Supplementary: AR-TTA: A Simple Method for Real-World Continual Test-Time Adaptation

Damian Sójka[1,2], Sebastian Cygert[2,3], Bartłomiej Twardowski[2,4,5], and Tomasz Trzciński[1,2,6]
[1]Poznań University of Technology, [2]IDEAS NCBR, [3]Gdańsk University of Technology, [4]Autonomous University of Barcelona, [5]Computer Vision Center, [6]Tooploox

This document describes our experimental procedure in detail and provides more results.

1. Experimental details

We test the method in a continual manner on every benchmark, which means that the methods continually adapt the models without the reset to the source state in between the domains, unless it is a part of a tested method, as proposed in [8].

1.1. SHIFT-C benchmark details

The SHIFT-C benchmark is created using the SHIFT dataset [5]. The dataset consists of multiple types of autonomous driving data from the CARLA Simulator [1]. We used RGB images from the front view of a car, discrete domain shifts, and bounding box annotations. More specifically, we download the required data with the script from SHIFT's website `https://www.vis.xyz/shift/`, using the following command:

```
python download.py --view "front" \
--group "[img, det_2d]" \
--split "[train, val]" \
--framerate "images" \
--shift "discrete" TARGET_DIR
```

To load the data for experiments, we utilized *shift-dev* repository: `https://github.com/SysCV/shift-dev`.

Following the CLAD-C benchmark [6], we create an image classification task by cutting out the bounding box annotations and using each of them as a separate data sample. Bounding boxes with fewer than 1024 pixels are discarded. We pad the images by their shortest axis (modify the aspect ratio to 1:1) and resize them to 32x32. Bounding boxes in the dataset are categorized into six classes, and so are the created images. Example images are displayed in Figure 1. We present a class distribution in Figure 2.

We distinguish between domains by the course annotations of time of day and weather. The source model is trained on images from train split, taken at daytime in clear weather. The TTA is also tested on data from the train split, but from different weather conditions and times of the day. Details about the size of each domain can be found in Table 1.

1.2. Compared TTA methods implementation details

Implementations of the compared methods were taken from their official code repositories. We use all hyperparameters and optimizers suggested by the papers or found in the code. We follow the standard model architectures used in TTA experiments and use WideResnet28 for CIFAR10C and ResNet50 for CLAD-C and SHIFT-C. Moreover, since we use a smaller batch size (BS) of 10 and benchmarks that have not been used before in TTA, we search for the optimal learning rate (LR) for each method. We focus on lowering the LR, considering the decreased batch size. Additionally, we search for the ϵ hyperparameter of EATA to correctly reject samples for adaptation. The results of the parameter search can be found in Table 2. The details and parameters used for each method are described below.

TENT [7] We use Adam optimizer with LR = 0.00003125 for every tested dataset. In the original paper, TENT uses LR = 0.001 for all the datasets except ImageNet, but it performed worse with this value on our setup.

CoTTA [8] Adam optimizer with LR = 0.00025 is used for every tested benchmark. The original implementation set LR to 0.001, but with an adjusted value, it achieved better results. We follow the suggestions for other hyperparameter values given by the authors. The restoration probability p is set to 0.01, the smoothing factor of the exponential moving average of teacher weights α is equal to 0.999, and

Figure 1. Example images sourced from various domains within the SHIFT-C benchmark.

Table 1. The number of samples in each domain in SHIFT-C benchmark

Domain nr	TIme of Day	Weather	Number of images
Source data	daytime	clean	57039
1		cloudy	41253
2		overcast	20497
3		rainy	59457
4		foggy	38590
5	dawn/dusk	clear	29543
6		cloudy	19985
7		overcast	9901
8		rainy	26677
9		foggy	20258
10	night	clear	28639
11		cloudy	18068
12		overcast	9471
13		rainy	32864
14		foggy	25464
Sum			437706

