2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

DA-RAW: Domain Adaptive Object Detection for Real-World Adverse Weather Conditions

Minsik Jeon*, Junwon Seo*, Jihong Min

***Abstract*— Despite the success of deep learning-based object detection methods in recent years, it is still challenging to make the object detector reliable in adverse weather conditions such as rain and snow. For the robust performance of object detectors, unsupervised domain adaptation has been utilized to adapt the detection network trained on clear weather images to adverse weather images. While previous methods do not explicitly address weather corruption during adaptation, the domain gap between clear and adverse weather can be decomposed into two factors with distinct characteristics: a style gap and a weather gap. In this paper, we present an unsupervised domain adaptation framework for object detection that can more effectively adapt to real-world environments with adverse weather conditions by addressing these two gaps separately. Our method resolves the style gap by concentrating on style-related information of high-level features using an attention module. Using self-supervised contrastive learning, our framework then reduces the weather gap and acquires instance features that are robust to weather corruption. Extensive experiments demonstrate that our method outperforms other methods for object detection in adverse weather conditions.**

I. INTRODUCTION

Object detection plays a crucial role in enabling machines, such as autonomous vehicles and surveillance systems, to perceive and comprehend their surrounding environment. While deep learning has significantly improved object detection capabilities, ensuring the accuracy of these systems under adverse weather conditions like rain and snow remains an ongoing challenge. To ensure the detector's dependability, it is necessary to develop a learning method that can adapt object detectors to adverse weather conditions.

Due to the laborious process of obtaining labeled data for real-world adverse weather conditions, various methods have utilized synthetic datasets to improve detection performance. By generating synthetic weather effects on clear weather images without degradation, fully annotated images of adverse weather are obtained. These images are utilized to train the robust model in a supervised manner [1]–[3], or the removal network can be trained to restore a clear image from adverse weather images [4]–[7]. However, prior knowledge of weather conditions cannot effectively capture the intricate characteristics of adverse weather conditions in the real world, which have diverse and complex effects on images. Therefore, relying on synthetic datasets does not significantly enhance the model's performance when applied to real-world environments. Recent works also suggest that separately trained removal networks do not help downstream tasks [8], [9], implying the need for methods that improve the performance of downstream tasks in adverse weather conditions.

Fig. 1: We propose a novel unsupervised domain adaptation method capable of adapting an object detector from clear weather to real-world adverse weather conditions with a significant domain gap. This gap can be divided into two distinct factors: the *Style Gap* and the *Weather Gap*. The *Style Gap* stems from environmental changes such as the image's background or color, whereas the *Weather Gap* is caused by weather corruptions like rain stains, which introduce random and localized image degradation. Due to the distinct characteristics of the two gaps, we employ separate modules to address each of them independently.

Recent studies have focused on Unsupervised Domain Adaptation (UDA) to enhance the robustness of object detectors in adverse weather conditions [10]–[17]. These methods adapt the model trained in the source domain of clear weather to the target domain of adverse weather by considering adverse weather as a factor contributing to the domain gap [18], [19]. Without requiring the ground truth labels of target domain images, most UDA methods align the feature distributions of the two domains globally in an adversarial manner [10]–[13], [20].

While most UDA methods regard the domain gap between clear and adverse weather data similarly to conventional domain adaptation settings, the gap can be broken down into two distinct factors: the *style gap* and the *weather gap* [21], as shown in Fig 1. Style gaps are caused by variations in the operating environment (e.g., background, color, texture), whereas weather gaps result from weather-induced corruption (e.g., rain stains, snowflakes). Unlike style gaps, which are caused by global and semantic factors, weather corrup-

This work was supported by the Agency For Defense Development Grant funded by the Korean Government in 2023.

Minsik Jeon, Junwon Seo, and Jihong Min are with the Agency for Defense Development, Republic of Korea {mikejeon001123, junwon.vision, happymin77}@gmail.com

*These authors contributed equally to this work.

Our video can be found at https://youtu.be/vsUSrFsbuu8

tion produces arbitrary and localized image degradation that is hard to characterize using prior knowledge [9]. Existing UDA methods consider weather corruption as part of the image's style and align the source and target distributions globally. As some features are arbitrarily and severely distorted by weather corruption, these methods frequently lead to suboptimal alignment under adverse weather conditions. Consequently, they are only effective on synthetic datasets with minor domain gaps [10], whereas their performance degrades when applied to real-world datasets with a large style gap and complex weather corruption [21]. Separately addressing the two aspects of the domain gap improves domain alignment and enables robust object detection in real-world adverse weather conditions.

In this paper, we propose an unsupervised domain adaptation method to enhance the robustness of object detection in real-world adverse weather conditions. Specifically, we resolve the style and weather gaps separately to achieve optimal feature alignment. To bridge the style gap, our method aligns high-level style-related features using an attention module. Moreover, self-supervised contrastive learning is employed to resolve the weather gap. Based on the assumption that each instance consists of an object and random weather corruption, our model encourages the similarity between instance features within the same class, resulting in a robust representation against corruption. To demonstrate the efficacy of our method in a variety of real-world scenarios, we collect actual driving data in a wide range of environments and weather conditions. Through extensive experiments, we demonstrate that our method effectively adapts to various real-world datasets.

