AR-TTA: A SIMPLE METHOD FOR REAL-WORLD CONTINUAL TEST-TIME ADAPTATION

Damian Sójka [1,2] Bartłomiej Twardowski [1,3,4] Tomasz Trzciński [1,5,6] Sebastian Cygert [1,7]

[1]IDEAS NCBR [2]Poznań University of Technology [3]Computer Vision Center [4]Autonomous University of Barcelona [5]Warsaw University of Technology [6]Tooploox [7]Gdańsk University of Technology

BMVC 2024

TL;DR

- Continual Test-Time Adaptation (TTA) methods allow models to adapt to changing data distributions without supervision.
- Current techniques are often evaluated on benchmarks that oversimplify real-world scenarios.
- We evaluate current test-time adaptation methods on realistic, continual domain shift image classification data from autonomous driving.
- We observe that they struggle with varying degrees of domain shifts, often resulting in performance drops below that of the source model.
- We propose a method that obtains state-of-the-art performance on multiple benchmarks with both artificial distortions and real-life domain shifts.

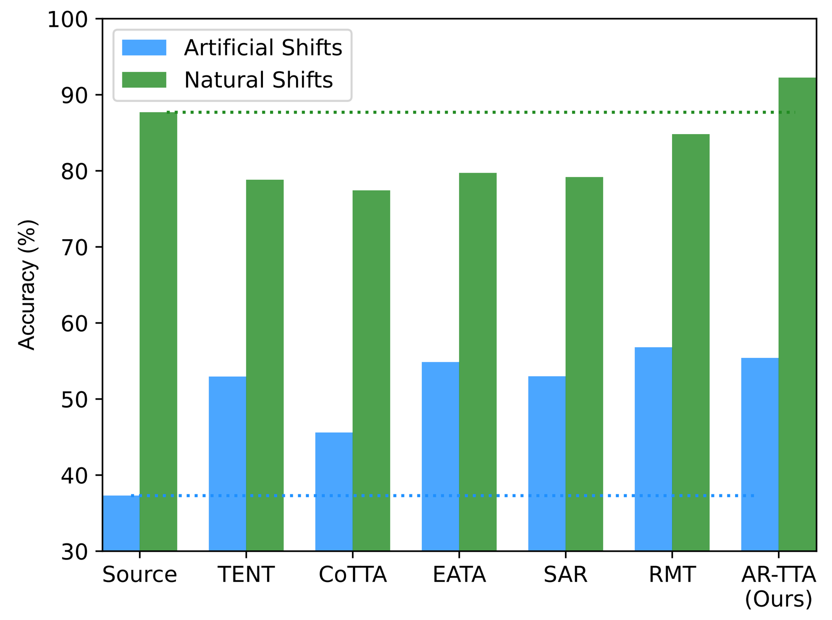

Figure 1. Continual test-time adaptation methods evaluated on artificial and natural domain shifts. Our method is the only one that consistently allows to improve over the naive strategy of using the (frozen) Source model.

Natural domain shifts

Figure 2. Example images from various domains within the CLAD-C benchmark.

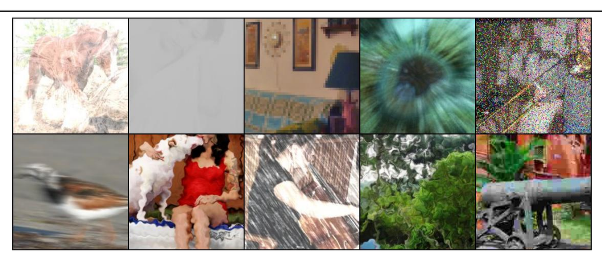

Figure 3. Example images with different corruptions from the ImageNet-C dataset.

- The most popular setting for test-time adaptation includes using different classes of synthetic corruptions.
- In practical applications, the target distribution can easily change in a different manner, perpetually over time, e.g., due to changing weather, lighting conditions, or traffic intensity.
- Hence, we propose to use two benchmarks that consist of data with domain shifts that can occur in real-world applications - the CLAD-C benchmark [5] and the SHIFT dataset [4].

