performance drops reaching over 70% despite moderate performances under a few cases, indicating its instability in environments that reduce visibility.

Regarding adverse effects, some single ones such as Crack, Rain, and Night cause noticeable performance drops than others in the same group. Mixed weather conditions result in the most significant degradation overall. Even the best-performing YOLO models are affected; for example, YOLOv6-l6 suffers a 30% drop under Haze + Night. These results show that while some models maintain reasonable accuracy under isolated effects, mixed conditions remain a major challenge, emphasizing the importance of datasets like ADCD for improving detection in real-world scenarios.

5.2 Class impact on detection performance in adverse weather

The mAP of models was influenced by the AP score of each class, as illustrated in Tables 5 and 6. Each class achieves its highest score on the base dataset and experiences varying degrees of performance drop under different conditions.

Table 3 Overall mAP scores (for six classes) under all weather conditions

	Base	Crack	Flare	Haze	Raindrop	Snow	Sunset	Night	Rain
Yolov5-x6	43.52	35.99	41.45	38.92	41.28	41.72	40.16	35.30	35.23
Yolov6-l6	**46.91**	**39.70**	**45.00**	**42.99**	**45.02**	**45.68**	**44.63**	**39.43**	**39.46**
Yolov7-e6e	37.74	31.32	35.01	34.15	36.19	36.91	35.77	30.28	32.82
Yolov8-x	38.35	31.53	36.09	34.93	36.64	37.53	36.45	30.14	33.27
Yolov9-e	39.23	32.01	35.98	34.62	37.24	37.95	37.27	31.11	33.59
Yolov10-x	37.17	30.91	34.81	33.45	35.58	36.36	35.19	28.46	31.90
Yolov11-x	38.82	32.00	36.48	34.62	36.80	37.74	36.17	29.60	32.84
DETR	34.91	27.10	16.25	15.95	29.60	33.00	32.65	11.65	21.43
RCNN	40.32	34.45	33.54	31.69	38.05	35.94	38.18	30.88	32.90
Faster-RCNN	39.69	33.78	32.70	31.29	37.35	36.30	37.54	30.58	32.38
RetinaNet	32.00	26.44	24.95	23.66	29.20	26.47	28.70	22.08	24.42
SSD	5.44	4.81	3.13	3.07	5.19	4.04	4.82	3.84	4.50

	Base	Haze raindrop	Haze night	Rain raindrop	Crack flare
Yolov5-x6	43.52	39.25	33.91	32.95	30.40
Yolov6-l6	**46.91**	**43.23**	**38.40**	**37.42**	**33.74**
Yolov7-e6e	37.74	34.52	28.98	30.91	24.89
Yolov8-x	38.35	35.12	28.19	30.87	25.25
Yolov9-e	39.23	34.94	29.15	31.30	24.83
Yolov10-x	37.17	33.80	26.67	29.47	24.85
Yolov11-x	38.82	27.35	28.31	30.27	26.41
DETR	34.91	16.38	7.89	15.42	8.43
RCNN	40.32	33.11	27.96	29.17	21.60
Faster-RCNN	39.69	32.57	27.26	28.16	20.93
RetinaNet	32.00	24.36	19.77	20.35	15.20
SSD	5.44	3.37	2.39	4.14	2.07

The top performance for each condition is highlighted in bold

Table 4 Overall mAP scores (for six classes) under all weather conditions

	Crack	Flare	Haze	Raindrop	Snow	Sunset	Night	Rain	Haze Raindrop	Haze Night	Rain Raindrop	Crack Flare
Yolov5-x6	17.3	4.76	10.57	5.15	4.14	7.72	18.89	19.05	9.81	22.08	24.29	30.15
Yolov6-l6	15.37	4.07	8.36	4.03	2.62	4.86	15.95	15.88	7.84	18.14	20.23	28.08
Yolov7-e6e	17.01	6.7	7.23	4.32	2.04	5.64	16.91	16.24	8.53	23.21	18.1	34.05
Yolov8-x	17.78	5.89	8.92	4.46	2.14	4.95	21.41	11.55	8.42	26.49	19.5	34.16
Yolov9-e	18.4	8.34	11.88	10.12	6.53	9.02	21.18	20.04	10.94	25.69	20.21	36.71
Yolov10-x	16.84	6.35	9.66	6.91	2.07	5.38	23.43	16.06	9.07	28.25	20.72	33.15
Yolov11-x	17.39	6.18	11.75	5.56	2.78	6.34	21.66	15.51	29.55	27.07	22.02	31.97
DETR	15.07	18.05	18.68	7.24	3.57	13.35	21.47	26.64	48.67	75.27	51.68	73.58
RCNN	20.7	13.71	22.56	14.42	5.72	16.47	26.15	24.27	19.09	31.67	28.71	47.21
Faster-RCNN	14.89	17.61	21.16	5.9	8.54	5.42	22.95	18.42	17.94	31.32	29.05	47.27
RetinaNet	17.38	22.03	26.06	8.75	17.28	10.31	31	23.69	23.88	38.22	36.41	52.5
SSD	11.58	42.46	43.57	4.6	25.74	11.4	29.41	17.28	38.05	56.07	23.9	61.95

*Lower values, or brighter colors, indicate better * robustness

Additionally, most models follow a similar class-wise pattern: high scores for cars, average performance for trucks, motorcycles, persons, and traffic lights, while bicycles show the lowest scores.