II. RELATED WORKS

A. Object Detection in Adverse Weather Condition

For the reliable perception of environments, numerous methods attempt to train robust object detectors under adverse weather [18], [22]–[24]. The most intuitive approach is to utilize annotated datasets of adverse weather conditions [9], [25], [26]. Due to the difficulty of acquiring labeled data for real-world adverse weather, synthetic weather effects are generated on labeled clear images using prior knowledge of image formulation under adverse weather condition [1]–[3]. The object detector is then trained in a supervised manner using this synthetic dataset. Other methods train removal networks to restore clear images from adverse weather images using paired data of clear and synthetic weather images with the same background [4], [27], [28], or using unpaired data [7], [29], [30]. To acquire more realistic synthetic data, some methods jointly train a synthetic data generation model and its removal network [5], [6].

In real-world environments, however, the efficacy of methods that utilize synthetic data decreases due to the complexity and diversity of real-world weather corruptions [9]. In addition, removal networks are computationally intensive to be attached to the front of the detection network, and they are trained independently to downstream tasks, which provides insufficient performance improvement for these tasks on real-world images [8], [9]. While some methods have attempted to jointly train the removal network and downstream tasks [31]–[34], they still rely on synthetic data or impose a computational burden.

B. Unsupervised Domain Adaptation for Object Detection

Unsupervised domain adaptation can be used to directly adapt a detector trained on the source domain of clear weather to the target domain of adverse weather [14]–[16], [19], [35], [36]. Most UDA methods jointly train a domain classifier and a detector so that the classifier distinguishes between the source and target features, while the detector is optimized to confuse the classifier and align the feature distributions globally [10], [37]. Alignment can be performed on image-level features from various backbones, such as ResNet [11] or Feature Pyramid Network (FPN) [20]. In addition, aligning instance-level features extracted from Region-of-Interest (RoI) can improve domain alignment [13], [38], [39]. Diverging from conventional UDA approaches, some methods employ mathematical formulations of adverse weather conditions to enhance the alignments of features [12], [17]. However, these methods consider the weather corruption as a part of an image's style and do not distinguish between the style and the weather gap [40], [41], despite their distinct characteristics. This oversight results in suboptimal adaptation performance in real adverse weather conditions with both significant style and weather gaps [21].

III. METHODS

A. Preliminaries

Given labeled images of clear weather conditions from the source domain $\mathcal{S}$ and unlabeled images of adverse weather conditions from the target domain $\mathcal{T}$, each minibatch consists of the same number of source and target data. Note that the source and target data are distinct in terms of both weather conditions and the surrounding environment.

We utilize the Faster R-CNN [42] pipeline with an FPN [43] backbone during training. The FPN backbone employs pyramid architecture to generate multi-scale feature maps (P_2, P_3, P_4, P_5) from an image, allowing the efficient detection of objects of varying scales. The Region Proposal Network (RPN) proposes RoI on these features and extracts instance features from each RoI, following the Region Classification Network (RCN) which makes final class and bounding box predictions. The supervised loss $\mathcal{L}_{\text{sup}}$ obtained from RPN and RCN is applied only to the source data [10].

The overall architecture of our method is depicted in Fig 2. We aim to train a robust object detector that performs well in real-world environments with adverse weather. Based on the architecture of the FPN-based Faster R-CNN framework, we propose two components for domain adaptation to handle both style and weather gaps. First, an image-level style alignment is used to reduce the style gap through adversarial training. An instance-level weather alignment is then utilized to reduce the weather gap and learn the corruption-invariant features. The entire model is trained simultaneously in an end-to-end manner.

