Multi-Stream Cellular Test-Time Adaptation of Real-Time Models Evolving in Dynamic Environments

Benoît Gérin[1,*] Anaïs Halin[2,*] Anthony Cioppa[2,3,*] Maxim Henry[2]
Bernard Ghanem[3] Benoît Macq[1] Christophe De Vleeschouwer[1] Marc Van Droogenbroeck[2]
[1] UCLouvain [2] ULiège [3] KAUST
benoit.gerin@uclouvain.be, anais.halin@uliege.be, anthony.cioppa@uliege.be

Abstract

In the era of the Internet of Things (IoT), objects connect through a dynamic network, empowered by technologies like 5G, enabling real-time data sharing. However, smart objects, notably autonomous vehicles, face challenges in critical local computations due to limited resources. Lightweight AI models offer a solution but struggle with diverse data distributions. To address this limitation, we propose a novel Multi-Stream Cellular Test-Time Adaptation *(MSC-TTA) setup where models adapt on the fly to a dynamic environment divided into cells. Then, we propose a real-time adaptive student-teacher method that leverages the multiple streams available in each cell to quickly adapt to changing data distributions. We validate our methodology in the context of autonomous vehicles navigating across cells defined based on location and weather conditions. To facilitate future benchmarking, we release a new multi-stream large-scale synthetic semantic segmentation dataset, called* DADE*, and show that our multi-stream approach outperforms a single-stream baseline. We believe that our work will open research opportunities in the IoT and 5G eras, offering solutions for real-time model adaptation.*

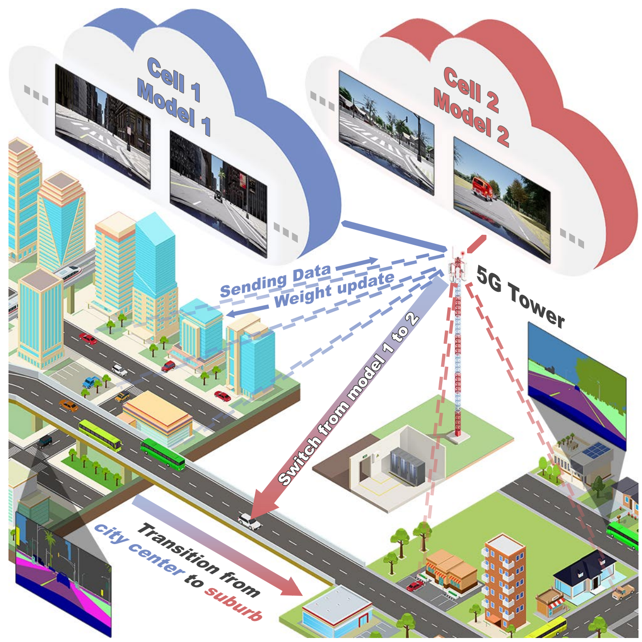

Figure 1. **Multi-Stream Cellular Test-time Adaptation (MSC-TTA) of real-time models.** We consider a set of agents (*e.g.*, autonomous vehicles) evolving in a dynamic environment divided into cells (*e.g.*, city center or suburb) that perform the same task (*e.g.*, semantic segmentation) in real time on their own unlabeled data stream (*e.g.*, recorded images) using an on-board model. We propose a first method in which agents share part of their data stream through an IoT network (*e.g.*, a connection to a 5G tower). Cell-based lightweight models are then trained on the fly (in our case through an adaptive student-teacher method) and their weights are regularly broadcasted to the agents to improve their performance over time. When agents transitions between cells, the agent's model is immediately switched to the one of the new cell, effectively adapting the predictions of the transiting agent.

1. Introduction

In the contemporary digital era, inanimate objects have gained the capability to connect and engage with each other via the Internet. This phenomenon has given rise to a dynamic network of interconnected objects known as the *Internet of Things* (IoT). This revolution is further driven by the advent of telecommunication technologies such as 5G, offering remarkable bandwidth ranging from 100MB to 1GB per second and a mere 10 millisecond latency [21]. This new larger bandwidth offers unprecedented opportunities for smart objects, especially those relying on computer vision for autonomous navigation, allowing real-time sharing of recorded images or videos through the network.

However, some critical computations need to be per-

(*) Equal contributions. Code and data available at github.com/ULiege-driving/MSC-TTA and github.com/ULiege-driving/DADE.

formed locally. For instance, autonomous vehicles should analyze their environment and take appropriate actions despite a loss of connection to the network. This forces smart objects to include an on-board processing unit, especially for autonomous navigation. Due to limitations on battery capacities, these processing units are often limited in their computation capabilities. Furthermore, the entire processing power can not just be dedicated to the analysis of the environment, but also needs to ensure all other critical functions (*e.g*., risk assessment, navigation system, *etc*.).

Considering these limitations, the deployment of lightweight artificial intelligence models analyzing the environment becomes essential. Lightweight models offer the advantage of high inference speed, meeting the real-time constraint, and low power consumption. However, their representational capacity is limited [15] compared to larger state-of-the-art model, failing in effectively handling a wide range of data distributions or generalizing to unseen environments at test time [6]. In the case of moving objects such as autonomous vehicles, data distribution shifts generally occur as the vehicles transition between different areas. Even in the case of static objects such as surveillance cameras, the encountered data distribution may be dynamic, due, for example, to changes in weather conditions or object density and occlusions. Fortunately, autonomous vehicles are able to precisely position themselves through a combination of navigation systems, telecommunications, and sensors (such as IMUs). It is therefore possible to get prior knowledge on the encountered data distribution such as location (*e.g*., the region of the world, the city, or the neighbourhood) or local weather and traffic information. Hence, the environment can be divided into a set of *cells*, representing different dynamic data distributions. Also, multiple objects evolving in similar environments (*e.g*., a fleet of autonomous vehicles) sense the environment and collect *multiple streams*, allowing to sample the changing data distributions within each cell more quickly and comprehensively.

In this work, we propose a first *Multi-Stream Cellular Test-Time Adaptation (MSC-TTA)* setup in which a fleet of connected objects, called *agents*, adapt on the fly their model to their data stream with distribution shifts. Then, we propose a first real-time method on our MSC-TTA setup based on an adaptive student-teacher online training strategy that leverages the division of the environment into different cells. Specifically, each agent analyses its own data stream on board using a lightweight student model and offloads the heavy teacher inference process and student training remotely (*e.g*., on the cloud). As shown in Figure 1, data is collected by all agents and aggregated to train specialized student models for each cell. Finally, we study our new MSC-TTA setup in the practical real-world case of autonomous cars evolving in dynamic environments divided in different cells based on location (*e.g*., urban, suburbs, countryside, *etc*.) and weather conditions (*e.g*., sunny, rainy, foggy). To support our experiments and allow future benchmarking, we generate and publicly release a new large-scale synthetic semantic segmentation dataset based on the CARLA simulator [9] called *DADE*, and show improved performance of our proposed multi-stream and cell-based method over a single stream baseline. We believe that this new multi-stream cellular test-time adaptation setup will open research possibilities for the combined use of computer vision and machine learning technologies in the IoT and 5G eras, as provisioned in 5G roadmaps [29].

