An Efficient Domain-Incremental Learning Approach to Drive in All Weather Conditions

M. Jehanzeb Mirza[1,2] Marc Masana[1,3] Horst Possegger[1] Horst Bischof[1,2]

[1]Institute of Computer Graphics and Vision, Graz University of Technology.
[2]Christian Doppler Laboratory for Embedded Machine Learning.
[3]TU Graz - SAL Dependable Embedded Systems Lab, Silicon Austria Labs.
{muhammad.mirza, marc.masana, possegger, bischof}@icg.tugraz.at

Abstract

Although deep neural networks enable impressive visual perception performance for autonomous driving, their robustness to varying weather conditions still requires attention. When adapting these models for changed environments, such as different weather conditions, they are prone to forgetting previously learned information. This catastrophic forgetting *is typically addressed via incremental learning approaches which usually re-train the model by either keeping a memory bank of training samples or keeping a copy of the entire model or model parameters for each scenario. While these approaches show impressive results, they can be prone to scalability issues and their applicability for autonomous driving in all weather conditions has not been shown. In this paper we propose DISC – **D**omain **I**ncremental through **S**tatistical **C**orrection – a simple online* zero-forgetting *approach which can incrementally learn new tasks (*i.e. *weather conditions) without requiring re-training or expensive memory banks. The only information we store for each task are the statistical parameters as we categorize each domain by the change in first and second order statistics. Thus, as each task arrives, we simply 'plug and play' the statistical vectors for the corresponding task into the model and it immediately starts to perform well on that task. We show the efficacy of our approach by testing it for object detection in a challenging domain-incremental autonomous driving scenario where we encounter different adverse weather conditions, such as heavy rain, fog, and snow.*

1. Introduction

In order to entrust safety-critical systems such as autonomous vehicles with human lives, they must operate robustly in widely different environments. While recent

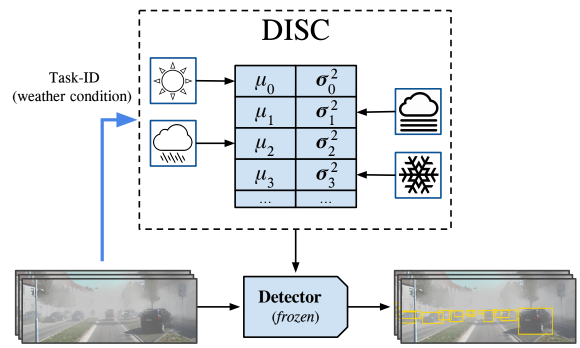


Figure 1. Robust object detection for autonomous driving across varying weather conditions with DISC. We only require domain-specific statistics (*i.e.* $\mu_{\text{ID}}, \sigma^2_{\text{ID}}$ per weather condition ID) which are inserted into a frozen detection model to achieve *zero-forgetting*, *i.e.* the detection performance on previously encountered weather conditions does not degrade over time.

deep learning-based approaches achieve very strong performance, *e.g.* for object detection [8,37,55,56], they are usually only trained and evaluated on data collected mostly in clear weather conditions [18,21,68]. However, a major real-world challenge for autonomous vehicles is to encounter deteriorating weather conditions while driving. As shown in [7,47,48], even slightly changing weather conditions can hamper the performance of object detectors considerably, making them unsafe for use in safety-critical systems.

A simple way to address such changing environmental conditions is to train different models for each individual weather condition. However, this requires extensive data collection, careful manual annotation and time-consuming re-training. Additionally, all these models need to be saved for future use which is sub-optimal because a significant amount of their learned weights might be redundant.

Instead of training a separate model for each different scenario, a better solution is to ideally have only a single model which can work in all weather scenarios. However, naively training a model for all weather scenarios individually can lead to a variety of problems, where the most critical is *catastrophic forgetting* [17, 19, 30, 46]. This means that as deep neural networks are trained for new domains (*e.g*. weather conditions), they forget information from the previous domain. For example, if an object detector trained on clear weather conditions is re-trained with foggy weather data, its performance improves for foggy weather but degrades for clear weather. This performance degradation during such model adaptation is typically addressed via Incremental Learning (IL) approaches [15, 44, 71]. These approaches learn a *sequence of tasks* (*e.g*. different weather conditions), one at a time, without having access to data from the previous tasks. They learn to perform good on a new task while retaining the performance on previously learned tasks. If performance does not degrade on previous tasks at all, then they are termed *zero-forgetting* approaches.

Numerous approaches try to improve autonomous driving algorithms for degrading weather conditions. For example, foggy weather conditions can be handled by fusing multi-modal input data [6] or combining synthetic and real data [13, 60]. Similarly, several approaches address "driving in the wild", *e.g*. [11,12,69]. Although these approaches often provide state-of-the-art results for the individual task which they are specifically designed for, their applicability to different tasks and, in fact, to different weather scenarios is not yet clear. Furthermore, these approaches mostly follow the *domain adaptation* paradigm which is different from the *incremental learning* setup and are often prone to issues like catastrophic forgetting.

We propose DISC, an efficient **D**omain-**I**ncremental learning approach which leverages **S**tatistical **C**orrection for robust object detection under varying weather conditions. DISC considers different weather conditions as distribution shifts and categorizes each condition according to its statistical difference. We achieve zero-forgetting by only retaining the weather-specific first- and second-order statistics calculated by the detection model during training and replacing these statistics when we encounter a weather change, as shown in Figure 1.

As we only store the domain-specific statistics of the model, our incremental learning approach is highly efficient, both in terms of computational cost and memory consumption, *i.e*. we neither need to store the entire model parameters nor any data samples from different domains. We show the applicability of our method in a challenging online setting where we let our system interact with multiple changes in domains (*i.e*. weather conditions) and show its effective zero-forgetting property.

Our contributions can be summarized as follows:

- We propose an incremental learning approach which considers different weather conditions as distribution shifts and thus, only needs to store the first and second order statistics for each weather condition.
- We show that by replacing the weather-specific statistics in an otherwise frozen model, we can efficiently realize a system which achieves zero-forgetting.
- We demonstrate strong performance gains using our DISC in both offline and online learning scenarios, highlighting its effectiveness.