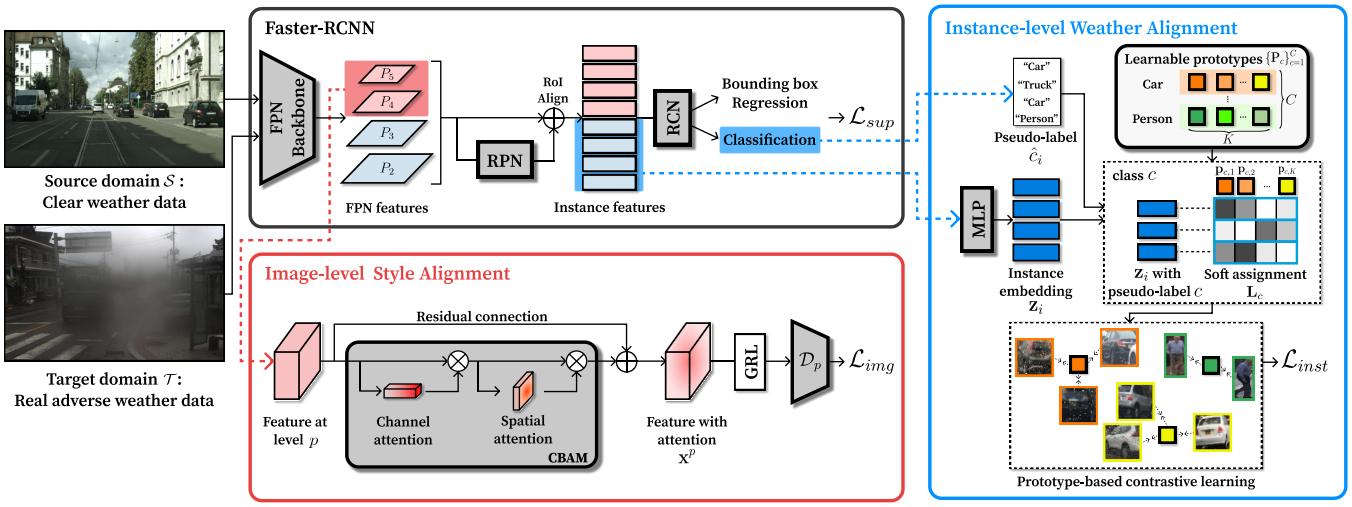

Fig. 2: Overall pipeline of the proposed method. Faster R-CNN with an FPN backbone is adopted for a detection network. *Image-level style alignment* reduces the style gap by aligning the FPN's high-level features. During alignment, they focus on style-related features by incorporating CBAM and highlighting important spatial and channel details. *Instance-level weather alignment* uses instance embedding and its corresponding pseudo-label from RCN to establish a soft assignment for each feature to learnable class prototypes. Using multi-prototype-based contrastive learning, it resolves the weather gap and constructs a weather-resistant feature representation by increasing the similarity between an instance embedding and its assigned prototypes.

B. Image-level Style Alignment

The distinct styles of the source and target domains' environments result in different feature distributions. This style gap between domains is resolved through the alignment of image-level features. Similar to [20], a domain classifier is attached to each layer of the FPN backbone to distinguish between domains. The backbone is then trained to generate the domain-invariant feature by confusing the domain classifier through adversarial training. These objectives are accomplished in a single backpropagation step by a Gradient Reversal Layer (GRL) with a weight coefficient λ.

We intend to perform image-level feature adaptation by focusing solely on the style properties of images. However, some features are severely degraded due to weather corruption, making it difficult to concentrate on the style differences. To enable the network to emphasize style-related features during alignment, the Convolutional Block Attention Module (CBAM) [44] is employed to emphasize features essential for domain alignment. CBAM is attached to each feature map and applies channel and spatial attention modules to acquire refined features $\mathbf{x}^p$ at feature level p. The attended feature is then fed into the discriminator $\mathcal{D}_p$ that predicts the domain of a feature, leading it to align features through the GRL by concentrating on essential information.

Since low-level features with fine-grained details are more susceptible to weather corruption, only high-level features are used for alignment. Therefore, image-level style alignment is performed on the P_4 and P_5 layers of the FPN backbone. The loss for image-level alignment, $\mathcal{L}_{\text{img}}$, is given by the following equation:

$$\mathcal{L}_{\text{img}} = -\sum_{p}\sum_{\mathbf{x}_i^p}\left[y_i \log \mathcal{D}_p\left(\mathbf{x}_i^p\right) + (1-y_i)\log\left(1-\mathcal{D}_p\left(\mathbf{x}_i^p\right)\right)\right], \tag{1}$$

where $p \in \{P_4, P_5\}$ represents feature level, $\mathbf{x}_i^p$ represents each feature of level p at location i after CBAM layers, and $y_i \in \{0,1\}$ indicates the domain label of each feature at location i.

C. Instance-level Weather Alignment

The weather corruption has a local effect on the image and substantially degrades the instance features that are essential for object detection. We aim to obtain corruption-invariant features and reduce the weather gap through prototype-based contrastive learning. In particular, we assume that an instance feature of the target domain image is composed of an object and an arbitrary pattern of weather corruption. Then, the similarity between instance features of object proposals within the same category is encouraged, resulting in instance features invariant to weather corruption.

From the source and target domain training images, instance features are obtained, and each of them is pseudo-labeled as class $\hat{c}_i$ using the classwise score of each instance provided by the RCN. To be utilized for contrastive learning, instance features are forwarded to an MLP head to generate instance embeddings $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^{N}$ with dimension D, where N is the total number of instances. Note that MLPs exist independently for each level of features without sharing weights, and only instance features with scores over a threshold δ from the low-level image feature are utilized to align fine-grained features.

To maximize the similarity between instance embeddings with the same pseudo-labels, prototype-anchored metric learning is used to design the contrastive loss [45]. Using learnable prototypes as representatives of each class, each instance embedding is assigned to prototypes, and a network is learned to increase their similarity. K learnable prototypes are used for each class c as $\mathbf{P}_c \in \mathbb{R}^{D\times K}$ to account for the intra-class variation of instance features, and each prototype

$\mathbf{p}_{c,k} \in \mathbb{R}^D$ serves as the k^{th} cluster center of a class c. Also, to further boost the performance of contrastive learning, instance embeddings with the background class also adopt the same number of learnable prototypes, which can be used as negative samples for other instances.

