Multi-weather city: Adverse weather stacking for autonomous driving

Valentina Muşat* Ivan Fursa† Paul Newman* Fabio Cuzzolin‡ Andrew Bradley†

valentina@robots.ox.ac.uk,17076662@brookes.ac.uk,pnewman@robots.ox.ac.uk,fabio.cuzzolin@brookes.ac.uk,abradley@brookes.ac.uk

Abstract

Autonomous vehicles make use of sensors to perceive the world around them, with heavy reliance on vision-based sensors such as RGB cameras. Unfortunately, since these sensors are affected by adverse weather, perception pipelines require extensive training on visual data under harsh conditions in order to improve the robustness of downstream tasks - data that is difficult and expensive to acquire. Based on GAN and CycleGAN architectures, we propose an overall (modular) architecture for constructing datasets, which allows one to add, swap out and combine components in order to generate images with diverse weather conditions. Starting from a single dataset with ground-truth, we generate 7 versions of the same data in diverse weather, and propose an extension to augment the generated conditions, thus resulting in a total of 14 adverse weather conditions, requiring a single ground truth. We test the quality of the generated conditions both in terms of perceptual quality and suitability for training downstream tasks, using real world, out-of-distribution adverse weather extracted from various datasets. We show improvements in both object detection and instance segmentation across all conditions, in many cases exceeding 10 percentage points increase in AP, and provide the materials and instructions needed to re-construct the multi-weather dataset, based upon the original Cityscapes dataset.

1. Introduction

Autonomous vehicles rely on a set of sensory information in order to correctly perceive the environment and ensure a safe journey. Unfortunately, adverse weather and lighting conditions can affect how the environment is perceived, thus impacting the performance of downstream tasks and, ultimately, the safety of the traffic participants. Cameras, which are one of the most cost-effective modalities in autonomous vehicles, are also among the most affected by adverse weather and illumination conditions [52], with matters made worse by the overlap with some of the causes of LIDAR performance degradation [17].

*Oxford Robotics Institute, University of Oxford
†Autonomous Driving Group, Oxford Brookes University
‡Visual Artificial Intelligence Laboratory, Oxford Brookes University

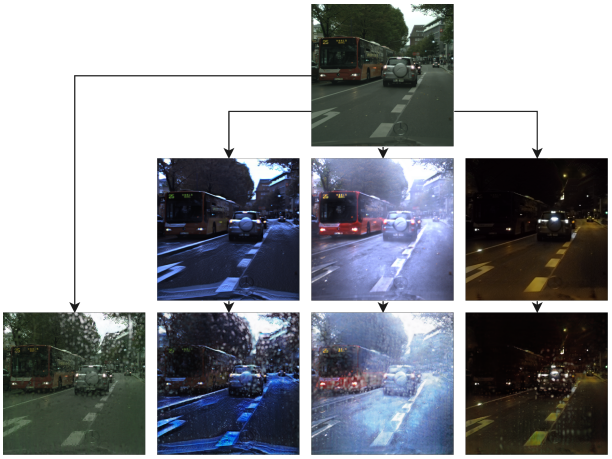


Figure 1: Concept of weather stacking: generated weather appearance, starting from real overcast.

Due to the increase in popularity of the autonomous driving industry, a lot of effort has been devoted to tackling these issues. While hardware solutions are being developed using the latest technology in order to ensure more robustness to adverse weather at the data acquisition stage [5], a large body of research focuses on improving the robustness of downstream tasks via domain adaptation, de-noising, de-weathering and sensor fusion, amongst others.

Unfortunately, both the aforementioned methods and the relevant AV-related tasks (such as semantic segmentation, object detection and depth estimation) require large training datasets, both in general and in each of the specific weather conditions the vehicle might encounter. Data availability, however, has become a serious bottleneck due to the cost, time and difficulty of obtaining it. To overcome this issue, significant work has recently been directed at the synthetic generation of weather conditions [13, 38] and the photo-realistic style-transfer of weather appearance [34, 33, 32, 2, 42]. For the purpose of testing downstream tasks in the wild, datasets have been designed to include scenes with diverse weather and the corresponding ground truth [45, 21]. Others have attempted to provide both clear weather and weather condition pairs for both static [10] and dynamic scenes [33, 35].

While these approaches are commonly benchmarked in isolation, rather than in combination, here we aim to show that combining these techniques can yield much more visu-

ally diverse outputs in a controlled and stackable way. In this work, in particular, we generate augmented imagery under 7 distinct various weather and illumination settings starting from a *single* dataset with ground truth (Cityscapes [8]), and test if the generated data is a good proxy for real weather. We do this by using the generated data as training data in the context of autonomous driving-related downstream tasks. As an extension, we propose as future work 7 other conditions based on the work of [38], and further present visual results.

The contributions of the paper are as follows:

1. We generate a number of weather conditions using a unified generator architecture for image translation, for both paired and unpaired settings, based on the work of [30] and [54], which results in imagery that not only is of increased realism and has fewer
2. We use the above-generated weather appearance as input to an additional network designed to add adherent droplets, thus resulting in a combination of more diverse weather appearances, again starting from only a single dataset with paired ground truth.
3. For a more extensive evaluation, we use multiple publicly available datasets comprising real adverse weather for validating the suitability of the data for instance segmentation and object detection, while also evaluating the quality of the images using the Inception Score and Fréchet Inception Distance.
4. We release the relevant materials and steps needed to recreate and use the multi-weather Cityscapes dataset, which can be found at https://github.com/vnmusat/multi-weather-city. Due to licensing restrictions, the dataset itself is distributed as a set of additive transformations that can be applied to the original Cityscapes dataset [8].

We would like to stress the facts that the purpose of this study is not to present an entirely novel image-to-image translation architecture, but to demonstrate a methodology for creating diverse data by starting from a single dataset with paired ground truth, using cascaded image translation models.