Contributions. We summarize our contributions as follows. **(i)** We define a new Multi-Stream Cellular Test-Time Adaptation (MSC-TTA) setup in which models adapt on the fly to a dynamic environment divided into cells. **(ii)** We propose a novel real-time adaptive student-teacher method to aggregate knowledge across different agents evolving in the same cell. **(iii)** We generate and release a new synthetic dataset, called *DADE*, for the semantic segmentation task on board autonomous vehicles and show improved performance of our proposed method compared to the baseline.

2. Related Work

2.1. Online learning

Online learning is a well-studied setup [4, 16, 18, 35, 40] defined as a game between a learner and an environment generating a stream of data. Based on past and current data generated by the environment, the learner tries to sequentially predict labels on the stream. At each step, the true label of the data is revealed and compared to the prediction of the learner. The learner then receives a regret score, used to penalize its mistakes. The learner's objective is thus to minimize future penalties by using previously observed data and labels. The field of online learning can count on multiple benchmarking datasets such as firehose [20] for language modeling and CLOC [3] and CLEAR [28] for image classification of objects whose representations evolved over the span of 10 years. In practice, online learning is relevant when the true label becomes available as time goes by, *e.g*., for the task of forecasting [30, 46, 50]. In this work, we assess an upper bound of our MSC-TTA method by extending the setup to multi-stream cellular online learning.

2.2. Test-time adaptation

Similarly, Test-Time Adaptation (TTA) aims to adapt a model on a data stream. However, the environment does not reveal the true label of previously observed data. Several setups, characterized by the data distribution of the stream, have been studied, such as Fully TTA [43], Continual TTA [44], Non-i.i.d TTA [14], or Practical TTA [52], in which the data stream contains distribution changes and correlated samples. These setups are suited for real-world applications, where the true labels are unavailable at test time. However, previous works only consider a single stream of data. In this work, we go further, by proposing a setup for multiple streams and introducing prior knowledge on cross-stream data distribution through the division of the environment into cells. In addition, our methodology brings a real-time aspect, a feature often overlooked in previous setups.

To leverage the information in the data stream, multiple methods have been developed [27]. Some works adapt the model's parameters by either fitting the batch normalization layers to the target domain [26, 33, 39], training the model with auxiliary tasks [8,42], or fine-tuning it using unsupervised objectives [37, 43, 53]. Some other works adapt the input data [12, 17, 22, 54] or weight the predictions of multiple models depending on the test distribution [10, 47]. However, few works ensure that the adaptation is real time.

In fact, in real-world applications, the model needs to adapt within limited time to leverage all samples of the data stream, due to finite computing capabilities. Alfara *et al*. [1] recently proposed an evaluation protocol to compare TTA methods under those constraints. To satisfy the real-time constraint, some works proposed a student-teacher architecture with a lightweight student model [6, 34]. Specifically, *ARTHuS* [6] proposed a first real-time method in which a lightweight student model is adapted on an unlabeled data stream at test time using pseudo-labels produced by a state-of-the-art but computation-expensive teacher model. The real-time constraint of the system is ensured by asynchronously processing the student and teacher inference and training at different frame rates. The fast lightweight student model therefore trains online on the changing data distributions using the teacher's slow predictions. However, in the case of rapid domain shifts, the student needs several batches to adapt. Houyon *et al*. [19] later tackled this issue by incorporating continual learning methods in the student online training to avoid catastrophic forgetting in the case of cyclic domain shifts. Nevertheless, in the case of multiple objects (*e.g*., autonomous vehicles), each data stream is treated independently. In this work, we extend ARTHuS [6] to multiple data streams and cell-divided environments.

2.3. Autonomous driving

Autonomous Vehicles (AVs) rely on advanced sensor arrays, high-resolution cameras and on-board computing power to perceive the environment and make informed decisions to navigate safely. Nowadays, perception is largely based on artificial intelligence and involves several computer vision tasks such as semantic segmentation [24,48,51, 55], object detection [2, 25] or depth estimation [5, 11, 13]. However, the road to fully self-driving cars remains challenging. For instance, it is still complex to operate AVs in diverse environments, such as varying weather conditions, traffic patterns and other unforeseen scenarios, and to process large amount of data while optimizing energy consumption in Electric Vehicles (EVs).

To adapt to several environments, some methods use domain adaptation strategies [23, 36, 45] to enhance system versatility and reliability. Also, cloud computing [38] or multi-access edge computing (MEC) [31, 49] provide the computational power and storage capacity for real-time data processing, enhancing energy efficiency and improving EV mileage. Similarly to MEC, our proposed method employs a hybrid approach. On-board processing handles immediate, low-latency operations, while resource-intensive computations are offloaded to external servers. Specifically, the heavy offloaded computations rely on the multiple streams of the fleet, while on-board, lightweight real-time perception is performed using models trained in the cloud, guaranteeing adaptability in dynamic environments.

3. Methodology

3.1. Problem statement

Given a finite set of N agents a_n forming a connected fleet $\mathbf{A}$, our proposed Multi-Stream Cellular Test-Time Adaptation (MSC-TTA) aims to adapt over time $t \in \{0, ..., T\}$ each agent's model $f^t_{a_n} \in \mathbf{F}$, pretrained on any source domain to perform a task τ, to the agent's data stream $\mathcal{X}_{a_n} \in \mathbf{X}$ of online unlabeled samples $\mathcal{X}_{a_n} = x^0_{a_n}, x^1_{a_n}, ..., x^t_{a_n}, ..., x^T_{a_n}$. As in [52], the samples are drawn from a distribution $\mathcal{P}_{a_n} \in \mathbf{P}$ shifting over time following $\mathcal{P}^0_{a_n}, \mathcal{P}^1_{a_n}, ..., \mathcal{P}^t_{a_n}, ..., \mathcal{P}^T_{a_n}$, in which consecutive samples $x^{t-1}_{a_n}, x^t_{a_n}, x^{t+1}_{a_n}$ may be highly correlated. At time t, the model $f^t_{a_n}$ receives a batch of unlabeled samples $\mathcal{B}^t_{a_n} = x^t_{a_n}, x^{t+1}_{a_n}, ..., x^{t+(b-1)}_{a_n}$, where b is the batch size, on which it makes predictions. Each model $f^t_{a_n}$ may be adapted to the current batch $\mathcal{B}^t_{a_n}$ by accumulating knowledge from previous samples of the multiple streams forming the following hyperspace $\cup x^{t'}_{a_n}, \forall a_n \in \mathbf{A} \times t' < (t+b)$. Let us note that samples $x^t_{a_n}$ may be unavailable for some time t for some agent a_n. This setup describes the general case of multiple sensors recording data streams and performing the same task, *e.g*., surveillance cameras placed in one or several cities on which crowd counting or car segmentation needs to be performed, with no assumptions on where the cameras are placed.

