TeSLA: Test-Time Self-Learning With Automatic Adversarial Augmentation Supplementary Material

Devavrat Tomar[1] Guillaume Vray[1] Behzad Bozorgtabar[1,2] Jean-Philippe Thiran[1,2]
[1]EPFL [2]CHUV
[1]{firstname}.{lastname}@epfl.ch

Overview. This Appendix provides important additional details about our proposed method **TeSLA**. In Appendix A, we provide hyperparameters details for each test-time adaptation experiment on the **common image corruptions**, **synthetic-to-real**, and **medical measurement shift** datasets. In Appendix B, we provide additional quantitative results, including class top-1 accuracies for the VisDA-C [10] and Kather-16 [5] datasets and corruption-wise error rates on the CIFAR-10-C/CIFAR-100-C [3], and ImageNet-C [3] datasets. In addition, we also provide segmentation class-wise mean Intersection over Union (mIoU) for the VisDA-S dataset [10] and class average Dice score for different sites of the target test domain on the spinal cord [11] and prostate dataset [7] for the competing test time adaptation methods. All quantitative results are included for both one-pass (**O**) and multi-pass (**M**) protocols. Appendix C provides an overall runtime computation cost of TeSLA along with other Test Time adaptation methods on VisDA-C [10] dataset, while Appendix D discusses TeSLA's equivalence to TENT [14] and [6] without mean teacher and adversarial augmentations. We include additional ablation experiments and hyperparameter sensitivity tests in Appendix E. Finally, we provide other qualitative results, including a sanity check on TeSLA's adversarial augmentations, uncertainty evaluation, and segmentation visualization in Appendix F.

A. Hyperparameter Settings

Table A.1 and Table A.2 present the hyperparameters' values of TeSLA used for individual experiments on different classification and segmentation datasets, respectively. These hyperparameters include the batch size B, learning rate, optimizer, EMA momentum coefficient α, number of epochs for test-time adaptation (M protocol), the number of weak augmentations $|\rho_w|$; the number of nearest neighbors n; class-wise queue size N_Q used by soft pseudo-label refinement (PLR) module, number of image operations for augmentation sub-policy N used by the adversarial augmentation module, the augmentation severity controller coefficient λ_1 and the knowledge distillation coefficient λ_2 for $\mathcal{L}_{\text{kd}}$ loss term.

Table A.1. **Hyperparameter setting** used for the proposed methods TeSLA/TeSLA-s on different classification datasets.

Hyperparameters	CIFAR10-C		CIFAR100-C		ImageNet-C		VisDA-C		Kather-16	
	O	M	O	M	O	M	O	M	O	M
Batch size B	128	128	128	128	128	128	128	128	32	32
Learning rate	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.005	0.005
Optimizer	Adam	Adam	Adam	Adam	SGD	SGD	SGD	SGD	Adam	Adam
Momentum coefficient α	0.99	0.999	0.99	0.999	0.9	0.996	0.9	0.996	0.9	0.96
Number of epochs	1	70	1	70	1	5	1	5	1	70
Number of weak augmented views $\lvert\rho_w\rvert$	5	5	5	5	5	5	5	5	5	5
Number of nearest neighbors n	1	4	1	1	1	1	10	10	8	8
Class-wise queue size $N_{\mathbf{Q}}$	1	256	1	1	1	1	256	256	32	256
Sub-policy dimension N	2	2	2	2	2	2	4	3	2	2
Augmentation severity controller λ_1	1	1	1	1	1	1	1	1	1	1
Knowledge distillation weight λ_2	1	1	1	1	1	1	1	1	1	1

B. Additional Quantitative Results

In Table B.1, we compare TeSLA against state-of-the-art test-time adaptation methods for the classification task on the Kather-16 dataset. We present the class top-1 accuracies (%) for each of the four tissue categories of **tumor**, **stroma**, **lymphocyte**, and **mucosa**. In addition, we report the class average accuracy (Avg.).

Furthermore, in Table B.2, we present the corruption-wise average class error rates for different competing test time adaptation baselines, including the proposed TesLA and TeSLA-s on the CIFAR10-C, CIFAR100-C and ImageNet-C. We use the following image corruptions for the evaluation at the maximum severity level of 5: [Gaussian Noise, Shot Noise, Impulse Noise, Defocus Blur, Glass Blur, Motion Blur, Zoom Blur, Snow, Frost, Fog, Brightness, Contrast, Elastic Transformation, Pixelate, JPEG Compression]. We also use the ResNet-50 backbone for all experiments.

In Table B.3, we include the overall and class-wise accuracies for test time adaptation of ResNet-101 trained on

Table A.2. **Hyperparameter setting** used for the proposed method TeSLA on different segmentation datasets.

Hyperparameters	VisDA-S		Spinal Cord		Prostate	
	O	M	O	M	O	M
Batch size B	8	8	16	16	16	16
Learning rate	0.001	0.001	0.0002	0.0002	0.0002	0.0002
Optimizer	AdamW	AdamW	AdamW	AdamW	AdamW	AdamW
Momentum coefficient α	0.996	0.999	0.996	0.996	0.996	0.996
Number of epochs	1	3	1	5	1	3
Number of weak augmented views $\lvert\rho_w\rvert$	3	3	5	5	5	5
Number of nearest neighbors n	1	1	1	1	1	1
Class-wise queue size $N_{\mathbf{Q}}$	1	1	1	1	1	1
Sub-policy dimension N	3	3	3	3	3	3
Augmentation severity controller λ_1	1	1	1	1	1	1
Knowledge distillation weight λ_2	1	1	1	1	1	1

synthetic vehicle images (**training**) and tested on the photo-realistic vehicle images (**validation**) of the VisDA-C dataset. The photo-realistic images are classified into 12 categories: **plane**, **bicycle**, **bus**, **car**, **horse**, **knife**, **motor-cycle**, **person**, **plant**, **skate-board**, **train**, and **truck**.

