Risk-Averse Predictions on Unseen Domains via Neural Style Smoothing

Anonymous Authors[1]

Abstract

Achieving high accuracy on data from domains unseen during training is a fundamental challenge in machine learning. While state-of-the-art neural networks have achieved impressive performance on various tasks, their predictions are biased towards domain-dependent information (ex. image styles) rather than domain-invariant information (ex. image content). This makes them unreliable for deployment in risk-sensitive settings such as autonomous driving. In this work, we propose a novel inference procedure, Test-Time Neural Style Smoothing (TT-NSS), that produces risk-averse predictions using a "style smoothed" version of a classifier. Specifically, the style smoothed classifier classifies a test image as the most probable class predicted by the original classifier on random re-stylizations of the test image. TT-NSS uses a neural style transfer module to stylize the test image on the fly, requires black-box access to the classifier, and crucially, abstains when predictions of the original classifier on the stylized images lack consensus. We further propose a neural style smoothing-based training procedure that improves the prediction consistency and the performance of the style-smoothed classifier on non-abstained samples. Our experiments on the PACS dataset and its variations, both in single and multiple source domain settings highlight the effectiveness of our methods at producing risk-averse predictions on unseen domains.

1. Introduction

A fundamental challenge in machine learning is to produce classifiers that can withstand a domain shift at test time without having any knowledge of the shift during training. Previous works (Bulusu et al., 2020; Hendrycks & Dietterich, 2019; Alcorn et al., 2019; Geirhos et al., 2018; Beery et al., 2018) have demonstrated that variations in styles/textures, weather changes, etc., unseen during training can drastically reduce the classifier's performance. Recent works (Geirhos et al., 2018; Nam et al., 2021; Hermann et al., 2020; Baker et al., 2018) brought to light the fact that predictions from state-of-the-art (SOTA) neural networks are biased towards the information unrelated to the content of the images but are dependent on the image styles, that can vary across domains. Due to the vast practical implications of this problem many works have studied this problem both analytically (Ben-David et al., 2007; 2010; Mansour et al., 2009; Shen et al., 2018; Zhao et al., 2019; Johansson et al., 2019; Blanchet & Murthy, 2019; Mehra et al., 2021b) and empirically (Albuquerque et al., 2019; Zhang et al., 2021a; Ganin et al., 2016; Zhao et al., 2018; Qiao et al., 2020; Gulrajani & Lopez-Paz, 2020; Mehra et al., 2022). However, in scenarios such as in autonomous driving, medical diagnoses, or using drones for rescue operations, where a risky misclassification could be catastrophic, augmenting the classifier with an abstaining mechanism or involving a human in the loop becomes crucial (Settles, 2009; Cortes et al., 2016). Thus, in this work we focus on problem of image classification under distribution shifts which comprise of differences in styles. To safeguard the classifier against risky misclassification (and enable risk-averse predictions) we augment the classifier with a capability to defer making a prediction on samples where it lacks confidence. However, since the softmax score of the classifier is known to be uncalibrated (Hein et al., 2019; Hendrycks et al., 2019; Hendrycks & Gimpel, 2016) on data from unseen domains, we propose a novel method that uses neural style information to estimate classifier's confidence in it's prediction under style changes.

Our inference procedure, Test-Time Neural Style Smoothing (TT-NSS), outlined in Fig. 1, first transforms a classifier (base classifier) into a style-smoothed classifier and then uses it to either predict the label of an incoming test sample or abstain on it. Specifically, the prediction of the style smoothed classifier, ψ, constructed from a base classifier f, on a test input x is defined as the class that the base classifier f would predict most often on stylized versions of the input. TT-NSS uses a style transfer network based on AdaIN (Huang & Belongie, 2017) to produce stylized versions of the test input in real time. While AdaIN can

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Under review by *AdvML-Frontiers'23: The 2nd Workshop on New Frontiers in Adversarial Machine Learning@ICML'23*, Honolulu, Hawaii, USA. Do not distribute.