To include cross-stream prior knowledge on data distributions, we consider the general case in which the agents evolve inside a dynamic environment split into a non-overlapping set of C cells $c \in \mathbf{E}$. We suppose that, at time t, each agent is located within one cell such that $e^t_{a_n} = c \in \mathbf{E}$, with agents being able to transition between cells over time. The cells c are predefined by a set of rules (*e.g*., based on the location, the weather, etc.) such that the expected data distribution of agents evolving in the same cell is similar, *i.e*., $\mathcal{P}^t_{a_n} \approx \mathcal{P}^t_{a_m}$ if $e^t_{a_n} = e^t_{a_m}$. Our setup therefore allows

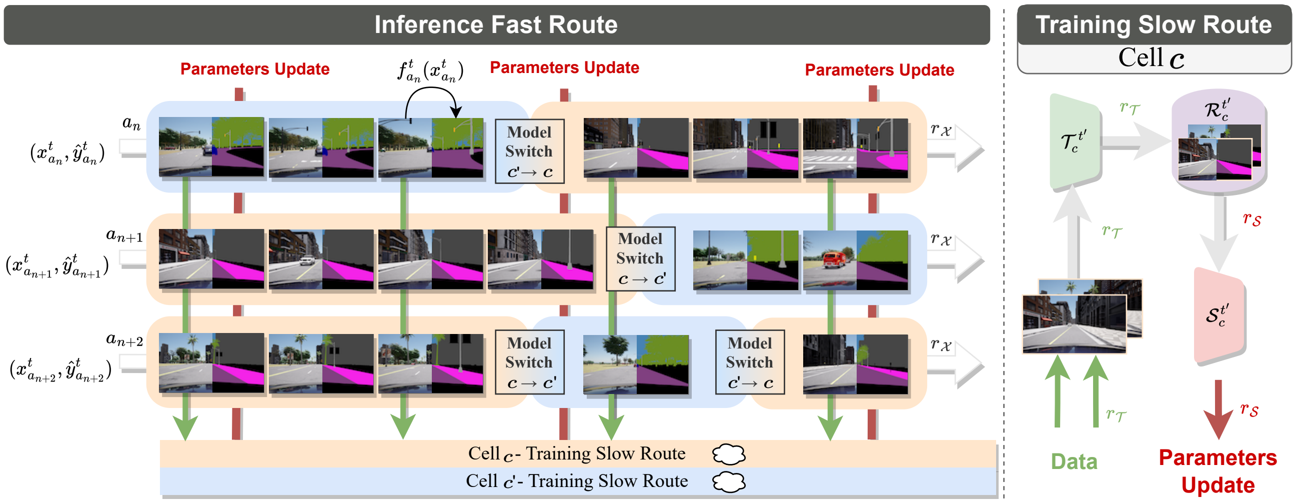

Figure 2. **Pipeline of our multi-stream cellular test-time adaptation of real-time models.** Our method is composed of a fast route for inference and a slow route for online training, as defined in [6, 19]. In the fast route, each agent a_n processes a stream of data samples $x^t_{a_n}$ and predicts labels $\hat{y}^t_{a_n} = f^t_{a_n}(x^t_{a_n})$ in real time (*i.e.*, at the data stream rate $r_{\mathcal{X}}$). Agents located within a cell c send a subset of their data samples at a slower rate $r_{\mathcal{T}}$) to a slow route operating on a remote server (*e.g.*, on the cloud) dedicated for each cell. In the slow route, a teacher model $\mathcal{T}_c^{t'}$ predicts pseudo labels on the received data and stores them in a replay buffer $\mathcal{R}_c^{t'}$. The replay buffer is then used to train on the fly a cell-specific student model $\mathcal{S}_c^{t'}$ at a rate $r_{\mathcal{S}}$. After each training epoch on the replay buffer, the parameters of $\mathcal{S}_c$ are transferred to all agent models f_{a_n} located within that cell. Finally, agents transiting between two cells have their model switched instantly.

the different data streams to share common data distribution properties at times that can be leveraged to effectively adapt the models. Naturally, in practice, this assumption may fail if the cells are incorrectly defined or estimated. This setup is particularly interesting in the real-world case of autonomous driving, in which vehicles evolve in different locations (*e.g.*, city centers, suburbs, highway, *etc.*) that they analyze through various sensors. Also, vehicles driving in the same environment may leverage the multiple streams of the fleet to better assess and adapt to the environment.

Let us note that considering the special case of $N = 1$ and $C = 1$ falls back to the original PTTA setup of Yuan *et al.* [52] in which a single model is adapted to its data stream. The case of $N \geq 1$ and $C = 1$ represents a Multi-Stream Test-Time Adaptation (MS-TTA) setup without division of the environment. Finally, the case of $C = 1$ and $f^t_{a_n} = f^t$, *i.e.*, in which a single model is adapted for all streams without prior knowledge on the environments, corresponds to a TTA setup in which samples from multiple streams are combined in the batch. In the following, we describe our adaptive method for the general case $N \geq 1$ and $C \geq 1$. To stay close to a real-world scenario, we add an extra real-time constraint on the method, *i.e.*, no delay accumulation or sample skipping when processing the multiple data streams.

3.2. Multi-stream cellular test-time adaptation

Our method, illustrated in Figure 2, produces a stream of predictions for every agent following $\hat{y}^t_{a_n} = f^t_{a_n}(x^t_{a_n})$, with the model $f^t_{a_n}$ operating in real time (*i.e.*, at the rate $r_{\mathcal{X}}$) on the data stream $\mathcal{X}_{a_n}$. To do so, we extend the adaptive real-time student-teacher method, ARTHuS, of Cioppa *et al.* [6], in which a lightweight student model $\mathcal{S}$ is adapted on-the-fly using pseudo labels produced by a state-of-the-art but computation-expensive teacher model $\mathcal{T}$. Particularly, we leverage the multiple streams and the division of the environment into cells. We allow agents evolving within the same cell to share their own data stream to produce a cell-specific data stream $\mathcal{X}_c^{t'} = \cup x^{t'}_{a_n}, \forall a_n \mid e^{t'}_{a_n} = c$ at a frame rate $r_{\mathcal{T}}$, producing samples $x_c^{t'}$.