2. Related Work

As our work lies at the intersection of incremental learning, unsupervised domain adaptation and autonomous driving in varying weather conditions, we summarize the current state-of-the-art in these research fields.

2.1. Incremental Learning

Incremental learning – learning a sequence of tasks one after the other without access to previously learned tasks – received increasing interest in recent years. Incremental learning can be divided into three main scenarios [25, 71]: task-incremental learning [15], class-incremental learning [40,44] and domain-incremental learning [1]. Throughout all these scenarios, existing approaches can be assigned to three main categories, namely replay-based, regularization-based and parameter isolation approaches.

Replay-based approaches explicitly store samples from previous tasks in a limited exemplar memory that can be rehearsed during the training of new tasks [10, 27, 54, 57]. Alternatively, some methods substitute the replay based on an exemplar memory by learning a generative model which is capable of describing previous distributions and sample from them [5, 33, 52, 65]. To counter the potential overfitting to the replayed memories, some approaches propose to constrain the interference between the new task and all previously learned ones [4, 9, 39].

Regularization-based approaches introduce extra regularization terms in the training loss which limit the change of important weights or large shifts on the activations. One of the most common strategies is knowledge distillation, which enforces the outputs of the network to shift as little as possible for previously learned classes, while allowing enough change to learn the new ones [28, 35, 53, 66, 78]. The second most common approach consists of calculating the importance of each parameter in the network and penalizing their update based on that importance [2, 30, 38, 77].

Parameter isolation approaches mostly assume no constraints on the model size. They usually isolate or freeze important model parameters from previous tasks and allow the models to introduce new parameters in order to exploit strong connections and avoid forgetting [3,51,59,75]. Other approaches enforce zero-forgetting by learning masks or paths for each parameter or each layer representation [16, 42, 45, 63].

Most of the incremental learning approaches are usually proposed for either task-IL or class-IL, and in general are framed inside a classification problem. In this paper, however, we demonstrate how to leverage domain-IL for adaptation to varying weather conditions. To this end, we propose DISC, a method which is designed for domain-IL and which does not rely on replay memories, regularization terms, or even long training sessions. Contrary to replay- or regularization-based methods, it provides zero-forgetting properties, and unlike parameter isolation models, it does not require the computationally expensive training or calculation of masks and paths.

2.2. Correcting Domain Statistics

Adapting the batch normalization [26] statistics for the target domain data has been extensively used for unsupervised domain adaptation. For example, Li *et al*. [34] recalculate the first and second order statistics of the batch normalization layer for domain adaptation, while Carlucci *et al*. [43] learn hyper-parameters during training to optimally mix the statistics from source and target domain. Other approaches, such as [50, 62, 67, 79] propose prediction-time batch normalization in order to reduce the statistical discrepancies between source and target domains. In addition to correcting the first and second order statistics for the target domain, Wang *et al*. [72] also learn the scale and shift parameters of the batch normalization layer by calculating the gradients from the entropy of predictions. Mirza *et al*. [49] introduce DUA, a highly data-efficient method to adapt the statistics in an online manner for the target domain for unsupervised domain adaptation.

2.3. Autonomous Driving in Varying Weather

Improving the robustness of autonomous vehicles to varying weather conditions also gained increased attention in recent years. Recent works propose approaches for semantic segmentation [13,60] and image defogging [76] during night. Sakaridis *et al*. [61] improve semantic segmentation for foggy scenes using synthetic data. Bijelic *et al*. [6] propose a multi-modal fusion architecture for driving in foggy conditions. Chen *et al*. [11, 12] propose object detection approaches for driving in the wild. RoyChowdhury *et al*. [58] propose a self-training approach for adaptation of object detectors to different weather conditions. Other works, such as [29,64] propose image de-hazing approaches in the context of autonomous driving.

Our work also focuses on driving in diverse weather conditions. However, in contrast to other task- and weather-specific approaches, *e.g*. [6, 11–13, 60], which mostly follow the *domain adaptation* paradigm, we propose a *zero-forgetting domain-incremental learning* approach which learns to drive in varying weather conditions as they are encountered while not reducing the performance on previous conditions. We consider different weather conditions as distribution shifts and employ DUA [49] to learn the weather-specific statistics. During testing we only need to replace these weather-specific statistics which makes our approach highly efficient and achieve zero-forgetting.

3. Domain-IL for Weather Conditions

In incremental learning scenarios, a sequence of tasks is learned one at a time within their own training sessions, without access to data from previously seen tasks. Following most common incremental scenarios [15,44,71], we can define an IL problem as the sequence of n tasks:

$$\mathcal{T} = [(C^1, D^1), (C^2, D^2), \dots, (C^n, D^n)], \tag{1}$$

where the set of classes $C^t = \{c_1^t, c_2^t ..., c_{n^t}^t\}$ represents each task t, learned with training data D^t. The training data consists of input features $x \in X$ (*i.e*. images in our setup) and corresponding ground truth labels y (*i.e*. bounding box annotations and class labels in our case). Therefore, we have $D^t = \{(x_1, y_1), (x_2, y_2), \dots, (x_k, y_k)\}$ all available pairs of input and label for a given task, where all $y_i \in C^t$.

In this paper, we focus on the variant of *domain-incremental* learning, where all tasks share the same classes, *i.e*. $C^t = C^1 \ \forall t \in \mathcal{T}$. As the sequence of tasks is learned, the same objects have to be detected while their domain and data distribution changes. At inference time, similar to task-incremental learning [15], we have access to the task-ID. For our use-case this means that we know the current weather condition.

The properties from the proposed domain-incremental learning scenario correspond to the real-world problem of object detection under different weather conditions. We aim to learn the same set of objects (classes) over different weather conditions (domains), while having access to the task-ID (current weather). A representation of the domain-IL scenario we adopt is depicted in Figure 2.

4. Method

Before presenting our domain-IL approach DISC, we first explore how correcting task-specific latent representation statistics can be used for domain adaptation.

