Supplementary: What Can Style Transfer and Paintings Do For Model Robustness?

Hubert Lin, Cornell University
Mitchell van Zuijlen, TU Delft
Sylvia C. Pont, TU Delft
Maarten W.A. Wijntjes, TU Delft
Kavita Bala, Cornell University

S1. Outline

In this supplementary, we provide additional details to enable reproducibility, and additional visualizations and results to complement the main findings in the paper. In Section S2, we specify the creation of the Materials dataset. In Section S3, we detail the experimental setup for the classification robustness experiments. In Section S4, we describe the implementation and parameters used for style transfer, and we show visualizations of these methods in Section S5. In Section S6, we extend the discussion in Section 4 of the main paper by analyzing the effect of stylization strength versus robustness. In Section S7, we visualize the power spectra of stylized images and compare them to natural images. Finally, in Section S8, we frame model robustness as domain generalization and discuss how domain-invariance can affect model robustness.

S2. Materials Dataset Details

In Section 3.2 of the main paper, we briefly described the two primary datasets on which we focused our experiments. PACS [7] is a standard benchmark dataset while Materials is a novel dataset of photographs and paintings that was created by sampling image patches from existing datasets with material annotations. In this section, we give additional information on the creation of Materials. This dataset is released for reproducibility at https://github.com/hubertsgithub/style_painting_robustness

Natural photographs. We acquired image patches from OpenSurfaces[1], COCO stuff [3], and MINC-2500 [2]. To create image patches for image classification from segmentation annotations, we constructed bounding boxes around segments, and cropped out these bounding boxes to form image patches. We constructed square bounding boxes with side length equal to 150% of the minimum side length of tight bounding box around the segment. Non-tight bounding boxes are used since it is important to include some context for the patch. We also sampled from MINC-2500 which already contains annotated image patches that do not require additional processing. Image crops that extend beyond the boundary of the full image are padded to square with ImageNet mean padding, and all final images patches are resized to 224×224. We sampled from OpenSurfaces and MINC first, before sampling from COCO if necessary. We created subsets of data of up to 60K photos, and each subset was created to be as-class-balanced-as-possible. For illustration, we provide per-class counts for two such subsets of data in Table 5.

Natural-10K	Count	Natural-60K	Count
Ceramic	1000	Ceramic	3132**
Fabric	1000	Fabric	8006
Foliage	1000	Foliage	8006
Glass	1000	Glass	7216**
Liquid	1000	Liquid	7174**
Metal	1000	Metal	7204**
Paper	1000	Paper	3258**
Skin	1000	Skin	2276**
Stone	1000	Stone	5716**
Wood	1000	Wood	8006

Table 5: Training datasets are sampled to be as class-balanced as possible. ** indicates that all training samples of that category are included in the training set, and no further samples exist. Natural-10K is a subset of Natural-60K. The test set contains 200 samples of each category.

Paintings. We sample paintings across the same material categories as above from [12]. We only sample patches that are at least 128×128 pixels in area to avoid very low-resolution annotations. The image patches are padded and resized in the same manner as above, and data is also sampled to be as-class-balanced-as-possible.

S3. Classification Parameters

For all classification experiments, we use the following setup. Code is released for reproducibility at https://github.com/hubertsgithub/style_painting_robustness

- Network architecture: ResNet18, ImageNet pretrained.
- Training hyperparameters: 30 epochs with initial learning rate (LR) 1e-3, LR reduced to 1e-4 at epoch 24. The LR of the classification layer is increased by 10×.
- Optimizer: SGD with 0.9 momentum.
- Training data augmentation: horizontal flipping, random scaling, color jitter, and ImageNet normalization.
- For experiments that train a model on both photos and stylized photos, all photos are stylized exactly once offline and included in the training set as an independent image from the original photo.
- Evaluation accuracies are averaged over 3 independent runs for each experiment.

Our results do not appear sensitive to the choice of training hyperparameters. Therefore, we train all networks with this configuration and evaluate the final model after training. Our experiments suggest that this training schedule is sufficient for convergence without overfitting across all datasets we experimented with. Increasing training epochs to 100 or more does not improve results. Increasing number of training epochs is required if starting from random initialization, but ImageNet pretraining is standard practice so we do not extensively experiment with random initialization.