In Table B.4, we present segmentation results (class Avg. volume-wise mean %Dice score) for test-time adaptation baselines on two multi-site magnetic resonance imaging (MRI) benchmarks - spinal cord [11] and prostate dataset [7]. For the spinal cord dataset, we report results for test-time adaptation of the U-Net segmentation model trained on **site 1** to **site 2**, **site 3**, and **site 4**. Similarly, we report results of the U-Net segmentation model trained on the sites **A and B**, which are adapted on the sites **D**, **site E**, and **site F**.

Table B.5 presents the results of competing test-time adaptation methods applied to the segmentation adaptation task from the synthetic images of GTA [12] to the photo-realistic images of Cityscapes [2] dataset. We report the class-wise mean Intersection over Union (mIoU) over 19 classes: **road**, **side-walk**, **building**, **wall**, **fence**, **pole**, **light**, **sign**, **vegetation**, **terrain**, **sky**, **person**, **rider**, **car**, **truck**, **bus**, **train**, **motor-cycle**, and **bicycle**.

C. Runtime Analysis

We compare the computational runtime cost of several test-time adaptation methods, including BN [4, 8], TTAC [13], SHOT-IM [6], TENT [14], AdaContrast [1] and our proposed method TeSLA in Table C.1. We also include overall TesLA runtimes using static, pre-optimized RandAugment (RA) / AutoAugment (AA) augmentation policies instead of the proposed Adversarial Augmentations.

Table B.1. **Comparison of state-of-the-art TTA methods under different protocols** on the Kather-16 dataset. We report the class top-1 accuracies (%) for each of the four classes and the per-class average accuracy (Avg.). Each result is averaged over ten seeds.

Method	Protocol	tumor	stroma	lymphocyte	mucosa	Avg.
Source	N	84.5±4.0	91.6±3.0	0.9±1.2	95.0±1.3	68.0±1.3
BN	N-O	89.3±2.5	85.5±2.6	61.7±2.2	90.2±0.6	81.7±1.0
Tent	N-O	89.8±4.0	89.3±3.4	67.2±2.2	88.9±1.0	83.8±1.8
SHOT	N-O	84.7±5.7	95.7±2.0	67.9±3.9	92.8±0.8	85.3±2.5
TTT++	Y-O	82.8±8.5	85.1±7.6	73.7±3.8	91.4±1.7	83.3±2.7
TTAC	Y-O	92.6±2.6	96.6±1.9	**78.3±4.3**	93.9±1.0	90.4±1.1
TeSLA	N-O	**93.5±1.8**	**98.2±1.2**	77.3±4.1	**94.3±0.7**	**90.8±1.1**
TeSLA-s	Y-O	90.7±4.6	98.0±1.0	77.9±5.6	94.0±1.7	90.1±1.4
BN	N-M	86.3±3.6	86.1±1.9	66.9±1.2	87.7±0.7	81.8±1.0
Tent	N-M	96.4±4.2	99.5±0.4	62.6±9.0	93.7±1.5	88.0±3.3
SHOT	N-M	84.6±4.4	98.5±0.6	77.1±5.2	91.7±0.8	88.0±2.4
TTT++	Y-M	95.6±1.4	93.9±2.8	85.2±5.3	93.6±1.8	92.1±2.0
TTAC	Y-M	92.9±12.6	98.1±1.1	92.4±4.7	94.4±1.8	94.5±4.7
TeSLA	N-M	97.1±1.0	**99.6±0.3**	94.4±2.0	95.6±0.9	96.7±0.5
TeSLA-s	Y-M	**97.4±0.4**	99.5±0.3	**95.1±2.0**	**95.7±1.0**	**96.9±0.6**

Table C.1. Runtime (GPU hours) per epoch on GeForce RTX-3090 for ResNet-101 with batch size of 128 on the VisDA-C. RA implies RandAugment [10], AA implies AutoAugment [9].

BN	TTAC	SHOT-IM	TENT	AdaContrast	**TeSLA**	**TeSLA (RA)**	**TeSLA (AA)**
0.04	0.14	0.16	0.05	0.22	**0.38**	**0.27**	**0.28**

D. Equivalence to other Test-Time Objectives

Our proposed flipped cross-entropy loss f-$\mathbb{CE}$ of Eq. 1 (*main paper*) without soft-pseudo labels from the teacher is equivalent to entropy minimization of TENT [14], while our final objective $\mathcal{L}_{\text{TeSLA}}$ of Eq. 14 (*main paper*) without the knowledge distillation from adversarial augmentation is equivalent to SHOT-IM [6] as $\mathcal{D}_{\text{KL}}\left(\mathbf{Y} \parallel \hat{\mathbf{Y}} \mid \mathbf{X}\right) = 0$ when the teacher network is an instant update of the student (momentum α is 0). Thus, without the mean-teacher and adversarial augmentation, our method would have similar shortcomings as that of TENT and SHOT. Incorporating the soft-pseudo labels from the mean teacher alone improves TeSLA's accuracy on VisDA-C from 82.0% to 86.5%.

E. Sensitivity Tests and Aditional Ablations

E.1. Sensitivity Tests

Automatic Adversarial Augmentation. We additionally provide the sensitivity test results for the hyperparameters of the automatic adversarial augmentation module (λ_1 and sub-policy dimension N) on the VisDA-C and VisDA-S datasets. In Fig. E.1, we show how the class average (Avg.) accuracy (%) varies with the hyperparameter λ_1 controlling the severity of augmentations and the sub-policy dimension N

Table B.2. **Comparison of state-of-the-art TTA methods under different protocols** on common image corruptions datasets, including CIFAR10-C, CIFAR100-C, and ImageNet-C. We report the error rates (%) on 15 test images' corruptions.