4.1. Statistical Correction for Domain Adaptation

In deep neural networks, batch normalization [26] is a strategy that aims to reduce the internal covariate shifts between each layer's input distribution during training. Essentially, it normalizes the input activations of each layer to have a zero-centered mean and unit variance, such that

$$\hat{x} = \frac{x - \mathbb{E}[X]}{\sqrt{\mathrm{Var}[X] + \epsilon}} \cdot \gamma + \beta, \tag{2}$$

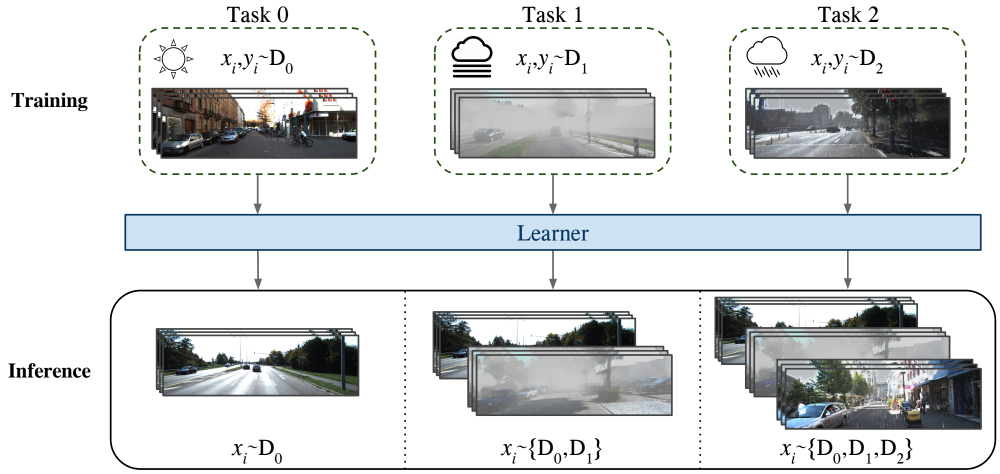


Figure 2. Domain-incremental learning setting used in this paper. Different tasks correspond to different weather conditions which are learned in the training phase. During inference, we evaluate the performance of the current task along with all the previously seen tasks.

where each input sample x is normalized by the mean and variance calculated from the activations during training. The parameters γ and β denote the distribution scale and shift, respectively, while $\epsilon > 0$ prevents division by zero. The expected value $\mathbb{E}[X]$ is estimated by the running mean

$$\hat{\mu}_k = (1 - \rho) \cdot \hat{\mu}_{k-1} + \rho \cdot \mu_k, \tag{3}$$

while the variance $\mathrm{Var}[X]$, *i.e*. the variability around the expectation is estimated by the running variance

$$\hat{\sigma}_k^2 = (1 - \rho) \cdot \hat{\sigma}_{k-1}^2 + \rho \cdot \sigma_k^2, \tag{4}$$

where μ and σ^2 denote the mean and variance of the current incoming batch of data samples, respectively. The running mean and variance are updated with each new batch of data. The momentum parameter ρ (default $\rho = 0.1$) controls the balance between the previously accumulated statistics and the effect of the current batch statistics.

By design, the behavior of batch normalization is inherently different between training and inference phases. During training, since data is forwarded through the network in batches, the statistics are updated by the batch normalization layer via the running mean and variance (Eqs. (3) and (4)). However, at inference time, the statistics accumulated during training are fixed and used for normalization following Eq. (2). Batch normalization works best when both train and test data are from a similar distribution [22, 23, 74]. However, when train and test data distributions are not well aligned, batch normalization can cause performance degradation [73].

Correcting the statistics when train and test data distributions are not aligned can be helpful for unsupervised domain adaptation [34,49,50,62,72,79]. Existing approaches either adapt or replace the statistics estimated by the batch normalization layers during training for the out-of-distribution test data, improving performance on the latter. Since different weather conditions can also be considered as being out-of-distribution w.r.t. the clear weather condition, we propose a similar technique for our domain-incremental learning through statistical correction approach. In particular, we draw inspiration from DUA [49] due to its data-efficient nature for domain adaptation.

4.2. DISC

Motivated by the promising results for domain adaptation, we leverage task-specific statistical correction for domain-incremental learning. Specifically, we propose that each time a new task from the sequence arrives, we freeze the parameters of the model and only calculate the running mean and variance at each batch normalization layer. While DUA [49] calculates the statistics from a fraction of the data for domain adaptation, for domain-IL we need to store task-specific statistics to overcome catastrophic forgetting.

Since our DISC only requires a quick, efficient, zero-forgetting storage of the statistics at each batch normalization layer for each task in the sequence, we divide its usage into two phases, namely adaptation & plug-and-play:

In the **Adaptation phase**, we require only a tiny fraction of unlabeled data from each weather condition to adapt the statistics estimated by the batch normalization layers via DUA [49]. DUA adapts the statistics by directly using the out-of-distribution test data without requiring backpropagation. It uses an adaptive momentum (ρ) scheme in order to adjust the statistics by using less than 1% of test data. After adapting to each of the weather conditions until convergence (*i.e*. when the statistics become stable), we store the weather-specific statistics calculated by this model and discard the remaining model parameters. This means that we only need to store the vectors of the running mean

$\hat{\mu}$ and variance $\hat{\sigma}^2$ for each batch normalization layer in the model. In contrast to DUA, we do not use the test data to adapt the statistics, but only use the training data collected in degrading weather to calculate the statistics.

In the **Plug-and-Play phase**, we apply the weather-specific statistics vectors to the initial model, which is only trained on the first task. At inference, we replace the batch normalization statistics in the initial model with the ones learned for the current task (*plug*) and forward the data through the network (*play*). We do not replace or modify any other model parameters other than the running mean and variance of the batch normalization layers. Thus, we can exactly recover the model's original performance on all previous tasks, even after encountering new tasks. Consequently, DISC belongs to the family of *zero-forgetting* incremental learning approaches [16, 41, 42, 45].