Each instance with a pseudo-label c is assigned to prototypes of the same class by computing the soft assignment matrix for class c, $\mathbf{L}_c \in \mathbb{R}_+^{K \times n_c}$. The soft assignment matrix satisfies the condition that the sum of soft assignment probabilities for each instance is one, *i.e.*, $\mathbf{L}_c^\top \cdot \mathbf{1}^K = \mathbf{1}^{n_c}$, where $\mathbf{1}^K$ and $\mathbf{1}^{n_c}$ denotes the vector of all ones with dimensions K and n_c, respectively. The assignment matrix can be obtained by maximizing the similarity between instance embeddings and the class prototypes, $\mathbf{Q}_c = \mathbf{P}_c^\top \mathbf{Z}_c \in \mathbb{R}^{K \times n_c}$, where $\mathbf{Z}_c \in \mathbb{R}^{D \times n_c}$ and n_c represent instance embeddings and the number of instances pseudo-labeled as class c, respectively.

To avoid a trivial solution in which all the instance embeddings are assigned to a single prototype, an equipartition constraint is added to ensure that instances are equally distributed among prototypes within a class. By adding the entropy regularization term [46] with a parameter κ that controls the smoothness of assignment, the objective for obtaining the assignment matrix for class c is as follows:

$$\max_{\mathbf{L}_c} \mathrm{Tr}\left(\mathbf{L}_c^\top \mathbf{Q}_c\right) + \kappa\, \mathcal{H}(\mathbf{L}_c), \quad \textit{s.t.} \quad \mathbf{L}_c \cdot \mathbf{1}^{n_c} = \frac{n_c}{K} \cdot \mathbf{1}^K, \tag{2}$$

which turns into an optimal transport problem. The solution can be computed by a few iterations of the *Sinkhorn-Knopp* algorithm [46], which outputs the re-normalization vectors $\mathbf{u} \in \mathbb{R}^K$ and $\mathbf{v} \in \mathbb{R}^{n_c}$:

$$\mathbf{L}_c = \mathrm{diag}(\mathbf{u}) \exp\left(\frac{\mathbf{Q}_c}{\kappa}\right) \mathrm{diag}(\mathbf{v}). \tag{3}$$

After obtaining the soft assignment matrix, the network is trained so that similarities between prototypes and instance embeddings correspond to the soft assignment matrix. The prototypes and instance embeddings are simultaneously optimized by minimizing the following cross-entropy loss between the similarity and assignment matrix:

$$\mathcal{L}_{inst} = -\frac{1}{N \cdot K} \sum_{i=1}^{N} \sum_{j=1}^{K} \mathbf{L}_{\hat{c}_i}^{i,j} \cdot \log \frac{\exp\left(\mathbf{z}_i \cdot \mathbf{p}_{\hat{c}_i,j}/\tau\right)}{\sum_c^C \sum_k^K \exp\left(\mathbf{z}_i \cdot \mathbf{p}_{c,k}/\tau\right)}, \tag{4}$$

where $\hat{c}_i$ is the pseudo-label for the i^{th} instance, and $\mathbf{L}_{\hat{c}_i}^{i,j}$ is a soft assignment of the instance to the j^{th} prototype of the pseudo-labeled class. Also, τ is a temperature parameter, and C denotes the number of classes, including the background class. For each instance $\mathbf{z}_i$, minimizing $\mathcal{L}_{inst}$ increases its similarity with assigned prototypes $\mathbf{p}_{\hat{c}_i,j}$, and decreases its similarity with all the others. Note that the loss is computed on both the source and target domain features using the same prototypes to reduce the domain gap.

As a result, instance embeddings are grouped around their assigned prototypes. This produces corruption-resistant instance features by promoting instance embeddings with similar semantics and variable weather corruption to become closer. The final objective of our method is as follows:

$$\mathcal{L} = \mathcal{L}_{sup} + \alpha \mathcal{L}_{img} + \beta \mathcal{L}_{inst}. \tag{5}$$

IV. EXPERIMENTS

In this section, we validate that our unsupervised domain adaptation method can effectively enhance object detection performance in real-world environments with adverse weather. Using publicly available datasets and our own datasets as evaluation, the results of our method are quantitatively and qualitatively compared to those of other methods for object detection under adverse weather conditions. In addition, ablation studies are conducted to assess the validity of each component of our methodology.

A. Datasets

Our source domain dataset is *Cityscapes* [48], which consists of real-world urban driving images captured under clear weather conditions. For the target domain dataset, multiple datasets are used to validate the efficacy of our method in various environments with adverse weather conditions. Using two synthetic weather datasets, we investigate the efficacy of other methods employing synthetic data or UDA in synthetic weather contexts. On clear images of the *Cityscapes*, the *Rain Rendering* [1] generates synthetic rain images with a physical particle simulator for each, and *RainCityscapes* [47] generates rain and fog effects subject to scene depth. To validate the efficacy of methods in real-world environments, *BDD 100K* [26], a real-world driving dataset captured in various weather conditions, is employed. The rainy and snowy subsets are used as our target domain to evaluate the efficacy of methods under diverse weather conditions.