Under single weather effects, cars are correctly detected across almost all models, with AP scores around 60. Their stability is comparable to that of persons, but the latter tends to have lower accuracy. Trucks and motorcycles, scoring from 30 to 35, decline moderately under Crack and Night conditions. Night also slightly affects bicycles, while traffic lights experience a noticeable drop in performance compared to other effects, possibly due to bright light spots being misidentified as traffic signals. In terms of models, DETR, as mentioned above, performs well under certain conditions but struggles significantly under others—a pattern clearly reflected in class-wise scores. Furthermore, R-CNN outperforms YOLOv6-l6 in detecting traffic lights under some conditions. There are also anomalies, such as YOLOv10-x performing unusually poorly on persons under the Night compared to other YOLO variants and effects.

Under mixed effects, nearly all models witness a noticeably lower score across all classes, especially with Crack+Flare. Similar patterns observed under single effects persist, such as the stability of cars and persons, as well as

Table 5 Class-wise AP scores under single conditions

	Model	Base	Crack	Flare	Haze	Raindrop	Snow	Sunset	Night	Rain
Car	Yolov5-x6	65.45	59.15	64.89	62.66	64.10	62.78	64.48	62.52	61.40
	Yolov6-l6	**69.12**	**63.45**	**68.90**	**67.10**	**68.27**	**67.37**	**68.69**	**66.67**	**66.10**
	Yolov7-e6e	65.09	57.83	64.00	62.49	63.59	62.72	64.67	61.47	59.89
	Yolov8-x	64.25	56.97	63.57	61.37	62.63	61.84	63.85	60.03	58.80
	Yolov9-e	64.10	57.16	63.68	61.42	62.57	62.05	63.89	60.72	59.10
	Yolov10-x	63.05	56.43	62.48	59.82	61.61	60.66	62.74	58.26	57.48
	Yolov11-x	64.07	57.11	63.50	60.96	62.33	61.94	63.59	59.92	58.31
	DETR	60.32	51.04	40.18	37.27	56.05	59.02	58.28	27.01	49.54
	RCNN	62.54	55.87	61.14	58.32	60.61	56.22	62.24	63.14	57.39
	Faster-RCNN	62.90	56.29	61.81	59.60	61.14	58.26	62.41	62.50	57.78
	RetinaNet	60.35	51.28	56.11	54.08	57.30	51.05	58.32	54.34	50.41
	SSD	16.14	14.75	12.41	11.91	15.55	13.66	15.21	14.08	14.30
Truck	Yolov5-x6	39.82	31.87	38.75	37.35	38.29	37.73	34.65	30.89	33.02
	Yolov6-l6	**42.12**	**34.92**	**40.74**	**40.01**	**40.77**	**40.09**	**38.14**	**35.24**	**36.91**
	Yolov7-e6e	37.24	29.33	35.98	35.24	36.37	35.06	34.64	29.55	32.58
	Yolov8-x	36.41	28.20	35.88	35.13	35.37	35.74	32.73	27.84	31.41
	Yolov9-e	37.71	28.78	34.73	34.05	36.22	35.10	34.29	28.57	32.02
	Yolov10-x	35.89	28.10	34.87	34.28	34.68	34.46	32.32	27.70	31.47
	Yolov11-x	36.51	28.33	35.96	34.44	35.69	34.44	32.02	29.05	30.65
	DETR	27.01	17.71	5.38	7.59	21.49	22.73	21.40	5.61	11.36
	RCNN	37.79	29.90	31.34	31.28	36.49	33.18	33.83	23.64	31.08
	Faster-RCNN	38.18	29.83	31.20	31.69	36.81	34.14	33.93	26.21	32.69
	RetinaNet	31.54	23.72	24.52	25.42	30.19	27.22	26.38	18.15	26.16
	SSD	7.44	6.12	2.68	2.73	7.29	4.00	5.98	3.77	5.37
Bicycle	**Yolov5-x6**	**25.62**	**20.82**	**24.02**	**22.47**	**25.15**	25.43	20.89	**21.74**	18.04
	Yolov6-l6	24.77	20.71	23.21	22.36	24.71	**25.81**	**21.37**	21.37	**19.00**
	Yolov7-e6e	17.95	14.67	16.34	15.87	17.10	18.16	13.73	12.41	14.67
	Yolov8-x	17.97	14.79	16.87	16.69	17.51	18.70	14.65	13.25	15.72
	Yolov9-e	18.83	15.06	17.61	16.89	17.87	19.93	15.45	13.89	15.05
	Yolov10-x	17.19	14.36	16.48	15.87	16.91	18.25	13.89	12.32	14.68
	Yolov11-x	18.55	15.48	17.71	17.02	18.27	19.57	14.78	13.03	15.63
	DETR	18.18	15.58	6.76	6.94	15.68	16.86	12.20	4.52	4.70
	RCNN	19.33	17.60	15.94	15.37	19.53	18.96	15.14	12.30	15.69
	Faster-RCNN	17.60	16.06	14.17	13.36	17.55	18.00	13.38	11.28	13.87
	RetinaNet	13.89	12.70	10.40	8.94	13.49	13.03	9.01	7.27	10.16
	SSD	1.14	1.01	0.29	0.38	1.12	0.90	0.83	0.55	0.92
Motorcycle	Yolov5-x6	47.14	35.09	44.45	39.87	42.89	44.67	41.48	38.45	34.57
	Yolov6-l6	**51.85**	**39.71**	**48.58**	**45.07**	**48.11**	**49.36**	**46.70**	**42.65**	**38.95**
	Yolov7-e6e	35.28	26.98	31.97	30.46	33.32	35.32	33.11	29.81	29.43
	Yolov8-x	38.27	28.60	34.82	32.89	36.22	37.38	35.21	30.70	31.28
	Yolov9-e	39.11	29.36	35.33	33.00	36.83	38.10	36.69	32.92	32.26
	Yolov10-x	35.79	27.54	33.07	30.83	34.09	35.24	33.52	29.08	28.93
	Yolov11-x	38.95	29.38	36.01	33.16	35.93	37.59	35.27	30.91	31.27
	DETR	32.88	23.12	6.78	7.80	24.36	24.07	33.46	10.01	14.46
	RCNN	35.26	29.09	26.38	23.20	31.62	29.44	32.05	29.79	26.33
	Faster-RCNN	34.67	28.15	24.96	22.86	31.19	30.94	30.86	29.57	24.93
	RetinaNet	23.12	18.78	15.43	13.24	20.13	18.88	20.22	17.86	16.44
	SSD	3.98	3.49	0.86	0.92	3.53	2.44	3.54	2.56	2.78