Our method is composed of a fast route and a slow route. In the fast route (inference), student models for each agent produce predictions $\hat{y}^t_{a_n} = \mathcal{S}^t_{a_n}(x^t_{a_n}) = f^t_{a_n}(x^t_{a_n})$ at the rate $r_{\mathcal{X}}$. In parallel in the slow route (training), a slow but high-performance teacher model $\mathcal{T}_c^{t'}$ for each cell produces pseudo-ground truths $\tilde{y}_c^{t'} = \mathcal{T}_c^{t'}(x_c^{t'})$ at the rate $r_{\mathcal{T}}$ on the cell data streams. The pair of data $(x_c^{t'}, \tilde{y}_c^{t'})$ are then stored in a replay buffer $\mathcal{R}_c^{t'}$ of size R using a First-In-First-Out (FIFO) strategy. One student network per cell $\mathcal{S}_c^{t'}$ is trained on the updated replay buffer $\mathcal{R}_c^{t'}$ using a loss function

$$\mathcal{L} = \sum_{i=1}^{R} L(\mathcal{S}_c^{t'}(x_c^i), \tilde{y}_c^i), \tag{1}$$

where L is a dissimilarity measure suited for task τ. After training for one epoch on the replay buffer, the weights of students in the fast route are updated with the weight of

Table 1. **Mean IoU performance on DADE-static weather.** The MSC-OL setup leverages the CARLA segmentation masks as pseudo labels while the MSC-TTA setup leverages pseudo labels from the teacher model. We compare several pretraining, scenarios, and adaptive (✓) versus frozen (❄) models. For each pretraining, the best score is shown in **bold** and the second is underlined.

			Multi-stream cellular online learning				Multi-stream cellular test-time adaptation			
			mIoU imminent		mIoU future		mIoU imminent		mIoU future	
Pretraining	**Scenario**	**Adapt**	3 hours	Last hour	3 hours	Last hour	3 hours	Last hour	3 hours	Last hour
Cityscapes [7]	Student	❄	.214	.218	.214	.218	.214	.218	.214	.218
	Teacher	❄	.668	.671	.668	.671	.668	.671	.668	.671
Scratch	*Baseline* [6]	✓	.223	.249	.208	.231	.274	.309	.244	.285
	Baseline+MIR [19]	✓	.173	.194	.164	.188	.181	.201	.171	.195
	Common	✓	.338	.483	.316	.461	.340	.363	.327	.373
	Spatial	✓	**.353**	**.513**	**.328**	**.485**	**.368**	**.440**	**.351**	**.413**
General	*Baseline* [6]	✓	.435	.442	.415	.446	.422	.442	.397	.425
	Baseline+MIR [19]	✓	.650	.656	.614	.626	.417	.432	.401	.423
	Common	✓	**.702**	.696	**.673**	.692	**.474**	**.517**	.461	.501
	Spatial	✓	.700	**.701**	.660	**.701**	.470	**.517**	**.462**	**.505**
	Common	❄	.650	.658	.650	.658	.454	.450	.454	.450
Cell	*Spatial*	✓	**.658**	**.681**	.597	**.682**	**.552**	.567	.522	.556
	Spatial	❄	.634	.660	**.634**	.660	.544	**.572**	**.544**	**.572**

Figure 3. **Images of the different locations in our dataset.** We define 7 different locations that are defined based on the GNSS data. From left to right: forest, countryside, rural farmland, highway, low density residential, community buildings, and high density residential.

the environment students in the slow route such that $\mathcal{S}^t_{a_n} = \mathcal{S}^t_c, \forall a_n | e^t_{a_n} = c$, at a slower rate $r_{\mathcal{S}}$. Since the slow route gathers information from several agents, the heavy teacher inference and student training processes can be offloaded to a dedicated server (*e.g.*, on the cloud). Hence, agents only perform the real-time inference with a lightweight model, greatly reducing computation requirements and saving precious battery power in the case of autonomous vehicles. Finally, considering the special case $C = N$ with each agent defining its own cell is equivalent to the original ARTHuS method [6], serving as baseline in our experiments.

4. Experiments

4.1. Dataset

To support our experiments, we generate and release the Driving Agents in Dynamic Environments (DADE) dataset, based on the CARLA simulator. DADE is tailored for the online training and evaluation of semantic segmentation methods in the context of autonomous driving agents navigating dynamic environments The first part of DADE contains 100 video sequences of agents evolving in 7 connected locations illustrated in Figure 3, with static weather conditions (clear day). The second part contains 300 video sequences in the same locations with dynamic weather conditions (clear, rainy, and foggy), during day and night. We provide video sequences, semantic segmentation masks, Global Navigation Satellite System (GNSS) data, and weather information. Each sequence is acquired by an agent within a 5-hours time frame. The first two hours are used for pretraining and the remaining three for adaptation.

To the best of our knowledge, our dataset, large of 150 GBytes, is the first to provide long videos of multiple agents evolving in diverse driving locations and weather conditions with ground truth labels for the task of semantic segmentation. Our video sequences contain between 188 and 7,200 frames acquired at 1 frame per second (fps), with an average sequence length of 40 minutes. Existing datasets, such as [7,32,41], feature short video sequences, lack multi-agent perspectives, do not include ground truth data, or lack a diverse range of weather conditions. More information about our DADE dataset may be found in the appendix.

4.2. Experimental settings

4.2.1 Environment division

We consider six scenarios based on the division of the environment into cells. (1) The *Baseline* scenarios correspond to multiple independent streams on which independent agents adapt ($C = N$), *i.e.*, ARTHuS [6] and

Table 2. **Mean IoU performance on DADE-dynamic weather.** The MSC-OL setup leverages the CARLA segmentation masks as pseudo labels while the MSC-TTA setup leverages pseudo labels from the teacher model. We compare several pretraining, scenarios, and adaptive (✓) versus frozen (❄) models. For each pretraining, the best score is shown in **bold** and the second is underlined.