S4. Style Transfer Parameters

For all applications of style transfer used in this work, we use pretrained models from publicly available implementations. The sources are provided here:

- AdaIN [6]: `https://github.com/bethgelab/stylize-datasets`
- ETNet [10]: `https://github.com/zhijieW94/ETNet`
- TPFR [11]: `https://github.com/nnaisense/conditional-style-transfer`
- SACL [9]: `https://github.com/CompVis/adaptive-style-transfer`

Our initial experiments showed that applying style transfer at 224×224 resolution yielded visually poor results (except for AdaIN). Therefore, we apply style transfer at a higher resolution and downsample the final result to 224×224. For AdaIN, ETNet, and SACL, we apply style transfer at 768×768 resolution. For TPFR, we apply style transfer at 512×512 instead of 768×768 due to GPU memory constraints. All other hyperparameters are set to the default settings found in the implementations for each respective method.

S5. Visualizations of Stylized Photos

We show examples of images stylized by various style transfer methods on PACS (Fig. 10) and Materials (Fig. 11). The visualizations also include examples of intradomain stylization in which images are stylized by photos instead of by paintings. Notice that intradomain stylization yields stylizations that are, in general, visually similar to stylizations with painting style images. Overall, stylizations across all methods are holistically similar to natural paintings.

S6. Style Distance vs Robustness

In Section 4.2, we found that arbitrary stylization with style images that share the same semantic content as the content image ("intraclass stylization") results in lower gains in robustness. Since images with similar semantic content may be more visually similar, this suggests that intraclass stylization will lead to less stylized images, i.e. weaker augmentation. To verify this, we measured style differences via the Gram matrix distance between stylized images and their original counterparts. Table 6 summarizes differences on PACS. While intraclass stylization does result in smaller differences in style for each method, the Gram matrix distance across methods is not necessarily correlated with gains in robustness. For example, ETNet produces the largest style differences overall, but AdaIN improves robustness more (Fig. 3,4). As such, the strength of stylization alone is not indicative of the downstream robustness learned by models trained on these images.

Method	Painting	Intradomain	Intradomain (Intraclass)
AdaIN	1.58±0.93	1.28±0.79	1.16±0.85
ETNet	2.33±1.09	2.13±1.04	1.81±1.03
TPFR	1.52±0.90	1.38±0.87	1.27±0.91

Table 6: **Style (Gram Matrix) Distance.** Gram matrices computed from ImageNet pretrained ResNet18 features on PACS. Mean distance between `(image, stylized image)` pairs is reported. ↑ distance implies ↑ style difference. ± denotes standard deviation across 1.5K pairs.

S7. Power Spectra of Different Image Types

In Section 6 of the main paper, we found that SACL improves robustness against noise with imperceptible high frequency signals in the stylized images. The results are shown in Table 7. Here we show the power spectra of stylized images and compare them to the spectra for natural photos and natural paintings. The radial power spectrum for an image is computed as:

$$\text{power}(r) = ||X_r||^2$$
$$\text{where } X_r = \underset{\sqrt{i^2+j^2}\in\mathcal{R}(r)}{\text{mean}} ||X_{ij}||$$

X_{ij} are the frequency components given by the 2D discrete Fourier transform. Since (i, j) are discrete, the radial frequency component X_r is computed as an average over $||X_{ij}||$ for (i, j) that fall in a bin $\mathcal{R}(r)$. In Fig. 9, we visualize the mean radial power spectra for natural photos, natural paintings, and SACL-stylized photos. We observe that stylized photos contain higher magnitude high-frequency components relative to natural photos and natural paintings. As noted in Section 6 of the main paper, reducing the magnitude of sufficiently high-frequency components does not affect the perceptual quality of images.

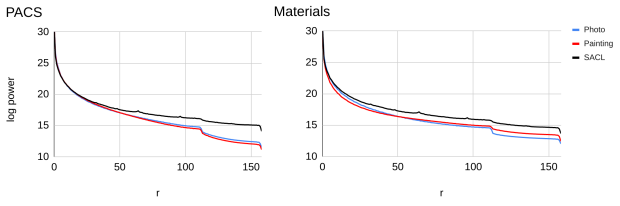

Figure 9: **Power Spectrum of Images**. Left: PACS, Right: Materials. The plots depict the mean power spectrum for different sets of images. Photos stylized by SACL have larger magnitude high frequency components than natural photos or natural paintings.