Method	Protocol	Gaus.	Shot	Impu.	Defo.	Glas.	Moti.	Zoom	Snow	Fros.	Fog	Brig.	Cont.	Elas.	Pixe.	Jpeg	Avg.
CIFAR10-C																	
Source	N	48.7	44.0	57.0	11.8	50.8	23.4	10.8	21.9	28.2	29.4	7.0	13.3	23.4	47.9	19.5	29.1
BN	N-O	18.2	17.2	28.1	9.8	26.6	14.2	8.0	15.5	13.8	20.2	7.9	8.3	19.3	13.3	13.8	15.6
Tent	N-O	16.0	14.8	24.5	9.2	23.8	13.1	7.7	14.9	13.0	16.5	8.2	8.3	17.9	10.9	13.3	14.1
SHOT	N-O	16.5	15.3	23.6	9.0	23.4	12.7	7.5	14.0	12.4	16.1	7.5	8.0	17.4	12.5	13.1	13.9
TTT++	Y-O	18.0	17.1	30.8	10.4	29.9	13.0	9.9	14.8	14.1	15.8	7.0	7.8	19.3	12.7	16.4	15.8
TTAC	Y-O	17.9	15.8	22.5	**8.5**	23.5	**11.2**	7.6	**11.9**	12.9	**13.3**	**6.9**	7.6	17.3	12.3	12.6	13.4
TeSLA	N-O	13.3	12.5	20.8	8.8	21.1	11.8	7.3	12.6	11.2	15.6	7.6	7.6	16.2	9.7	11.6	12.5
TeSLA-s	Y-O	**13.0**	**12.2**	**20.3**	**8.5**	**20.8**	**11.2**	**7.2**	12.0	**11.0**	15.5	7.3	**7.2**	**15.6**	**9.1**	**11.3**	**12.1**
BN	N-M	17.3	16.2	28.0	9.8	26.1	14.0	7.9	16.1	13.7	20.4	8.3	8.3	19.6	11.8	14.0	15.4
Tent	N-M	15.1	13.7	22.2	8.5	22.4	11.8	7.1	12.7	11.9	12.9	7.6	7.6	16.9	9.8	12.6	12.9
SHOT	N-M	15.8	14.8	24.9	9.2	23.6	13.2	7.5	14.5	12.8	17.5	8.1	8.2	18.1	10.8	13.4	14.2
TTT++	Y-M	13.2	11.8	**11.1**	7.9	16.5	8.9	6.6	9.5	9.7	8.6	**5.2**	**5.6**	13.1	8.8	11.1	9.8
TTAC	Y-M	11.6	10.3	15.8	**6.8**	15.9	**7.5**	**5.8**	**8.7**	9.0	**8.5**	5.6	5.7	**12.7**	8.0	9.7	**9.4**
TeSLA	N-M	10.7	**9.8**	15.2	7.0	**15.8**	9.1	6.1	10.0	**8.9**	10.9	6.0	6.2	13.0	**7.9**	9.6	9.7
TeSLA-s	Y-M	**10.4**	**9.8**	14.9	7.3	16.1	9.0	6.2	9.5	9.1	11.5	5.9	5.8	12.9	**7.9**	**9.5**	9.7
CIFAR100-C																	
Source	N	80.8	77.8	87.8	39.6	82.3	54.2	38.4	54.6	60.2	68.1	28.9	50.9	59.5	72.3	50.0	60.4
BN	N-O	48.2	46.4	61.1	33.8	58.2	41.4	31.9	46.1	42.5	54.7	31.3	33.3	48.4	39.0	39.6	43.7
Tent	N-O	43.3	41.2	52.7	31.2	50.8	36.1	29.3	41.9	38.9	43.6	30.1	31.0	43.5	34.4	36.5	39.0
SHOT	N-O	44.1	41.8	53.3	31.5	50.6	36.0	29.6	40.7	40.1	41.9	29.5	33.6	44.0	34.9	36.6	39.2
TTT++	Y-O	50.2	47.7	66.1	35.8	61.0	38.7	35.0	44.6	43.8	48.6	28.8	30.8	49.9	39.2	45.5	44.4
TTAC	Y-O	47.7	45.7	58.1	32.5	55.3	36.6	31.2	40.3	40.8	**44.7**	30.0	39.9	47.1	37.8	38.3	41.7
TeSLA	N-O	40.0	38.9	51.5	32.2	49.1	36.9	29.7	40.4	37.4	46.0	29.3	30.7	42.7	32.9	34.6	38.2
TeSLA-s	Y-O	**39.1**	**38.5**	**50.0**	**30.6**	**48.6**	**35.9**	**29.1**	**38.9**	**36.4**	46.2	**28.3**	**29.7**	**41.9**	**32.1**	**33.9**	**37.3**
BN	N-M	47.4	45.5	60.0	33.9	56.9	40.8	31.8	46.4	42.6	54.2	32.3	33.1	48.5	37.2	39.4	43.3
Tent	N-M	41.0	38.4	49.2	30.0	47.4	33.1	28.1	38.1	38.0	37.5	28.3	29.0	41.1	32.8	35.6	36.5
SHOT	N-M	41.6	40.6	51.7	31.4	49.5	36.2	29.3	42.4	38.4	45.4	29.9	31.3	43.1	33.5	36.0	38.7
TTT++	N-M	38.4	37.7	**41.3**	29.1	44.1	32.9	27.8	34.3	34.4	**34.7**	25.4	26.6	39.2	32.3	33.6	34.1
TTAC	N-M	37.8	36.8	45.1	28.2	45.3	**30.7**	26.6	35.3	35.7	36.7	26.8	27.4	39.6	30.6	34.2	33.6
TeSLA	N-M	34.4	33.5	42.2	28.0	**41.9**	32.1	**25.9**	35.1	32.6	38.3	25.0	27.4	37.5	28.6	30.6	32.9
TeSLA-s	Y-M	**33.9**	**33.0**	42.1	**27.5**	42.0	31.6	26.1	**34.2**	**32.2**	39.4	**24.8**	**26.3**	**36.8**	**28.1**	**30.3**	**32.6**
ImageNet-C																	
Source	N	97.0	96.3	97.4	82.1	90.3	85.3	77.5	83.4	76.9	76.0	40.9	94.6	83.5	79.1	67.4	81.8
BN	N-O	83.5	82.6	82.9	84.4	84.2	73.1	60.5	65.1	66.3	51.5	34.0	82.6	55.3	50.3	58.7	67.7
Tent	N-O	70.8	68.7	69.1	72.5	73.3	59.3	50.8	53.0	59.1	42.7	32.6	**74.5**	45.5	41.6	47.8	57.4
SHOT	N-O	77.0	74.6	76.4	81.2	79.3	72.5	61.7	65.7	66.3	55.6	56.0	92.7	57.1	56.3	58.2	68.7
TTAC	Y-O	71.5	67.7	70.3	81.2	77.3	64	54.4	51.1	56.9	45.4	32.6	79.1	46.0	43.7	48.6	59.3
TTT++	Y-O	69.4	66	69.7	84.2	81.7	65.2	53.2	49.3	56.2	44.4	32.8	75.7	43.9	41.6	46.9	58.7
TeSLA	N-O	65.0	62.9	63.5	69.4	69.2	55.4	49.5	49.1	56.6	41.8	33.7	77.9	43.3	40.4	46.6	55.0
TeSLA-s	Y-O	**61.4**	**58.8**	**60.3**	**67.3**	**66.2**	**54.0**	**48.2**	**46.9**	**53.1**	**40.9**	**32.4**	81.2	**41.1**	**39.2**	**44.8**	**53.1**
BN	N-M	83.4	82.6	82.8	84.4	84.2	73.2	60.3	64.9	66.4	51.2	34.0	82.6	54.9	49.9	58.8	67.6
Tent	N-M	66.1	63.7	64.2	68.9	69.6	52.6	47.4	48.4	58.4	39.8	**31.6**	77.9	41.7	**28.7**	44.5	54.2
SHOT	N-M	75.8	73.7	73.7	78.3	77.1	71.8	60.9	64.2	66.1	55.4	59.8	95.5	56.1	57.3	58.1	68.2
TeSLA	N-M	**62.3**	**60.9**	**60.6**	**64.3**	**65.7**	**50.4**	**46.2**	**46.1**	**54.7**	**39.1**	32.2	**68.5**	**40.9**	37.5	**43.5**	**51.5**