2. Related work

Adverse weather can affect the performance of computer vision tasks in multiple ways: temperature and temperature variations affect the optical, electronic and mechanical components used in capturing visual data, while ambient conditions affect light propagation and the appearance of the environment [7]. For example, cold temperatures or foggy conditions can result in condensation on the lens, blurring the view; raindrops on the windshield can act as a double lens or generate glares; static snow on roads may cover the lane markings, affecting detection of driveable areas, while wet road surfaces might result in reflections and artefacts due to water puddles, and deteriorated contrast between road features. As the success of autonomous vehicles depends on the ability to overcome the effects of these conditions, some studies have developed hardware solutions to tackle these problems. For example, [5] studies the performance of gated cameras, while [4] extends the study to combine stereo, gated and thermal cameras with Radar and LiDAR scanners, showing significant improvements for car detection at various levels of fog, rain and snow. Other studies use domain adaptation to 'change' the weather conditions as a post data-acquisition process. For example, [29] explores the effect of generated night-time and generated day-time rain images on road segmentation and traffic object detection, whereas [34] shows an improvement in localisation by using generated night-time imagery and [33] develops a de-raining model to improve semantic segmentation.

2.1. Real adverse weather capture

Among the first to provide a dataset with clear and weather-affected image pairs were the authors of [10], who used a transparent pane to add dirt and droplets to real-world scenery. Unfortunately, the dataset focuses only on static scenes. In the same category fall the works of [46], which uses 4 cameras attached to a vehicle to capture pairs of clear and images affected by soil; [33], which uses a stereo camera behind a bi-partite chamber with one clear lens and one lens affected by adherent droplets; and finally, [35] which uses a similar setup to [10], but captures outdoor images in an indoor environment.

A related but different category is represented by efforts to collect and annotate data in a series of target conditions such as: night-time, rainy night, heavy snow and other variations, such as [39, 28, 45, 31, 21, 44]. Whist these provide some of the most extensive datasets so far, the data is limited to specific road conditions in specific areas of the world, and the data collection process is heavily influenced by weather forecast. To facilitate the development of a truly weather-proof system using real data would require the collection of training imagery in all conditions, in all usage areas and at all times - which is a time-consuming and expensive undertaking. To overcome this difficulty, efforts have been made to provide a cost-effective and more scalable alternative, such as augmented visual data that is based upon physics models, synthesis or appearance style transfer.

2.2. Synthetic adverse weather generation

Physics-based approaches are often employed in generating synthetic weather, especially for fog and droplets. For example, [36] proposes a pipeline that uses a stereo pair and depth information to add synthetic fog on clear images,

while [13] creates a purely synthetic fog dataset based on Synscapes [49] (synthetic fog to synthetic images). Similarly, [14] uses a physics simulator to add rain streaks and fog on clear images and further tests object detection and semantic segmentation on real rainy imagery, while [33] uses a physics-based pipeline to add synthetic adherent raindrops to clear images, and further tests a lane-marking segmentation model.

Furthermore, [1] uses a computer graphics engine to render photo-realistic in-focus and defocused raindrops, and [22] develops a model for restoring images affected by heavy rain. Neither, however, tests the viability of restored images as training data, focusing instead on reconstruction metrics. On the other hand, [23] develops a decomposition network to split rain-affected images into a clean image and a rain layer and further trains the model on synthetic generated rain, but only tests it on 20 real-world images.

2.3. Weather appearance transfer

Due to the recent developments in GAN [12] and CycleGAN architectures [54], an increasing body of research has been devoted to applying these models for autonomous driving tasks. In the case of unpaired data, the first to use appearance transfer were the authors of [34], who trained a model to generate images with snow and diverse illumination in order to optimise feature matching for localisation. Later research includes that of [2], which generates day-time images from night-time images in order to improve retrieval-based localisation and [53], which learns to de-haze synthetic hazy images. The authors of [24] generate night-time images from day-time images, whereas [47] generates the soiled counterpart of a clear image and [11] adds synthetic fog to clear images. Similarly, [29] generates night-time images from day-time images and shows qualitative results on a day-time image with adherent droplets, while [9] is among the few papers that test semantic segmentation under rainy night conditions, with the drawback that their model requires paired data. While the aforementioned works provide multiple weather appearance pairs, they do not combine or stack conditions, and provision of a dataset is outside the scope of their work.

Other approaches involve direct image synthesis using paired data, with prominent examples being [20] and [48] which synthesize images from semantic or instance maps. These models, however, assume that a semantic segmentation ground truth exist in order to ensure higher-quality image generation. Later extensions that aim to improve realism include AdaIN [19] and SPADE [30], which propose improved normalization techniques to encourage the intermediate convolutional layers to make a better use of the input data. We describe how our work derives from existing methods in the following section.

3. Methodology

Our overall weather stacking methodology (Fig. 2) consists of two stages. In the first stage, a set of N models receives as input real overcast imagery, and outputs the same data but in N different weather styles. In the second stage, a single model receives the generated weather images and adds adherent raindrops.

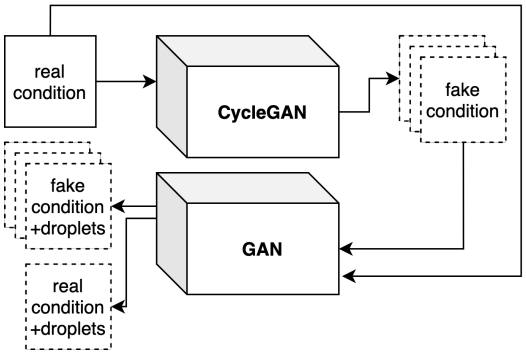


Figure 2: Overall methodology for weather stacking. First, an image translation CycleGAN model (trained using unpaired data) is used to create N weather and illumination conditions from a reference real condition. Then, a second image translation GAN model (trained using paired data) is used to apply adherent droplets to the N conditions. The current setup is *one* example of such a model stack, with both models being freely electable.

3.1. Datasets

We chose Cityscapes [8] as our source dataset, on which we transfer weather appearance (snowy, rainy/wet, night-time), using Oxford RobotCar [26] as a source of style for rainy/wet and snowy and the train set of Dark Zurich [39] for night-time appearances.