Table 5 (continued)

	Model	Base	Crack	Flare	Haze	Raindrop	Snow	Sunset	Night	Rain
Person	Yolov5-x6	49.60	40.90	46.61	42.85	45.71	45.56	45.00	36.39	38.10
	Yolov6-l6	**55.51**	**46.89**	**53.17**	**49.31**	**52.12**	**52.72**	**53.17**	**43.13**	**44.36**
	Yolov7-e6e	41.08	34.33	37.17	36.12	39.13	39.87	39.13	30.36	35.35
	Yolov8-x	42.50	35.40	38.92	37.32	39.72	40.40	40.17	30.87	35.56
	Yolov9-e	42.51	35.18	38.57	36.61	39.73	39.64	39.81	31.01	36.51
	Yolov10-x	39.64	33.28	36.06	34.28	37.31	37.92	37.21	27.70	33.29
	Yolov11-x	41.67	34.44	37.97	35.88	38.65	39.36	38.66	29.14	34.48
	DETR	42.61	33.83	25.93	23.22	37.32	43.43	39.80	19.27	31.36
	RCNN	46.86	40.18	39.32	35.53	43.18	42.94	44.51	34.38	38.77
	Faster-RCNN	46.76	39.96	38.83	35.10	42.98	42.72	44.61	33.81	38.33
	RetinaNet	33.55	27.97	27.02	24.44	29.87	27.70	29.79	21.41	27.33
	SSD	3.89	3.38	2.54	2.49	3.57	3.21	3.34	2.09	3.50
Traffic light	Yolov5-x6	33.48	28.12	29.98	28.33	31.55	34.17	34.43	21.83	26.22
	Yolov6-l6	38.06	32.51	**35.42**	**34.12**	36.14	**38.75**	39.73	**27.52**	**31.44**
	Yolov7-e6e	29.78	24.78	24.62	24.74	27.61	30.32	29.31	18.05	25.03
	Yolov8-x	30.72	25.25	26.48	26.18	28.39	31.13	32.12	18.14	26.86
	Yolov9-e	33.12	26.51	25.96	25.77	30.25	32.90	33.49	19.58	26.61
	Yolov10-x	31.44	25.75	25.92	25.63	28.89	31.63	31.47	15.72	25.53
	Yolov11-x	33.20	27.23	27.72	26.28	29.92	33.55	32.68	15.54	26.69
	DETR	28.44	21.31	12.49	12.85	22.72	31.89	30.73	3.46	17.13
	RCNN	**40.16**	**34.08**	27.14	26.45	**36.90**	34.88	**41.31**	22.04	28.14
	Faster-RCNN	38.00	32.37	25.22	25.14	34.42	33.77	40.03	20.14	26.69
	RetinaNet	29.52	24.16	16.20	15.82	24.25	20.97	28.50	13.44	16.03
	SSD	0.06	0.13	0.00	0.00	0.07	0.04	0.02	0.01	0.14

The top performance for each condition is highlighted in bold

the high false detection rate of traffic lights under Night (with Haze). In terms of models, DETR and SSD perform significantly worse across all conditions. Meanwhile, YOLOv6-l6 and YOLOv5-x6 show a substantial difference from other YOLO variants in certain classes, particularly bicycles and motorcycles.