Table B.3. **Comparison of state-of-the-art TTA methods under different protocols** on the VisDA-C dataset. We report the class top-1 accuracies (%) for each of the 12 classes. We also report the overall accuracy (Acc.) and the per-class average accuracy (Avg.). Each result is averaged over three seeds.

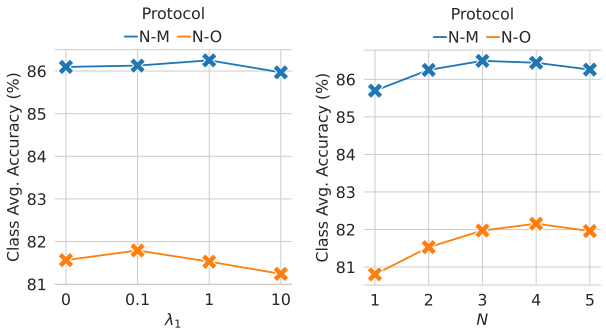
Method	Protocol	plane	bicycle	bus	car	horse	knife	mcycl	person	plant	sktbrd	train	truck	Acc.	Avg.
Source	N	3.8±4.5	23.3±0.7	56.0±3.9	82.5±0.9	70.8±3.1	1.6±0.3	84.4±1.3	9.1±2.2	78.0±5.4	22.1±3.7	79.3±2.4	1.6±0.7	55.6±0.7	48.5±1.0
BN	N-O	86.9±2.2	57.8±2.3	75.4±1.2	52.9±1.3	86.7±0.6	54.2±4.0	85.5±0.9	55.4±2.0	64.9±2.7	41.6±2.3	85.7±1.2	28.8±2.5	64.5±0.3	64.6±0.5
Tent	N-O	86.9±2.2	57.7±3.0	77.4±1.4	56.8±1.5	87.3±0.8	62.4±3.8	86.6±0.8	62.9±2.9	71.2±1.7	39.9±2.8	84.8±1.2	24.7±3.4	66.3±0.3	66.5±0.6
SHOT	N-O	90.5±1.0	77.0±0.9	76.2±0.7	47.5±0.5	87.9±0.2	62.1±4.0	75.9±0.2	74.4±1.1	83.3±0.3	47.0±6.6	84.2±0.9	41.6±0.4	68.6±0.6	70.6±1.0
AdaContrast	N-O	95.2±0.3	78.2±0.3	81.8±0.1	67.9±1.2	94.9±0.5	87.4±3.3	**87.9±0.6**	82.0±1.5	90.7±0.7	36.8±16.1	**88.6±0.1**	31.5±3.6	76.2±0.7	76.9±1.4
TTT++	Y-O	86.4±1.5	60.5±2.6	75.7±2.2	51.7±3.6	86.5±0.9	55.3±2.1	85.2±2.7	55.8±1.1	64.5±2.7	41.3±2.1	86.4±1.9	28.4±2.6	64.4±0.8	64.8±0.7
TTAC	Y-O	90.0±1.2	64.7±12.5	69.7±0.9	48.5±1.7	84.3±1.8	82.8±3.6	84.7±4.1	64.7±7.2	72.1±1.3	40.2±6.3	86.5±1.2	25.5±5.6	65.5±1.6	67.8±2.1
TeSLA	N-O	**95.4±0.2**	**87.4±0.2**	**83.8±0.6**	**70.1±0.8**	**95.1±0.1**	90.0±1.0	84.8±3.1	**83.2±1.3**	**93.6±0.1**	**67.9±19.9**	85.4±0.8	**49.3±1.2**	**80.3±1.3**	**82.2±1.9**
TeSLA-s	Y-O	92.0±0.2	81.2±2.0	77.1±1.9	56.5±0.9	90.2±0.4	**91.0±0.9**	82.9±1.8	79.8±0.8	91.3±0.1	48.9±3.5	81.2±1.5	40.1±2.4	73.5±0.3	76.0±0.3
BN	N-M	87.2±1.4	58.0±1.1	76.4±1.4	53.7±1.9	87.2±1.3	54.2±3.6	86.2±0.3	55.5±1.6	64.9±2.3	42.1±2.7	85.6±1.3	29.3±2.2	64.9±0.1	65.0±0.4
Tent	N-M	89.1±2.0	56.4±5.9	82.4±1.0	69.2±0.5	89.3±1.3	95.2±0.5	**91.4±0.5**	79.5±1.0	86.1±0.3	16.3±1.9	84.7±0.4	8.4±3.5	70.9±0.4	70.7±0.6
SHOT	N-M	93.9±0.5	82.6±0.7	76.6±0.8	49.7±1.8	92.0±0.2	79.0±21.6	75.3±2.0	80.9±2.4	89.5±0.6	50.5±19.0	83.8±0.9	52.2±1.1	72.7±1.8	75.5±3.4
AdaContrast	N-M	95.6±0.6	82.8±1.0	76.5±2.4	**72.4±5.3**	**96.7±0.3**	91.3±2.2	88.6±1.2	**85.4±0.8**	95.3±0.5	30.1±51.3	**93.6±0.7**	48.9±2.1	79.7±1.3	79.8±3.9
TTT++	Y-M	87.2±2.0	61.8±2.0	74.7±1.3	52.7±3.6	86.1±1.7	65.0±7.0	84.9±2.3	62.1±6.0	67.2±1.6	36.6±1.3	86.2±0.1	27.1±3.6	65.3±0.3	65.9±1.0
TTAC	Y-M	86.8±4.2	73.5±1.3	69.3±2.1	44.2±2.5	78.8±5.1	73.1±6.7	84.7±1.6	67.3±8.6	78.6±5.6	52.9±4.1	84.7±2.6	33.2±3.9	66.0±2.0	68.9±2.4
TeSLA	N-M	**96.6±0.2**	**91.3±0.1**	**85.1±1.0**	69.3±0.0	**96.7±0.3**	**97.1±0.8**	88.0±0.9	85.2±0.4	**96.3±0.2**	**87.7±9.3**	87.4±0.2	**57.3±0.8**	**83.4±0.6**	**86.5±0.9**
TeSLA-s	Y-M	96.1±0.4	89.4±0.4	83.0±0.4	62.4±0.5	94.4±0.1	94.5±1.1	87.3±0.3	83.3±0.5	95.5±0.2	63.9±14.9	85.7±0.7	49.4±3.3	79.3±1.0	82.1±1.5