			Multi-stream cellular online learning				Multi-stream cellular test-time adaptation			
			mIoU imminent		mIoU future		mIoU imminent		mIoU future	
Pretraining	**Scenario**	**Adapt**	3 hours	Last hour	3 hours	Last hour	3 hours	Last hour	3 hours	Last hour
Cityscapes [7]	Student	❄	.159	.130	.159	.130	.159	.130	.159	.130
	Teacher	❄	.611	.542	.611	.542	.611	.542	.611	.542
Scratch	*Baseline* [6]	✓	.204	.197	.167	.167	.212	.190	.173	.173
	Baseline+MIR [19]	✓	.144	.137	.125	.118	.147	.133	.129	.110
	Common	✓	.278	.352	.249	.323	.278	.257	.253	.243
	Spatial	✓	**.307**	**.397**	**.269**	**.358**	**.312**	**.300**	**.278**	**.276**
	Weather	✓	.226	.295	.199	.279	.227	.216	.202	.197
	Daylight	✓	.245	.279	.176	.259	.182	.198	.150	.184
	Specific	✓	.22	.204	.203	.187	.233	.186	.218	.166
General	*Baseline* [6]	✓	.581	.546	.502	.502	.471	.406	.409	.409
	Baseline+MIR [19]	✓	.567	.531	.527	.480	.455	.386	.427	.347
	Common	✓	.644	.595	.613	.565	.506	.427	.483	.405
	Spatial	✓	**.654**	**.622**	.606	**.589**	**.516**	**.442**	.473	.405
	Weather	✓	.641	.586	.611	.562	.507	.429	**.484**	.408
	Daylight	✓	.636	.603	.572	.585	.498	.430	.477	**.413**
	Specific	✓	.632	.602	.596	.559	.500	.437	.471	.393
	Common	❄	.618	.581	**.618**	.581	.476	.403	.476	.403
Cell	*Spatial*	✓	**.662**	**.642**	.609	.590	**.527**	**.461**	.484	**.423**
	Weather	✓	.634	.580	.607	.551	.509	.427	.483	.409
	Daylight	✓	.645	.592	.620	.577	.507	.432	**.488**	.415
	Specific	✓	.612	.582	.589	.554	.500	.438	.485	.412
	Spatial	❄	.642	.606	**.642**	**.606**	.488	.409	**.488**	.409
	Weather	❄	.565	.528	.565	.528	.443	.384	.443	.384
	Daylight	❄	.563	.485	.563	.485	.421	.362	.421	.362
	Specific	❄	.447	.400	.447	.400	.349	.298	.349	.298

Houyon *et al*. [19]. (2) The *Common* scenario aggregates the multiple data streams into a single stream, on which one common model adapts ($C = 1$). (3) The *Spatial* scenario leverages the different locations of our dataset to split the environment into cells ($C = 7$). (4) The *Weather* and (5) *Daylight* scenarios temporally divide the environment based on the weather ($C = 3$: clear, rainy, foggy) and the time period ($C = 2$: day, night). (6) The *Specific* scenario considers each combination of location, weather condition, and time period ($C = 42$).

4.2.2 Pretraining

We choose the same model architecture for all agent models f_{a_n} and cell-specific student models $\mathcal{S}_c$. Following the work of Cioppa *et al*. [6], we select TinyNet: a lightweight semantic segmentation model operating in real time. The pretraining set is divided into a training set and a validation set using a 90-10% split. For each scenario, we evaluate 3 pretraining techniques. The *General* pretraining consists in training the student model on all samples of the training set, irrespective of the division into cells. The *Cell* pretraining considers a separate model for each cell c, trained on cell-specific samples. Finally, *Scratch* assigns random weights (*i.e*., no pretraining). The models are pretrained with a learning rate of 10^{-4} using the Adam optimizer and the best performing model over the validation set is selected. The number of epochs is set to 3 for the *General* pretraining and scaled for each cell for the *Cell* pretraining to match the number of backward passes and ensure a fair comparison.

4.2.3 Testing

For a given scenario and pretraining procedure, we compare the online performance (*i.e*., our adaptive method) with the offline performance (*i.e*., a frozen pretrained model). We choose the teacher model as a frozen state-of-the-art SegFormer [48] model trained on Cityscapes [7] that produces pseudo labels at a rate of $r_{\mathcal{T}} = 1/3$ [Hz]. The replay buffers are chosen as FIFO buffers with a size $R = 100$, updated at the same rate $r_{\mathcal{T}}$. Finally, the cell-specific student models are trained online at a rate $r_{\mathcal{S}} = 1/30$ [Hz], with a learn-

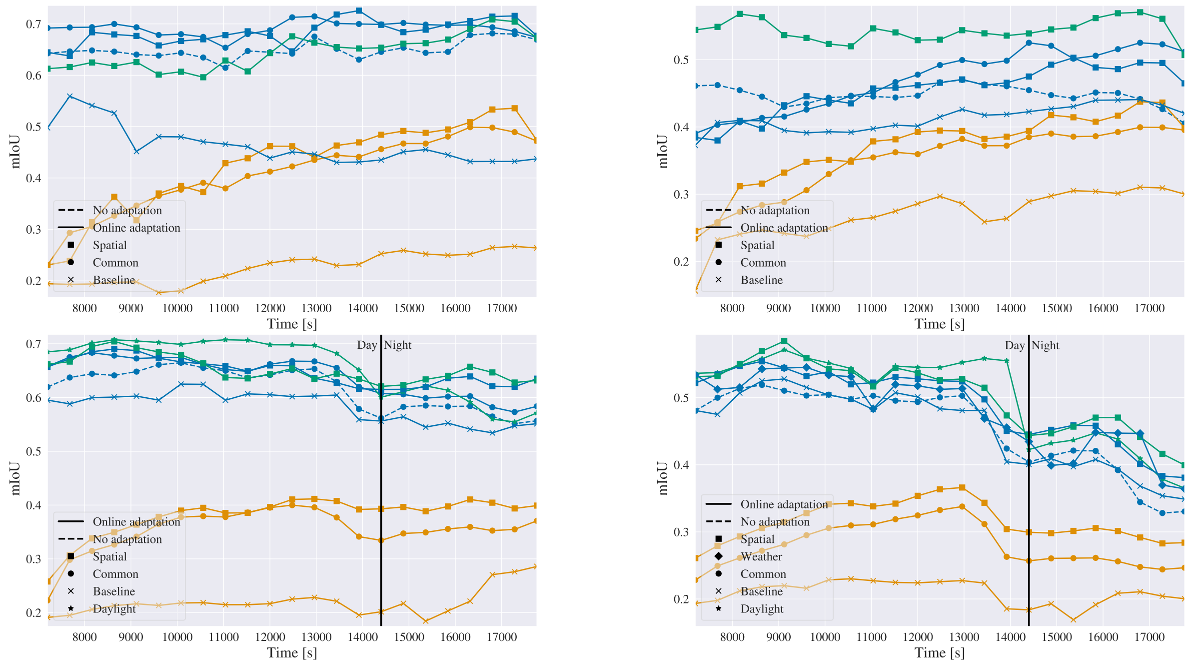

Figure 4. **Evolution of the fleet performance over time on DADE-static weather (top) and DADE-dynamic weather (bottom).** Comparison of the performance in the MSC-OL setup (left) and the MSC-TTA (right) setup of the best adaptive settings along with the baseline for each pretraining (*Scratch*, *General*, and *Cell*).

ing rate of 10^{-4}, batch size of 25 with the Adam optimizer, and the cross-entropy loss. The model is only trained if the buffer contains new samples to prevent overfitting.