The proposed approach provides several advantages. First, the lack of update on the model parameters completely avoids catastrophic forgetting when learning new tasks, effectively turning our proposed approach into a zero-forgetting one. Second, only having to calculate the statistics means that we do not need computationally expensive training sessions. In particular, only a single forward pass through the model with a fraction of the current unlabeled training data is sufficient. This makes our approach extremely efficient and easily adaptable. Third, the storage of the statistics for each layer grows linearly with the number of tasks, but it is still quadratically more efficient than storing parameters for each weight or a memory of exemplars, as most IL methods do. Next, since the estimate of the statistics is only based on the available samples from D^t, our method can be compared under both online and offline domain-IL scenarios. Finally, our approach only relies on the x_i samples from the data, without the need for the labels y_i – except for the starting task model of the sequence which might require supervised training. This allows our method to be applicable in practical scenarios where the subsequent tasks in the sequence (*i.e.* after the first), can be learned in an unsupervised manner.

A requirement of DISC and the proposed domain-IL scenario is to have access to the task-ID during inference, as it is done in task-incremental learning approaches [15]. However, especially for weather conditions, this task-ID can easily be inferred by either training a separate classifier, or simply querying additional sensors available in modern cars, such as (typically infrared-based) rain sensors. We also provide results for using incorrect task-IDs in Sec. 6.2.

5. Experimental Setting

In this section, we first introduce the dataset and tasks on which we evaluate our domain-incremental learning approach. Then, we provide details about the experimental protocols and the baselines we compare to.

	car	van	truck	ped	sit	cyc	tram	misc	sum
train	12884	1307	475	1905	101	688	258	398	18016
val	1381	150	54	262	16	83	34	67	2047
test	14477	1457	565	2320	105	856	219	508	20507

Table 1. Annotated instances of classes for the different dataset splits of KITTI used in our experiments.

5.1. Dataset and Tasks

We conduct all our experiments on the widely used KITTI [18] autonomous driving dataset. The KITTI dataset consists of 8 object classes manually annotated for training both 2D and 3D object detectors. The dataset consists of real-world driving scenes captured in Germany in both rural and urban areas. Separate training and testing splits are provided for the KITTI dataset. However, publicly available annotations are only provided for the training split, whereas the testing split is used for the KITTI object detection challenge which is evaluated through their private server.

Thus, we follow the common protocol [31, 32] and divide the 7,481 images in the KITTI training dataset into train (3,740 images) and test (3,741 images) splits. Further, we use 10% of this train split as validation split during training to optimize hyper-parameters. Details about the exact number of instances present for each class in each split is provided in Table 1. These splits are fixed for both training and inference across all experiments and methods.

Recently, several works introduced approaches to change the input images realistically to simulate driving in different weather scenarios, such as fog [20], rain [70], and snow [24]. We use these approaches to create different weather scenarios for our domain-incremental learning setup. Thus, we define the following four-task scenario:

Task 0 - Clear: The initial task in our incremental learning setup is driving in clear weather. Since the KITTI dataset is recorded almost exclusively in bright daylight conditions, we use the original KITTI data as input for this task.

Task 1 - Fog: The second task in the sequence is driving in foggy weather. Halder *et al.* [20] introduce realistic physics-based rendering of fog. In particular, they use the depth information from the LiDAR sensor, re-projected into the camera view to simulate 7 different fog levels. The fog severity is categorized by the visibility in meters and varies from 30 m, corresponding to the most severe fog, to a visibility of 750 m, corresponding to the least severe fog. For this task, we apply their method to simulate fog on KITTI at the maximum severity level, *i.e.* 30 m visibility.

Task 2 - Rain: The third task to learn is the rainy weather condition. Tremblay *et al.* [70] propose a physics-based rain model to simulate photo-realistic rain at 8 different levels of severity. They also leverage the re-projected LiDAR-based depth measurements and apply a particle simulation framework [14] to approximate the real-world dynamics of raindrops. The degradation severity can be specified on the basis of rain intensity measured in mm/hr, ranging from light rain at 1 mm/hr to heavy rain at 200 mm/hr. For a challenging scenario, we use the highest severity level at 200 mm/hr in all our experiments to simulate heavy rain on KITTI data.

Task 3 - Snow: The final task we address in our sequence of weather conditions is snow. Hendrycks and Dietterich [24] evaluate the robustness of various deep neural networks in several different scenarios. In their benchmark, they also introduce an approach to simulate snow on images which we use to superimpose snow on the KITTI images.

5.2. Implementation Details

Throughout all experiments we use the open source PyTorch implementation[1] of the YOLOv3 [55] object detector. For statistical correction of the batch normalization layers in YOLOv3 we use the official DUA implementation[2]. We use the default settings from the implementations except for the learning rate and batch sizes. For training all models we use a batch size of 16 while learning rates are handled by an early-stopping criteria. When the validation error does not improve for 5 epochs, we decrease the learning rate by a factor of 3. If the validation error still does not improve after 3 learning rate changes, we stop training the model further. The initial learning rate value is 0.01.

5.3. Experimental Protocols and Metrics

For all experiments we report the mean average precision (mAP) over all object classes, evaluated at 50% intersection over union (IOU). We follow this setting for reporting all object detection results unless stated otherwise.

In IL, it is common to provide metrics which evaluate over all the tasks seen so far while advancing through the sequence of tasks. Thus we adopt this protocol. In most IL setups, a weighted average over the task is provided, since the amount of classes and data might be variable. However, in our experimental setting we have the same amount of samples per weather condition (task), therefore a simple average over all seen tasks mAPs is sufficient.

Since our proposed domain-IL scenario is task-aware (*i.e.* we know the weather condition present in each image), we allow all compared methods to use this information and thus, apply or replace the corresponding task-specific parts of each model. For example, for Freezing we use the detector head trained for the specific weather condition. In Sec. 6.2, we additionally analyse how well weather-specific statistics or models can perform across the different weather conditions.