Method	Noise	Blur	Weather	Digital	OOD
Materials (30K Samples/Domain)					
Photo-Only	43.70±0.65	58.76±0.14	55.25±0.33	61.20±0.69	41.33±0.62
Photo + SACL	61.87±0.16	64.36±0.20	57.49±0.24	66.55±0.17	34.54±0.91
Photo + Painting	49.82±0.56	61.03±0.13	56.69±0.10	64.15±0.14	43.92±0.47
Photo+SACL (LF)	45.82±1.36	64.24±0.39	57.06±0.13	66.37±0.29	36.92±1.15
Photo+Painting (LF)	44.95±0.66	60.87±0.29	56.82±0.23	63.69±0.46	41.21±0.56
PACS (1.5K Samples/Domain)					
Photo-Only	62.64±1.48	72.75±0.04	83.24±0.22	86.33±0.14	82.57±0.00
Photo + SACL	85.98±0.56	84.61±0.15	89.73±0.33	88.74±0.48	77.43±0.84
Photo + Painting	68.83±0.83	75.80±0.95	86.88±0.66	87.07±0.14	85.43±0.70
Photo+SACL (LF)	77.55±2.60	85.4±0.11	88.93±0.22	88.53±0.15	77.43±0.47
Photo+Painting (LF)	71.16±1.31	75.97±0.71	86.82±0.37	87.35±0.36	83.71±0.40

Table 7: **Robustness without High Frequency Signals.** "LF" denotes filtered low frequency images. Photos are always unfiltered. Filtering invisible high frequency components mainly impacts noise robustness. (blue) Filtering stylized photos significantly reduces noise robustness while (red) filtering paintings has a relatively smaller effect. ± indicates standard deviations over 3 runs.

S8. Domain-Invariant Feature Learning

Our results from Section 5 of the main paper provide evidence that models can learn more robust feature representations from the addition of paintings to a dataset of photographs. We can take this further by explicitly enforcing similar (or domain-invariant) feature representations across photos and paintings. Domain-invariance is a common approach to the problem of domain generalization, where models are trained on multiple domains with the goal of generalizing to unseen domains, e.g. [5, 8]. In our setting, we can consider images with common corruptions to be the set of unseen domains. Perfect domain-invariant feature extraction can be harmful if it prevents useful features in photos from being extracted due to an underrepresentation of such features in paintings. Since the target task is recognition of photos, losing robust photo-specific signals can be detrimental. Therefore, we explore the following:

- **Hypothesis H1S:** Explicitly learning domain-invariance from paintings and photos may negatively impact model robustness.

We use an adversarial domain discriminator to learn domain invariant features [4, 8]. In Table 8, we find that explicitly learning domain invariant features from paintings results in lower robustness than unrestricted feature learning with paintings. However, learning domain-invariant features does still improve robustness over the photo-only baseline. Existing work in domain generalization has shown that domain-invariance is an effective method for learning to recognize images from unseen domains, e.g. [8]. Our finding here suggests that in the special case of domain generalization to corrupted versions of natural photographs, it is advantageous to retain photo-specific features for recognition. This is consistent with our hypothesis and discussion above – an underrepresentation of any particular photo-specific features in paintings can result in such features being ignored entirely when domain-invariance is enforced, even if such features are useful for robust recognition.

Answer to H1S: *Explicitly learning domain-invariant features from paintings negatively impacts model robustness with respect to unrestricted feature learning with paintings. However, domain-invariant features do still improve robustness relative to photos only.*

Method	MEAN	Noise	Blur	Weather	Digital
Materials (30K Samples/Domain)					
Photo-Only	54.73	43.71	58.76	55.25	61.20
Photo + Painting	**57.92**	**49.82**	**61.03**	**56.68**	**64.15**
Photo + Painting (DA)	55.99	46.97	59.60	54.51	62.90
PACS (1.5K Samples/Domain)					
Photo-Only	76.16	62.64	72.75	83.24	86.33
Photo + Painting	**78.99**	68.04	**74.72**	**86.26**	**86.92**
Photo + Painting (DA)	77.44	**68.86**	72.59	84.09	84.23

Table 8: **Effect of Domain-Invariant Features.** "DA" refers to feature learning with an adversarial domain discriminator loss [4]. Learning domain-invariant features (red) reduces robustness relative to unrestricted feature learning from paintings (blue), but still improves robustness over photo-only.