Cityscapes was chosen as it is a widely used dataset for training downstream tasks, with high-quality instance annotations and additional sources of ground-truth such as disparity. Additionally, many of the methods adapted in this work have been either trained or tested on Cityscapes or its derivatives. The choice for a source dataset is however open and free and should be consistent with the target applications. Since RainyScreens [35] contains imagery captured through a transparent pane with added droplets, it makes a good source of paired data for training a droplet generation model. Finally, to evaluate the night and night+droplets generated data, we extract diverse real images with adverse weather from Mapillary [27], BDD100K [51], DAWN [21] and ACDC [37] to cover the conditions of interest.

3.2. Models

Generative Adversarial Networks [12] are a class of generative models where a generator and a discriminator compete against each other: the generator G learns to generate data from a particular distribution $p_{data}(x)$, whereas the discriminator D learns to detect which data comes from the same distribution. The learning setting can thus be formulated as a minimax game, where each of the models tries to

minimise its own losses:

$$\min_G \max_D L(D,G) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{y\sim p_y(y)}[\log(1-D(G(y)))]. \quad (1)$$

The generator seeks to minimise its own loss by generating images with high fidelity. Thus, its loss will be minimal when the discriminator is fooled, i.e., $D(G(y)) = 1$. On the other hand, the loss of the discriminator will be minimal when it is able to correctly identify real images ($D(x) = 1$) from generated images ($D(G(y)) = 0$).

Cycle-consistency GANs are an extension of GAN models, developed in order to allow image translation between unpaired datasets. Training a CycleGAN involves optimizing simultaneously two generators and two discriminators, where one generator learns the mapping function from a domain A to a domain B, while the other learns the mapping from domain B to domain A. Since the supervision of the two discriminators is not enough to ensure transfer, an additional reconstruction loss is used in order to enforce cycle consistency, by forcing the two generators to reconstruct each other's output back into the original domain.

We use N CycleGANs ($N = 3$ in our case) to train image translation from $(real, overcast, daytime) \rightarrow (fake, night)$, $(real, overcast, daytime) \rightarrow (fake, wet)$ and $(real, overcast, daytime) \rightarrow (fake, snow)$, using the official architecture [54] but with a SPADE-based generator, as it was shown to generate images with higher fidelity due to improved normalization layers [30].

For the paired image translation task (in this case, applying adherent droplets to the generated conditions) we use a pix2pix-like architecture [20, 48], again with a SPADE-based generator [30]. As we have pairs of clear and droplet-affected images of the Cityscapes dataset (from the Rainyscreen dataset [35]), we employ one single GAN to learn the $(real, overcast, daytime) \rightarrow (fake\text{-}droplet, overcast, daytime)$ mapping, and at inference time we run it on the N conditions generated as previously explained.

3.3. Evaluation

3.3.1 Perceptual quality evaluation

Following image quality assessment methods as used in [30] and [41], we evaluate the perceptual quality of the generated styles using Fréchet Inception Distance (FID) [18], but also Inception Score (IS) [40], both based on the Inception-v3 network [43], and shown to be in line with human judgements [6]. Whereas IS is computed by taking into account the predicted class probabilities of generated images via [43], the FID score analyses the last pooling layer (prior to classification) and models the activations of real and generated images as two multi-variate Gaussian distributions, calculating the distance between the two distributions using the Fréchet distance. An image with high diversity and high quality would have a high IS, whereas an image with a low FID would correlate with high quality.

Dataset	O	D	W	WD	S	SD	N	ND
BDD100K	100	37	70	68	63	12	46	14
Mapillary	65	-	99	9	385	-	20	-
DAWN	-	-	-	200	-	204	-	-
ACDC	-	-	400	-	400	-	400	-

Table 1: Number of images used in out-of-sample testing of Mask-RCNN

3.3.2 Quantitative evaluation

The suitability of the generated images as training data for relevant downstream tasks is extensively tested on various real weather conditions, in terms of (i) object detection, (ii) semantic segmentation and (iii) instance segmentation performance. Due to the large number of condition-and-dataset combinations, we chose to use and fine-tune Mask-RCNN [15], as it performs all tasks from the same backbone, while training relatively efficiently. We would like to stress that our goal is not to produce state-of-the-art results, but instead to assess the suitability of our generated training data while keeping all other variables constant. Any other recent or state-of-the-art model could be a substitute for MaskRCNN, yielding potentially better results overall. Table 1 contains a summary of the datasets, weather conditions and number of images extracted and used for testing.

In order to evaluate the suitability of each generated condition, we start with a Mask-RCNN model pre-trained on Cityscapes real overcast images, which is then further fine-tuned for each generated condition (7 different instances of the same initial pre-trained model). After fine-tuning each model, we test it out-of-sample and out-of-distribution on the real conditions extracted previously, and note the changes in results. Finally, to test the performance in the case of a monolithic model instead of individual models, we fine-tune one model on all the generated conditions at once, and test again out-of-sample on all real conditions.

The image-translation models were trained on images that have been resized to 512×1024 and randomly cropped to 512×512. In this way we enforce a uniform standard across all analyses and ensure that the ground truth is processed to reflect the changes.

We use the Detectron2 [50] implementation of Mask-RCNN with a ResNet+FPN backbone [16, 25]), which outputs both predicted masks and bounding boxes. We start with the official pre-trained model on ImageNet, COCO and Cityscapes for instance segmentation and bounding box detection.

We report our results in terms of mean Average Precision (AP), AP@50 and AP@75[1], for Object detection, Semantic segmentation and Instance segmentation, depending on the ground truth availability of the test dataset.

[1] AP at IoU=.50/IoU=.75, where only candidates with an area at least 50%/70% compared to GT area are considered.