[1] https://github.com/ultralytics/yolov3, commit: d353371
[2] https://github.com/jmiemirza/DUA, commit: 7a0240c

5.4. Baselines

We propose to compare with several baselines which can be implemented in two distinct categories: offline and online. While offline approaches are allowed to train until convergence on each new task, online approaches are only trained for a single epoch on the new task. Thus, for offline approaches we can generally expect a better performance on the newest task, with an increased amount of forgetting of previous tasks. On the other hand, as online approaches are optimized only for a single epoch on the new tasks, it is expected that they maintain a stronger stability and reduced amount of forgetting. We evaluate all baselines in both online and offline settings. The baselines we consider are:

Source-Only maintains the initial model trained on clear weather data completely fixed throughout all tasks. This baseline does not perform any adaptation, modification or replacement to its parameters and serves as a lower bound.

Freezing only allows to update the parameters for the three YOLOv3 heads – which are specialized on detecting objects at different spatial resolutions – while keeping the representations learned by the remaining layers frozen. To train the separate sets of heads for each of the tasks, we use the learning rate at which the training for the initial task terminated. Similar to DISC, freezing is also a zero-forgetting approach.

Disjoint trains a separate model for each task. During inference, the model which matches the corresponding task-ID is used. To train this baseline, we initialize each of the disjoint YOLOv3 models with weights pre-trained on the MS COCO [36] object detection dataset and fine-tune for the corresponding weather condition until convergence.

Fine-tuning learns each task incrementally in a fully-supervised manner. When a new task arrives, the current model will be fine-tuned on the task-specific training data until convergence. The learning rate used is the one at which the initial task terminated.

Joint-training trains the model with all data from all tasks seen so far in a fully-supervised manner, breaking one of the key incremental learning properties and thus, provides an upper bound. The objective is to provide a comparison to a model which can learn a weather-invariant representation which performs well under all weather conditions.

6. Results

We now present detailed results comparing DISC to different baselines, for both offline and online training.

6.1. Comparison in the Four-Task Scenario

First, we evaluate the four different weather conditions (introduced in Sec. 5.1) when presented in the domain-incremental learning setting (described in Sec. 3). The results are summarized in Table 2.

Method	clear → fog → rain → snow
Source-Only	91.7 → 58.9 → 61.9 → 51.8
Freezing	91.7 → 61.4 → 63.7 → 54.0
Disjoint	91.7 → 56.7 → 66.9 → 71.4
Fine-tuning	91.7 → 59.4 → **80.0** → **76.2**
DISC	91.7 → **66.2** → 68.8 → 59.7
Joint-training	91.7 → 95.3 → 97.2 → 97.2

(a) Offline

Method	clear → fog → rain → snow
Source-Only	91.7±0.0 → 58.9±0.0 → 61.9±0.0 → 51.8±0.0
Freezing	91.7±0.0 → 60.8±0.9 → 63.6±0.3 → 53.6±0.4
Disjoint	91.7±0.0 → 38.4±1.0 → 66.0±1.2 → 57.3±1.0
Fine-tuning	91.7±0.0 → 34.0±3.9 → 67.6±1.3 → 39.6±1.1
DISC	**91.7±0.0** → **66.2±0.0** → **68.8±0.0** → **59.7±0.0**
Joint-training	91.7±0.0 → 79.7±1.4 → 85.4±0.6 → 84.9±0.7

(b) Online

Table 2. Mean Average Precision (mAP@50) on KITTI averaged over all object classes. Following the common IL protocol, we report the results as the mean over the current and all previously seen tasks. (a) Offline setting, where all baselines are optimized until convergence. (a) Online setting, where all baselines are only trained for a single epoch. We report the mean and standard deviation over 10 runs.

In Table 2a we compare the *offline setting*, where all baselines are trained in a supervised manner until convergence. Results show the mAP@50 averaged over all object classes, with DISC obtaining the best results after learning the fog task, and Fine-tuning obtaining the best results after the remaining tasks of the sequence. DISC obtains better results than the other zero-forgetting approach, Freezing, across all the sequence, showing that adapting the intermediate layer representations is better than freezing them. Due to our challenging adverse weather simulation settings, there is a notable gap between all approaches and the upper performance bound (*i.e.* Joint-training), which opens the door for further improved strategies in the future. Note that the performance gap in comparison to the Disjoint and Fine-tuning baselines after the rain and snow tasks is expected, because in contrast to DISC, both baselines can fully adapt the detection model. DISC, however, provides a simple and efficient way to notably improve the initial model performance in varying weather conditions, without requiring extensive re-training or memory banks.

In Table 2b, we provide results for the *online setting*, where all baselines are trained for a single epoch on subsequent tasks. To ensure a fair comparison in this setting, all models are initialized with a model pre-trained on the clear weather condition. In this case, DISC obtains the best results across all tasks and throughout the whole sequence. This demonstrates that DISC is well suited for the practical online scenario, since it only needs to store the statistics after forwarding the samples through the network, and does not need any other training for subsequent tasks. Comparing all methods to the lower bound (*i.e.* Source-Only) shows that this learning setting is not trivial: Without using exemplars or any other technique to avoid catastrophic forgetting, both Fine-tuning and Disjoint degrade significantly at the most challenging fog and snow conditions.

6.2. Interaction between Tasks

To investigate how different the weather condition tasks are, we conduct an experiment where we use the task-specific statistics/models on all other tasks. For DISC, we use each weather-specific statistics and evaluate the performance under all other weather conditions. Similarly for Disjoint, we evaluate each of the separate weather models on all weather tasks. Through this experiment we want to observe how good the learned representations are when presented with a significant domain shift, which could happen by incorrectly estimating the current weather task-ID.

The results of this experiment are presented in Fig. 4. The diagonal entries show the results obtained when using the methods with the correct task-ID, while the first row contains the results corresponding to the Source-Only baseline. In all cases, using the statistics/models for the weather condition they are trained on provides the best results. For both DISC and Disjoint, we can observe that fog and snow are the most challenging weather conditions, most likely due to some objects becoming barely visible in the deep fog/heavy snow. The Disjoint results demonstrate that, obviously, a weather-specific model has a clear advantage on highly adverse conditions (especially snow). This can be attributed to the large visual appearance gap (*e.g.* consider clear weather *vs.* almost "white-out" snow scenario) which seemingly requires different feature representations to be learned by the detection model.