5.3 Comparison between non-effect and restored images

We further compare the performance of object detection methods on non-effect images and restored images. We aim to restore the hazy images to a state resembling the Base dataset. In particular, we utilized state-of-the-art dehazing models, SFNet [62] and CORUN [63], on the Haze dataset (generated from the original non-effect images).

Then, we assess model performance across three settings, namely Base (non-effect images), Haze, and Dehazing. As shown in Fig. 5, restoring the Haze images with SFNet and CORUN does yield a clear improvement over the Haze input. However, both methods still fall short of the results on "Base", remaining much closer to the Haze baseline than to the clean images. This may be attributed to information loss during the dehazing process, domain shifts introduced by restoration artifacts, and visual inconsistencies such as unnatural lighting or contrast. These results indicate that image restoration alone is insufficient to bridge the gap in our task, underscoring the value of comprehensive datasets like ADCD for both training and evaluation under diverse weather conditions.

5.4 Discussion

The aforementioned experiments emphasize the critical impact of weather effects on the evaluation of the model. All models experience performance degradation compared to the base dataset's score, emphasizing the need for reinforcing the dual importance of advanced architectures and high-quality datasets in developing robust object detection systems in the real world. It also reveals that certain weather effects can cause specific models to degrade significantly, even if those models perform well under other conditions, indicating that performance may be condition-specific. Moreover, the need to examine class-wise AP scores becomes evident, as it helps identify bottlenecks in model performance under specific effects. For example, under night conditions, models may falsely detect light spots, such as headlights or reflections, revealing class-specific vulnerabilities that could be masked when only considering overall metrics. Furthermore, the imbalance in performance across different classes suggests that certain categories are more prone to degradation than others, pointing to a potential

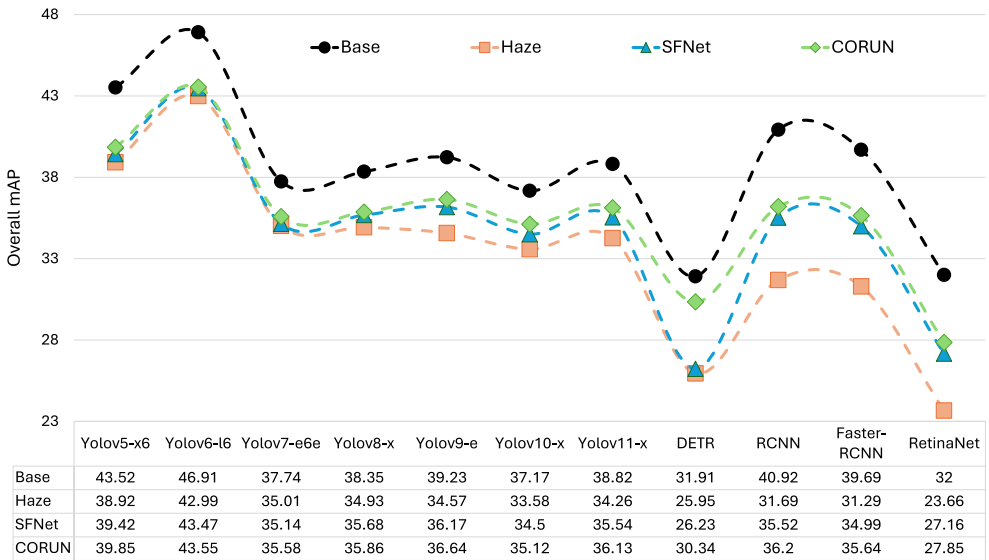
Table 6 Class-wise AP scores under mixed conditions