4. Results

4.1. Qualitative results

Using the IS and FID metrics described in section 3.3.1, the results are reported in Table 2. The Inception Score is provided as a means for performing a rough comparison with other approaches, but needs to be used carefully when comparing models, as outlined in [3]. On the other hand the FID score may be used to check the degree of alignment (how similar the conditions or their distributions might be) between datasets or conditions, and in our case has a surprising amount of correlation with the Quantitative results reported in Table 3, under "Improvement on individual fine-tuned models". Cityscapes' synthetic conditions that have a comparatively lower (better) FID score with respect to BDD (such as *wet*, *snow* and *night*) also perform much better on their corresponding BDD conditions, with the ranking predicted by the FID score being a good indicator of object detection and instance segmentation performance across various conditions.

Dataset	IS	FID
Real overcast (**O**)	3.75	69.74
Fake droplets on real overcast (**D**)	4.04	124.66
Fake wet (**W**)	4.13	87.10
Fake droplets on fake wet (**WD**)	3.21	182.24
Fake snow (**S**)	4.12	116.40
Fake droplets on fake snow (**SD**)	3.72	227.00
Fake night (**N**)	3.27	86.56
Fake droplets on fake night (**ND**)	3.45	152.48

Table 2: Qualitative results for Inception Score and Fréchet Inception Distance. FID is computed wrt. the selected BDD100K train set.

4.2. Quantitative results

We split our quantitative analysis into four parts: testing against the BDD, Mapillary, DAWN and ACDC datasets, respectively. We would like to point out that all our testing, except for the initial baseline, is done on out-of-distribution data in order to strengthen the validity of the results and to act as a better proxy for real world performance.

4.2.1 BDD

Table 3 shows our results on various conditions extracted from the BDD dataset. We begin by benchmarking the performance of the model fine-tuned on half-resolution, center-cropped Cityscapes overcast images against the Cityscapes validation set, in order to establish a baseline (1st row).

First, we note that the performance is slightly lower that the official baseline in [50] due to the use of half-resolution images. We then assess the loss of performance due to domain shift by testing the same model on BDD real *overcast* imagery. We note that the drop in performance is less pronounced for Object detection as compared to Instance segmentation, but still significant. After establishing these two **overcast baselines**, we then assess the performance of the model (fine-tuned on *overcast* images) on the 7 representative conditions extracted from BDD, establishing our **condition baselines**. We notice particularly low performance for *real droplets on real night* and unusually high performance for *real droplets on real snow*. This is potentially due to the low number of samples used for these two conditions (see Table 1), and should be assessed with care.

We then test models trained on the 7 individual synthetic Cityscapes conditions on their respective BDD conditions. On first analysis, the mixed results (potentially discounting the two aforementioned conditions with low number of samples) might be surprising, with *fake night* and *fake wet* showing large increases, *fake snow* and *fake droplets on fake wet* remaining largely the same, and *fake droplets on real overcast* showing much lower performance. However, the FID scores in Table 2 may contain an indication for this behavior: we notice that *fake night* and *fake wet* have relatively good (low) FID scores, appearing to be aligned with their respective BDD conditions, while the other 5 conditions seem much more unaligned with their respective BDD conditions.

We analyze this claim in the next block of rows, where we show results for testing a model trained on all the Cityscapes conditions against the individual BDD conditions. Because the model now has to learn to generalise across a much wider set of distributions of conditions instead of only one potentially misaligned distribution, we would expect to see significant gains against both the results on individual models and against the baselines. And indeed, we observe gains across all conditions, with large improvements (discounting the two conditions with reduced samples) for *fake droplets on real overcast*, *fake night*, *fake snow*, *fake wet* and *fake droplets on fake wet*. Additionally, we test this model on the Cityscapes overcast validation set and show that it outperforms the original baseline, by up to 3.3 percentage points.

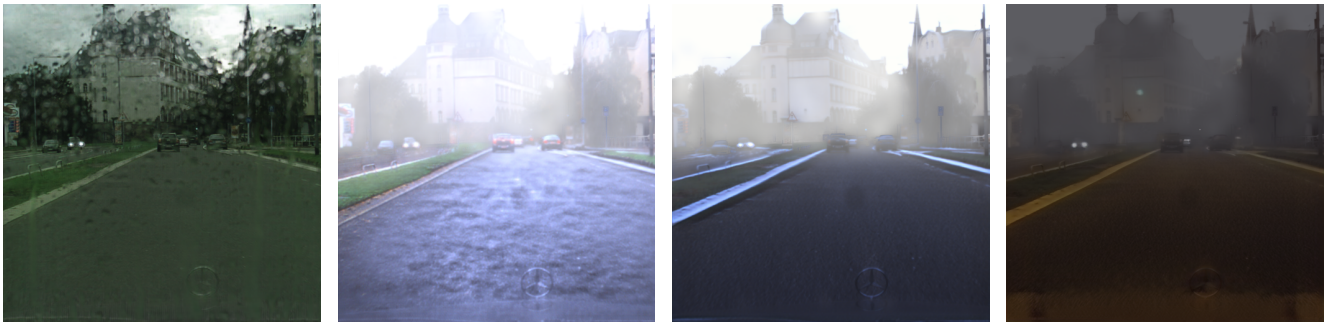
4.2.2 Mapillary, DAWN and ACDC

To make up for the reduced number of samples for certain conditions in BDD, we also test on conditions extracted from the Mappillary dataset, with results presented in Table 4. We follow the same procedure as for BDD, establishing baselines, observing mixed results for individual models, and finally obtaining significant increases for all conditions when using the model trained on all synthetic Cityscapes conditions (for example an almost 11 percentage points increase in AP@50 when testing on snow). The DAWN [21] dataset contains examples of harsh weather conditions, and specifically covers *real droplets with snow* (which was underrepresented in BDD) and *real droplets with wet*. Our results are reported in Table 5. Again, we begin by establishing baselines for the model trained on overcast data. We then obtain mixed results for the individually trained models, with *fake droplets on fake wet* improving considerably. Finally, we show significant improvements on both conditions when using the model trained on all synthetic