Considering the DISC results in this cross-task experiment, we see that the statistics learned from rain and snow conditions seem to have good transferability, as the performance is always comparable or better than using the initial clear weather statistics (*i.e.* Source-Only results). Fog statistics, on the other hand, perform significantly better than clear weather statistics only when evaluated on fog data. For both DISC and Disjoint, we see that using fog-specific statistics/model on snow data, the performance decreases. This degradation is especially significant for Disjoint. In general, it seems that Disjoint models are very good at solving the weather condition they have been trained on, while mostly having less transferability to other domains than DISC.

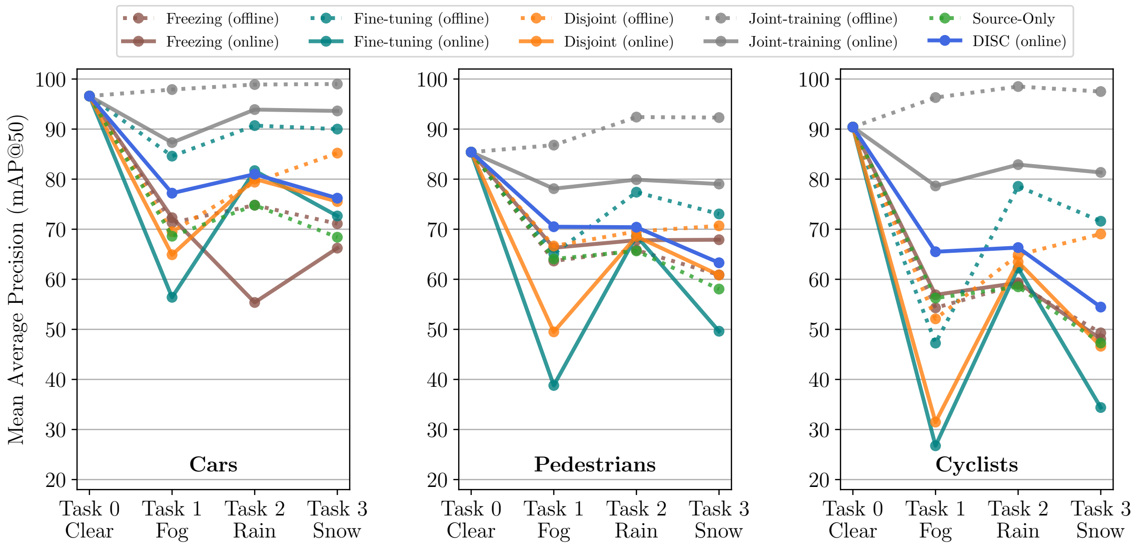


Figure 3. Mean Average Precision (mAP@50) for the three most common traffic participant classes, *i.e*. cars, pedestrians, and cyclists. We show the results for all baselines for both online and offline training. We follow the common IL evaluation protocol and report the performance at a particular task as the mean over this task and all previously seen tasks.

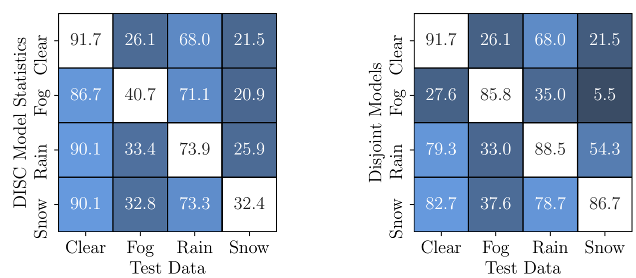
6.3. Results for Individual Classes

Following the common evaluation protocol for object detection in autonomous driving scenarios, *e.g*. [31,32,49], we separately analyse the results for the three most frequently occurring traffic participants, *i.e*. cars, pedestrians, and cyclists. From Fig. 3 we can observe several trends: In general, the performance for all classes drops after adjusting to fog. Moving to the next weather change (rain), we see that the performance of the zero-forgetting methods (DISC & Freezing) stays more or less the same, whereas offline approaches improve (as they can adjust the model until convergence) and competing online approaches degrade even further. Finally, when encountering the challenging snow scenario all approaches – except for the offline Disjoint variant – drop in performance. Overall, these results also confirm that a realistic and practical online learning scenario is significantly more challenging than the offline setting.

While this behaviour is similar to the previously observed results in Table 2a and 2b, here we can observe some object class-specific differences. When driving in deep fog, reliably detecting cyclists is most challenging for all approaches, indicated by the large performance drops for this class. Furthermore, considering the upper performance bound (*i.e*. Joint-training), future research should focus on pedestrian detection in adverse weather conditions to enable robust and safe autonomous driving systems.

7. Conclusion

We proposed a novel zero-forgetting domain-incremental learning approach to efficiently address varying weather conditions in autonomous driving scenarios. By only correcting the weather-specific domain statistics when encountering weather changes, we achieve robust object detection even in adverse weather conditions. Our evaluations on challenging consecutive weather changes show promising results of our DISC approach for both online and offline learning scenarios.



(a) DISC (b) Disjoint

Figure 4. Cross-task evaluation using mAP@50 averaged over all classes. We apply (a) DISC's weather-specific statistics and (b) the per-weather optimized models on all different weather conditions. Entries along the diagonal show results at the condition for which the corresponding statistics/models were trained for.

Acknowledgments This work was partially funded by the Christian Doppler Laboratory for Embedded Machine Learning and the Austrian Research Promotion Agency (FFG) under the project High-Scene (884306). Marc Masana acknowledges the support by the "University SAL Labs" initiative of Silicon Austria Labs (SAL).

