

Supplementary Material to ‘Adversarial Style Augmentation for Domain Generalization’

Yabin Zhang¹, Bin Deng², Ruihuang Li¹, Kui Jia², Lei Zhang¹

¹ The Hong Kong Polytechnic University ² South China University of Technology
csybzhang@comp.polyu.edu.hk

The following materials are provided in this supplementary file:

- Domain generalization results in the segmentation task (cf. Section 4 in the main paper).
- Domain generalization results to images with corruptions (cf. Section 4 in the main paper).
- More domain generalization results under the leave-one-domain-out setting (cf. Section 4.1 in the main paper).
- In-depth analyses on the adversarial style augmentation.
- More visualization of feature representations (cf. Section 4.3 in the main paper).

A1 Domain generalization results in the segmentation task

Implementation details. We validate the effectiveness of our ASA method with AdvStyle module on cross-domain semantic segmentation task. Specifically, we train the model on GTA5 (Richter et al. 2016) and test its generalization performance on real-world samples. The GTA5 is a synthetic dataset generated from the Grand Theft Auto 5 game engine, including 24,966 synthetic images shared by 19 semantic classes. All images are with pixel-level semantic annotations and collected from the car perspective in the American-style virtual cities. We adopt the Cityscapes dataset (Cordts et al. 2016), whose samples are collected from 50 cities, as the real-world dataset. All experiments are conducted with a DeepLab-v2 (Chen et al. 2017) segmentation model, where ResNet-101 is used as the backbone. We closely follow FADA (Wang et al. 2020) code base to conduct the comparisons and adopt the Mean Intersection over Union (mIoU) and mean Accuracy (mAcc) as the evaluation metric.

Results. As illustrated in Tab. A1, our AdvStyle outperforms existing style augmentation modules in term of both mIoU and mAcc. Specifically, our method outperforms the closest competitor DSU by 1.6% on mIoU, demonstrating the effectiveness of adversarial style augmentation. Visualization results of different methods can be found in Fig. A1. Our method generates more accurate segmentation results on the flat road and challenging cars in the distance, illustrating the advantage of our method clearly.

A2 Domain generalization results to images with corruptions

Implementation details. We also validate the generalization performance of our method on a specific domain divergence, image corruptions. Specifically, we train models on the training data of vanilla Cifar10 and Cifar100 datasets and test the model performance on the corrupted Cifar10-C and Cifar100-C datasets (Hendrycks and Dietterich 2019), which include 15 corruptions: Gaussian Noise, Shot Noise, Impulse Noise, Defocus Blur, Frosted Glass Blur, Motion Blur, Zoom Blur, Snow, Frost, Fog, Brightness, Contrast, Elastic, Pixelate, JPEG. These corruptions are classified into four super-categories: Noise, Blur, Weather, and Digital. We conduct all experiments based on a WideResNet backbone. Besides the widely-used mean error across corruptions, we also report the standard deviation of errors (*i.e.*, Std. Error) across corruptions, which measures the model fluctuation on various corruptions. In other words, a smaller Std. Error represents smaller performance fluctuation across corruptions and, therefore, more robust generalization ability.

Results. As illustrated in Tab. A2, generalization to image corruptions could also be enhanced by augmenting styles in the feature space. Our AdvStyle achieves the lowest mean error across different image corruptions among all style augmentation-based methods, justifying the efficacy of adversarial style augmentation. As an additional benefit, our AdvStyle also achieves the lowest Std. Error, demonstrating the lowest performance fluctuation across corruptions and the most robust generalization ability. We also find that although some methods (*e.g.*, DSU (Li et al. 2022)) reduce the mean error on image corruptions, the standard deviation of errors is increased. In other words, the performance divergence on different corruptions is enlarged. It is also noted that such methods prefer certain corruptions.

A3 More domain generalization results under the leave-one-domain-out setting

The results are shown in Tab. A3. Compared to the representative DG algorithms (Motiian et al. 2017; Li et al. 2018; Carlucci et al. 2019; Shankar et al. 2018; Balaji, Sankaranarayanan, and Chellappa 2018; Li et al. 2019; Zhou et al. 2020), the methods based on style augmentations present comparable performance with less modification on model