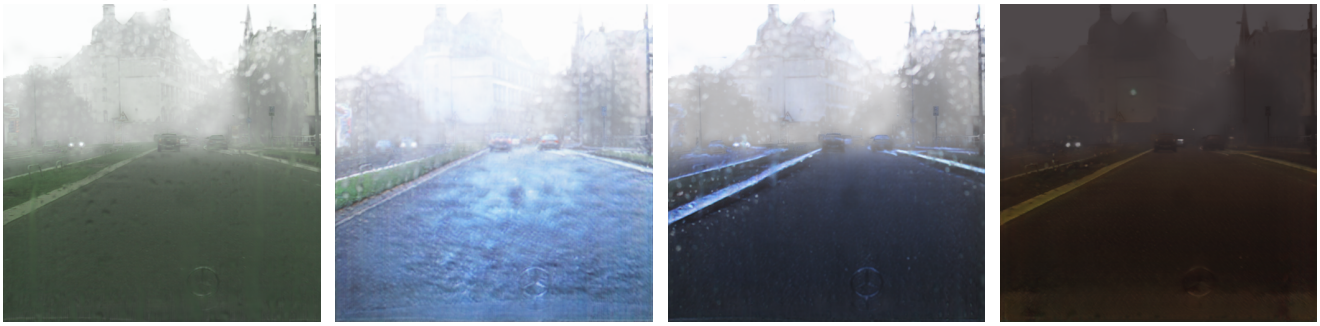
(a) real overcast **(b)** fake snow **(c)** fake rain/wet **(d)** fake night

(e) fake droplets (real overcast) **(f)** fake droplets (fake snow) **(g)** fake droplets (fake rain/wet) **(h)** fake droplets (fake night)

Figure 3: Generated conditions



(a) drop(real) **(b)** fog(wet(real)) **(c)** fog(snow(real)) **(d)** fog(night(real))



(e) fog(drop(real)) **(f)** fog(drop(wet(real))) **(g)** fog(drop(snow(real))) **(h)** fog(drop(night(real)))

Figure 4: Extension: fog applied on generated conditions, with a fog coefficient of 0.01

Cityscapes conditions, with *fake droplets on fake wet* gaining more than 10 percentage points over the real *overcast* baseline, and *fake droplets on fake snow* more than 6 percentage points.

The ACDC dataset [37] contains examples of night, snow and wet conditions with semantic segmentation ground truth. We report results in Table 6. Similarly to previous experiments, we observe mixed results for individual models and an increase in performance across the board for the monolithic model, reinforcing the trend observed in previous experiments.

Description	Fine-tune set Cityscapes	Test condition	Object detection AP	Object detection AP@50	Object detection AP@75	Instance segmentation AP	Instance segmentation AP@50	Instance segmentation AP@75
Domain shift to BDD	Real **O** train	City real **O** val	31.83	52.73	30.77	27.14	48.15	24.95
	Real **O** train	BDD **O**	29.22	51.92	25.28	20.99	37.52	19.32
Weather baselines	Real **O** train	BDD **D**	26.00	47.82	16.06	19.57	43.12	7.49
	Real **O** train	BDD **N**	20.60	29.90	23.67	15.11	27.69	21.50
	Real **O** train	BDD **ND**	7.75	19.12	3.61	4.02	14.74	1.23
	Real **O** train	BDD **S**	24.95	40.40	27.08	20.50	34.37	19.48
	Real **O** train	BDD **SD**	39.38	57.60	51.67	37.40	52.02	45.98
	Real **O** train	BDD **W**	22.09	39.99	20.95	16.58	34.54	15.55
	Real **O** train	BDD **WD**	17.57	37.65	17.04	13.76	32.10	10.99
Results on individual fine-tuned models	Synth **D** train	BDD **D**	21.03	35.92	22.97	15.86	32.51	8.88
	Synth **N** train	BDD **N**	26.45	36.48	25.82	22.03	33.56	22.70
	Synth **ND** train	BDD **ND**	16.86	34.21	9.48	8.35	22.41	3.07
	Synth **S** train	BDD **S**	25.07	42.62	26.46	17.33	27.35	19.40
	Synth **SD** train	BDD **SD**	8.77	15.68	9.66	4.44	8.05	3.58
	Synth **W** train	BDD **W**	25.12	43.30	25.35	18.73	37.13	17.52
	Synth **WD** train	BDD **WD**	17.10	35.64	15.91	15.22	30.26	9.13
Results on all-weathers fine-tuned models	Synth all train	BDD **D**	31.46	52.71	42.89	21.43	48.13	13.01
	Synth all train	BDD **N**	26.73	46.08	32.75	25.43	42.48	24.09
	Synth all train	BDD **ND**	26.27	45.32	26.78	13.66	38.36	4.63
	Synth all train	BDD **S**	36.59	62.38	33.28	28.29	56.72	28.50
	Synth all train	BDD **SD**	36.33	49.62	44.47	29.88	43.65	36.40
	Synth all train	BDD **W**	35.13	58.62	35.14	28.94	52.37	27.45
	Synth all train	BDD **WD**	23.98	49.39	23.56	24.88	42.54	30.06
Re-test	Synth all train	City real **O** val	35.18	56.42	35.37	25.68	44.63	24.66

Table 3: Object detection and instance segmentation results on BDD conditions

Description	Fine-tune set Cityscapes	Test condition	Object detection AP	Object detection AP@50	Object detection AP@75	Instance segmentation AP	Instance segmentation AP@50	Instance segmentation AP@75
Domain shift to Mapillary	Real **O** train	City real **O** val	31.83	52.73	30.77	27.14	48.15	24.95
	Real **O** train	Mapillary **O**	24.02	38.44	25.48	20.20	35.49	19.05
Weather baselines	Real **O** train	Mapillary **N**	10.40	19.08	11.07	7.85	19.34	6.52
	Real **O** train	Mapillary **S**	11.12	18.18	11.17	10.51	16.91	11.14
	Real **O** train	Mapillary **W**	17.67	29.03	17.64	15.06	29.11	13.07
	Real **O** train	Mapillary **WD**	12.90	17.94	14.37	13.90	27.13	13.98
Results on individual fine tuned models	Synth **N** train	Mapillary **N**	8.44	17.34	6.55	6.15	11.66	5.74
	Synth **S** train	Mapillary **S**	7.75	12.27	8.59	7.04	11.51	7.09
	Synth **W** train	Mapillary **W**	16.09	27.78	15.54	15.10	26.24	15.04
	Synth **WD** train	Mapillary **WD**	13.16	19.96	12.95	15.18	29.53	16.89
Results on all-weathers fine tuned models	Synth all train	Mapillary **N**	11.14	19.94	9.11	9.80	16.62	10.25
	Synth all train	Mapillary **S**	13.69	23.22	13.28	13.22	21.10	14.40
	Synth all train	Mapillary **W**	18.22	30.89	18.69	16.80	32.41	12.73
	Synth all train	Mapillary **WD**	16.52	25.60	15.04	17.12	23.73	18.64