Table B.4. **Segmentation results** for test-time adaptation methods (class Avg. volume-wise mean Dice score in %) on the spinal cord dataset (site $\{1\} \rightarrow 2, 3, 4$) and prostate dataset (sites $\{A, B\} \rightarrow D, E, F$), respectively.

Method	Protocol	Spinal Cord				Prostate			
Sites		$\{1\} \rightarrow \{2\}$ Class Avg.	$\{1\} \rightarrow \{3\}$ Class Avg.	$\{1\} \rightarrow \{4\}$ Class Avg.	$\{1\} \rightarrow \{2,3,4\}$ Avg.	$\{A,B\} \rightarrow \{D\}$ Class Avg.	$\{A,B\} \rightarrow \{E\}$ Class Avg.	$\{A,B\} \rightarrow \{F\}$ Class Avg.	$\{A,B\} \rightarrow \{D,E,F\}$ Avg.
Source	N	77.4±6.6	64.8±11.7	85.9±3.8	76.0 ±11.8	75.8±8.9	65.9±18.5	38.4±32.3	60.5±27.0
BN	N-O	85.2±2.1	70.6±3.6	88.9±1.7	81.6±8.3	75.9±9.4	74.4±7.4	65.7±22.4	72.1±15.2
TENT	N-O	85.7±1.8	68.7±2.8	88.9±1.7	81.1±9.1	78.8±6.2	77.9±6.9	67.0±28.4	74.7±17.9
PL	N-O	85.3±2.1	71.0±3.6	88.9±1.7	81.7±8.6	76.1±9.4	74.8±7.5	66.2±22.4	72.4±15.2
OptTTA	N-O	84.4±2.3	80.2±5.1	87.5±2.0	84.1±4.8	84.9±6.9	**80.3±8.4**	84.0±6.6	83.1±7.7
TeSLA	N-O	**86.3±1.5**	**80.3±7.3**	**89.3±1.4**	**85.3±5.8**	**86.1±3.3**	79.8±7.5	**84.3±6.3**	**83.5±6.5**
BN	N-M	85.5±1.6	78.5±3.2	88.8±1.5	84.3±4.8	77.8±9.6	77.3±7.2	63.8±26.7	73.1±18.0
TENT	N-M	85.5±1.6	79.0±3.3	88.8±1.5	84.4±4.7	81.6±7.7	79.0±10.4	82.8±9.2	81.2±9.3
PL	N-M	85.5±1.7	78.8±3.3	88.8±1.5	84.3±4.7	81.2±7.9	79.1±10.1	82.8±9.1	81.1±9.2
OptTTA	N-M	84.3±2.5	**80.7±4.9**	87.7±2.0	84.3±4.4	**86.2±5.2**	78.6±8.6	85.0±6.7	83.4±7.7
TeSLA	N-M	**86.4±1.7**	80.4±3.2	**89.3±1.7**	**85.4±4.4**	85.9±4.0	**81.2±6.7**	**85.6±5.4**	**84.3±5.8**

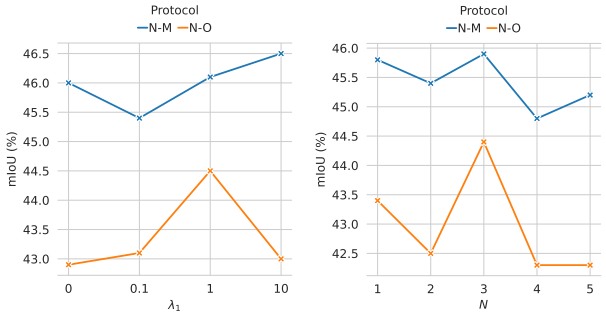
Table B.5. **Segmentation results** for test-time adaptation methods (mIoU%) for adaptation from synthetic GTA5 dataset to Cityscapes dataset (O and M protocols).