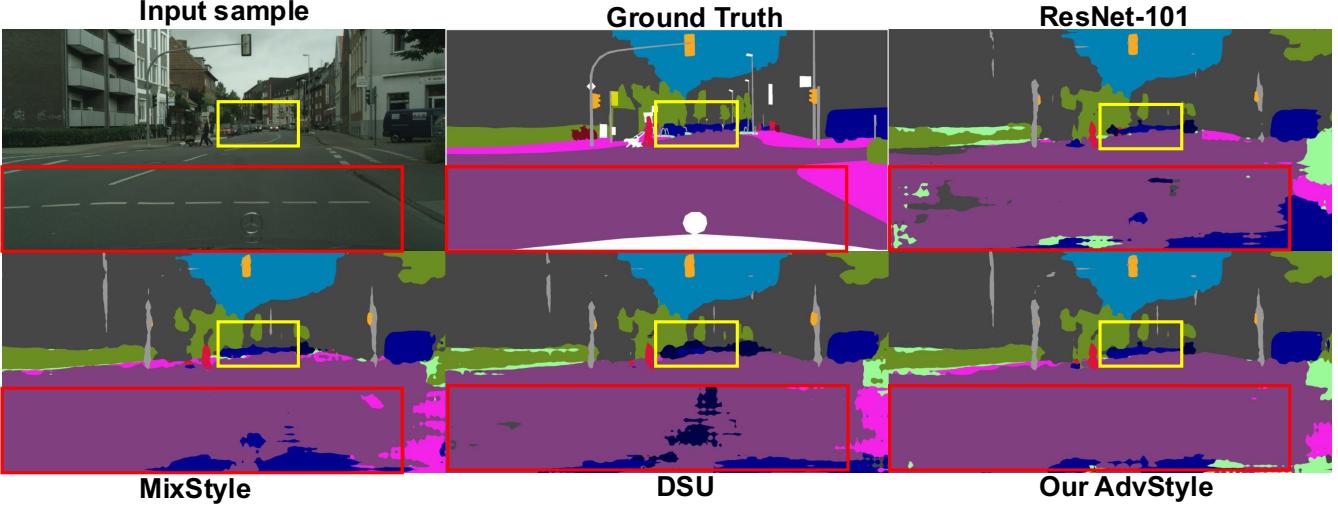


Figure A1: Visualization on the real-world dataset of Cityscapes, where models are trained on the synthetic GTA5 dataset.

	Ro.	SW	Bu.	Wa.	Fe.	Po.	TL	TS	Ve.	Te.	Sk.	PR	Ri.	Ca.	Tr.	Bu.	Tr.	Mo.	Bi.	mIoU
ResNet-101	57.9	17.4	71.5	19.3	18.3	25.4	32.5	16.8	82.3	28.2	55.3	31.3	71.6	19.1	26.8	9.2	26.3	13.7	37.0	
+ pAdaIN	57.2	20.2	71.6	28.3	19.1	26.1	33.6	13.0	82.1	29.0	69.5	56.7	33.0	67.5	27.8	35.1	17.6	33.7	14.5	38.7
+ MixStyle	60.4	17.3	77.2	30.0	20.4	27.0	31.6	15.5	81.2	28.4	73.3	56.0	32.9	68.1	27.7	28.6	16.5	29.2	17.1	38.8
+ DSU	75.4	19.8	69.3	24.2	20.1	24.6	33.8	16.0	82.4	26.6	79.6	56.7	33.7	69.0	29.1	30.9	18.3	31.2	25.1	40.3
+ AdvStyle	76.7	21.7	74.6	29.1	20.9	28.3	34.4	16.5	83.4	28.4	79.3	57.7	33.5	77.3	30.0	33.9	13.5	33.5	23.2	41.9

Table A1: Cross-domain semantic segmentation results on GTA5 → Cityscapes. .

construction or optimization strategy. The style augmentation methods significantly outperform traditional augmentation strategies (Zhang et al. 2017; Verma et al. 2019; Ghiasi, Lin, and Le 2018; Yun et al. 2019; DeVries and Taylor 2017) in DG by explicitly taking the domain divergence into consideration. Compared to other style augmentation strategies (Nuriel, Benaim, and Wolf 2021; Zhou et al. 2021; Li et al. 2022), our method presents clear advantages, which is consistent to the observations under the single source domain generalization setting in the main paper.