	Model	Base	Haze raindrop	Haze night	Rain raindrop	Crack flare
Car	Yolov5-x6	65.45	63.11	61.06	58.58	53.38
	Yolov6-l6	**69.12**	**67.72**	**65.92**	**64.43**	**57.93**
	Yolov7-e6e	65.09	62.84	59.30	57.43	50.03
	Yolov8-x	64.25	61.66	56.80	55.88	48.99
	Yolov9-e	64.10	61.73	58.29	55.99	49.10
	Yolov10-x	63.05	60.27	55.42	54.28	49.21
	Yolov11-x	64.07	51.31	57.55	55.29	51.02
	DETR	60.32	41.10	20.83	39.62	24.69
	RCNN	62.54	58.88	57.43	52.82	44.01
	Faster-RCNN	62.90	60.17	56.94	52.64	44.24
	RetinaNet	60.35	54.23	48.71	44.44	36.99
	SSD	16.14	12.55	9.86	13.30	8.75
Truck	Yolov5-x6	39.82	37.75	30.44	31.55	26.60
	Yolov6-l6	**42.12**	**40.38**	**34.39**	**35.24**	**28.73**
	Yolov7-e6e	37.24	35.78	28.23	30.35	22.21
	Yolov8-x	36.41	35.34	26.97	29.00	21.66
	Yolov9-e	37.71	34.63	27.16	29.38	20.07
	Yolov10-x	35.89	34.96	26.72	29.38	21.25
	Yolov11-x	36.51	30.62	28.20	28.46	22.62
	DETR	27.01	6.58	3.28	5.97	1.74
	RCNN	37.79	33.67	22.86	28.00	15.62
	Faster-RCNN	38.18	34.28	23.82	28.31	15.37
	RetinaNet	31.54	27.40	18.65	20.48	10.82
	SSD	7.44	3.65	1.71	5.05	1.55
Bicycle	**Yolov5-x6**	**25.62**	**22.76**	19.96	17.00	**17.64**
	Yolov6-l6	24.77	22.75	**20.21**	**18.04**	17.55
	Yolov7-e6e	17.95	15.51	11.61	13.57	12.26
	Yolov8-x	17.97	16.38	12.38	14.08	12.45
	Yolov9-e	18.83	16.74	12.54	13.84	12.57
	Yolov10-x	17.19	15.94	10.94	13.41	12.31
	Yolov11-x	18.55	13.72	11.97	14.36	13.64
	DETR	18.18	6.28	3.08	2.20	4.00
	RCNN	19.33	16.33	10.64	13.63	11.93
	Faster-RCNN	17.60	14.16	9.37	11.79	10.80
	RetinaNet	13.89	9.42	5.81	7.91	7.99
	SSD	1.14	0.31	0.15	0.84	0.13
Motorcycle	Yolov5-x6	47.14	40.20	36.36	31.47	29.41
	Yolov6-l6	**51.85**	**45.16**	**40.88**	**35.71**	**32.73**
	Yolov7-e6e	35.28	31.32	28.38	27.29	21.03
	Yolov8-x	38.27	33.97	28.44	28.36	22.49
	Yolov9-e	39.11	33.72	30.58	29.44	22.88
	Yolov10-x	35.79	31.90	26.93	25.94	21.67
	Yolov11-x	38.95	18.45	28.98	28.19	24.30
	DETR	32.88	6.65	3.88	7.49	1.85
	RCNN	35.26	25.89	27.63	21.01	15.52
	Faster-RCNN	34.67	25.07	26.91	19.68	14.48
	RetinaNet	23.12	14.72	15.03	12.73	8.90
	SSD	3.98	1.00	1.02	2.24	0.38

Table 6 (continued)

	Model	Base	Haze raindrop	Haze night	Rain raindrop	Crack flare
Person	Yolov5-x6	49.60	42.88	34.91	34.89	34.46
	Yolov6-l6	**55.51**	**49.33**	**42.19**	**41.92**	**40.53**
	Yolov7-e6e	41.08	36.78	29.43	33.67	27.65
	Yolov8-x	42.50	37.59	29.03	33.20	28.83
	Yolov9-e	42.51	36.77	29.47	34.15	28.15
	Yolov10-x	39.64	34.67	26.27	30.66	26.93
	Yolov11-x	41.67	24.43	27.71	31.45	28.27
	DETR	42.61	24.63	14.40	25.63	13.62
	RCNN	46.86	37.85	31.72	35.04	27.10
	Faster-RCNN	46.76	37.55	30.71	34.63	26.54
	RetinaNet	33.55	26.00	20.02	23.91	17.77
	SSD	3.89	2.68	1.62	3.20	1.59
Traffic light	Yolov5-x6	33.48	28.79	20.74	24.18	20.91
	Yolov6-l6	38.06	**34.05**	**26.78**	**29.17**	**24.98**
	Yolov7-e6e	29.78	24.91	16.95	23.16	16.17
	Yolov8-x	30.72	25.80	15.55	24.68	17.09
	Yolov9-e	33.12	26.07	16.84	24.98	16.23
	Yolov10-x	31.44	25.03	13.72	23.17	17.72
	Yolov11-x	33.20	25.58	15.43	23.86	18.59
	DETR	28.44	13.05	1.87	11.59	4.66
	RCNN	**40.16**	26.06	17.46	24.50	15.44
	Faster-RCNN	38.00	24.22	15.83	21.89	14.17
	RetinaNet	29.52	14.41	10.39	12.62	8.75
	SSD	0.06	0.00	0.00	0.18	0.00

The top performance for each condition is highlighted in bold



	Yolov5-x6	Yolov6-l6	Yolov7-e6e	Yolov8-x	Yolov9-e	Yolov10-x	Yolov11-x	DETR	RCNN	Faster-RCNN	RetinaNet
Base	43.52	46.91	37.74	38.35	39.23	37.17	38.82	31.91	40.92	39.69	32
Haze	38.92	42.99	35.01	34.93	34.57	33.58	34.26	25.95	31.69	31.29	23.66
SFNet	39.42	43.47	35.14	35.68	36.17	34.5	35.54	26.23	35.52	34.99	27.16
CORUN	39.85	43.55	35.58	35.86	36.64	35.12	36.13	30.34	36.2	35.64	27.85

Fig. 5 Comparison of object detection performance on Base, Haze, and the dehazed images from two dehazing models, SFNet [62] and CORUN [63]. SSD is excluded for clearer visualization, as its low mAP (∼0–5%) would compress the scale relative to other models (∼ 20–50%)

need for class-balanced training or evaluation strategies. In short, all these insights suggest that training models on more targeted, condition-specific data could improve their robustness and mitigate such failure cases.