Method	Protocol	road	sidewalk	building	wall	fence	pole	light	sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motocycle	bicycle	mIoU
Source	N	72.9	20.0	81.4	21.7	22.9	19.2	25.3	10.6	78.9	26.4	85.9	54.9	20.7	53.0	30.6	16.2	1.9	20.0	7.5	35.3
BN	N-O	84.3	31.8	79.2	24.1	20.5	21.5	23.5	10.5	74.6	32.0	75.3	52.1	14.4	77.6	28.1	20.6	6.0	14.9	6.2	36.7
PL	N-O	84.0	31.1	80.6	25.5	20.7	21.5	24.7	11.4	77.3	34.0	**79.4**	54.2	17.1	78.3	30.1	21.7	9.7	18.5	8.1	38.3
Tent	N-O	88.0	34.3	80.7	27.7	17.8	19.3	22.1	10.0	**80.1**	**40.5**	77.6	51.8	15.7	81.8	**32.6**	24.0	8.9	18.8	5.8	38.8
CoTTA	N-O	85.8	35.3	79.1	26.5	20.3	19.8	21.7	9.9	76.7	36.2	74.6	53.2	14.4	77.8	29.0	19.3	3.6	13.2	5.8	37.0
TeSLA	N-O	**90.4**	**52.2**	**82.5**	**29.6**	**25.5**	**28.1**	**32.5**	**29.7**	79.7	39.0	75.2	**59.0**	**21.3**	**84.0**	29.3	**24.6**	**14.5**	**23.0**	**26.1**	**44.5**
BN	N-M	84.3	31.1	80.7	25.4	21.0	22.6	25.6	11.8	76.7	32.7	77.6	54.8	17.2	79.7	29.7	21.7	9.3	18.7	8.5	38.4
PL	N-M	85.1	30.6	80.9	25.7	20.6	21.5	24.7	11.3	77.9	33.9	**80.2**	54.4	17.4	80.0	30.0	21.9	9.2	19.0	8.3	38.6
Tent	N-M	89.0	35.1	81.0	28.4	17.0	19.5	22.9	9.8	**80.8**	**41.8**	76.7	52.4	16.3	83.6	**33.0**	24.8	7.1	20.6	5.7	39.2
CoTTA	N-M	88.6	40.2	80.6	**30.0**	20.4	19.2	25.9	16.0	77.1	32.7	75.3	55.7	23.1	82.8	30.1	19.4	9.8	19.9	11.3	39.9
TeSLA	N-M	**90.1**	**51.4**	**83.1**	29.0	**27.7**	**28.7**	**34.8**	**34.0**	78.7	35.7	73.0	**62.0**	**26.5**	**83.9**	28.5	**25.0**	**25.7**	**27.3**	**29.4**	**46.0**



(a) (b)

Figure E.1. **Sensitivity test for adversarial augmentation hyperparameters** of TeSLA on the VisDA-C dataset for classification task on various TTA protocols. We plot the class Avg. accuracy (%) on the VisDA-C dataset for (a) augmentation severity controller $\lambda_1 \in \{0, 0.1, 1, 10\}$ and (b) sub-policy dimension $N \in [1, 5]$.

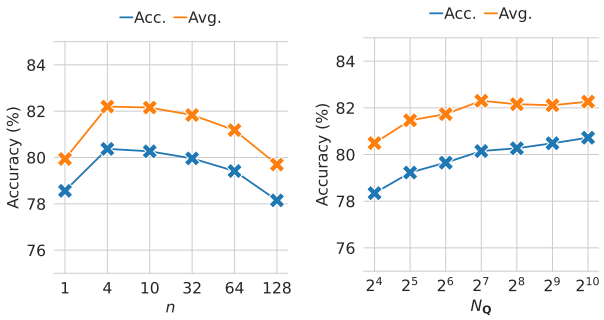


(a) (b)

Figure E.2. **Sensitivity test for adversarial augmentation hyperparameters** of TeSLA on the VisDA-S dataset for segmentation task on various TTA protocols. We plot the mIoU (%) on the VisDA-S dataset for (a) augmentation severity controller $\lambda_1 \in \{0, 0.1, 1, 10\}$ and (b) sub-policy dimension $N \in [1, 5]$, respectively.

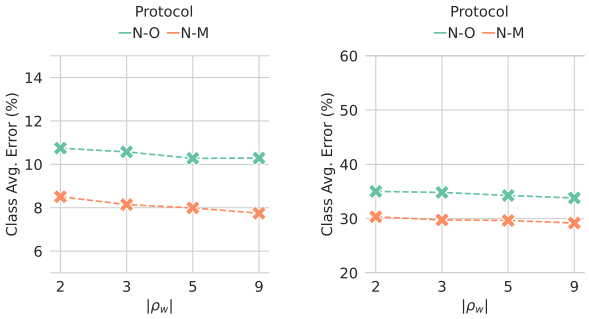
on the VisDA-C dataset. Similarly, Fig. E.2 shows the effect of changing λ_1 and N on the segmentation scores measured by mIoU (%) on the VisDA-S dataset. We observe that the performance of our method TeSLA is stable over a wide range of λ_1 and N for both classification and segmentation tasks.

PLR hyperparameters. We present sensitivity tests for the hyperparameters of the soft pseudo-label refinement (PLR) module. In Fig. E.3, we show the test-time adaptation classification performance of TeSLA on the VisDA-C (N-O) for varying numbers of nearest neighbors $n \in \{1, 4, 10, 32, 64, 128\}$, and class memory queue size $N_{\mathbf{Q}} \in \{16, 32, 64, 128, 256, 512, 1024\}$. TeSLA outperforms competing baselines under a wide range of choices. Moreover, the number of examples in the queue can be as small as



(a) (b)

Figure E.3. **Sensitivity test for PLR hyperparameters**: (a) the number of nearest neighbors n and (b) the class memory queue size $N_{\mathbf{Q}}$ on the VisDA-C dataset. We report the overall accuracy (Acc.) and the class average accuracy (Avg.) in % on the ViSDA-C under the N-O protocol.

