Supplementary Materials for “AuxAdapt: Stable and Efficient Test-Time Adaptation for Temporally Consistent Video Semantic Segmentation”

Yizhe Zhang [*†]
yizhe.zhang.cs@gmail.com

Shubhankar Borse [‡§]
sborse@qti.qualcomm.com

Hong Cai [‡§]
hongcai@qti.qualcomm.com

Fatih Porikli [‡]
fporikli@qti.qualcomm.com

1. Additional Results

1.1. Per-Class Temporal Consistency

As shown in Table 1, AuxAdapt provides consistent TC improvement across classes.

Method	road	sdwk.	bldg.	wall	fence	pole	t-light	t-sign	veg.	terrain
No Adapt	96.5	82.8	87.6	53.6	66.3	30.2	57.8	58.5	88.9	76.5
AuxAdapt	97.3	85.4	89.8	65.2	75.8	30.2	60.2	61.4	90.5	81.2

Method	sky	person	rider	car	truck	bus	train	m-bike	bike	**mean**
No Adapt	91.9	60.6	58.7	85.7	81.1	85.9	83.0	56.1	68.1	72.1
AuxAdapt	92.6	62.7	62.0	87.5	85.7	90.4	86.7	65.1	71.3	75.8

Table 1: Per-class temporal consistency on Cityscapes. MainNet: HRNet-w48-s4. AuxNet: HRNet-w18-s8.

1.2. When Pretrained AuxNet is Unavailable

During test time, there can be cases where only the main segmentation network (MainNet) is provided and no pretrained auxiliary networks (AuxNets) are available. In such cases, it is still possible to apply AuxAdapt to improve temporal consistency, by creating a lower-resolution copy of MainNet to serve as AuxNet. More specifically, we create a copy of MainNet and add a down-sampling layer at the beginning of it. In this way, AuxNet shares the same architecture and weights as MainNet, but works with down-sampled inputs, thus considerably saving computation. At the end of AuxNet, a corresponding up-sampling layer is added such that the size of its output matches that of MainNet.

In Table 2, it can be seen that in this setting where AuxNet is obtained from MainNet (denoted as “OFM"), AuxAdapt provides considerable improvement to temporal consistency while maintaining segmentation accuracy. As compared to the case where a pretrained AuxNet is available (denoted as “PT"), AuxAdapt in the OFM setting provides very similar performance. Overall, the OFM option makes AuxAdapt more widely applicable while providing comparable adaptation performance. Note that, in this OFM setup, the additional computational cost for test-time adaptation could be higher than using a well-designed and pretrained AuxNet since now AuxNet is directly derived from MainNet, and is not optimized for its efficiency.

Method	TC	mIoU	GMAC/$_F$
Cityscapes			
HRNet-w18-s4 [1]	70.5	76.2	78
w/ AuxAdapt (PT)	75.3	76.6	128
w/ AuxAdapt (OFM)	75.2	76.4	136
CamVid			
HRNet-w18-s4 [1]	75.8	73.2	26
w/ AuxAdapt (PT)	79.1	73.2	42
w/ AuxAdapt (OFM)	78.9	73.2	45
WRN38 [2]	78.1	80.6	1920
w/ AuxAdapt (PT)	79.4	80.8	1995
w/ AuxAdapt (OFM)	79.7	80.7	2280

Table 2: AuxAdapt using MainNet-derived AuxNet on Cityscapes and CamVid. OFM indicates that AuxNet is obtained from MainNet, with an additional down-sampling operation at the beginning. PT denotes the setting where AuxNet is pretrained using the corresponding architectures described in Table 1 of the main paper.

1.3. Input Down-sampling for AuxNet

In the main paper, 2× down-sampling is applied to AuxNet’s input (see Fig. 2 of main paper), which reduces computation. In this part, we study the effect of further down-sampling the input to AuxNet.

Table 3 shows the results with a more aggressive down-sampling ratio (3×). It can be seen that the TC improvement is similar and the computation cost is reduced. However, the segmentation accuracy slightly drops, as the further-down-sampled input now contains less information.

1.4. Standalone Performance of AuxNet

In Table 4, we report the **standalone** performance of the lightweight models which are used as AuxNets in our experiments. The TC, mIoU, and GMAC numbers reported here are based on using the AuxNet model alone, without MainNet and adaptation.

[*] Work done at Qualcomm AI Research.
[†] Nanjing University of Science and Technology, Nanjing, China
[‡] Qualcomm AI Research, an initiative of Qualcomm Technologies, Inc.
[§] Equal contribution.

Method	TC	mIoU	GMAC/$_F$
Cityscapes			
HRNet-w48-s4 [1]	72.1	81.0	749.9
w/ AuxAdapt (2× ↓)	75.8	81.0	808.2
w/ AuxAdapt (3× ↓)	76.7	80.5	776.2
KITTI			
HRNet-w48-s4 [1]	57.4	65.9	175.8
AuxAdapt (2× ↓)	63.5	65.8	189.4
AuxAdapt (3× ↓)	63.4	64.0	181.9

Table 3: Effect of AuxNet's input resolution. The numbers in the parentheses indicate how much the input image is down-sampled via average pooling. MainNet is HRNet-w48-s4. For 2× (3×) down-sampling, AuxNet is HRNet-w18-s8 (HRNet-w18-s12), where the number following "s" indicates the upsampling ratio at the output.

Networks	TC	mIoU	GMAC/$_F$
Cityscapes			
HRNet-w18-s8	71.9	72.6	19
HRNet-w16-s8	71.4	74.3	17
CamVid			
HRNet-w18-s8	76.8	69.4	6.3
HRNet-w16-s8	76.5	70.8	5.6
KITTI			
HRNet-w18-s8	54.1	57.5	4.3

Table 4: Standalone performance of the lightweight AuxNet models on Cityscapes, CamVid, and KITTI.

References

[1] J. Wang, K. Sun, T. Cheng, B.i Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao. Deep high-resolution representation learning for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[2] Y. Zhu, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam, A. Tao, and B. Catanzaro. Improving semantic segmentation via video propagation and label relaxation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.