S9. Additional Architectures

We expect our findings to hold across architectures and datasets. As a sanity check, we have extended Table 2 with two additional architectures. The results (Table 9) follow similar trends to those found in Table 2. For example, SACL outperforms both AdaIN and Paintings on Noise.

Resnet-18

Method	Noise	Blur	Weather	Digital
Materials (30K Samples/Domain)				
Photo-Only	43.70±0.65	58.76±0.14	55.25±0.33	61.20±0.69
Photo + AdaIN	47.33±0.22	65.09±0.21	61.78±0.18	61.41±0.16
Photo + SACL	61.87±0.16	64.36±0.20	57.49±0.24	66.55±0.17
Photo + Painting	49.82±0.56	61.03±0.13	56.69±0.10	64.15±0.14
PACS (1.5K Samples/Domain)				
Photo-Only	62.64±1.48	72.75±0.04	83.24±0.22	86.33±0.14
Photo + AdaIN	70.17±1.70	81.18±0.20	88.37±0.23	89.32±0.19
Photo + SACL	85.98±0.56	84.61±0.15	89.73±0.33	88.74±0.48
Photo + Painting	68.83±0.83	75.80±0.95	86.88±0.66	87.07±0.14

WideResnet-50-2

Method	Noise	Blur	Weather	Digital
Materials (30K Samples/Domain)				
Photo + AdaIN	57.80±1.79	73.77±0.11	67.75±0.51	66.96±0.06
Photo + SACL	69.39±0.72	70.00±0.34	64.00±0.54	73.05±0.30
Photo + Painting	60.72±0.83	68.09±0.49	61.15±0.23	70.98±0.24
PACS (1.5K Samples/Domain)				
Photo + AdaIN	82.05±1.33	86.89±0.64	93.98±0.15	94.39±0.30
Photo + SACL	93.79±1.35	89.64±0.36	95.19±0.17	93.63±0.11
Photo + Painting	83.92±1.81	85.38±0.27	94.19±0.08	92.63±0.24

Densenet-121

Method	Noise	Blur	Weather	Digital
Materials (30K Samples/Domain)				
Photo + AdaIN	54.32±0.23	71.08±0.24	67.31±0.37	66.47±0.13
Photo + SACL	67.22±0.16	68.89±0.16	63.08±0.33	71.87±0.62
Photo + Painting	54.83±1.20	68.21±0.38	61.29±0.39	70.66±0.13
PACS (1.5K Samples/Domain)				
Photo + AdaIN	76.96±4.12	85.79±0.50	94.96±0.13	92.34±0.19
Photo + SACL	91.33±0.28	88.92±0.37	94.18±0.49	94.12±0.54
Photo + Painting	76.65±2.22	83.22±0.19	94.00±0.62	91.72±0.14

Table 9: **Per-Corruption Accuracy (Additional Architectures).** Trends across different architectures are generally consistent. For example, SACL (blue) greatly outperforms AdaIN and paintings (red) for noise robustness. ± indicates standard deviation over 3 runs.

References

[1] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. OpenSurfaces: A richly annotated catalog of surface appearance. *ACM Transactions on Graphics (TOG)*, 32(4), 2013. 1

[2] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the Materials in Context database. *Computer Vision and Pattern Recognition (CVPR)*, 2015. 1

[3] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018. 1

[4] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. 3

[5] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020. 3

[6] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 2

[7] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 1

[8] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *AAAI*, pages 11749–11756, 2020. 3

[9] Artsiom Sanakoyeu, Dmytro Kotovenko, Sabine Lang, and Bjorn Ommer. A style-aware content loss for real-time hd style transfer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 698–714, 2018. 2

[10] Chunjin Song, Zhijie Wu, Yang Zhou, Minglun Gong, and Hui Huang. Etnet: Error transition network for arbitrary style transfer. *arXiv preprint arXiv:1910.12056*, 2019. 2

[11] Jan Svoboda, Asha Anoosheh, Christian Osendorfer, and Jonathan Masci. Two-stage peer-regularized feature recom-

bination for arbitrary image style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13816–13825, 2020. 2