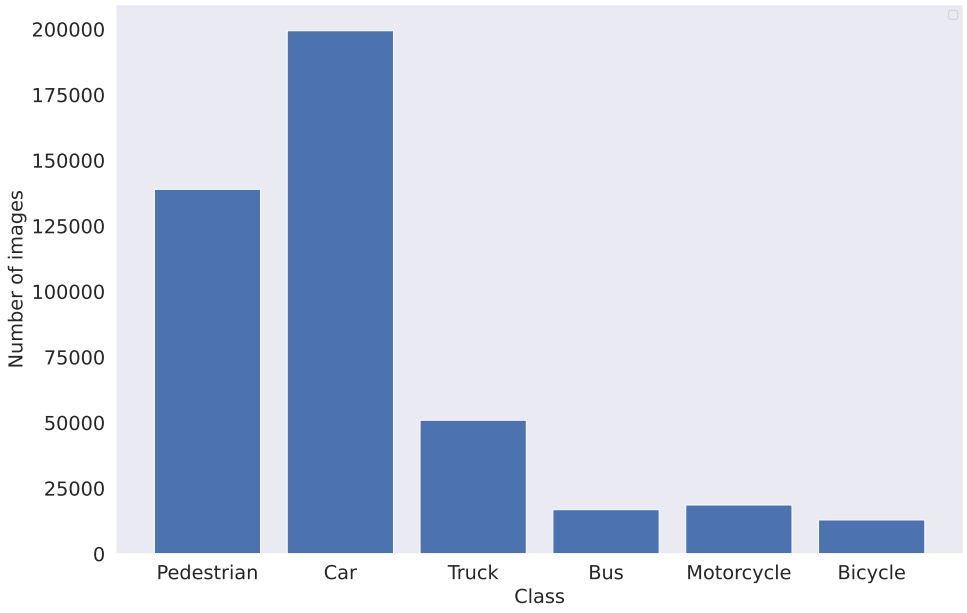

Figure 2. SHIFT-C benchmark class distribution.

the confidence threshold for applying augmentations p_{th} is set to 0.92.

EATA [2] We use the SGD optimizer with the momentum of 0.9 and LR of 0.00025 for CIFAR10C and CLAD-C, and 0.00003125 for SHIFT-C. The original EATA paper uses an LR value of 0.005/0.00025 for CIFAR10C/ImageNet, but they used BS = 64. After the search for the optimal ϵ parameter value for filtering redundant samples, we set it to 0.4/0.05/0.6 for CIFAR10C/CLAD-C/SHIFT. The authors used ϵ equal to 0.4/0.05 for CIFAR10C/ImageNet. The entropy constant E_0 is set to the standard value of $0.4\times \ln C$, where C was the number of classes, following the original paper and [3]. The trade-off parameter β is equal to 1, and 2000 samples are used to calculate the fisher importance of model weights as for the CIFAR10 dataset in the original paper.

SAR [3] SGD optimizer is used with the momentum of 0.9 and LR = 0.001 for CIFAR10C, and LR = 0.00025 for CLAD-C and SHIFT-C. It almost aligns with the authors' choice since, in original experiments, they used a learning rate equal to 0.00025/0.001 for ResNet/Vit models. The parameter E_0 is set to $0.4\times \ln C$, as in the paper, similarly to EATA. We follow the authors' choice of a constant reset threshold value e_0 of 0.2, and a moving average factor equal to 0.9. The radius parameter ρ is set to the default value of 0.05.

AR-TTA (Ours) We use SGD optimizer with momentum of 0.9 and LR of 0.001 for every dataset. The scale hyper-parameter γ is set to 0.1 for CLAD-C, and 10 for CIFAR10C and SHIFT-C. α value for weighting the exponential moving average of β is equal to 0.2. We set the initial β value to 0.1. The ψ parameter used for Beta distribution to sample λ for mixup is equal to the standard value of 0.4. We store 2000 of exemplars from source data for memory replay.