To further validate our model across a wider range of environments, we collect *Our Dataset* in adverse weather conditions using our platform, which is equipped with an external vehicle RGB camera [49], [50]. Several datasets with adverse weather scenarios primarily focus on urban scenes and lack diversity in background environments [51], [52]. To address significant domain gaps, our datasets include data collected from rural and mountainous environments. In comparison to the *BDD 100K*, which uses a camera mounted inside the windshield of the vehicle, our dataset utilizes an external camera that is consistent with the source domain dataset. In addition, raindrops and snowfalls on the lens result in much more severe blurring of the images. Our dataset is divided into *Rainy* and *Snowy* subsets. The rainy subset consists of 2845 training images and 677 validation images, and the snowy subset consists of 1656 training images and 598 validation images. Our dataset comprises three classes: *person*, *car*, and *motorcycle*. For alignment with *Cityscapes* dataset, *Cityscapes* classes are assigned to our dataset during the experiment as follows: 1) *person*, *rider* to *person*, 2) *car*, *truck*, *bus* to *car*, and 3) *motorcycle*, *bicycle* to *motorcycle*.

B. Experimental Setup

Implementation Details. We adopt Faster R-CNN with ImageNet-pretrained ResNet-50 [53] and FPN as our object detection network. Initially, the model is trained with only the $\mathcal{L}_{\text{sup}}$ using source data in order to obtain pseudo-labels, and $\mathcal{L}_{\text{img}}$ and $\mathcal{L}_{\text{inst}}$ are applied after 7.5k iterations with $\alpha = 1.0$, $\beta = 1.0$. The image-level style alignment is performed on

TABLE I: Quantitative results on both synthetic and real-world datasets. mAP (%) is used as an evaluation metric. **Synthetic Data** column indicates whether synthetically generated adverse weather data are utilized during training, and the **Target Data** column indicates whether unlabeled data from the target domain is incorporated during training. Our method outperforms other methods when applied to datasets with real-world adverse weather conditions.

Method	Source Data	Synthetic Data	Target Data	Rainy				Snowy	
				Synthetic		Real-World		Real-World	
				RainCityscapes [47]	*Rain Rendering* [1]	*BDD 100K* [26]	*Our Dataset*	*BDD 100K* [26]	*Our Dataset*
Source Only	✓	✗	✗	35.0	31.4	31.6	49.4	27.9	57.8
Physics-based [1]	✓	✓	✗	**40.5**	41.8	22.1	35.1	18.1	42.0
MPRNet [28]	✓	✓	✗	37.7	**46.9**	12.8	38.6	12.3	41.8
SADA [20]	✓	✗	✓	38.7	40.1	29.1	48.2	27.6	53.5
SWDA [11]	✓	✗	✓	37.7	36.7	31.1	49.3	28.4	58.4
Ours	✓	✗	✓	37.7	35.6	**34.5**	**51.2**	**30.3**	**62.6**

Fig. 3: Qualitative results on real-world target datasets. Compared to other methods, *Ours* successfully detects the objects even in the presence of severe weather corruption and style variations. Even though *MPRNet* removes raindrops in the first two images, the detector performance remains low, indicating images generated by the removal network do not consistently help object detection. In the remaining images, the removal network fails to remove corruptions and instead creates some artifacts due to the disparity between real weather data and synthetic weather data, which *MPRNet* was trained on. While *SWDA* directly adapts the network to the target domain, it fails to detect objects under severe weather corruption and environmental differences. More qualitative results are available in our multimedia material.

the FPN features at levels P_4 and P_5, with λ of GRL set as 0.01. The instance-level weather alignment is performed on features at levels P_2 and P_3. We execute three iterations of the Sinkhorn-Knopp algorithm with smoothness parameter $\kappa = 0.05$, each taking about 0.1 milliseconds. The number of prototypes K for each class is set to 5, with other hyper-parameters empirically set to $\tau = 0.05$, $\delta = 0.8$, $D = 128$. During training, stochastic gradient descent (SGD) is used as an optimizer with a weight decay of $5e^{-4}$ and momentum of 0.9. Each batch consists of 16 images, eight from the source domain and eight from the target domain. We resized all the input images so that the shorter side has a length of 800 pixels and applied random horizontal flipping with a probability of 0.5. The entire model is trained with an initial learning rate of $2.5e^{-3}$ for $9.5k$ iterations and then reduced to $2.5e^{-4}$ for another $2k$ iterations.

Comparison Methods. To demonstrate the efficacy of our method under real-world adverse weather conditions, our method is compared to other detection methods designed for such conditions. For comparison with the method utilizing synthetic weather data, *Physics-based* [1] is adopted, which trains the model in a supervised manner using synthetic rainy images. Note that *Physics-based* and *Rain Rendering* dataset uses the same method to synthesize rain. The rain removal network, *MPRNet* [28], is also utilized. The removal network is trained using synthetic paired images of *Cityscapes* and *Rain Rendering*, and evaluation is conducted on restored images from the network using a detector trained only with clear source domain data. We also include results from several UDA methods that were trained from scratch. *SADA* [20] directly aligns source and target features at both the image-level and instance-level through adversarial training, whereas *SWDA* [11] focuses solely on aligning image-level features. However, none of the aforementioned methods address the style and weather gaps separately.