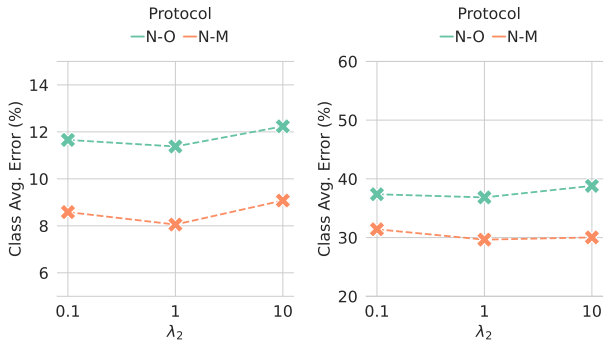


(a) CIFAR10-C (b) CIFAR100-C

Figure E.4. **Sensitivity test on the number of weak augmentation** ($|\rho_w|$) for ensembling soft-pseudo labels for Soft-Pseudo Label Refinement (PLR) on (a) CIFAR10-C and (b) CIFAR100-C datasets. We report, for each case, the average error rate over 4 validation corruptions.

less than 0.5% of the dataset size and still maintains on-par performance.

Fig. E.4 shows the classification performance of TeSLA on the CIFAR10-C and CIFAR100-C and various corruptions [Gaussian Blur, Spatter, Speckle Noise, Saturate] under multiple protocols and the number of weak augmentation for ensembling $|\rho_w| \in \{2, 3, 5, 9\}$. These plots show that increasing the number of views decreases the average error rate in both protocols. While we report the results with $|\rho_w| = 5$ in the main, we observe we could further decrease the error rate with $|\rho_w| = 9$. However, this choice would multiply the computational cost by two as the number of forward passes is linearly proportional to this hyperparameter. For this reason, we opt $|\rho_w| = 5$, which gathers the benefit of ensembling and reasonable computational cost.



(a) CIFAR10-C (b) CIFAR100-C

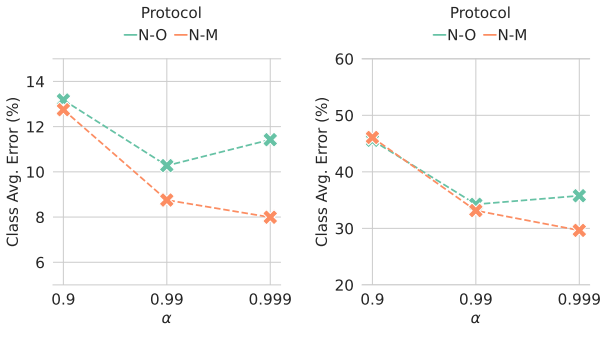
Figure E.5. **Sensitivity test on the scalar coefficient** λ_2 of $\mathcal{L}_{\text{kd}}$ term of the test time loss $\mathcal{L}_{\text{TeSLA}}$. We report, for each case, the average error rate over the 4 validation corruptions.

Knowledge distillation coefficient. Finally, we report the sensitivity test results for the knowledge distillation weight λ_2. In Fig. E.5, we show the classification adaptation performance of TeSLA on the CIFAR10-C and CIFAR100-C datasets is not very sensitive to the selection of $\lambda_2 \in \{0.1, 1, 10\}$.

E.2. Ablations

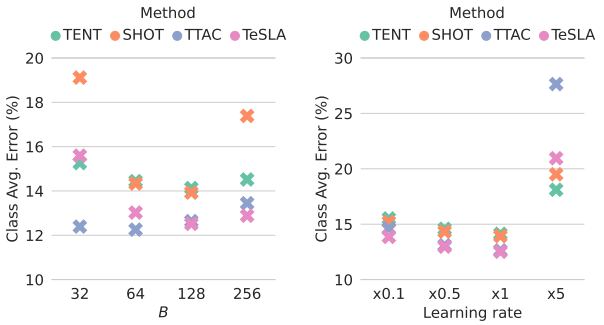
EMA coefficient of teacher model. In Fig. E.6, we show the effect of changing the EMA coefficient α used for updating the teacher model from the student model for the one-pass (O protocol) and multi-pass (M protocol) on the CIFAR10-C and CIFAR100-C datasets. We observe that for the multi-pass protocol (M), decreasing α leads to better performance, while for the one-pass protocol (O), optimal α depends on the number of test images observed in one epoch. If α is large (close to 1.0), the teacher is updated very slowly and thus requires more updates to reach better performance. Therefore, for a one-pass online evaluation, the accuracy decreases. On the other hand, if we set α to a minimal value, it results in unstable convergence.

Batch size and learning rate. In Fig. E.7, we show the effect of batch size and learning rate on the proposed method TeSLA along with TENT [14], SHOT [6], and TTAC [13] on CIFAR10-C for N-O protocol. We observe that increasing batch size helps reduce test time error rates, and the model performs best with the same batch size used during source model training. Similarly, increasing the learning rate reduces the error rate until it becomes too large for unstable gradient model updates.



(a) CIFAR10-C (b) CIFAR100-C

Figure E.6. **Sensitivity test on the EMA coefficient of the teacher model** α on the (a) CIFAR10-C and (b) CIFAR100-C datasets. We report the average error rate over four corruptions for each dataset.



(a) (b)

Figure E.7. **Ablation study** for the roles played by the (a) batch size B and (b) the learning rate scale with respect to the default value. We report, for each baseline, the average error rate (in %) over four validation corruption sets of the CIFAR10-C under the N-O protocol.

F. Additional Qualitative Results

F.1. Sanity Check for Adversarial Augmentation

To assess the adversarial effect of the proposed automatic augmentations, we conduct a sanity check for the optimized sub-policies. in particular, in Fig. F.1, we rank the sub-policies optimized by our automatic augmentation module on the VisDA-C for N-M protocol after one epoch by decreasing the order of sampling probability. Then, we evaluate the performance of the student model on the test-test images from VisDA-C that are augmented using the above sub-policies. We observe that reducing the hardness level of sub-policies, the more the student model is accurate in recognizing the images. This is supported by the Pearson correlation of 0.7 between the sub-policy rank and the accuracy ($p = 0.02$), demonstrating our module's capability to optimize and sample adversarial examples.