Proposed method (AR-TTA)

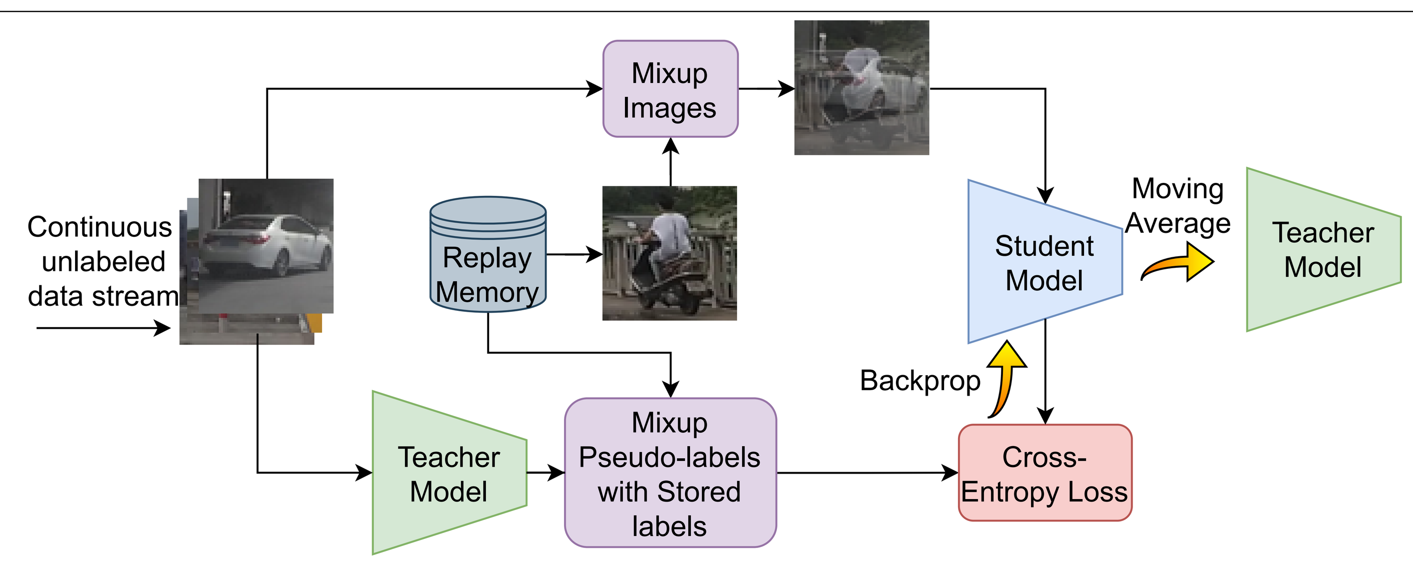

Figure 4. Overall method diagram.

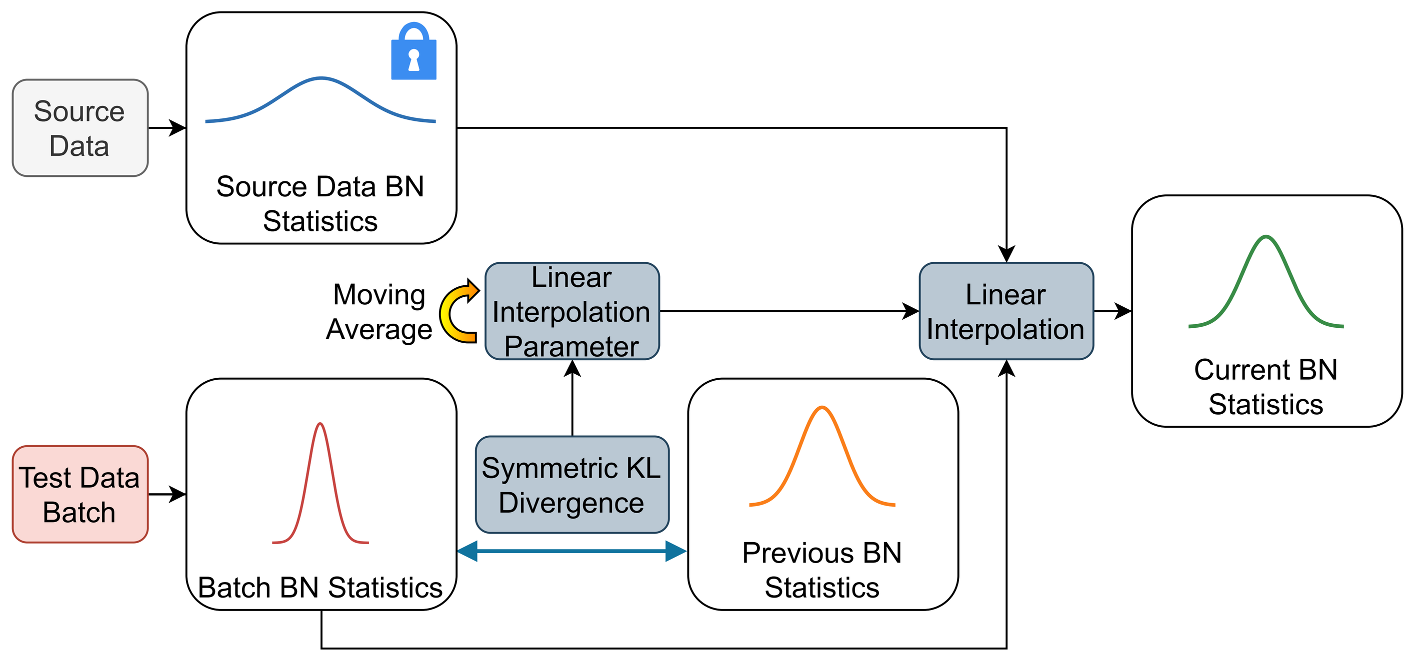

Figure 5. Our batch normalization statistics update scheme.