[12] Mitchell Van Zuijlen, Hubert Lin, Kavita Bala, Sylvia C Pont, and Maarten WA Wijntjes. A database of painterly material depictions. *Journal of Vision*, 20(11):1127–1127, 2020. 1

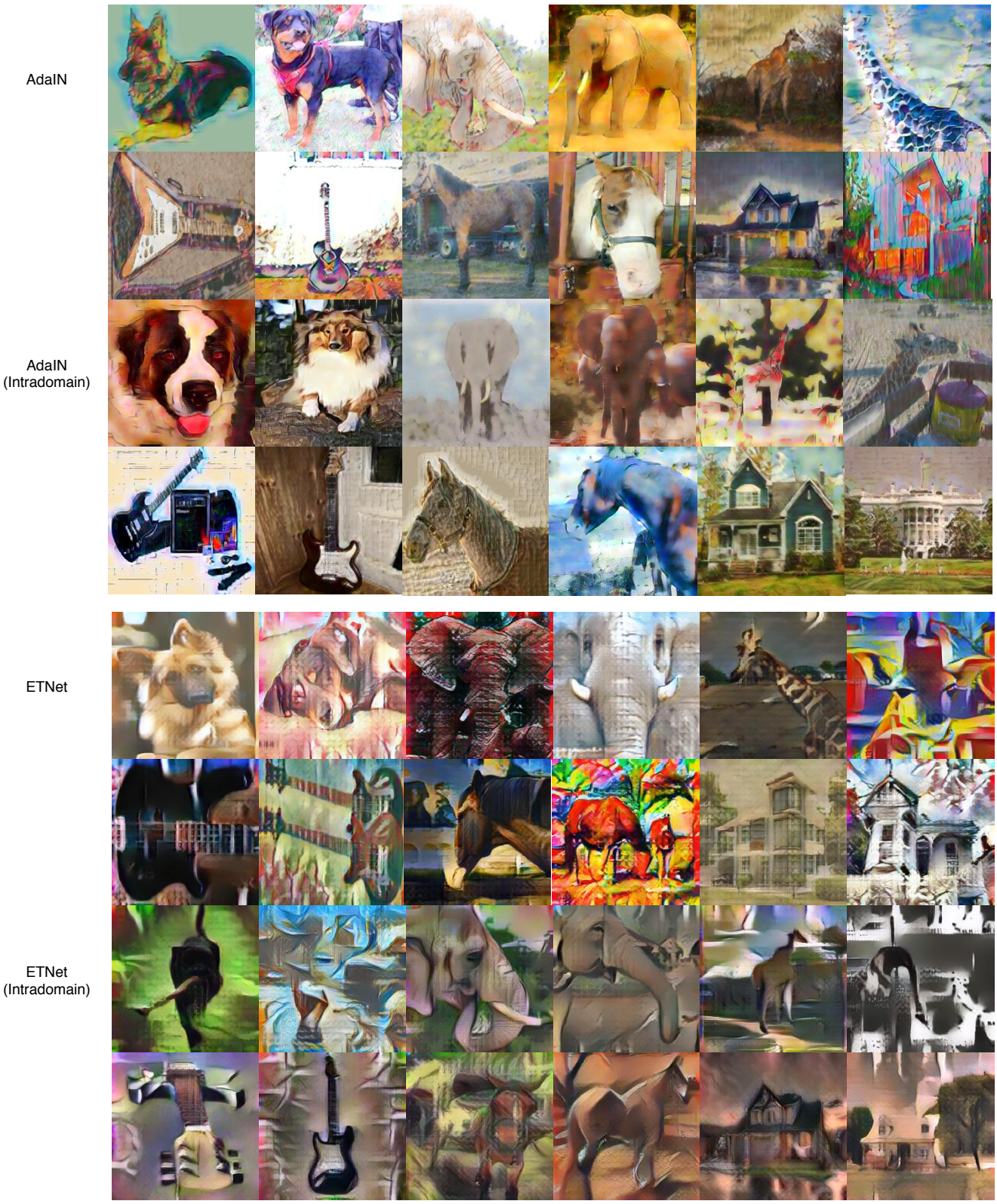

Figure 10: **Stylized Photos (PACS) (1/2)**. Intradomain refers to stylization with photos as style images instead of paintings as style images. SACL is a learned style transfer method that is applied with different models pretrained to transfer the style of different artists. (*Continued on next page*)

Figure 10: **Stylized Photos (PACS) (2/2)**. Intradomain refers to stylization with photos as style images instead of paintings as style images. SACL is a learned style transfer method that is applied with different models pretrained to transfer the style of different artists.