A4 Analyses on the adversarial style augmentation

By expanding the style representation space via the proposed adversarial statistics perturbation, we significantly improved the cross-domain generalization performance as shown in Tab. 1 and Tab. 2 of the main paper. We also note that the performance on easy tasks may be deteriorated if the model is over-adapted to the worst-case domain, as presented in Tab. A4 here. To tackle the easy tasks and hard tasks simultaneously, we propose to combine the advantages of the existing style augmentation strategy and our proposed adversarial style augmentation. In other words, we can expand the training data with both strategies of style augmentation in practice. As illustrated in Tab. A4, taking the advantages of both style augmentation strategies results in more balanced performance across tasks, which is adopted as the default setting in this paper. Moreover, these results validate the complementarity of our adversarial style augmentation and existing style augmentation strategies.

A5 More visualization of feature representations

As illustrated in Fig. A2 and Fig. A3, our AdvStyle consistently outperforms the vanilla ResNet50 baseline and the closest competitor-DSU.

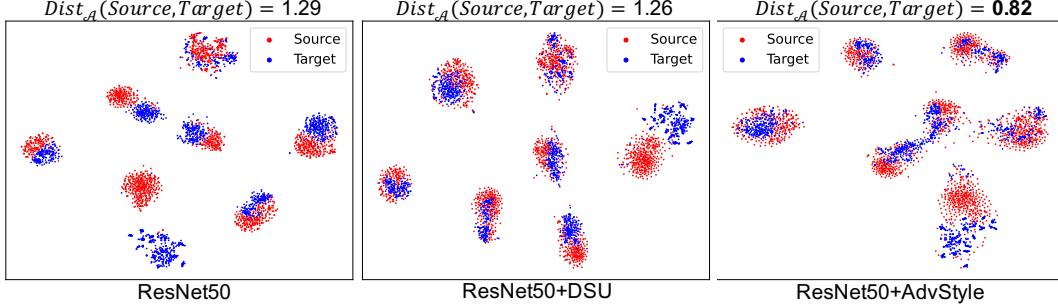


Figure A2: T-SNE (Van der Maaten and Hinton 2008) and \mathcal{A} -distance (*i.e.*, $Dist_{\mathcal{A}}$) (Ben-David et al. 2010) of the feature representations on the PACS dataset, where we adopt the Art and Photo as the source and target domains, respectively.

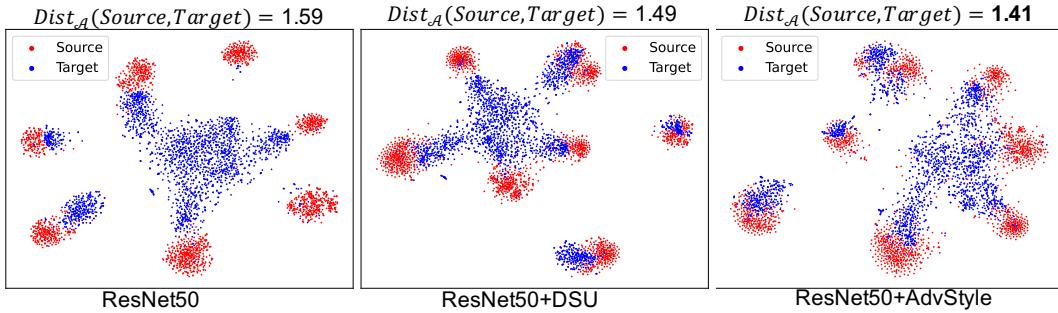


Figure A3: T-SNE (Van der Maaten and Hinton 2008) and \mathcal{A} -distance (*i.e.*, $Dist_{\mathcal{A}}$) (Ben-David et al. 2010) of the feature representations on the PACS dataset, where we adopt the Art and Cartoon as the source and target domains, respectively.

	Cifar10-C																
	Noise			Blur				Weather				Digital				mCE \downarrow	Std. Error \downarrow
	N1	N2	N3	B1	B2	B3	B4	W1	W2	W3	W4	D1	D2	D3	D4		
Baseline	52.9	40.8	43.2	18.1	46.0	21.6	24.6	16.5	20.9	11.4	6.0	21.9	16.5	25.4	21.8	25.8	13.6
+ pAdaIN	49.6	38.1	40.8	16.6	42.5	18.7	18.1	16.9	19.1	10.4	6.7	18.1	15.2	23.2	20.3	23.6	12.7
+ MixStyle	50.0	39.1	42.6	14.6	42.1	17.8	17.0	16.5	18.8	10.7	6.5	17.8	14.6	22.1	20.5	23.4	12.9
+ DSU	53.9	41.7	41.8	12.8	43.9	17.3	17.2	15.5	17.6	8.7	5.9	14.3	15.3	22.7	21.8	23.4	14.6
+ AdvStyle	40.5	31.9	33.5	13.3	45.7	18.8	18.9	18.7	21.0	9.6	6.8	9.6	16.4	26.1	19.2	22.0	11.4