Evaluation Metric. The mean Average Precision (mAP) of all categories is used for evaluation with an Intersection over Union (IoU) threshold of 0.5 to compute the Average Precision (AP). Due to class imbalance issues in our dataset, class-agnostic AP is calculated for all the bounding boxes with an IoU threshold of 0.5 for a fair comparison.

C. Experimental Results

Comparisons with Other Methods. The quantitative and qualitative results are summarized in Table I and Fig 3. In both rainy and snowy conditions, our approach outperforms other methods when applied to real-world datasets. Existing UDA methods such as *SADA* and *SWDA* show a significant improvement in performance on synthetic datasets, but their

TABLE II: Results of the ablation studies. mAP (%) is used as an evaluation metric. Incorporating each module improves detection performance.

Module		Rainy		Snowy	
Style	Weather	*BDD 100K* [26]	*Our Dataset*	*BDD 100K* [26]	*Our Dataset*
✗	✗	31.6	49.4	27.9	57.8
✓ (w.o. CBAM)	✗	31.5	50.2	27.1	59.2
✓ (w. CBAM)	✗	32.4	50.7	28.7	59.6
✗	✓	33.7	51.0	29.3	61.5
✓ (w. CBAM)	✓	**34.5**	**51.2**	**30.3**	**62.6**

performance on real-world datasets is only marginally improved or even decreases. This implies that existing methods that globally align distributions are ineffective when adapting to real-world datasets with a large style gap and severe weather corruption, in contrast to synthetic datasets with a small domain gap from synthetic weather.

In addition, *SWDA* outperforms *SADA*, despite the fact that *SADA* incorporates instance-level alignment while *SWDA* focuses solely on image-level alignment. This suggests that directly aligning the instance-level features that are severely contaminated in real adverse weather conditions reduces performance, necessitating the use of alternative alignment methods. Using image-level style alignment and instance-level weather alignment, our method optimizes feature alignment by resolving both style and weather gaps, thereby improving detection performance.

Efficacy of Synthetic Weather Dataset. Methods that utilize synthetic weather images during training perform well when evaluated on synthetic data. However, their performance decreases when evaluated on real-world data, indicating that synthetic weather fails to accurately represent the complexities of real weather. The use of removal networks on real-world datasets also has a negative effect on performance, despite requiring more computation. As shown in Fig 3, the removal network trained on synthetic data has difficulty restoring a clear image from real-world images under both rainy and snowy conditions, showing its inability to remove complex real rain and generalize to other weather conditions. In addition, detection performance decreases even in visually restored areas. This suggests that the features obtained through the removal network do not contribute to improving the detection performance. By directly adapting to downstream tasks, our method achieves superior performance on real-world datasets without relying on synthetic priors.

Ablation Studies on Each Component. To validate the efficacy of each component, we conducted an ablation analysis on real-world datasets. The results are shown in Table II. Image-level style alignment increases performance, indicating that high-level feature alignment bridges the domain gap between image-level features effectively. Particularly, there is a significant performance increase when CBAM is present, implying that CBAM improves alignment by focusing on essential features. Incorporating instance-level weather alignment further enhances performance. This suggests that instance features obtained by self-supervised contrastive learning are robust to weather corruption and domain-invariant. Overall, combining both components achieves the highest performance, which effectively addresses both gaps.

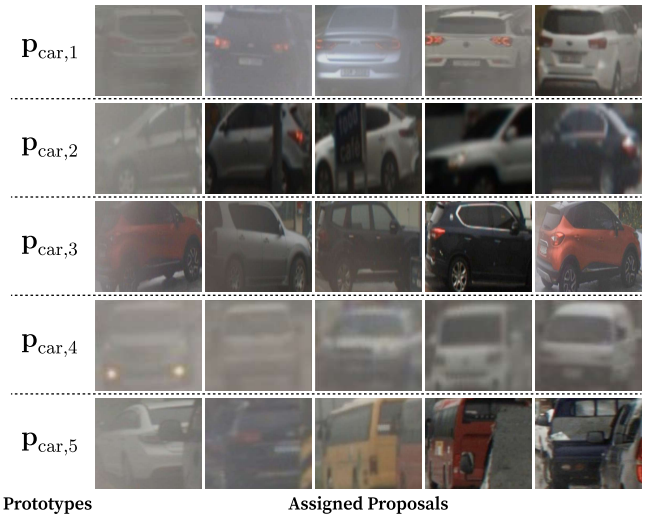

Fig. 4: Visualization of proposals assigned to each prototype in our rainy dataset. Each row displays the proposals whose instance embedding is highly similar to each car class prototype. Similar-shaped objects with diverse corruption and styles are assigned to identical prototypes, indicating the effectiveness of prototype-based contrastive learning. For example, the first row contains car proposals captured from a rear-view perspective and showing varying degrees of corruption.