2. Proposed AR-TTA method analysis

The γ is a scale parameter of the distance between distributions $D(\phi^S, \phi_t^T)$. It determines the magnitude of the calculated values of β, which is used for linear interpolation between the saved source batch normalization (BN) statistics ϕ^S and the BN statistics calculated from the current batch ϕ_t^T. The higher the value of γ, the higher the values of β tend to be. At the same time, the higher the β values, the more influence BN statistics from current batch have on interpolation and calculation of the finally used BN statistics. In Figure 3 we show the relationship between γ parameter value and mean accuracy of our AR-TTA method for CIFAR10-to-CIFAR10C and CLAD-C benchmarks. We can see the contradicting trend between the two benchmarks. This suggests that the discrepancy in the data distribution between the source domain and the estimated distribution for each test data batch is more prominent in CIFAR10C compared to CLAD-C. This is in agreement with the results of the BN stats adapt [4] baseline method. BN stats adapt discards the BN statistics from the source data. Its performance was significantly better on CIFAR10C and worse on CLAD-C, compared to the fixed source model.

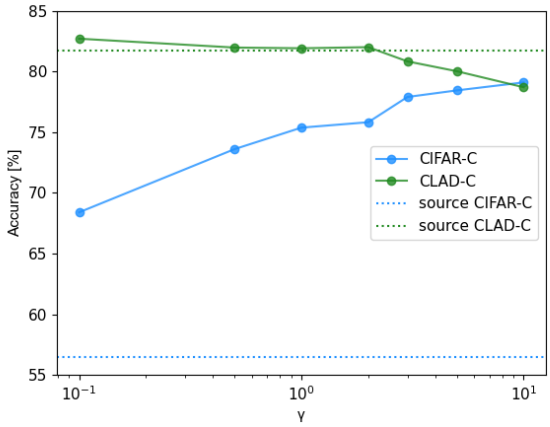

Figure 3. The relationship between mean classification accuracy (%) and the value of parameter γ for CIFAR10C and CLAD-C benchmarks.

3. Additional results

We present a batch-wise accuracy plots for CIFAR10C benchmark in Figure 6 and SHIFT-C benchmark in Figure 5.

Table 2. Mean classification accuracy (%) for CIFAR10C, CLAD-C, and SHIFT-C continual test-time adaptation task for compared state-of-the-art methods with different learning rates and EATA's ϵ parameter.

Method	learning rate	ϵ	Mean CIFAR10C	Mean CLAD-C	Mean SHIFT-C
CoTTA [8]	0.001	-	49.3	71.5	74.3
	0.00025	-	75.7	71.8	78.6
	0.00003125	-	74.5	71.8	76.2
TENT-continual [7]	0.001	-	24.3	64.4	63.4
	0.00025	-	72.3	71.0	75.3
	0.00003125	-	76.7	71.1	82.7
SAR [3]	0.001	-	75.2	70.6	86.0
	0.00025	-	75.1	70.6	86.0
	0.00003125	-	75.0	70.6	86.0
EATA [2]	0.001	0.60	68.6	70.1	80.4
	0.001	0.40	76.3	70.6	80.4
	0.001	0.10	75.0	70.6	86.0
	0.001	0.05	74.9	70.6	86.0
	0.00025	0.60	77.8	70.5	85.6
	0.00025	0.40	78.2	70.6	86.1
	0.00025	0.10	74.9	70.6	86.0
	0.00025	0.05	74.9	70.7	86.0
	0.00003125	0.60	76.5	70.6	86.1
	0.00003125	0.40	76.5	70.6	86.0
	0.00003125	0.10	74.9	70.6	86.0
	0.00003125	0.05	74.9	70.6	86.0