Table 4: Object detection and instance segmentation results on Mapillary conditions

Description	Fine-tune set Cityscapes	Test condition	AP	AP@50	AP@75
Weather baselines	Real **O** train set	DAWN **SD**	9.27	25.19	6.84
	Real **O** train set	DAWN **WD**	10.39	18.24	11.11
Results on individual fine tuned models	Synth **SD** train set	DAWN **SD**	8.13	16.04	7.74
	Synth **WD** train set	DAWN **WD**	14.49	24.96	15.76
Results on all-weathers fine tuned models	Synth all train set	DAWN **SD**	16.55	37.21	14.46
	Synth all train set	DAWN **WD**	21.20	39.19	21.62

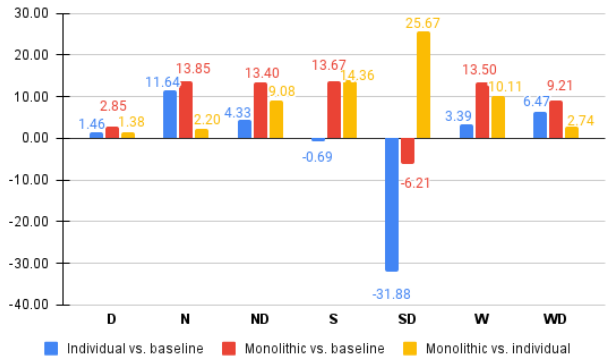
Table 5: Object detection results on DAWN conditions

Description	Fine-tune set Cityscapes	Test condition	AP	AP@50	AP@75
Weather baselines	Real **O** train set	ACDC **N**	1.63	4.68	1.12
	Real **O** train set	ACDC **S**	9.29	20.96	6.31
	Real **O** train set	ACDC **W**	10.60	21.74	8.04
Results on individual fine tuned models	Synth **N** train set	ACDC **N**	3.03	8.24	1.81
	Synth **S** train set	ACDC **S**	5.95	14.15	4.32
	Synth **W** train set	ACDC **W**	9.76	20.16	7.22
Results on all-weathers fine tuned models	Synth all train set	ACDC **N**	3.95	10.51	2.10
	Synth all train set	ACDC **S**	12.07	27.09	9.41
	Synth all train set	ACDC **W**	11.69	25.48	8.10

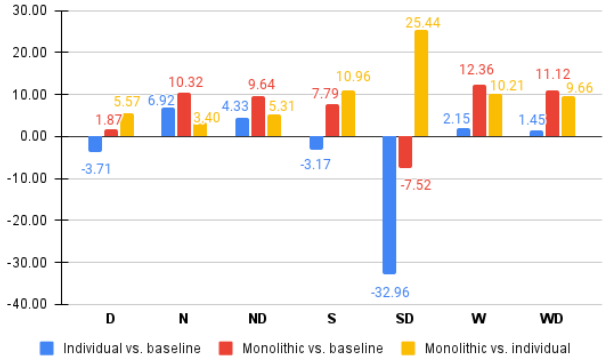
Table 6: Semantic segmentation results on ACDC conditions

5. Conclusions and proposed work

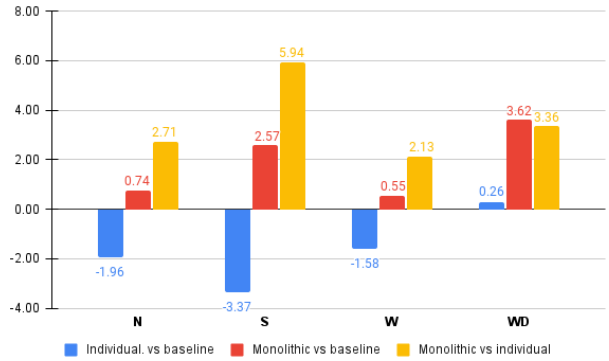
In this work we propose a modular architecture aimed to unlock diverse and stackable weather conditions with the purpose of weather appearance synthesis for improving perception downstream tasks. We generate 7 different weather styles starting from a *single* dataset with ground truth and show significant improvements in AP in both object detection and instance segmentation, in many cases exceeding 10 percentage points increase in AP. Due to the difficulty of finding aligned real-world conditions with existing ground truth from available datasets, we also train a monolithic model and show significant improvements not only over the weather baselines, but also over the original real overcast Cityscapes baseline. Finally, we publish instructions to reconstruct the dataset. As future work, we propose an ex-



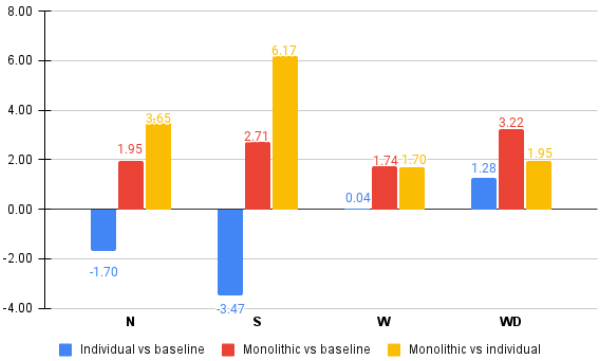
(a) Percentage points improvements in AP for Object detection, evaluated on BDD weather conditions



(b) Percentage points improvements in AP for Instance segmentation, evaluated on BDD weather conditions