Efficacy of Instance-level Weather Alignment. To evaluate the impact of the instance-level weather alignment on feature embeddings, we visualize proposals whose instance embeddings are highly similar to each of the car class prototypes. As shown in Fig 4, proposals with similar object shapes but varying degrees of corruption are assigned to the same prototype. This demonstrates that our weather align module contributed to extracting semantically meaningful features that are resilient to corruption. Moreover, the fact that objects with similar shapes are gathered together in the same prototype demonstrates the efficacy of employing multiple prototypes to address intra-class variation.

V. CONCLUSION

This paper presents a novel framework for domain adaptive object detection that improves robustness under real-world adverse weather conditions. The proposed method effectively addresses two distinct aspects of the domain gap, the style gap and the weather gap, by using image-level style alignment and instance-level weather alignment, respectively. Diverging from previous approaches that were mostly evaluated in synthetic datasets, our method shows robust performance on real-world datasets, which have been validated through extensive experiments.

We believe that our method can expand the range of applications for machines as it can detect objects in a variety of real adverse weather conditions. To make our method more applicable to real-world applications, we are investigating techniques that can adapt to dynamic adverse weather conditions during inference time. Furthermore, we also aim to expand our framework with minimal requirements for target domain data.

REFERENCES

[1] S. S. Halder, J.-F. Lalonde, and R. d. Charette, "Physics-based rendering for improving robustness to rain," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 10 203–10 212.

[2] M. Tremblay, S. S. Halder, R. De Charette, and J.-F. Lalonde, "Rain rendering for evaluating and improving robustness to bad weather," *International Journal of Computer Vision*, vol. 129, pp. 341–360, 2021.

[3] G. Volk, S. Müller, A. Von Bernuth, D. Hospach, and O. Bringmann, "Towards robust cnn-based object detection through augmentation with synthetic rain variations," in *IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 285–292.

[4] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2482–2491.

[5] H. Wang, Z. Yue, Q. Xie, Q. Zhao, Y. Zheng, and D. Meng, "From rain generation to rain removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 14 791–14 801.

[6] Y. Ye, Y. Chang, H. Zhou, and L. Yan, "Closing the loop: Joint rain generation and removal via disentangled image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 2053–2062.

[7] Y. Ye, C. Yu, Y. Chang, L. Zhu, X.-l. Zhao, L. Yan, and Y. Tian, "Unsupervised deraining: Where contrastive learning meets self-similarity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 5821–5830.

[8] Y. Pei, Y. Huang, Q. Zou, Y. Lu, and S. Wang, "Does haze removal help cnn-based image classification?" in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 682–697.

[9] S. Li, I. B. Araujo, W. Ren, Z. Wang, E. K. Tokuda, R. H. Junior, R. Cesar-Junior, J. Zhang, X. Guo, and X. Cao, "Single image deraining: A comprehensive benchmark analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3838–3847.

[10] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Domain adaptive faster r-cnn for object detection in the wild," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 3339–3348.

[11] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Strong-weak distribution alignment for adaptive object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 6956–6965.

[12] V. A. Sindagi, P. Oza, R. Yasarla, and V. M. Patel, "Prior-based domain adaptive object detection for hazy and rainy conditions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020, pp. 763–780.

[13] M. Xu, H. Wang, B. Ni, Q. Tian, and W. Zhang, "Cross-domain detection via graph-induced prototype alignment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 12 355–12 364.

[14] J. Deng, W. Li, Y. Chen, and L. Duan, "Unbiased mean teacher for cross-domain object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 4091–4101.

[15] Y.-J. Li, X. Dai, C.-Y. Ma, Y.-C. Liu, K. Chen, B. Wu, Z. He, K. Kitani, and P. Vajda, "Cross-domain adaptive teacher for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 7581–7590.

[16] S. Cao, D. Joshi, L.-Y. Gui, and Y.-X. Wang, "Contrastive mean teacher for domain adaptive object detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 23 839–23 848.

[17] J. Li, R. Xu, J. Ma, Q. Zou, J. Ma, and H. Yu, "Domain adaptive object detection for autonomous driving under foggy weather," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 612–622.

[18] M. Hnewa and H. Radha, "Object detection under rainy conditions for autonomous vehicles: A review of state-of-the-art and emerging techniques," *IEEE Signal Processing Magazine*, vol. 38, no. 1, pp. 53–67, 2020.

[19] P. Oza, V. A. Sindagi, V. V. Sharmini, and V. M. Patel, "Unsupervised domain adaptation of object detectors: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[20] Y. Chen, H. Wang, W. Li, C. Sakaridis, D. Dai, and L. Van Gool, "Scale-aware domain adaptive faster r-cnn," *International Journal of Computer Vision*, vol. 129, no. 7, pp. 2223–2243, 2021.

[21] X. Ma, Z. Wang, Y. Zhan, Y. Zheng, Z. Wang, D. Dai, and C.-W. Lin, "Both style and fog matter: Cumulative domain adaptation for semantic foggy scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 18 922–18 931.