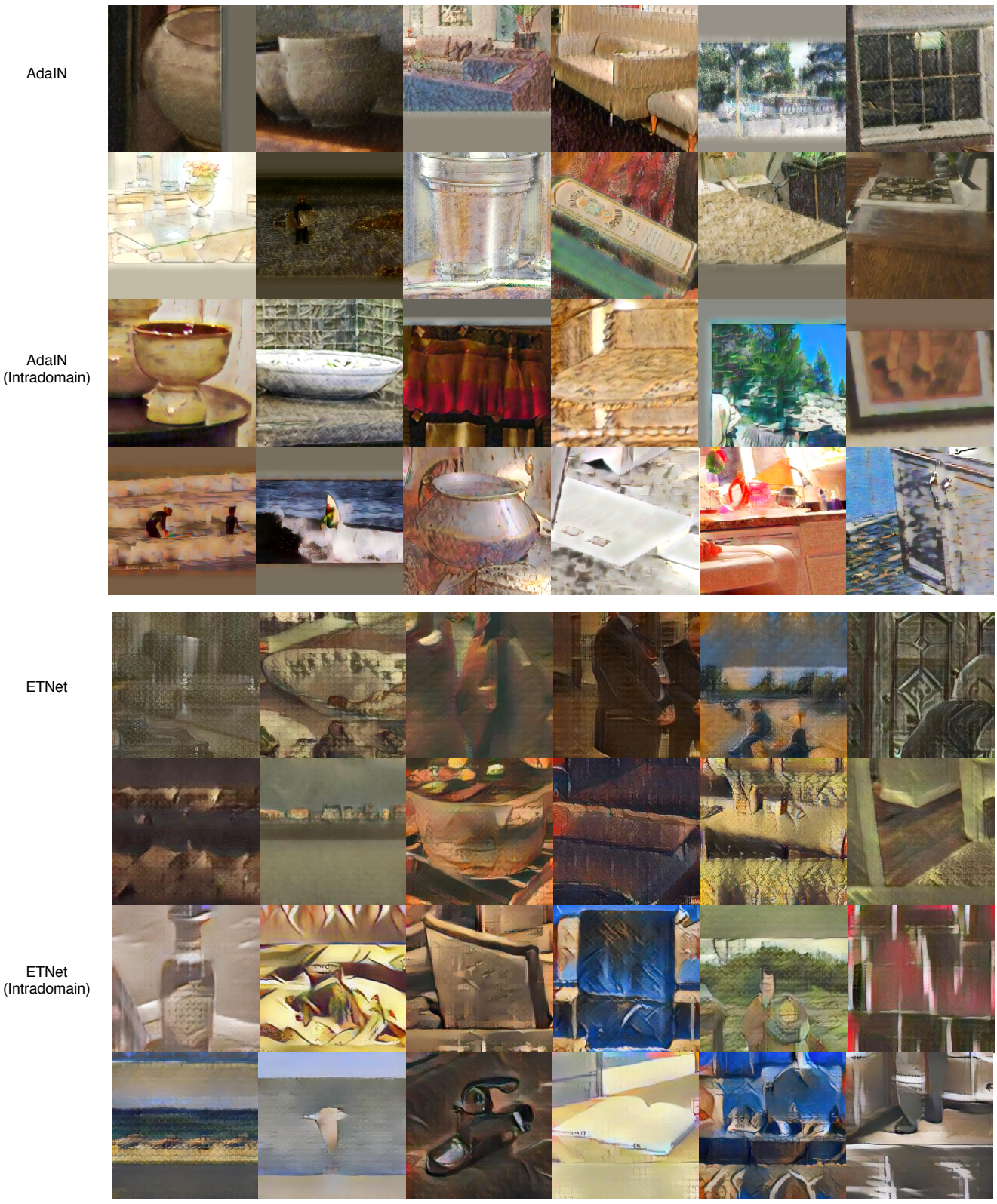

Figure 11: **Stylized Photos (Materials) (1/2)**. Intradomain refers to stylization with photos as style images instead of paintings as style images. SACL is a learned style transfer method that is applied with different models pretrained to transfer the style of different artists. (*Continued on next page*)