Results

Table 1. Classification accuracy (%) for all of the tested online continual test-time adaptation tasks. Methods that use exemplars are in the right section. The red color indicates accuracy lower compared to a source model. The blue color indicates dataset with artificial domain shifts and green dataset with natural ones.

Method	CIFAR10C	ImageNet-C	CIFAR10.1	CLAD-C	SHIFT-C	Average
Source	56.5	18.1	88.3	81.3	93.5	67.5
BN-1	75.0	26.9	81.3	71.1	85.1	67.9
TENT [6]	76.7	29.2	82.3	71.5	82.7	68.5
EATA [2]	78.2	31.5	82.9	71.1	85.1	69.8
CoTTA [7]	75.7	15.5	82.3	72.6	77.4	64.7
SAR [3]	75.2	30.8	81.3	71.1	85.1	68.7
RMT w/o replay [1]	77.6	21.7	80.6	75.1	92.2	69.4
RMT [1]	**83.1**	30.5	83.3	75.3	**95.9**	73.6
AR-TTA (Ours) w/o replay	$77.3_{\pm0.07}$	$30.0_{\pm0.45}$	$88.2_{\pm0.10}$	$\mathbf{83.9}_{\pm0.30}$	$92.4_{\pm0.25}$	74.4
AR-TTA (Ours)	$\mathbf{78.8}_{\pm0.13}$	$\mathbf{32.0}_{\pm0.07}$	$\mathbf{88.3}_{\pm0.05}$	$83.7_{\pm0.64}$	$94.8_{\pm0.03}$	**75.5**

- **BN-1** significantly improves the result on corrupted images but does not improve the performance over the Source model on natural domain shifts.
- Similarly, the state-of-the-art TTA methods achieve lower accuracy than the Source model on natural domain shifts.
- Our method outperforms state-of-the-art methods and achieves higher accuracy than the Source model on both types of benchmarks.

Ablation study

Table 2. Classification accuracy and average mean class accuracy (AMCA) (%) for the CLAD-C continual test-time adaptation task.

Method	T1	T2	T3	T4	T5	Mean	AMCA
Source	75.6	85.9	73.3	87.5	66.2	81.3	57.6
BN-1	73.2	69.9	75.0	75.5	59.7	71.1	48.3
TENT [6]	73.4	69.8	76.5	76.1	59.7	71.5	47.6
EATA [2]	73.3	69.9	75.0	75.6	59.7	71.1	48.4
CoTTA [7]	75.2	69.3	80.2	77.0	62.7	72.6	44.8
SAR [3]	73.2	69.9	75.0	75.5	59.7	71.1	48.3
RMT w/o replay [1]	**87.1**	70.9	**86.6**	76.9	64.3	75.1	48.4
RMT [1]	83.8	71.3	85.0	77.6	66.4	75.3	48.8
AR-TTA (Ours) w/o replay	76.9	**86.7**	81.4	87.9	**73.5**	$\mathbf{83.9}_{\pm0.30}$	$59.6_{\pm2.92}$
AR-TTA (Ours)	77.2	**86.7**	80.0	**89.6**	70.7	$83.7_{\pm0.64}$	$\mathbf{63.1}_{\pm3.32}$

($t \longrightarrow$ over T1–T5)

Table 3. Classification accuracy (%) for CIFAR10C and CLAD-C tasks for different configurations of the proposed method.

Method	CIFAR10C	CLAD-C
A: Weight-avg. teacher	$75.7_{\pm0.07}$	$71.1_{\pm0.53}$
B: A + Replay memory	$77.3_{\pm0.16}$	$69.0_{\pm0.66}$
C: B + Mixup	$78.5_{\pm0.13}$	$72.2_{\pm0.31}$
D: A + Dynamic BN stats	$77.3_{\pm0.07}$	$83.8_{\pm0.82}$
E: D + Replay memory	$79.8_{\pm0.03}$	$82.8_{\pm1.09}$
AR-TTA (Ours): E + Mixup	$78.8_{\pm0.13}$	$83.7_{\pm0.64}$

Table 4. Classification accuracy and average mean class accuracy (AMCA) (%) results for state-of-the-art methods with simple **replay method added**.