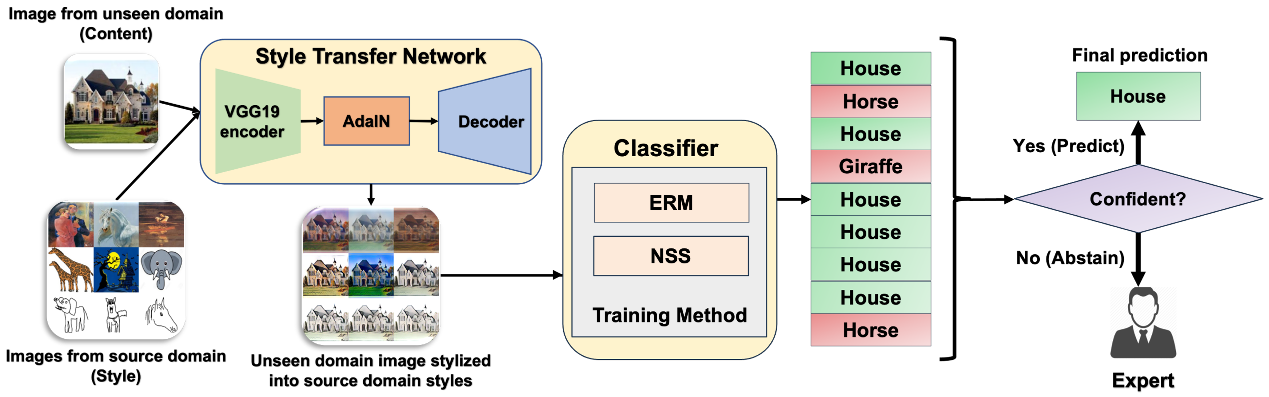


Figure 1. Overview of our Test-Time Neural Style Smoothing (TT-NSS) inference procedure for obtaining risk-averse predictions. TT-NSS works by stylizing a test sample into source domain styles and classifies the sample as the most probable class assigned by the classifier to the stylized samples if that class is much more likely than the other classes. Otherwise, it abstains from making a prediction and refers the sample to an expert thereby avoiding a risky misclassification.

transform the style of x to any arbitrary style, we specifically transform it into the style of the data from the domain used for training. This choice is based on the fact f can be easily made agnostic to the styles of a domain used for training. Moreover, changing the styles of x to arbitrary styles, unknown to f, can worsen the classifier's performance due to a widened distribution shift. TT-NSS can be used to evaluate any classifier with only black-box access to it, i.e., it does not require the knowledge of weights, architecture, or training procedure used to train the classifier and only needs its predictions on stylized samples. However, computing the prediction of a style-smoothed classifier requires computing the probability with which the base classifier classifies the stylized images of x. Following works in Randomized Smoothing (Cohen et al., 2019), we propose a Monte Carlo algorithm to estimate this probability. When this estimated probability exceeds a set threshold it implies that the predictions of the classifier f on stylized images of x achieve a desired level of consensus and the prediction is reliable. In other cases, TT-NSS abstains due to lack of consensus.

Furthermore, we propose a novel training method that improves the consistency of the predictions of the classifier on stylized images. The improved consistency leads to lower abstaining rates and improved performance on non-abstained samples thereby improving the reliability of the predictions from the classifier. Our training method creates a style smoothed version of the soft base classifier and uses stylized versions of the source domain data (generated by stylizing the source domain images into random styles of other source domain images) to train the base classifier. Similar to previous works (Jeong & Shin, 2020; Sohn et al., 2020; Sun et al., 2021), we incorporate consistency regularization during training to further boost the performance of the classifier on non-abstained samples at various abstaining rates. Similar to TT-NSS, our NSS-based training losses can be combined with any training method and can improve the reliability of the classifier's predictions without significantly degrading their accuracy. We present results of using our inference and training procedures on PACS (Li et al., 2017) dataset and its variations generated by applying style changes and common corruptions, in both single and multiple source domain settings. Our results show that our proposed methods improve the reliability of the classifier's predictions on unseen domains. Our main contributions are:

- We propose a simple and effective inference procedure based on neural style smoothing for obtaining risk-averse predictions. Our method returns the prediction of the style-smoothed classifier in real time with only black-box access to the underlying classifier.
- We propose a novel training procedure to improve the performance of style-smoothed classifier by incorporating neural style smoothing during training and enforcing prediction consistency under random stylizations of the source domain data.
- We evaluate the effectiveness of methods on benchmark datasets and their novel variations created by stylizing and applying common corruptions to them.