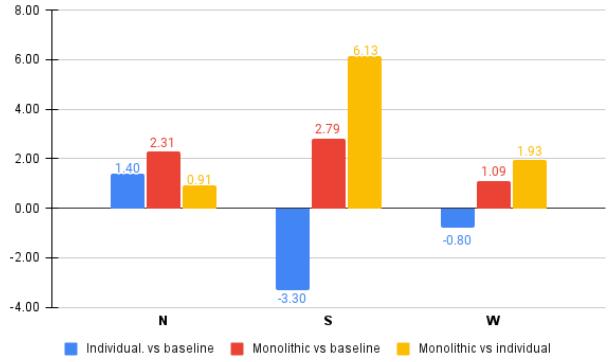


(c) Percentage points improvements in AP for Object detection, evaluated on Mapillary weather conditions

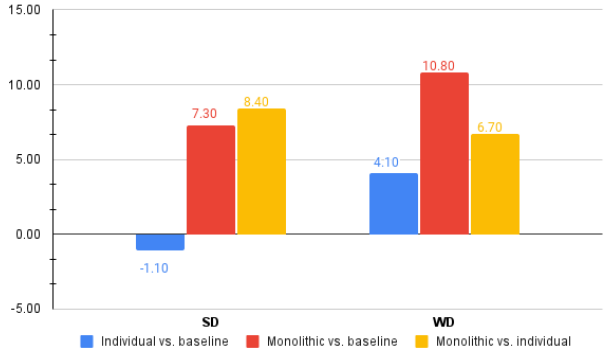


(d) Percentage points improvements in AP for Instance segmentation, evaluated on Mapillary weather conditions

Figure 5: Due to misalignment between the train conditions and test conditions, models trained on individual conditions only may exhibit loss of performance (Blue). However, training the model on all weathers leads to large improvements in performance across all conditions, compared to the baseline model trained on the original Cityscapes overcast imagery.



(a) Percentage points improvements in AP for Semantic segmentation, evaluated on ACDC weather conditions



(b) Percentage points improvements in AP for Object detection, evaluated on DAWN weather conditions

Figure 6: Reinforcing the trend observed in previous experiments, we observe that models trained with all conditions perform significantly better than either the baselines or models trained with individual conditions.

tension to our current overall methodology based on Foggy Cityscapes [38], which applies synthetic fog on real images with good weather conditions. Since their fog pipeline is based on stereo pairs from Cityscapes, we are able to use the authors' provided demo in order to add synthetic fog to our generated weathers. While quantitative analysis of the extended foggy conditions is out of the scope of this paper, we present visual results in Fig.4.

ACKNOWLEDGMENT

The authors wish to thank Alexander Rast, Peter Ball and Matthias Rolf for fruitful discussions and support throughout this work, and Izzeddin Teeti and Valeriu Plamadeala for helping out with test data management. This project has received funding from the European Union's Horizon 2020 research and innovation programme, under grant agreement No. 964505 (E-pi).

References

[1] S. Alletto, C. Carlin, L. Rigazio, Y. Ishii, and S. Tsukizawa. Adherent raindrop removal with self-supervised attention maps and spatio-temporal generative adversarial networks. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 2329–2338, 2019. 3

[2] Asha Anoosheh, Torsten Sattler, Radu Timofte, Marc Pollefeys, and Luc Van Gool. Night-to-day image translation for retrieval-based localization. pages 5958–5964, 05 2019. 1, 3

[3] Shane Barratt and Rishi Sharma. A Note on the Inception Score. *arXiv e-prints*, page arXiv:1801.01973, Jan. 2018. 5

[4] Mario Bijelic, Tobias Gruber, Fahim Mannan, Florian Kraus, Werner Ritter, Klaus Dietmayer, and Felix Heide. Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather. pages 11679–11689, 06 2020. 2

[5] Mario Bijelic, Tobias Gruber, and W. Ritter. Benchmarking image sensors under adverse weather conditions for autonomous driving. *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1773–1779, 2018. 1, 2

[6] Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179, 02 2018. 4

[7] Pak Chan, Gunwant Dhadyalla, and Valentina Donzella. A framework to analyze noise factors of automotive perception sensors. *IEEE Sensors Letters*, PP:1–1, 05 2020. 2

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3

[9] Shuai Di, Qi Feng, Chun-Guang Li, Mei Zhang, Honggang Zhang, Semir Elezovikj, Chiu Tan, and Haibin Ling. Rainy night scene understanding with near scene semantic adaptation. *IEEE Transactions on Intelligent Transportation Systems*, PP:1–9, 02 2020. 3

[10] David Eigen, Dilip Krishnan, and Rob Fergus. Restoring an image taken through a window covered with dirt or rain. pages 633–640, 12 2013. 1, 2

[11] Rui Gong, Dengxin Dai, Yu-Hua Chen, Wen Li, and L. Gool. Analogical image translation for fog generation. *ArXiv*, abs/2006.15618, 2020. 3

[12] Ian J. Goodfellow, Jean Pouget-Abadie, M. Mirza, B. Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial networks. *ArXiv*, abs/1406.2661, 2014. 3

[13] M. Hahner, D. Dai, C. Sakaridis, J. Zaech, and L. V. Gool. Semantic understanding of foggy scenes with purely synthetic data. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3675–3681, 2019. 1, 3

[14] Shirsendu Sukanta Halder, Jean-Franccois Lalonde, and Raoul de Charette. Physics-based rendering for improving robustness to rain. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10202–10211, 2019. 3

[15] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017. 4

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 4

[17] R. Heinzler, P. Schindler, J. Seekircher, W. Ritter, and W. Stork. Weather influence and classification with automotive lidar sensors. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 1527–1534, 2019. 1

[18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. 12 2017. 4

[19] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1510–1519, 2017. 3

[20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei Efros. Image-to-image translation with conditional adversarial networks. pages 5967–5976, 07 2017. 3, 4

[21] Mourad A. Kenk and M. Hassaballah. Dawn: Vehicle detection in adverse weather nature dataset. *ArXiv*, abs/2008.05402, 2020. 1, 2, 3, 5