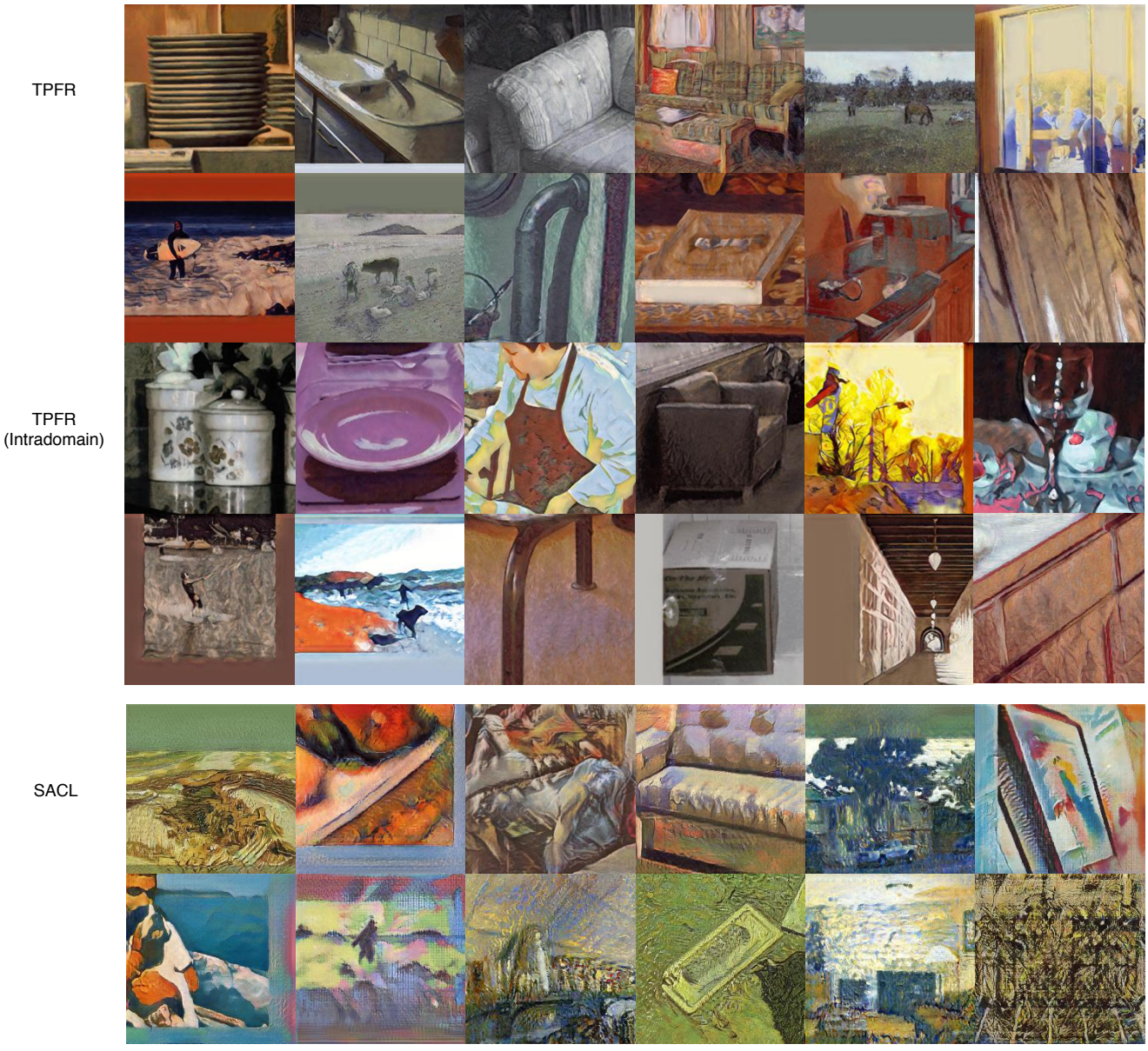

Figure 11: **Stylized Photos (Materials) (2/2)**. Intradomain refers to stylization with photos as style images instead of paintings as style images. SACL is a learned style transfer method that is applied with different models pretrained to transfer the style of different artists.