	Cifar100-C																
	Noise			Blur				Weather				Digital				mCE \downarrow	Std. Error \downarrow
	N1	N2	N3	B1	B2	B3	B4	W1	W2	W3	W4	D1	D2	D3	D4		
Baseline	78.1	69.6	75.3	39.4	77.9	46.7	46.2	45.2	52.0	36.4	27.0	45.2	41.2	50.0	52.4	52.2	15.8
+ pAdaIN	75.3	68.1	70.1	38.6	77.7	44.3	45.7	47.5	50.1	35.7	28.0	37.1	40.9	49.4	51.2	50.7	15.1
+ MixStyle	74.3	68.5	69.8	38.4	77.4	44.0	45.9	47.8	49.1	36.1	28.2	37.4	41.0	49.0	50.6	50.4	15.0
+ DSU	78.2	69.9	73.4	37.4	78.5	42.8	43.8	45.1	49.7	34.1	27.4	38.0	40.7	48.9	53.6	50.5	16.5
+ AdvStyle	71.5	63.9	65.9	39.0	77.4	44.0	44.4	47.0	50.0	36.7	28.8	35.3	41.9	48.7	49.1	49.8	14.1

Table A2: Corruption-wise generalization errors on Cifar10-C and Cifar100-C (Hendrycks and Dietterich 2019), where N1, N2, N3, B1, B2, B3, B4, W1, W2, W3, W4, D1, D2, D3, D4 represent corruptions of Gaussian Noise, Shot Noise, Impulse Noise, Defocus Blur, Frosted Glass Blur, Motion Blur, Zoom Blur, Frost, Fog, Brightness, Contrast, Elastic, Pixelate, JPEG, respectively.

Method	Art	Cartoon	Photo	Sketch	Mean ↑	Std. ↓
MMD-AAE (Li et al. 2018)	75.2	72.7	96.0	64.2	77.0	13.5
CCSA (Motiian et al. 2017)	80.5	76.9	93.6	66.8	79.4	11.1
JiGen (Carlucci et al. 2019)	79.4	75.3	96.0	71.6	80.5	10.8
CrossGrad (Shankar et al. 2018)	79.8	76.8	96.0	70.2	80.7	11.0
Epi-FCR (Li et al. 2019)	82.1	77.0	93.9	73.0	81.5	9.1
Metareg (Balaji, Sankaranarayanan, and Chellappa 2018)	83.7	77.2	95.5	70.3	81.7	10.7
L2A-OT (Zhou et al. 2020)	83.3	78.2	96.2	73.6	82.8	9.8
ResNet-18	77.0±0.6	75.9±0.6	96.0±0.1	69.2±0.6	79.5	11.5
+ Manifold Mixup (Verma et al. 2019)	75.6±0.7	70.1±0.9	93.5±0.7	65.4±0.6	76.2	12.3
+ Cutout (DeVries and Taylor 2017)	74.9±0.4	74.9±0.6	95.9±0.3	67.7±0.9	78.3	12.2
+ CutMix (Yun et al. 2019)	74.6±0.7	71.8±0.6	95.6±0.4	65.3±0.8	76.8	13.1
+ Mixup (Zhang et al. 2017)	76.8±0.7	74.9±0.7	95.8±0.3	66.6±0.7	78.5	12.3
+ DropBlock (Ghiasi, Lin, and Le 2018)	76.4±0.7	75.4±0.7	95.9±0.3	69.0±0.3	79.2	11.6
Following tables: ResNet + Style augmentation modules						
+ pAdaIN (Nuriel, Benaim, and Wolf 2021)	81.7±0.4	76.6±0.8	96.3±0.2	75.1±0.5	82.5	9.7
+ MixStyle (Zhou et al. 2021)	82.4±0.2	79.4±0.8	96.2±0.1	72.3±0.6	82.6	10.0
+ DSU (Li et al. 2022)	81.5±2.3	78.3±0.6	95.8±0.1	75.1±0.4	82.7	9.1
+ AdvStyle (ours)	81.5±0.7	78.3±0.8	95.8±0.2	76.1±0.2	83.0	8.9
ResNet-50	84.4±0.9	77.1±1.4	97.6±0.2	70.8±0.7	82.5	11.5
+ pAdaIN (Nuriel, Benaim, and Wolf 2021)	87.5±0.9	80.3±0.6	98.0±0.2	75.5±0.8	85.3	9.8
+ MixStyle (Zhou et al. 2021)	88.7±0.7	81.4±0.7	98.0±0.2	75.0±0.6	85.8	9.9
+ DSU (Li et al. 2022)	88.6±0.5	81.7±0.6	97.3±0.2	77.4±0.9	86.3	8.7
+ AdvStyle (ours)	85.2±0.5	83.3±0.1	96.2±0.4	83.4±0.7	87.0	6.2