[22] Ruoteng Li, L. Cheong, and R. Tan. Heavy rain image restoration: Integrating physics model and conditional adversarial learning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1633–1642, 2019. 3

[23] Siyuan Li, Wenqi Ren, Jiawan Zhang, Jinke Yu, and Xiaojie Guo. Single image rain removal via a deep decomposition–composition network. *Computer Vision and Image Understanding*, 186:48–57, 2019. 3

[24] C. Lin, S. Huang, Y. Wu, and S. Lai. Gan-based day-to-night image style transfer for nighttime vehicle detection. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–13, 2020. 3

[25] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. 4

[26] Will Maddern, Geoff Pascoe, Chris Linegar, and Paul Newman. 1 Year, 1000km: The Oxford RobotCar Dataset. *The International Journal of Robotics Research (IJRR)*, 36(1):3–15, 2017. 3

[27] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulò, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *International Conference on Computer Vision (ICCV)*, 2017. 3

[28] Lukáš Neumann, Michelle Karg, Shanshan Zhang, Christian Scharfenberger, Eric Piegert, Sarah Mistr, Olga Prokofyeva, Robert Thiel, Andrea Vedaldi, Andrew Zisserman, and Bernt Schiele. *NightOwls: A Pedestrians at Night Dataset*, pages 691–705. 05 2019. 2

[29] Vladislav Ostankovich, R. Yagfarov, Maksim Rassabin, and S. Gafurov. Application of cyclegan-based augmentation for

autonomous driving at night. *2020 International Conference Nonlinearity, Information and Robotics (NIR)*, pages 1–5, 2020. 2, 3

[30] T. Park, M. Liu, T. Wang, and J. Zhu. Semantic image synthesis with spatially-adaptive normalization. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2332–2341, 2019. 2, 3, 4

[31] Matthew Pitropov, Danson Evan Garcia, Jason Rebello, Michael Smart, Carlos Wang, Krzysztof Czarnecki, and Steven Waslander. Canadian adverse driving conditions dataset. *The International Journal of Robotics Research*, 0(0):0278364920979368, 0. 2

[32] Horia Porav, Tom Bruls, and P. Newman. Don't worry about the weather: Unsupervised condition-dependent domain adaptation. *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 33–40, 2019. 1

[33] H. Porav, T. Bruls, and P. Newman. I can see clearly now: Image restoration via de-raining. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7087–7093, 2019. 1, 2, 3

[34] H. Porav, W. Maddern, and P. Newman. Adversarial training for adverse conditions: Robust metric localisation using appearance transfer. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1011–1018, 2018. 1, 2, 3

[35] Horia Porav, Valentina-Nicoleta Musat, Tom Bruls, and P. Newman. Rainy screens: Collecting rainy datasets, indoors. *ArXiv*, abs/2003.04742, 2020. 1, 2, 3, 4

[36] Christos Sakaridis, Dengxin Dai, and L. Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126:973–992, 2018. 2

[37] Christos Sakaridis, Dengxin Dai, and L. Gool. Acdc: The adverse conditions dataset with correspondences for semantic driving scene understanding. *ArXiv*, abs/2104.13395, 2021. 3, 6

[38] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, Sep 2018. 1, 2, 8

[39] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019. 2, 3

[40] Tim Salimans, Ian J. Goodfellow, W. Zaremba, Vicki Cheung, A. Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, 2016. 4

[41] Edgar Schönfeld, Vadim Sushko, Dan Zhang, Juergen Gall, Bernt Schiele, and Anna Khoreva. You only need adversarial supervision for semantic image synthesis. In *International Conference on Learning Representations*, 2021. 4

[42] Lei Sun, Kaiwei Wang, Kailun Yang, and Kaite Xiang. See clearer at night: towards robust nighttime semantic segmentation through day-night image conversion. page 8, 09 2019. 1

[43] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and ZB Wojna. Rethinking the inception architecture for computer vision. 06 2016. 4

[44] X. Tan, Yiheng Zhang, Ying Cao, Lizhuang Ma, and R. Lau. Night-time semantic segmentation with a large real dataset. *ArXiv*, abs/2003.06883, 2020. 2

[45] F. Tung, J. Chen, L. Meng, and J. J. Little. The raincouver scene parsing benchmark for self-driving in adverse weather and at night. *IEEE Robotics and Automation Letters*, 2(4):2188–2193, 2017. 1, 2

[46] Michal Uricar, Jan Ulicny, Ganesh Sistu, Hazem Rashed, Pavel Krizek, David Hurych, Antonin Vobecky, and Senthil Yogamani. Desoiling dataset: Restoring soiled areas on automotive fisheye cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, Oct 2019. 2

[47] Michal Uřičář, Pavel Křížek, Ganesh Sistu, and Senthil Yogamani. Soilingnet: Soiling detection on automotive surround-view cameras. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, page 67–72. IEEE Press, 2019. 3

[48] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8798–8807, 2018. 3, 4

[49] Magnus Wrenninge and J. Unger. Synscapes: A photorealistic synthetic dataset for street scene parsing. *ArXiv*, abs/1810.08705, 2018. 3

[50] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 4, 5

[51] F. Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, V. Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *ArXiv*, abs/1805.04687, 2018. 3

[52] S. Zang, M. Ding, D. Smith, P. Tyler, T. Rakotoarivelo, and M. A. Kaafar. The impact of adverse weather conditions on autonomous vehicles: How rain, snow, fog, and hail affect the performance of a self-driving car. *IEEE Vehicular Technology Magazine*, 14(2):103–111, 2019. 1

[53] Jingming Zhao, Juan Zhang, Z. Li, J. Hwang, Yongbin Gao, Zhijun Fang, Xiaoyan Jiang, and Bo Huang. Dd-cyclegan: Unpaired image dehazing via double-discriminator cycle-consistent generative adversarial network. *Eng. Appl. Artif. Intell.*, 82:263–271, 2019. 3

[54] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017. 2, 3, 4