For the online evaluation, we aggregate the confusion matrices over a sliding window of 30 [s] (imminent performance) for every agent in every cell and compute the mean-Intersection-over-Union (mIoU) as defined in Houyon *et al*. [19]. Additionally, we propose to evaluate the current model 5 minutes in the future (future performance) to assess the capacity of the model to generalize to future samples. Finally, we also compute the overall mIoU for the entire test set (3 hours) and for the last hour to assess the long-term performance. As an upper bound, we also evaluate our method in a Multi-Stream Cellular Online Learning (MSC-OL) setup by replacing the teacher pseudo labels by the true ground-truth labels. We also compare our approach to the best method proposed in Houyon *et al*. [19], which is equivalent to our *Baseline* with a Maximal Interfered Retrieval (MIR) buffer, and report the offline performances of the frozen teacher and student models both trained on Cityscapes [7]. More information on the evaluation, such as details about the object classes, is provided in the appendix.

4.3. Results

4.3.1 Quantitative performances

Table 1 shows the mean performance of the fleet in the different settings on DADE-static. We observe that the baseline setups [6, 19] are outperformed by our method for every scenario and pretraining, highlighting the benefits of using multiple streams when adapting the models. For no pretraining (*Scratch*), the *Spatial* division of the environment leads to the best results, indicating that leveraging cellular information improves the models. For *General* pretraining, the adapted *Common* and *Spatial* scenarios show better performance than the frozen pretrained model, highlighting the benefits of adapting the model online. In the MSC-TTA setup, the *Cell* pretraining outperforms the *General* one while it is the opposite in the MSC-OL setup, indicating that clean generic labels compensate for cell-specific ones.

We also provide the mean performance on DADE-dynamic in Table 2. As can be seen, our method still outperforms the baselines. Again, from *Scratch*, the *Spatial* scenario brings the best results, followed by the *Common* scenario. However, temporal divisions such as *Weather*, *Daylight*, and *Specific* lead to lower performances. While DADE includes at least one vehicle in almost every location over time, the same weather and daylight are applied to all locations simultaneously, leading to discontinuities in the availability of samples for time-based cells. This temporarily stops the adaptation and slows down model convergence. Longer sequences, would allow the models to better explore those cells. Finally, the *Cell* pretraining shows the best overall performance for the MSC-OL/TTA setups, showing the advantage of dividing the environment into cells.

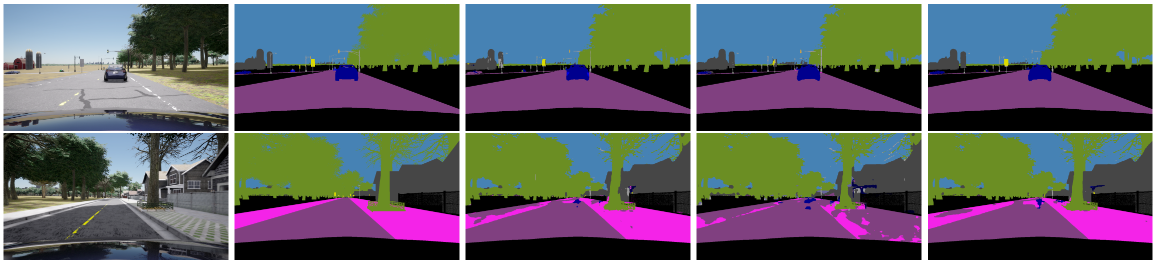

Figure 5. **Qualitative results.** Comparison of different segmentation masks. From left to right: RGB image, ground truth, *Baseline*, *Common* scenario with *General* pretraining, and *Spatial* scenario with *Cell* pretraining. Black areas correspond to non-evaluated classes.

4.3.2 Evolution of the fleet performances

The evolution of the fleet performance over time is shown in Figure 4. For visualization purposes, we aggregate the confusion matrices in sliding windows of 8 minutes to compute the mIoU. Regarding the static weather (top row), the *Baseline* is outperformed by all settings of our method. Interestingly, even if the baseline starts from pretrained weights and our method from scratch, we outperform the baseline in the MSC-OL setup and reach similar performances in the MSC-TTA setup. Additionally, *Cell* pretraining with the *Spatial* scenario reaches the best performance in the MSC-TTA setup for the whole duration, keeping steady performance. This is crucial for autonomous vehicles that need to operate similarly in all conditions.

We also show the performance for DADE-dynamic in Figure 4 (bottom row). For both the MSC-OL and MSC-TTA setups, the *Daylight* scenario with *Cell* pretraining produces the best performance before nightfall, after which it drops while other scenarios, such as *Spatial* with *Cell* pretraining, become better options. This is due to the fact that the night models are not updated before nightfall while the location-based models are constantly updated, during day, dusk, and night. Nevertheless, it can be seen that the performance drops regardless of the scenario or pretraining during nightfall, leaving room for improvement in future works.

4.3.3 Qualitative results

We qualitatively show the improvement of our multi-stream cellular method over the ARTHuS [6] baseline. To do so, we display in Figure 5 the segmentation masks predicted by our method in two scenarios: the *Common* scenario with *General* pretraining and the *Spatial* scenario with *Cell* pretraining, and compare them to the masks predicted by the *Baseline* and the ground truth labels. On the top row, we show a vehicle driving in the countryside under static (clear) weather at the end of the online training. We can see that the baseline confuses some building with poles and a car is misclassified as being part of the road, while our method is able to correctly segment it. The *Spatial* model produces the most accurate segmentation masks as it is able to precisely segment the city and vegetation in the background and the cars on the left. On the bottom row, we show a vehicle driving in the low density residential location under static (clear) weather also at the end of the online training. As can be seen, the *Common* model fails in this cell because it needs to learn a broader data distribution and looses accuracy due to its limited learning capacity. Contrarily, the *Spatial* model is able to better learn that particular cell data distribution and therefore produces the best results.