[22] M. J. Mirza, C. Buerkle, J. Jarquin, M. Opitz, F. Oboril, K.-U. Scholl, and H. Bischof, "Robustness of object detectors in degrading weather conditions," in *IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021, pp. 2719–2724.

[23] W. Wu, H. Chang, Y. Zheng, Z. Li, Z. Chen, and Z. Zhang, "Contrastive learning-based robust object detection under smoky conditions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4295–4302.

[24] T. Rothmeier, D. Wachtel, T. von Dem Bussche-Hünnefeld, and W. Huber, "I had a bad day: Challenges of object detection in bad visibility conditions," in *IEEE Intelligent Vehicles Symposium (IV)*, 2023, pp. 1–6.

[25] M. Hassaballah, M. A. Kenk, K. Muhammad, and S. Minaee, "Vehicle detection and tracking in adverse weather using a deep learning framework," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4230–4242, 2020.

[26] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 2636–2645.

[27] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 270–12 279.

[28] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, M.-H. Yang, and L. Shao, "Multi-stage progressive image restoration," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 14 821–14 831.

[29] Y. Wei, Z. Zhang, Y. Wang, M. Xu, Y. Yang, S. Yan, and M. Wang, "Deraincyclegan: Rain attentive cyclegan for single image deraining and rainmaking," *IEEE Transactions on Image Processing*, vol. 30, pp. 4788–4801, 2021.

[30] X. Chen, J. Pan, K. Jiang, Y. Li, Y. Huang, C. Kong, L. Dai, and Z. Fan, "Unpaired deep image deraining using dual contrastive learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 2017–2026.

[31] Y. Lee, Y. Ko, Y. Kim, and M. Jeon, "Perception-friendly video enhancement for autonomous driving under adverse weather conditions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022, pp. 7760–7767.

[32] K. Wang, T. Wang, J. Qu, H. Jiang, Q. Li, and L. Chang, "An end-to-end cascaded image deraining and object detection neural network," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9541–9548, 2022.

[33] W. Liu, G. Ren, R. Yu, S. Guo, J. Zhu, and L. Zhang, "Image-adaptive yolo for object detection in adverse weather conditions," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, vol. 36, no. 2, 2022, pp. 1792–1800.

[34] S. Kalwar, D. Patel, A. Aanegola, K. R. Konda, S. Garg, and K. M. Krishna, "Gdip: Gated differentiable image processing for object detection in adverse conditions," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7083–7089.

[35] G. Eskandar, R. A. Marsden, P. Pandiyan, M. Döbler, K. Guirguis, and B. Yang, "An unsupervised domain adaptive approach for multimodal 2d object detection in adverse weather conditions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 10 865–10 872.

[36] H. Zhang, L. Xiao, X. Cao, and H. Foroosh, "Multiple adverse weather conditions adaptation for object detection via causal intervention," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[37] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of

neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[38] F. Rezaeianaran, R. Shetty, R. Aljundi, D. O. Reino, S. Zhang, and B. Schiele, "Seeking similarities over differences: Similarity-based domain alignment for adaptive object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 9204–9213.

[39] V. Vs, V. Gupta, P. Oza, V. A. Sindagi, and V. M. Patel, "Mega-cda: Memory guided attention for category-aware unsupervised domain adaptive object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 4516–4526.

[40] S. Choi, S. Jung, H. Yun, J. T. Kim, S. Kim, and J. Choo, "Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11 580–11 590.

[41] S. Lee, T. Son, and S. Kwak, "Fifo: Learning fog-invariant features for foggy scene segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 18 911–18 921.

[42] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 28, 2015.

[43] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2117–2125.

[44] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[45] T. Zhou, W. Wang, E. Konukoglu, and L. Van Gool, "Rethinking semantic segmentation: A prototype view," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 2582–2593.

[46] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 26, 2013.

[47] X. Hu, C.-W. Fu, L. Zhu, and P.-A. Heng, "Depth-attentional features for single-image rain removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8022–8031.

[48] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3213–3223.

[49] S. Sim, J. Min, S. Ahn, J. Lee, J. S. Lee, G. Bae, B. Kim, J. Seo, and T. S. Choe, "Build a multi-sensor dataset for autonomous driving in adverse weather conditions," *The Journal of Korea Robotics Society*, vol. 17, no. 3, pp. 245–254, 2022.

[50] J. Seo, S. Sim, and I. Shim, "Learning off-road terrain traversability with self-supervisions only," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4617–4624, 2023.

[51] M. A. Kenk and M. Hassaballah, "Dawn: vehicle detection in adverse weather nature dataset," *arXiv preprint arXiv:2008.05402*, 2020.

[52] M. Pitropov, D. E. Garcia, J. Rebello, M. Smart, C. Wang, K. Czarnecki, and S. Waslander, "Canadian adverse driving conditions dataset," *The International Journal of Robotics Research*, vol. 40, no. 4-5, pp. 681–690, 2021.

[53] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.