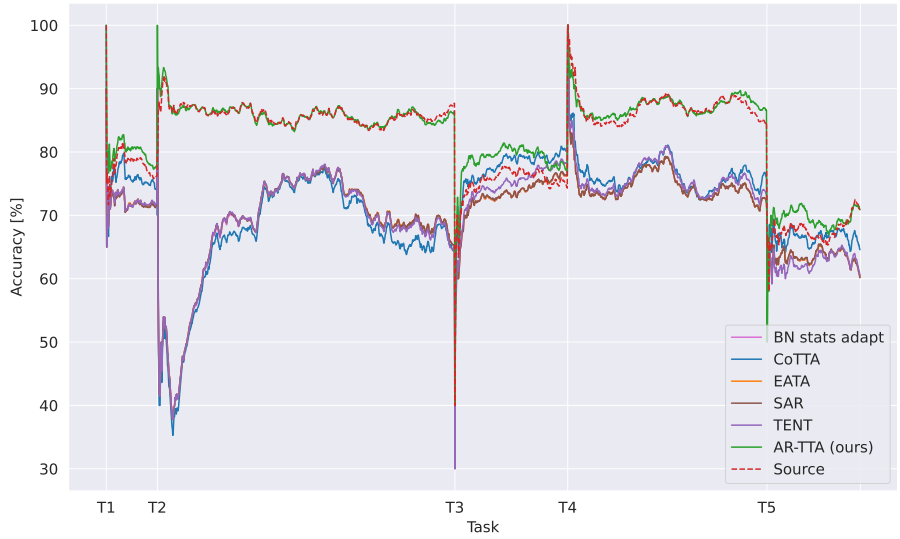

Figure 4. Batch-wise classification accuracy (%) averaged in a window of 100 batches on CLAD-C benchmark for the chosen methods continually adapted to the sequences of data. The ticks on the x-axis symbolize the beginning of the next sequence and, at the same time, a different domain. The window to calculate the average values is cleared in between the sequences. Best viewed in color.

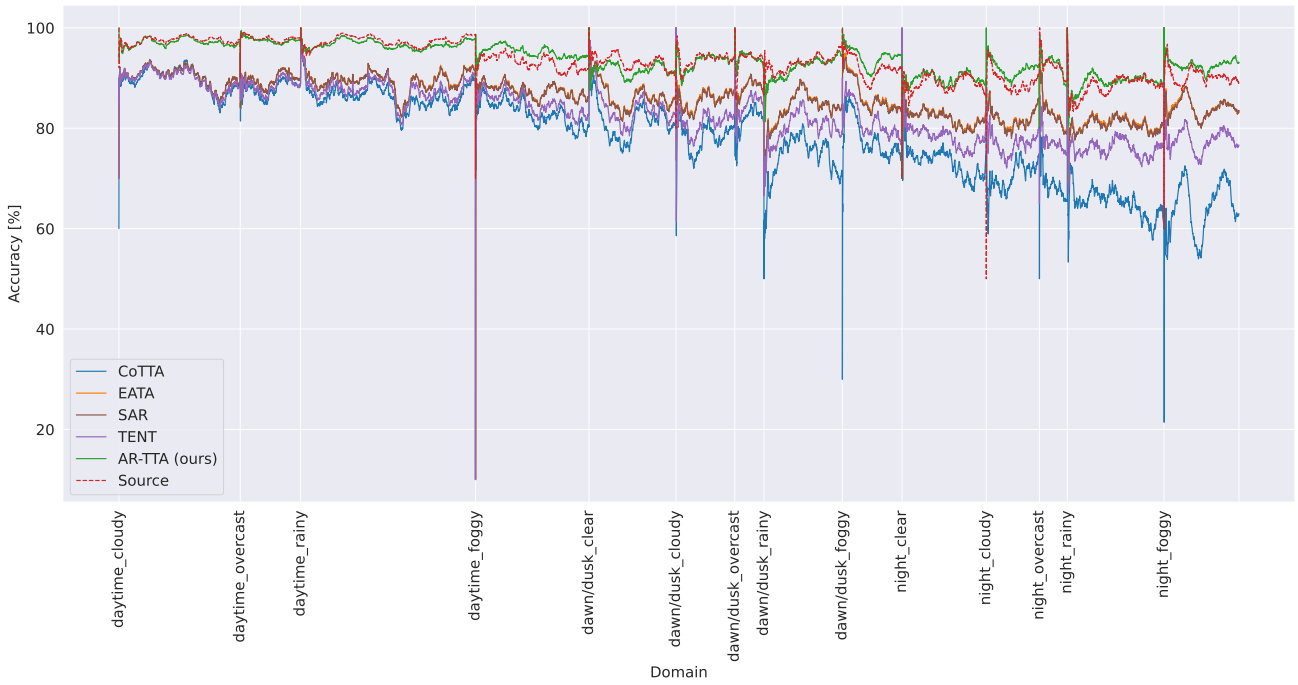

Figure 5. Batch-wise classification accuracy (%) averaged in a window of 500 batches on SHIFT-C benchmark for the chosen methods continually adapted to the sequences of data. The ticks on the x-axis symbolize the beginning of the next sequence and, at the same time, a different domain. The window to calculate the average values is cleared in between the sequences. Best viewed in color.