Table A3: Domain generalization results of classification on the PACS dataset under the leave-one-domain-out setting, where the listed domain is adopted for testing and models are trained on the remaining three domains.

Method	Acc. (%)
Leave-one-domain-out generalization results	
DSU Only (Li et al. 2022)	86.3
AdvStyle Only	86.5
Both	87.0
Single source generalization results	
DSU Only (Li et al. 2022)	57.3
AdvStyle Only	68.2
Both	67.1

Table A4: Domain generalization performance on the PACS dataset with various style augmentation strategies. The test domain and training domain are listed in the leave-one-domain-out setting and single source generalization setting, respectively. We randomly activate the DSU module or the AdvStyle model as the style augmentation module in the forward pass in the ‘Both’ setting.

References

- Balaji, Y.; Sankaranarayanan, S.; and Chellappa, R. 2018. Metareg: Towards domain generalization using meta-regularization. *Advances in Neural Information Processing Systems*, 31: 998–1008.
- Ben-David, S.; Blitzer, J.; Crammer, K.; Kulesza, A.; Pereira, F.; and Vaughan, J. W. 2010. A theory of learning from different domains. *Machine learning*, 79(1): 151–175.
- Carlucci, F. M.; D’Innocente, A.; Bucci, S.; Caputo, B.; and Tommasi, T. 2019. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2229–2238.
- Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; and Yuille, A. L. 2017. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4): 834–848.
- Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; and Schiele, B. 2016. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3213–3223.
- DeVries, T.; and Taylor, G. W. 2017. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*.
- Ghiasi, G.; Lin, T.-Y.; and Le, Q. V. 2018. Dropblock: A regularization method for convolutional networks. *arXiv preprint arXiv:1810.12890*.
- Hendrycks, D.; and Dietterich, T. 2019. Benchmarking neural network robustness to common corruptions and perturbations. *ICLR*.
- Li, D.; Zhang, J.; Yang, Y.; Liu, C.; Song, Y.-Z.; and Hospedales, T. M. 2019. Episodic training for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1446–1455.
- Li, H.; Pan, S. J.; Wang, S.; and Kot, A. C. 2018. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5400–5409.
- Li, X.; Dai, Y.; Ge, Y.; Liu, J.; Shan, Y.; and Duan, L.-Y. 2022. Uncertainty Modeling for Out-of-Distribution Generalization. *ICLR*.
- Motiian, S.; Piccirilli, M.; Adjeroh, D. A.; and Doretto, G. 2017. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE international conference on computer vision*, 5715–5725.
- Nuriel, O.; Benaim, S.; and Wolf, L. 2021. Permuted AdaIN: reducing the bias towards global statistics in image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9482–9491.
- Richter, S. R.; Vineet, V.; Roth, S.; and Koltun, V. 2016. Playing for data: Ground truth from computer games. In *European conference on computer vision*, 102–118. Springer.
- Shankar, S.; Piratla, V.; Chakrabarti, S.; Chaudhuri, S.; Jyothi, P.; and Sarawagi, S. 2018. Generalizing across domains via cross-gradient training. *ICLR*.
- Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).
- Verma, V.; Lamb, A.; Beckham, C.; Najafi, A.; Mitliagkas, I.; Lopez-Paz, D.; and Bengio, Y. 2019. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, 6438–6447. PMLR.
- Wang, H.; Shen, T.; Zhang, W.; Duan, L.-Y.; and Mei, T. 2020. Classes matter: A fine-grained adversarial approach to cross-domain semantic segmentation. In *European conference on computer vision*, 642–659. Springer.
- Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6023–6032.
- Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.
- Zhou, K.; Yang, Y.; Hospedales, T.; and Xiang, T. 2020. Learning to generate novel domains for domain generalization. In *European Conference on Computer Vision*, 561–578. Springer.
- Zhou, K.; Yang, Y.; Qiao, Y.; and Xiang, T. 2021. Domain generalization with mixstyle. *ICLR*.