References

[1] M Acharya, T Hayes, and C Kanan. RODEO: Replay for Online Object Detection. In *Proc. BMVC*, 2020. 2

[2] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory Aware Synapses: Learning what (not) to Forget. In *Proc. ECCV*, 2018. 2

[3] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert Gate: Lifelong Learning with a Network of Experts. In *Proc. CVPR*, 2017. 2

[4] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient Based Sample Selection for Online Continual Learning. In *NeurIPS*, 2019. 2

[5] C Atkinson, B McCane, L Szymanski, and A Robins. Pseudo-Recursal: Solving the Catastrophic Forgetting Problem in Deep Neural Networks. *arXiv preprint arXiv:1802.03875*, 2018. 2

[6] 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. In *Proc. CVPR*, 2020. 2, 3

[7] Mario Bijelic, Paula Kysela, Tobias Gruber, Werner Ritter, and Klaus Dietmayer. Recovering the Unseen: Benchmarking the Generalization of Enhancement Methods to Real World Data in Heavy Fog. In *Proc. CVPR Workshops*, 2019. 1

[8] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 1

[9] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient Lifelong Learning with A-GEM. In *Proc. ICLR*, 2019. 2

[10] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and M Ranzato. Continual Learning with Tiny Episodic Memories. *arXiv preprint arXiv:1902.10486*, 2019. 2

[11] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain Adaptive Faster R-CNN for Object Detection in the Wild. In *Proc. CVPR*, 2018. 2, 3

[12] Chen, Yuhua and Wang, Haoran and Li, Wen and Sakaridis, Christos and Dai, Dengxin and Van Gool, Luc. Scale-aware Domain Adaptive Faster R-CNN. *IJCV*, 129(7):2223–2243, 2021. 2, 3

[13] Dengxin Dai and Luc Van Gool. Dark Model Adaptation: Semantic Image Segmentation from Daytime to Nighttime. In *Proc. ITSC*, 2018. 2, 3

[14] Raoul De Charette, Robert Tamburo, Peter C Barnum, Anthony Rowe, Takeo Kanade, and Srinivasa G Narasimhan. Fast Reactive Control for Illumination through Rain and Snow. In *Proc. ICCP*, 2012. 6

[15] Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A Continual Learning Survey: Defying Forgetting in Classification Tasks. *TPAMI*, 2021. 2, 3, 5

[16] Chrisantha Fernando, Dylan Banarse, Charles Blundell, Yori Zwols, David Ha, Andrei A Rusu, Alexander Pritzel, and Daan Wierstra. Pathnet: Evolution Channels Gradient Descent in Super Neural Networks. *arXiv preprint arXiv:1701.08734*, 2017. 3, 5

[17] Robert M French. Catastrophic Forgetting in Connectionist Networks. *Trends in Cognitive Sciences*, 1999. 2

[18] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision Meets Robotics: The KITTI Dataset. *IJR*, 32(11):1231–1237, 2013. 1, 5

[19] Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An Empirical Investigation of Catastrophic Forgetting in Gradient-based Neural Networks. In *Proc. ICLR*, 2014. 2

[20] Shirsendu Sukanta Halder, Jean-François Lalonde, and Raoul de Charette. Physics-Based Rendering for Improving Robustness to Rain. In *Proc. ICCV*, 2019. 5

[21] Jianhua Han, Xiwen Liang, Hang Xu, Kai Chen, Lanqing Hong, Chaoqiang Ye, Wei Zhang, Zhenguo Li, Chunjing Xu, and Xiaodan Liang. SODA10M: Towards Large-Scale Object Detection Benchmark for Autonomous Driving. In *NeurIPS*, 2021. 1

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proc. CVPR*, 2016. 4

[23] Zhenwei He and Lei Zhang. Multi-adversarial Faster-RCNN for Unrestricted Object Detection. In *Proc. ICCV*, 2019. 4

[24] Dan Hendrycks and Thomas Dietterich. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. In *Proc. ICLR*, 2019. 5, 6

[25] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating Continual Learning Scenarios: A Categorization and Case for Strong Baselines. In *NeurIPS Workshops*, 2018. 2

[26] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proc. ICML*, 2015. 3

[27] David Isele and Akansel Cosgun. Selective Experience Replay for Lifelong Learning. In *Proc. AAAI*, 2018. 2

[28] Heechul Jung, Jeongwoo Ju, Minju Jung, and Junmo Kim. Less-Forgetting Learning in Deep Neural Networks. *arXiv preprint arXiv:1607.00122*, 2016. 2

[29] Guisik Kim, Jinhee Park, Suhyeon Ha, and Junseok Kwon. Bidirectional Deep Residual learning for Haze Removal. In *Proc. CVPR Workshops*, 2019. 3

[30] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming Catastrophic Forgetting in Neural Networks. *Proceedings of the national academy of sciences*, 2017. 2

[31] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d Proposal Generation and Object Detection from View Aggregation. In *Proc. IROS*, 2018. 5, 8

[32] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast Encoders for Object Detection from Point Clouds. In *Proc. CVPR*, 2019. 5, 8

[33] Frantzeska Lavda, Jason Ramapuram, Magda Gregorova, and Alexandros Kalousis. Continual Classification Learning using Generative Models. *arXiv preprint arXiv:1810.10612*, 2018. 2

[34] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting Batch Normalization For Practical Domain Adaptation. In *Proc. ICLR Workshops*, 2016. 3, 4

[35] Zhizhong Li and Derek Hoiem. Learning Without Forgetting. *TPAMI*, 40(12):2935–2947, 2017. 2

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In *Proc. ECCV*, 2014. 6

[37] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single Shot MultiBox Detector. In *Proc. ECCV*, 2016. 1

[38] Xialei Liu, Marc Masana, Luis Herranz, Joost Van de Weijer, Antonio M Lopez, and Andrew D Bagdanov. Rotate your Networks: Better Weight Consolidation and Less Catastrophic Forgetting. In *ICPR*, 2018. 2

[39] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient Episodic Memory for Continual Learning. In *NeurIPS*, 2017. 2

[40] Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online Continual Learning in Image Classification: An Empirical Survey. *Neurocomputing*, 469:28–51, 2022. 2

[41] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a Single Network to Multiple Tasks by Learning to Mask Weights. In *Proc. ECCV*, 2018. 5

[42] Arun Mallya and Svetlana Lazebnik. Packnet: Adding Multiple Tasks to a Single Network by Iterative Pruning. In *Proc. CVPR*, 2018. 3, 5

[43] Fabio Maria Carlucci, Lorenzo Porzi, Barbara Caputo, Elisa Ricci, and Samuel Rota Bulo. AutoDIAL: Automatic DomaIn Alignment Layers. In *Proc. ICCV*, 2017. 3