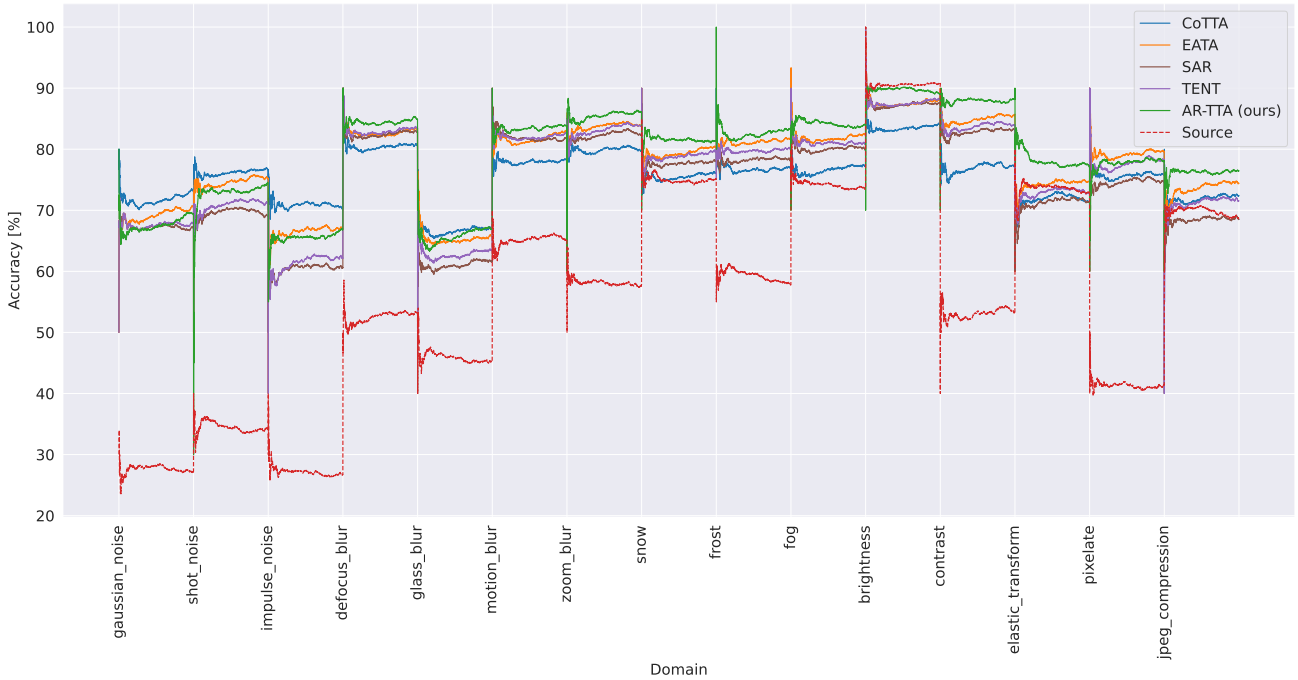

Figure 6. Batch-wise classification accuracy (%) averaged in a window of 500 batches on CIFAR10C benchmark for the chosen methods continually adapted to the sequences of data. The ticks on the x-axis symbolize the beginning of the next sequence and, at the same time, a different domain. The window to calculate the average values is cleared in between the sequences. Best viewed in color.

References

[1] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017.

[2] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 16888–16905. PMLR, 2022.

[3] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. In *Internetional Conference on Learning Representations*, 2023.

[4] Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. *Advances in Neural Information Processing Systems*, 33:11539–11551, 2020.

[5] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21371–21382, June 2022.

[6] Eli Verwimp, Kuo Yang, Sarah Parisot, Lanqing Hong, Steven McDonagh, Eduardo Pérez-Pellitero, Matthias De Lange, and Tinne Tuytelaars. Clad: A realistic continual learning benchmark for autonomous driving. *Neural Networks*, 161:659–669, 2023.

[7] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno A. Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[8] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 7191–7201. IEEE, 2022.