Method	CIFAR10C	CLAD-C	
	Mean	Mean	AMCA
Source	56.5	81.3	57.6
TENT [6]	77.3	70.3	49.2
EATA [2]	78.6	71.1	48.4
CoTTA [7]	**79.9**	72.6	51.0
SAR [3]	75.3	71.1	48.3
AR-TTA (Ours)	78.8	**83.7**	**63.1**

Table 5. The wall-clock time (seconds) and memory usage (MB) measured for processing 10,000 images of CIFAR10C on a single RTX 4080 GPU.

Method	Time [s]	Memory [MB]
Source	8.0	304
BN-1	8.3	304
TENT [6]	16.3	506
EATA [2]	24.3	505
CoTTA [7]	319.4	1532
SAR [3]	30.8	506
RMT [1] w/o replay	55.5	1576
RMT [1]	163.7	3039
AR-TTA (Ours) w/o replay	66.2	1098
AR-TTA (Ours)	66.6	1098

Effect Of Exemplars. Average mean class accuracy (AMCA) values show that the usage of replay memory might be crucial for high mean per-class accuracy.

Baselines With Simple Replay Memory. While the proposed method performs slightly worse than CoTTA with simple replay memory on the CIFAR10C, it performs significantly better on the natural domain shift dataset. Most importantly, our method is the only one that constantly improves over the source model.

Computational Efficiency While our method does not rank as the most computationally efficient, it achieves a balance between computational demands and performance. Despite incorporating exemplars, we maintain a consistent computational budget.

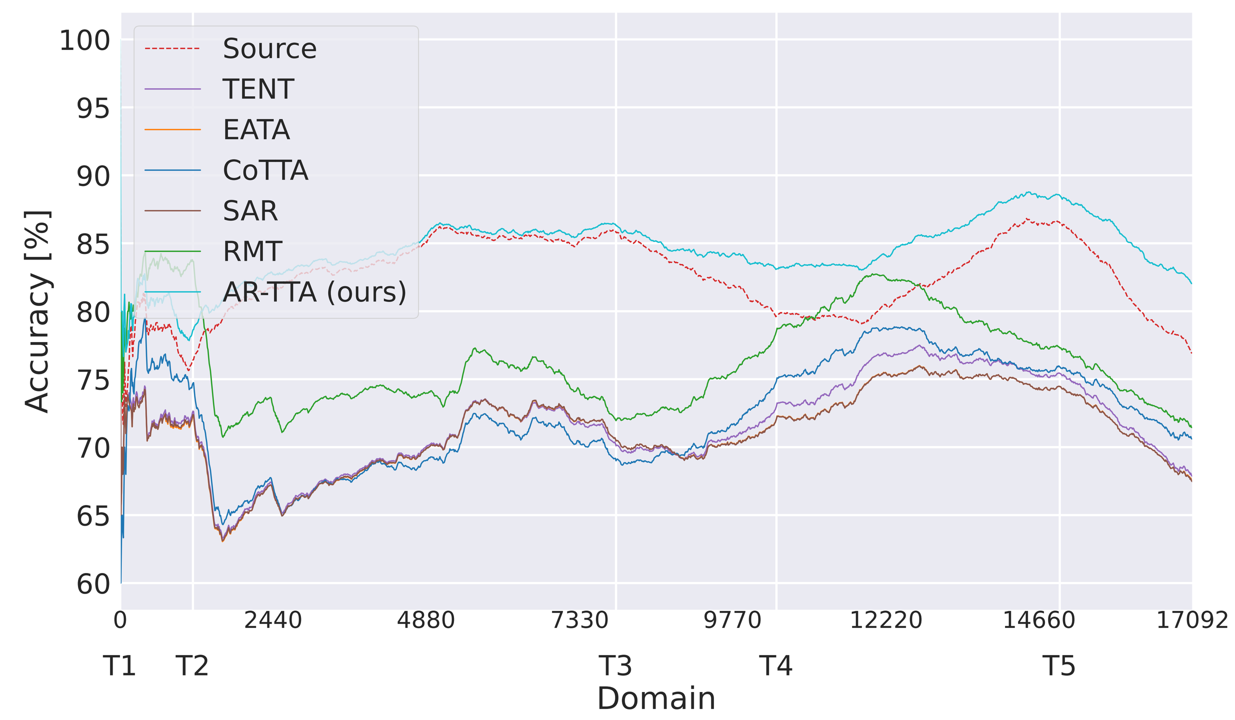

Figure 6. Batch-wise classification accuracy (%) averaged in a window of 400 batches on CLAD-C benchmark for the chosen methods continually adapted to the sequence of data, with major ticks on the x-axis symbolizing the beginning of a different domain and minor ticks indicating image number. Best viewed in color.

References

[1] Mario Döbler, Robert A. Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. *CoRR*, abs/2211.13081, 2022.

[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] 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.

[5] 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.

[6] 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.

[7] 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.

Read the paper at