Geometric and Textural Augmentation for Domain Gap Reduction: *Supplementary Material*

Xiao-Chang Liu Yong-Liang Yang Peter Hall
University of Bath
{XL2546,yy753,maspmh}@bath.ac.uk

This supplementary material comprises:

- Examples of geometric augmentations on objects from different datasets (Section 1);
- Additional ablation studies on Office-Home and Digits-DG (Section 2): Results show that the sensitivity of different dataset to the augmentation spaces are different, a threshold value near 0.5 is expected to induce a better average performance;
- Additional feature visualizations on Office-Home and Digits-DG (Section 3): By adding geometric and textural augmentation, object clusters become denser and easier to separate;
- Discussions and comparisons between the effectiveness of distribution based augmentation and set/mini-batch based augmentation (Section 4): Results show that representation diversity plays a key role in model robustness improvement. Compared with set/mini-batch based augmentation, leveraging distributions lead to a general improvement in performance due to the significantly improved representation diversity.

1. Geometric Augmentations on Different Objects

We present some geometric augmentation effects on objects from different datasets, as shown in Figure 1. It can be seen that different objects have different deformation types which are suitable for them.

2. Additional Experimental Results

In the paper, our ablation study (Section 5.3) on the influence of different probability thresholds for augmentation on classification (Figure 4 in the main paper) and the visualization of the feature statics are on PACS [2] dataset. Here we show some more results on Office-Home [6] and Digits-DG [7].

In Figure 2 and Figure 4, we show the influence of different probability thresholds on Office-Home and Digits-DG, respectively. Basically, the effectivenesses of geometry and texture augmentations vary between two datasets. For each dataset, it can be seen that in terms of performance on each target domain, the best threshold varies across target domains and augmentation spaces. This is consistent with our conclusions in the paper.

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

Figure 1. Examples of geometric augmentations on objects from different datasets. The first two are from PACS [2]. The middle two are from Office-Home [6]. The last two are from Digits-DG [7]. (a) Original images. (b) – (e) Various geometric deformation effects.

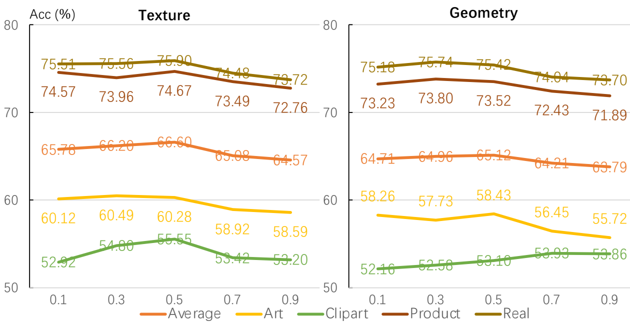

Figure 2. Influence of different probability thresholds for augmentation on classification (left for textural augmentation and right for geometric augmentation). Results shown are for multi-source domain test on Office-Home [6] with ResNet-18 backbone. Statistics on domains with different styles are colored differently.

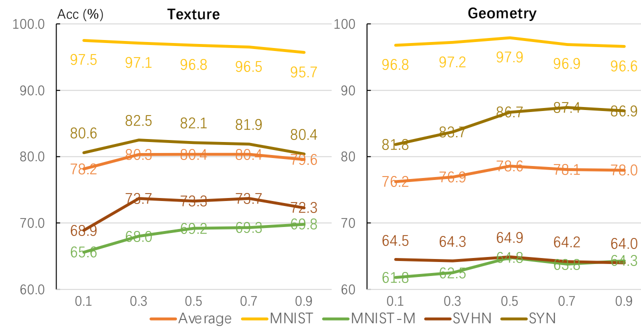

Figure 4. Influence of different probability thresholds for augmentation on classification (left for textural augmentation and right for geometric augmentation). Results shown are for multi-source domain test on Digits-DG [7] with the same backbone as [7]. Statistics on domains with different styles are colored differently.

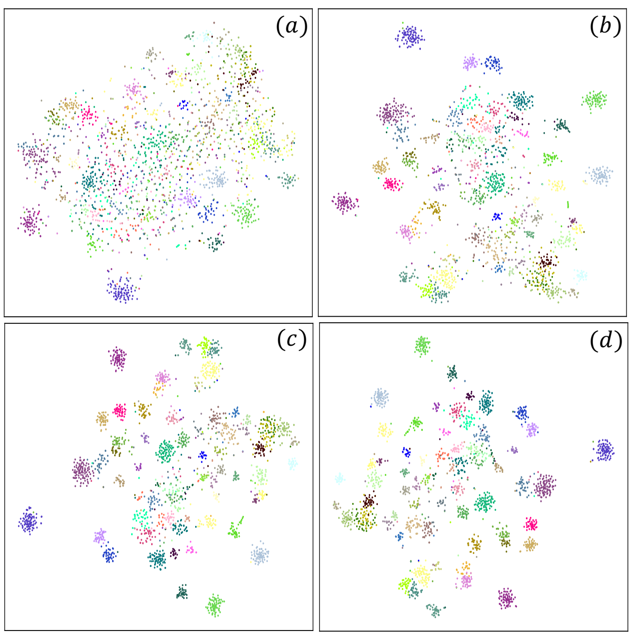

Figure 3. T-SNE visualization of the feature statistics using different augmentations on Office-Home [6]. Each color represents a different class. Legend labels are not included due to over many classes (65 in total). (a) Basic training without any augmentation. (b) Add only geometric augmentation. (c) Add only textural augmentation. (d) Using both geometric and textural augmentation.