5. Conclusion

Our novel Multi-Stream Cellular Test-Time Adaptation (MSC-TTA) setup addresses multi-stream model adaptation in dynamic environments. We focus on environments where data distribution shifts pose significant challenges. To do so, we divide the environments into cells, characterized by similar conditions such as location and weather. Then, we propose a real-time method based on an adaptive student-teacher approach, leveraging the multiple streams and cellular information. Experimental validation on autonomous vehicles illustrates the benefits of our MSC-TTA setup, showcasing better performance compared to a single-stream baseline. Our novel DADE dataset supports our experiments and provides a comprehensive benchmark for future studies in test-time adaptation of semantic segmentation models for autonomous vehicles. This work represents a significant step forward in the field of test-time adaptation, holding promise for substantial contributions to IoT and autonomous driving.

Acknowledgments. B. Gérin and A. Halin are funded by the Walloon region under grant No. 2010235 (ARIAC by DIGITALWALLONIA4.AI). A. Cioppa is funded by the F.R.S.-FNRS. M. Henry is funded by PIT MecaTech under grant No. C8650 (ReconnAIssance). The present research benefited from computational resources made available on Lucia, infrastructure funded by the Walloon Region under grant No. 1910247.

References

[1] Motasem Alfarra, Hani Itani, Alejandro Pardo, Shyma Alhuwaider, Merey Ramazanova, Juan C. Pérez, Zhipeng Cai, Matthias Müller, and Bernard Ghanem. Revisiting test time adaptation under online evaluation. *arXiv*, abs/2304.04795, 2023. 3

[2] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal speed and accuracy of object detection. *arXiv*, abs/2004.10934, 2020. 3

[3] Zhipeng Cai, Ozan Sener, and Vladlen Koltun. Online continual learning with natural distribution shifts: An empirical study with visual data. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 8261–8270, Montréal, Can., Oct. 2021. Inst. Electr. Electron. Eng. (IEEE). 2

[4] Nicolò Cesa-Bianchi and Francesco Orabona. Online learning algorithms. *Annu. Rev. Stat. Its Appl.*, 8(1):165–190, Mar. 2021. 2

[5] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 5410–5418, Salt Lake City, UT, USA, Jun. 2018. Inst. Electr. Electron. Eng. (IEEE). 3

[6] Anthony Cioppa, Adrien Deliege, Maxime Istasse, Christophe De Vleeschouwer, and Marc Van Droogenbroeck. ARTHuS: Adaptive real-time human segmentation in sports through online distillation. In *IEEE Int. Conf. Comput. Vis. Pattern Recognit. Work. (CVPRW), CVsports*, pages 2505–2514, Long Beach, CA, USA, Jun. 2019. Inst. Electr. Electron. Eng. (IEEE). 2, 3, 4, 5, 6, 7, 8, 19

[7] 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 *IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3213–3223, Las Vegas, NV, USA, Jun. 2016. Inst. Electr. Electron. Eng. (IEEE). 5, 6, 7, 12, 19

[8] Antonio D'Innocente, Francesco Cappio Borlino, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. One-shot unsupervised cross-domain detection. In *Eur. Conf. Comput. Vis. (ECCV)*, volume 12361 of *Lect. Notes Comput. Sci.*, pages 732–748. Springer Int. Publ., 2020. 3

[9] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Annual Conference on Robot Learning*, volume 78 of *Proc. Mach. Learn. Res.*, pages 1–16, Mountain View, CA, USA, Nov. 2017. 2, 12

[10] Simon Föll, Alina Dubatovka, Eugen Ernst, Siu Lun Chau, Martin Maritsch, Patrik Okanovic, Gudrun Thäter, Joachim M. Buhmann, Felix Wortmann, and Krikamol Muandet. Gated domain units for multi-source domain generalization. *arXiv*, abs/2206.12444, 2022. 3

[11] Michaël Fonder, Damien Ernst, and Marc Van Droogenbroeck. Parallax inference for robust temporal monocular depth estimation in unstructured environments. *Sensors*, 22(23):1–22, Dec. 2022. 3

[12] Jin Gao, Jialing Zhang, Xihui Liu, Trevor Darrell, Evan Shelhamer, and Dequan Wang. Back to the source: Diffusion-driven adaptation to test-time corruption. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11786–11796, Vancouver, Can., Jun. 2023. Inst. Electr. Electron. Eng. (IEEE). 3

[13] Clement Godard, Oisin Mac Aodha, Michael Firman, and Gabriel Brostow. Digging into self-supervised monocular depth estimation. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 3827–3837, Seoul, South Korea, Oct. 2019. Inst. Electr. Electron. Eng. (IEEE). 3

[14] Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. NOTE: Robust continual test-time adaptation against temporal correlation. *arXiv*, abs/2208.05117, 2022. 2

[15] Ian J. Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT Press, Cambridge, MA, USA, 2016. 2

[16] Elad Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016. 2

[17] Yufan He, Aaron Carass, Lianrui Zuo, Blake E. Dewey, and Jerry L. Prince. Self domain adapted network. In *Int. Conf. Medical Image Comput. Comput. Interv.*, volume 12261 of *Lect. Notes Comput. Sci.*, pages 437–446. Springer Int. Publ., 2020. 3

[18] Steven C.H. Hoi, Doyen Sahoo, Jing Lu, and Peilin Zhao. Online learning: A comprehensive survey. *Neurocomputing*, 459:249–289, Oct. 2021. 2

[19] Joachim Houyon, Anthony Cioppa, Yasir Ghunaim, Motasem Alfarra, Anaïs Halin, Maxim Henry, Bernard Ghanem, and Marc Van Droogenbroeck. Online distillation with continual learning for cyclic domain shifts. In *IEEE International Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Workshop on Continual Learning in Computer Vision*, Vancouver, Canada, June 2023. IEEE. 3, 4, 5, 6, 7, 19

[20] Hexiang Hu, Ozan Sener, Fei Sha, and Vladlen Koltun. Drinking from a firehose: Continual learning with Web-scale natural language. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(5):5684–5696, May 2023. 2

[21] ITU. IMT vision – framework and overall objectives of the future development of IMT for 2020 and beyond. Technical report, Int. Telecommun. Union, Sept. 2015. Recommendation ITU-R M.2083-0. 1

[22] Neerav Karani, Ertunc Erdil, Krishna Chaitanya, and Ender Konukoglu. Test-time adaptable neural networks for robust medical image segmentation. *Medical Image Anal.*, 68:1–14, Feb. 2021. 3

[23] Mikhail Kennerley, Jian-Gang Wang, Bharadwaj Veeravalli, and Robby T. Tan. 2PCNet: Two-phase consistency training for day-to-night unsupervised domain adaptive object detection. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 11484–11493, Vancouver, Can., Jun. 2023. Inst. Electr. Electron. Eng. (IEEE). 3

[24] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. PointRend: Image segmentation as rendering. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9796–9805, Seattle, WA, USA, Jun. 2020. Inst. Electr. Electron. Eng. (IEEE). 3