[44] Marc Masana, Xialei Liu, Bartłomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost van de Weijer. Class-Incremental Learning: Survey and Performance Evaluation. *arXiv preprint arXiv:2010.15277*, 2020. 2, 3

[45] Marc Masana, Tinne Tuytelaars, and Joost Van de Weijer. Ternary Feature Masks: Zero-Forgetting for Task-Incremental Learning. In *Proc. CVPR Workshops*, 2021. 3, 5

[46] Michael McCloskey and Neal J Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. *Psychology of Learning and Motivation*, 1989. 2

[47] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking Robustness in Object Detection: Autonomous Driving when Winter is Coming. *arXiv preprint arXiv:1907.07484*, 2019. 1

[48] Muhammad Jehanzeb Mirza, Cornelius Buerkle, Julio Jarquin, Michael Opitz, Fabian Oboril, Kay-Ulrich Scholl, and Horst Bischof. Robustness of Object Detectors in Degrading Weather Conditions. In *Proc. ITSC*, 2021. 1

[49] Muhammad Jehanzeb Mirza, Jakub Micorek, Horst Possegger, and Horst Bischof. The Norm Must Go On: Dynamic Unsupervised Domain Adaptation by Normalization. *arXiv preprint arXiv:2112.00463*, 2021. 3, 4, 8

[50] Zachary Nado, Shreyas Padhy, D Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating Prediction-Time Batch Normalization for Robustness under Covariate Shift. *arXiv preprint arXiv:2006.10963*, 2020. 3, 4

[51] Jathushan Rajasegaran, Munawar Hayat, Salman Khan, Fahad Shahbaz Khan, and Ling Shao. Random Path Selection for Incremental Learning. In *NeurIPS*, 2019. 2

[52] Jason Ramapuram, Magda Gregorova, and Alexandros Kalousis. Lifelong Generative Modeling. *Neurocomputing*, 2020. 2

[53] Amal Rannen, Rahaf Aljundi, Matthew B Blaschko, and Tinne Tuytelaars. Encoder Based Lifelong Learning. In *Proc. ICCV*, 2017. 2

[54] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental Classifier and Representation Learning. In *Proc. CVPR*, 2017. 2

[55] Joseph Redmon and Ali Farhadi. YOLOv3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1, 6

[56] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *NeurIPS*, 2015. 1

[57] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience Replay for Continual Learning. In *NeurIPS*, 2019. 2

[58] Aruni RoyChowdhury, Prithvijit Chakrabarty, Ashish Singh, SouYoung Jin, Huaizu Jiang, Liangliang Cao, and Erik Learned-Miller. Automatic Adaptation of Object Detectors to New Domains using Self-training. In *Proc. CVPR*, 2019. 3

[59] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive Neural Networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

[60] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Guided Curriculum Model Adaptation and Uncertainty-aware Evaluation for Semantic Nighttime Image Segmentation. In *Proc. CVPR*, 2019. 2, 3

[61] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic Foggy Scene Understanding with Synthetic Data. *IJCV*, 126(9):973–992, 2018. 3

[62] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving Robustness Against Common Corruptions by Covariate Shift Adaptation. In *NeurIPS*, 2020. 3, 4

[63] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming Catastrophic Forgetting with Hard Attention to the Task. In *Proc. ICML*, 2018. 3

[64] Jiawei Shen, Zhuoyan Li, Lei Yu, Gui-Song Xia, and Wen Yang. Implicit Euler ODE Networks for Single-Image Dehazing. In *Proc. CVPR Workshops*, 2020. 3

[65] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual Learning with Deep Generative Replay. In *NeurIPS*, 2017. 2

[66] Daniel L Silver and Robert E Mercer. The Task Rehearsal Method of Life-Long Learning: Overcoming Impoverished Data. In *Conference of the Canadian Society for Computational Studies of Intelligence*, 2002. 2

[67] Saurabh Singh and Abhinav Shrivastava. EvalNorm: Estimating Batch Normalization Statistics for Evaluation. In *Proc. ICCV*, 2019. 3

[68] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in Perception for Autonomous Driving: Waymo Open Dataset. In *Proc. CVPR*, 2020. 1

[69] Luan Tran, Kihyuk Sohn, Xiang Yu, Xiaoming Liu, and Manmohan Chandraker. Gotta Adapt'em All: Joint Pixel and Feature-level Domain Adaptation for Recognition in the Wild. In *Proc. CVPR*, 2019. 2

[70] Maxime Tremblay, Shirsendu S. Halder, Raoul de Charette, and Jean-François Lalonde. Rain Rendering for Evaluating and Improving Robustness to Bad Weather. *IJCV*, 2020. 5

[71] Gido M van de Ven and Andreas S Tolias. Three Scenarios for Continual Learning. In *NeurIPS Workshops*, 2018. 2, 3

[72] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully Test-time Adaptation by Entropy Minimization. In *Proc. ICLR*, 2020. 3, 4

[73] Yuxin Wu and Justin Johnson. Rethinking "Batch" in BatchNorm. *arXiv preprint arXiv:2105.07576*, 2021. 4

[74] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated Residual Transformations for Deep Neural Networks. In *Proc. CVPR*, 2017. 4

[75] Ju Xu and Zhanxing Zhu. Reinforced Continual Learning. In *NeurIPS*, 2018. 2

[76] Wending Yan, Robby T Tan, and Dengxin Dai. Nighttime Defogging using High-low Frequency Decomposition and Grayscale-color Networks. In *Proc. ECCV*, 2020. 3

[77] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual Learning through Synaptic Intelligence. In *Proc. ICML*, 2017. 2

[78] Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C-C Jay Kuo. Class-Incremental Learning via Deep Model Consolidation. In *Proc. WACV*, 2020. 2

[79] M. Zhang, H. Marklund, N. Dhawan, A. Gupta, S. Levine, and C. Finn. Adaptive risk minimization: Learning to adapt to domain shift. In *NeurIPS*, 2021. 3, 4