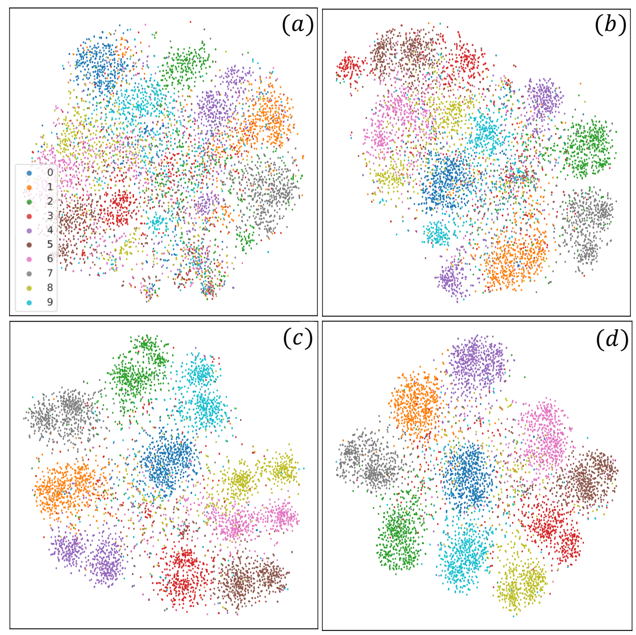

Figure 5. T-SNE visualization of the feature statistics using different augmentations on Digits-DG [7]. (a) Basic training without any augmentation. (b) Add only geometric augmentation. (c) Add only textural augmentation. (d) Using both geometric and textural augmentation.

3. Visualizations of Feature Statistics

To better illustrate the effect of each augmentation space on the model, we also visualize the feature statistics on Office-Home and Digits-DG using the T-SNE method, as shown in Figure 3 and Figure 5. Similar to the results on PACS (Figure 5 in the paper), after employing geometric and texture augmentations during the training, boundaries among different class objects become cleaner while the same class object cluster closer together.

4. Distribution based Augmentation versus Set/mini-batch based Augmentation

As described in the paper (Section 4), in terms of using style transfer methods for data augmentation, unlike most approaches [1,3,5,8] that sample an image from the training data or a mini-batch to transfer the texture of another one, we leverage constructed distributions instead of finite sets

Table 1. Comparisons between distribution based augmentation and set/mini-batch based augmentation. Results are shown for multi-source domain test accuracy (%) on PACS with ResNet-18 backbone. Each column indicates the target domain. **Ours$^{\mathbf{T1}}$** refers to the set-based version of our texture augmentation (using samplings from V (Equation 10)). **Ours$^{\mathbf{T}}$** refers to the distribution-based texture augmentation as we used in the paper (using samplings from $\mathcal{N}_V(\mu, \Sigma)$ (Equations 11–12). **Ours$^{\mathbf{G1}}$** refers to the set-based version of our geometric augmentation (using samplings from W_k (Equation 5)). **Ours$^{\mathbf{G}}$** refers to the distribution-based geometric augmentation as we used in the paper (using samplings from $\mathcal{N}_{W_k}(\mu_k, \Sigma_k)$ (Equations 6–7)). Compared with set/mini-batch based augmentation, leveraging distributions lead to a general improvement in performance due to the significantly improved representation diversity.

Augmentation Types		Methods	Art	Cartoon	Photo	Sketch	Average
Texture Augmentation	Set/Mini-batch based	SagNet [5]	83.58	77.66	95.47	76.30	83.25
		MixStyle [8]	84.10	78.80	96.10	75.90	83.70
		Ours$^{\mathrm{T1}}$	85.84	78.71	95.63	80.20	85.10
	Distribution based	Ours$^{\mathrm{T}}$	**86.34**	**80.12**	**96.38**	**81.42**	**86.07**
Geometric Augmentation	Set based	Ours$^{\mathrm{G1}}$	81.35	77.58	**96.11**	74.03	82.27
	Distribution based	Ours$^{\mathrm{G}}$	**82.74**	**78.22**	95.96	**75.26**	**83.05**

to augment the training image. The greatest benefit of such a strategy is to boost the geometric and textural style representations. From the experimental results in the paper (Tables 1–3), we can see that in terms of the individual performance of texture style augmentation, our distribution-based texture augmentation exceeds the set/mini-batch based texture augmentation approaches [5, 8]. And this is consistent with the conclusion in [4]: Representation diversity plays a key role in style transfer based model robustness improvement.

To further verify the effectiveness of the representation diversity. We compare the set based versions (Ours$^{\mathrm{G1}}$, Ours$^{\mathrm{T1}}$) of our augmentation with others, as shown in Table 1. **Ours$^{\mathbf{G1}}$**: Set based geometric augmentation, using samplings from W_k (Equation 5) instead of $\mathcal{N}_{W_k}(\mu_k, \Sigma_k)$ (Equations 6–7). **Ours$^{\mathbf{T1}}$**: Set based texture augmentation, using samplings from V (Equation 10) instead of $\mathcal{N}_V(\mu, \Sigma)$ (Equations 11–12). We can see that compared with set/mini-batch based augmentation, leveraging distributions lead to a general improvement in the results.

References

[1] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*, 2019. 2

[2] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, 2017. 1

[3] Yingwei Li, Qihang Yu, Mingxing Tan, Jieru Mei, Peng Tang, Wei Shen, Alan Yuille, and Cihang Xie. Shape-texture debiased neural network training. *ICLR*, 2021. 2

[4] Hubert Lin, Mitchell van Zuijlen, Sylvia C Pont, Maarten WA Wijntjes, and Kavita Bala. What can style transfer and paintings do for model robustness? In *CVPR*, 2021. 3

[5] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap by reducing style bias. In *CVPR*, 2021. 2, 3

[6] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, 2017. 1, 2

[7] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, 2020. 1, 2

[8] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021. 2, 3