[25] Peixuan Li and Jieyu Jin. Time3D: End-to-end joint monocular 3D object detection and tracking for autonomous driving. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3875–3884, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 3

[26] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. *arXiv*, abs/1603.04779, 2016. 3

[27] Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *arXiv*, abs/2303.15361, 2023. 3

[28] Zhiqiu Lin, Jia Shi, Deepak Pathak, and Deva Ramanan. The CLEAR benchmark: Continual LEArning on Real-world imagery. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2021. 2

[29] Thomas Linget. A visionary roadmap for advanced driving use cases, connectivity technologies, and radio spectrum needs. Technical report, 5G Automation Association (5GAA), Nov. 2022. 2

[30] Chenghao Liu, Steven C.H. Hoi, Peilin Zhao, and Jianling Sun. Online ARIMA algorithms for time series prediction. In *AAAI Conf. Artif. Intell.*, volume 30(1), pages 1867–1873. Association for the Advancement of Artificial Intelligence (AAAI), Feb. 2016. 2

[31] Shaoshan Liu, Liangkai Liu, Jie Tang, Bo Yu, Yifan Wang, and Weisong Shi. Edge computing for autonomous driving: Opportunities and challenges. *Proc. IEEE*, 107(8):1697–1716, Aug. 2019. 3

[32] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman. 1 year, 1000 km: The Oxford RobotCar dataset. *Int. J. Robot. Res.*, 36(1):3–15, Jan. 2017. 5, 12

[33] M. Jehanzeb Mirza, Jakub Micorek, Horst Possegger, and Horst Bischof. The norm must go on: Dynamic unsupervised domain adaptation by normalization. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 14745–14755, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 3

[34] Ravi Teja Mullapudi, Steven Chen, Keyi Zhang, Deva Ramanan, and Kayvon Fatahalian. Online model distillation for efficient video inference. In *IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 3572–3581, Seoul, South Korea, Oct. 2019. Inst. Electr. Electron. Eng. (IEEE). 3

[35] Francesco Orabona. A modern introduction to online learning. *arXiv*, abs/1912.13213, 2019. 2

[36] Sébastien Piérard, Anthony Cioppa, Anaïs Halin, Renaud Vandeghen, Maxime Zanella, Benoît Macq, Saïd Mahmoudi, and Marc Van Droogenbroeck. Mixture domain adaptation to improve semantic segmentation in real-world surveillance. In *IEEE/CVF Winter Conf. Appl. Comput. Vis. Work. (WACVW)*, pages 22–31, Waikoloa, HI, USA, Jan. 2023. Inst. Electr. Electron. Eng. (IEEE). 3

[37] Nikhil Reddy, Abhinav Singhal, Abhishek Kumar, Mahsa Baktashmotlagh, and Chetan Arora. Master of all: Simultaneous generalization of urban-scene segmentation to all adverse weather conditions. In *Eur. Conf. Comput. Vis. (ECCV)*, volume 13699 of *Lect. Notes Comput. Sci.*, pages 51–69. Springer Nat. Switz., 2022. 3

[38] Peter Schafhalter, Sukrit Kalra, Le Xu, Joseph E. Gonzalez, and Ion Stoica. Leveraging cloud computing to make autonomous vehicles safer. *arXiv*, abs/2308.03204, 2023. 3

[39] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 33, pages 11539–11551. Curran Associates, Inc., 2020. 3

[40] Shai Shalev-Shwartz. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2011. 2

[41] Tao Sun, Mattia Segu, Janis Postels, Yuxuan Wang, Luc Van Gool, Bernt Schiele, Federico Tombari, and Fisher Yu. SHIFT: A synthetic driving dataset for continuous multi-task domain adaptation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 21339–21350, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 5, 12

[42] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *Int. Conf. Mach. Learn. (ICML)*, volume 119 of *Proc. Mach. Learn. Res.*, pages 9229–9248, 2020. 3

[43] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *Int. Conf. Learn. Represent. (ICLR)*, May 2021. 2, 3

[44] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 7191–7201, New Orleans, LA, USA, Jun. 2022. Inst. Electr. Electron. Eng. (IEEE). 2

[45] Wei Wang, Zhun Zhong, Weijie Wang, Xi Chen, Charles Ling, Boyu Wang, and Nicu Sebe. Dynamically instance-guided adaptation: A backward-free approach for test-time domain adaptive semantic segmentation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 24090–24099, Vancouver, Can., Jun. 2023. Inst. Electr. Electron. Eng. (IEEE). 3

[46] Xinying Wang and Min Han. Online sequential extreme learning machine with kernels for nonstationary time series prediction. *Neurocomputing*, 145:90–97, Dec. 2014. 2

[47] Xinyi Wang, Yulia Tsvetkov, Sebastian Ruder, and Graham Neubig. Efficient test time adapter ensembling for low-resource language varieties. In *Findings of the Association for Computational Linguistics*, pages 730–737, Punta Cana, Dominican Republic, 2021. Assoc. Comput. Linguistics. 3

[48] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, volume 34, pages 12077–12090, 2021. 3, 6

[49] Bo Yang, Xuelin Cao, Kai Xiong, Chau Yuen, Yong Liang Guan, Supeng Leng, Lijun Qian, and Zhu Han. Edge intelligence for autonomous driving in 6g wireless system: Design challenges and solutions. *IEEE Wirel. Commun.*, 28(2):40–47, Apr. 2021. 3

[50] Haimin Yang, Zhisong Pan, and Qing Tao. Online learning for time series prediction of AR model with missing data. *Neural Process. Lett.*, 50(3):2247–2263, Feb. 2019. 2

[51] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. BiSeNet V2: Bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vis.*, 129(11):3051–3068, Sept. 2021. 3

[52] Longhui Yuan, Binhui Xie, and Shuang Li. Robust test-time adaptation in dynamic scenarios. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 15922–15932, Vancouver, BC, Canada, Jun. 2023. Inst. Electr. Electron. Eng. (IEEE). 2, 3, 4

[53] Marvin Zhang, Sergey Levine, and Chelsea Finn. MEMO: test time robustness via adaptation and augmentation. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, New Orleans, LA, USA, 2022. 3

[54] Xingchen Zhao, Chang Liu, Anthony Sicilia, Seong Jae Hwang, and Yun Fu. Test-time Fourier style calibration for domain generalization. In *Int. Jt. Conf. Artif. Intell. (IJCAI)*, pages 1721–1727, Vienna, Austria, Jul. 2022. 3

[55] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H.S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 6877–6886, Nashville, TN, USA, Jun. 2021. Inst. Electr. Electron. Eng. (IEEE). 3