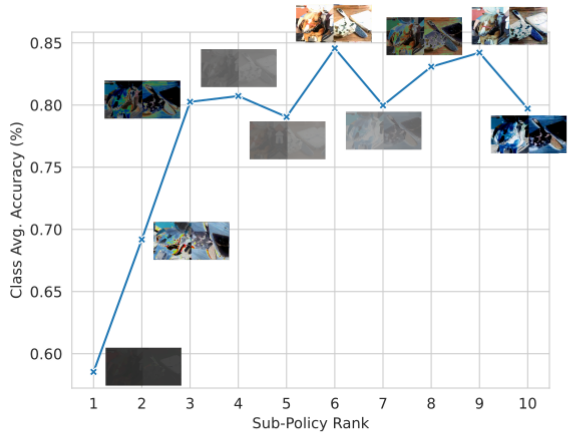


Figure F.1. **Adversarial augmentation sanity check.** We report the per-class average accuracy (%) of TeSLA's student model on the VisDA-C augmented by the ten most adversarial sub-policies optimized by our automatic augmentation module.

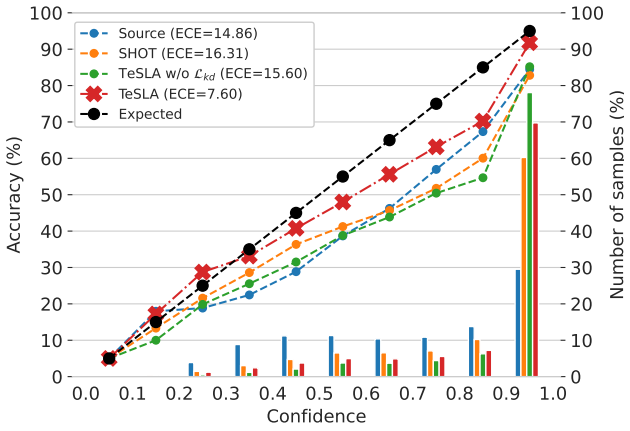
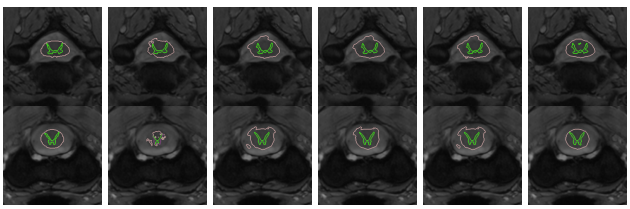
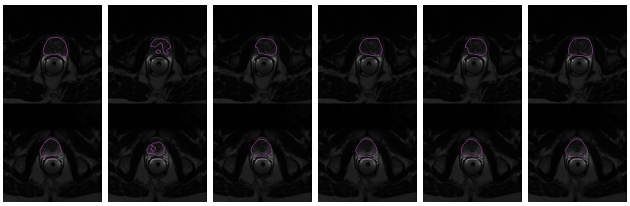


Figure F.2. **Calibration performance comparison** of TeSLA (with and without adversarial augmentations) against other baselines via a reliability diagram on the VisDA-C dataset for N-O protocol.



(a) (b) (c) (d) (e) (f)

Figure F.3. **Qualitative segmentation results** of test-time adaptation methods trained on **site 1** and tested on **site 3** of the spinal cord dataset. From left to right: (a) Ground Truth, (b) Source Model, (c) BN [8], (d) TENT [14], (e) PL, and (f) TeSLA, respectively.



(a) (b) (c) (d) (e) (f)

Figure F.4. **Qualitative segmentation results** of test-time adaptation method trained on **site A** and **site B** and tested on **site F** of the prostate dataset. From left to right: (a) Ground Truth, (b) Source Model, (c) BN [8], (d) TENT [14], (e) PL, and (f) TeSLA, respectively.

F.2. Uncertainty Evaluation

We evaluate the model reliability on the ViSDA-C classification adaptation task (N-O protocol). In Fig. F.2, we show the reliability diagram (dividing the probability range [0, 1.0] into ten bins) and report the expected calibration error (ECE) [9] for the Source model without adaptation, SHOT [6], and TeSLA with and without adversarial augmentations. The proposed TeSLA gives the lowest calibration error with an 8.71% improvement over SHOT. It is interesting to observe that the ECE of TeSLA without adversarial augmentations is on-par with the SHOT method. The adversarial augmentation module improves the TeSLA's ECE by 8%, which shows the benefit of test-time adversarial augmentation on the model's reliability.

F.3. Qualitative Segmentation Results

In Fig. F.3 and Fig. F.4, we show the qualitative segmentation results of TeSLA for test-time adaptation on the spinal cord and prostate MRI datasets and compare it with TENT [14], BN [8], and Pseudo Labeling (PL). Compared to other baselines, TeSLA outputs more accurate segmentation results closer to the provided ground truth.

References

[1] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305, 2022. 2

[2] 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 *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[3] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 1

[4] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. 2

[5] Jakob Nikolas Kather, Frank Gerrit Zöllner, Francesco Bianconi, Susanne M Melchers, Lothar R Schad, Timo Gaiser, Alexander Marx, and Cleo-Aron Weis. Collection of textures in colorectal cancer histology, May 2016. 1

[6] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020. 1, 2, 6, 7

[7] Quande Liu, Qi Dou, Lequan Yu, and Pheng Ann Heng. Ms-net: Multi-site network for improving prostate segmentation with heterogeneous mri data. *IEEE Transactions on Medical Imaging*, 2020. 1, 2

[8] 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. 2, 7

[9] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632, 2005. 7

[10] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge, 2017. 1

[11] Ferran Prados, John Ashburner, Claudia Blaiotta, Tom Brosch, Julio Carballido-Gamio, Manuel Jorge Cardoso, Benjamin N Conrad, Esha Datta, Gergely Dávid, Benjamin De Leener, et al. Spinal cord grey matter segmentation challenge. *Neuroimage*, 152:312–329, 2017. 1, 2

[12] Stephan R. Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *European Conference on Computer Vision (ECCV)*, volume 9906 of *LNCS*, pages 102–118. Springer International Publishing, 2016. 2

[13] Yongyi Su, Xun Xu, and Kui Jia. Revisiting realistic test-time training: Sequential inference and adaptation by anchored clustering. In *Thirty-Sixth Conference on Neural Information Processing Systems*, 2022. 2, 6

[14] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. 1, 2, 6, 7