Regarding the restoration approach, although improving model performance is essential, single-purpose restoration may be impractical under mixed effects, where multiple types of noise co-occur, while most models are designed to address a specific kind of visual degradation (e.g., Haze, Snow, Rain), which limits their generalizability. Meanwhile, combining multiple restoration models can add complexity, especially in real-time applications where rapid perception is critical. Moreover, some visual effects, such as Crack and Flares, still remain challenging to restore. These limitations highlight the need for more robust object detection systems that can handle degraded inputs. In this context,

comprehensive datasets like ADCD are valuable for both training and evaluation under diverse conditions.

For future work, beyond the dataset itself, the proposed work presents promising opportunities for broader impact. While expanding with additional effects, such as sand to simulate dusty conditions, snow-covered vehicles, or complex combinations such as snow at night, can help challenge object detection models further, a transferable construction approach opens even broader possibilities. Instead of being confined to a specific domain, such methods can be adapted to build diverse datasets for other real-world challenges. For example, wildlife object detection often involve capturing images under adverse conditions such as rain, haze, or night surveillance. These conditions may cause data imbalance, as there is typically less available data for these scenarios compared to clear images. The work we propose can help diversify and balance such datasets, making it possible to build more robust models.

6 Conclusion

In this paper, we assessed the robustness of state-of-the-art object detection models under adverse weather conditions. To this end, we collected the Adverse Driving Conditions Dataset (ADCD) which comprises 50,000 images augmented with 12 unique weather effects. Through experiments, YOLOs consistently demonstrated superior performance across most weather scenarios among the evaluated models. However, all models experienced notable performance degradation under adverse conditions, which was evidenced through detailed analysis, including percentage drops in mAP and class-wise AP, emphasizes the need for more resilient architectures capable of addressing multiple environmental challenges. Moreover, while image restoration methods have shown some promise, their impact remains limited, especially in cases involving mixed or severe effects. By identifying specific gaps in model performance and proposing adaptable methodology for dataset construction, our work contributes to advancing the safety and reliability of computer vision systems in critical applications such as autonomous vehicles and traffic monitoring.

Acknowledgements This research was supported by the National Science Foundation (NSF) under Grant 2025234.

Author contributions Ba-Thinh Tran-Le: Writing—original draft, Methodology, Data curation, Visualization, Validation, Investigation, Conceptualization. Vatsa Patel: Writing—original draft, Data curation, Investigation, Conceptualization. Viet-Tham Huynh: Writing—original draft, Methodology, Data curation, Investigation, Conceptualization. Mai-Khiem Tran: Methodology, Validation. Kunal Agrawal: Software, Data curation, Investigation, Conceptualization. Minh-Triet Tran: Supervision, Writing—original draft, Methodology, Resources, Project administration, Conceptualization. Tam V. Nguyen: Supervision, Writing—original draft, Methodology, Project administration, Investigation, Conceptualization, Funding acquisition.

Availability of data and materials The data is available on request.

Declarations

Competing interests The authors declare no competing interests.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision, pp. 213–229. Springer (2020)
2. Paulin, G., Ivasic-Kos, M.: Review and analysis of synthetic dataset generation methods and techniques for application in computer vision. Artif. Intell. Rev. **56**(9), 9221–9265 (2023)
3. He, R., Sun, S., Yu, X., Xue, C., Zhang, W., Torr, P., Bai, S., Qi, X.: Is synthetic data from generative models ready for image recognition? arXiv preprint arXiv:2210.07574 (2022)
4. Johnson-Roberson, M., Barto, C., Mehta, R., Sridhar, S.N., Rosaen, K., Vasudevan, R.: Driving in the matrix: can virtual worlds replace human-generated annotations for real world tasks? arXiv preprint arXiv:1610.01983 (2016)
5. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
6. Patel, V.S., Agrawal, K., Nguyen, T.V.: A comprehensive analysis of object detectors in adverse weather conditions. In: 2024 58th Annual Conference on Information Sciences and Systems (CISS), pp. 1–6 (2024)
7. Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., Yang, R.: The apolloscape dataset for autonomous driving. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (2018)
8. Varma, G., Subramanian, A., Namboodiri, A., Chandraker, M., Jawahar, C.: Idd: A dataset for exploring problems of autonomous navigation in unconstrained environments. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1743–1751. IEEE (2019)
9. Geyer, J., Kassahun, Y., Mahmudi, M., Ricou, X., Durgesh, R., Chung, A.S., Hauswald, L., Pham, V.H., Mühlegg, M., Dorn, S., et al.: A2d2: audi autonomous driving dataset. arXiv preprint arXiv:2004.06320 (2020)
10. Redmon, J., Divvala, S.K., Girshick, R.B., Farhadi, A.: You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)
11. Li, C., Li, L., Geng, Y., Jiang, H., Cheng, M., Zhang, B., Ke, Z., Xu, X., Chu, X.: YOLOv6 v3.0: a full-Scale Reloading arXiv preprint arXiv:2301.05586 (2023)
12. 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. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 7464–7475 (2023)
13. Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics YOLOv8 (2023). https://github.com/ultralytics/ultralytics
14. Wang, C.-Y., Yeh, I.-H., Liao, H.-Y.M.: YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information (2024)

15. Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., Ding, G.: Yolov10: real-time end-to-end object detection. arXiv preprint arXiv:2405.14458 (2024)
16. Jocher, G., Qiu, J.: Ultralytics YOLO11 (2024). https://github.com/ultralytics/ultralytics
17. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587 (2014)
18. Ren, S.: Faster r-cnn: Towards real-time object detection with region proposal networks. arXiv preprint arXiv:1506.01497 (2015)
19. Lin, T.: Focal loss for dense object detection. arXiv preprint arXiv:1708.02002 (2017)
20. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., Berg, A.C.: Ssd: Single shot multibox detector. In: Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, pp. 21–37. Springer (2016)
21. Janai, J., Güney, F., Behl, A., Geiger, A., et al.: Computer vision for autonomous vehicles: problems, datasets and state of the art. Found. Trends® Comput. Graph. Vis. **12**(1–3), 1–308 (2020)
22. Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
23. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (VOC) challenge. Int. J. Comput. Vis. **88**, 303–338 (2010)
24. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: common objects in context. In: European Conference on Computer Vision, pp. 740–755. Springer (2014)
25. Leal-Taixé, L.: Motchallenge 2015: towards a benchmark for multi-target tracking. arXiv preprint arXiv:1504.01942 (2015)
26. Milan, A.: Mot16: a benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 (2016)
27. Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P.: High-resolution stereo datasets with subpixel-accurate ground truth. In: Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2–5, 2014, Proceedings 36, pp. 31–42. Springer (2014)
28. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. Int. J. Comput. Vis. **47**, 7–42 (2002)
29. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings, vol. 1. IEEE (2003)
30. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 406–413 (2014)
31. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the kitti dataset. Int. J. Robot. Res. **32**(11), 1231–1237 (2013)
32. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: a multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11621–11631 (2020)
33. Chang, M.-F., Lambert, J., Sangkloy, P., Singh, J., Bak, S., Hartnett, A., Wang, D., Carr, P., Lucey, S., Ramanan, D., et al.: Argoverse: 3d tracking and forecasting with rich maps. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8748–8757 (2019)
34. Mishra, R., Patel, V., Kim, H., Nguyen, T.V.: Road surface material recognition from dashboard cameras. In: 2024 International Symposium on Visual Computing (ISVC), pp. 359–370 (2024)
35. Houston, J., Zuidhof, G., Bergamini, L., Ye, Y., Chen, L., Jain, A., Omari, S., Iglovikov, V., Ondruska, P.: One thousand and one hours: self-driving motion prediction dataset. In: Conference on Robot Learning, pp. 409–418. PMLR (2021)
36. Tarel, J.-P., Hautiere, N., Cord, A., Gruyer, D., Halmaoui, H.: Improved visibility of road scene images under heterogeneous fog. In: 2010 IEEE Intelligent Vehicles Symposium, pp. 478–485. IEEE (2010)
37. Tarel, J.-P., Hautiere, N., Caraffa, L., Cord, A., Halmaoui, H., Gruyer, D.: Vision enhancement in homogeneous and heterogeneous fog. IEEE Intell. Transp. Syst. Mag. **4**(2), 6–20 (2012)
38. Sakaridis, C., Dai, D., Van Gool, L.: Semantic foggy scene understanding with synthetic data. Int. J. Comput. Vis. **126**, 973–992 (2018)
39. Fu, X., Huang, J., Zeng, D., Huang, Y., Ding, X., Paisley, J.: Removing rain from single images via a deep detail network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3855–3863 (2017)
40. Yang, W., Tan, R.T., Feng, J., Liu, J., Guo, Z., Yan, S.: Deep joint rain detection and removal from a single image. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1357–1366 (2017)
41. Liu, Y.-F., Jaw, D.-W., Huang, S.-C., Hwang, J.-N.: Desnownet: context-aware deep network for snow removal. IEEE Trans. Image Process. **27**(6), 3064–3073 (2018)
42. Gaidon, A., Wang, Q., Cabon, Y., Vig, E.: Virtual worlds as proxy for multi-object tracking analysis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4340–4349 (2016)
43. Ros, G., Sellart, L., Materzynska, J., Vazquez, D., Lopez, A.M.: The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3234–3243 (2016)
44. Richter, S.R., Vineet, V., Roth, S., Koltun, V.: Playing for data: ground truth from computer games. In: Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II 14, pp. 102–118. Springer (2016)
45. Richter, S.R., Hayder, Z., Koltun, V.: Playing for benchmarks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2213–2222 (2017)
46. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, vol. 27 (2014)
47. Radford, A.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
48. Denton, E.L., Chintala, S., Fergus, R., et al.: Deep generative image models using a laplacian pyramid of adversarial networks. In: Advances in Neural Information Processing Systems, vol. 28 (2015)
49. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: International Conference on Machine Learning, pp. 1857–1865. PMLR (2017)
50. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2794–2802 (2017)
51. Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In:

Proceedings of the IEEE International Conference on Computer Vision, pp. 2223–2232 (2017)
52. Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1125–1134 (2017)
53. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International Conference on Machine Learning, pp. 2256–2265 (2015). PMLR
54. Dhariwal, P., Nichol, A.: Diffusion models beat GANs on image synthesis. Adv. Neural. Inf. Process. Syst. **34**, 8780–8794 (2021)
55. Brooks, T., Holynski, A., Efros, A.A.: Instructpix2pix: learning to follow image editing instructions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18392–18402 (2023)
56. Parmar, G., Park, T., Narasimhan, S., Zhu, J.-Y.: One-step image translation with text-to-image models. arXiv preprint arXiv:2403.12036 (2024)
57. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Adv. Neural. Inf. Process. Syst. **33**, 1877–1901 (2020)
58. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. **21**, 1–67 (2020)
59. Chowdhery, A., et al.: Palm: scaling language modeling with pathways. arXiv preprint arXiv:2204.02311 (2022)
60. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Jiang, Q., Li, C., Yang, J., Su, H., et al.: Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In: European Conference on Computer Vision, pp. 38–55. Springer (2025)
61. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.-Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4015–4026 (2023)
62. Cui, Y., Tao, Y., Bing, Z., Ren, W., Gao, X., Cao, X., Huang, K., Knoll, A.: Selective frequency network for image restoration. In: The Eleventh International Conference on Learning Representations (2023)
63. Fang, C., He, C., Xiao, F., Zhang, Y., Tang, L., Zhang, Y., Li, K., Li, X.: Real-world image dehazing with coherence-based pseudo labeling and cooperative unfolding network. In: The Thirty-eighth Annual Conference on Neural Information Processing Systems (2024)

Ba-Thinh Tran-Le is a Master's student in the Department of Computer Science at the University of Dayton, where his research focuses on computer vision, machine learning, and AI system optimization for real-world applications. He earned his Bachelor's degree in Computer Science with distinction from the Vietnam National University - Ho Chi Minh City University of Science. Prior to graduate school, he spent three years as a Software Engineer, developing large-scale, user-centric platforms. His current work aims to bridge academic research with practical deployment, enhancing the efficiency of modern AI systems.

Vatsa S. Patel holds a Ph.D. in Computer Science from the University of Dayton, specializing in AI, machine learning, and computer vision. He is a Research Scientist at the University of Dayton Research Institute (UDRI). He has contributed to research in object detection, UAV signal processing, and quantum-enhanced learning, combining deep learning, generative models, and multimodal analysis. With over a dozen peer-reviewed publications, Vatsa is known for bridging theory and application through innovative, real-world solutions. He is an upcoming Assistant Professor of Computer Science at Penn State Harrisburg.

Viet-Tham Huynh is a researcher of Software Engineering Lab at the University of Science, Vietnam National University Ho Chi Minh City. He obtained his Bachelor and Master degrees from University of Science. He is also a visiting researcher to National Institute of Informatics, Japan in 2024. His research areas are computer vision, virtual reality and augmented reality.

Khiem M. Tran is a research staff at Software Engineering Laboratory at the University of Science, Vietnam National University Ho Chi Minh city. He obtained his Master's degree from John Von Neumann Institute in 2024. His research topics include artificial intelligence, multimedia content analysis, computer networking, security and embedded systems. He has participated in more than 30 studies.

Kunal Agrawal is a PhD student at the Department of Computer Science, University of Dayton (UD). Prior to that, he obtained his Master of Computer Science degree at UD in 2024. His research areas are artificial intelligence, computer vision, and machine learning.

Minh-Triet Tran obtained his B.Sc., M.Sc., and Ph.D. degrees in computer science from University of Science, VNU-HCM, in 2001, 2005, and 2009. He joined the University of Science, VNU-HCM, in 2001. His research interests include cryptography and security, computer vision and machine learning, and human-computer interaction. He was a visiting scholar at National Institutes of Informatics (NII, Japan) in 2008, 2009, and 2010, and at University of Illinois at Urbana Champaign (UIUC) in 2015–2016. He is currently the Vice President of University of Science, VNU-HCM. He is also the Director of Software Engineering Laboratory, University of Science, VNU-HCM. He is the Vice Chairperson of Vietnam Information Security Association (South Branch) and Chair of Ho Chi Minh ACM SIGCHI Chapter.

Tam V. Nguyen is an Associate Professor and the Director of the Vision and Mixed Reality Lab at the Department of Computer Science, University of Dayton (UD). He is also a Co-Director of the UD Science and Engineering Catalyst Center. He obtained his Ph.D. degree from the National University of Singapore in 2013. His research topics include artificial intelligence, computer vision, machine learning, multimedia content analysis, and mixed reality. He has authored and co-authored 160+ research papers with 4,000+ citations according to Google Scholar.