2. Neural style smoothing

2.1. Background

Problem setup: Given N^i data samples from N_S source domains as $\mathcal{D}^i_{\text{source}} = \{(x^i_j, y^i_j)\}_{j=1}^{N^i}$ each following a distribution $P^i_S(X,Y)$, the goal is to learn a classifier $f(X)$ whose performance does not degrade on a sample from an unseen test domain with distribution $P_T(X,Y) \neq P^i_S(X,Y)$, for all $i \in \{1, \cdots, N_S\}$. Based on the number of source domains available during training we consider a single and a multiple source domain setting. The lack of information about the target domain makes the problem challenging and previous works have proposed training methods focusing on capturing domain invariant information from source domain data to generalize well to unseen domains at test time. However, learning a classifier by minimizing its empirical risk on all available source domains has been shown to achieve competitive performance on various benchmark datasets (Gulrajani & Lopez-Paz, 2020), especially in the multiple source domain setting.

Neural style transfer with AdaIN (Huang & Belongie, 2017): Given a content image, x_c and a style image x_s, AdaIN generates an image having the content of x_c and style of x_s. AdaIN works by first extracting the intermediate features (output of `block4_conv1`) of the style and content image by passing them through a VGG-19 (Simonyan & Zisserman, 2014) encoder, g, pretrained on Imagenet. Using these features AdaIN aligns the mean (μ) and variance (σ) of the two feature maps using

$$\begin{aligned} t &= \text{AdaIN}(g(x_c), g(x_s)) \\ &= \sigma(g(x_s)) \left(\frac{g(x_c) - \mu(g(x_c))}{\sigma(g(x_c))} \right) + \mu(g(x_s)). \end{aligned} \tag{1}$$

A decoder, h, is then used to map the AdaIN-generated feature back to the input space to produce a stylized image $x_{\text{stylized}} = h(t)$. We follow the design of the decoder as proposed in (Huang & Belongie, 2017) and train the decoder to minimize the content loss between the features of the stylized image, $g(x_{\text{stylized}})$ and the AdaIN transformed features of the content image, i.e.

$$\mathcal{L}_{\text{content}} = \|g(x_{\text{stylized}}) - t\|_2^2, \tag{2}$$

along with a style loss that measures the distance between the feature statistics of the style and the stylized image using L layers of the pretrained VGG-19 network, ϕ. In particular, the style loss is computed as

$$\begin{aligned} \mathcal{L}_{\text{style}} = &\sum_{i=1}^{L} \|\mu(\phi_i(x_s)) - \mu(\phi_i(x_{stylized})\|_2^2 \\ &+ \sum_{i=1}^{L} \|\sigma(\phi_i(x_s)) - \sigma(\phi_i(x_{\text{stylized}})\|_2^2. \end{aligned} \tag{3}$$

We measure the style loss, using `block1_conv1`, `block2_conv1`, `block3_conv1`, and `block5_conv1` layers of the VGG-19 network. We pre-train the decoder using images from MS-COCO (Lin et al., 2014) as content images and images from Wikiart (Nichol., 2016) as style images.

2.2. Neural style smoothing-based inference

Consider a classification problem from $\mathbb{R}^d$ to the label space $\mathcal{Y}$. Neural style smoothing produces an output, for a test image x, that a base classifier, f is most likely to return when x is stylized into the style of the source domain data, i.e., the data used for training f. Formally, given a base classifier f, we construct a style smoothed classifier $\psi : \mathbb{R}^d \to \mathcal{Y}$, whose prediction on a test image x is the most probable output of f on x converted into the style of the source domain data, i.e.,

$$\psi(x) := \arg\max_{y \in \mathcal{Y}} \mathbb{P}(f(h(t)) = y), \tag{4}$$

where $t = \text{AdaIN}(g(x), g(x_s))$, $x_s \sim P_S$, and P_S is the distribution of the source domain. When data from multiple source domains are available we combine the data from all the domains and use the combined data as source domain data. If the base classifier, f, correctly classifies the test image x, stylized into the styles of the source domain, then the style-smoothed classifier also correctly classifies that sample. However, computing the actual prediction of the style-smoothed classifier requires computing the exact probabilities assigned by the base classifier to stylized test samples. Thus, following (Cohen et al., 2019), we propose a Monte Carlo algorithm to estimate these probabilities and the prediction of the style-smoothed classifier. The first step in estimating the prediction of the style smoothed classifier on a test image x is to generate a stylized version of the image using the styles from the source domain. To achieve the style conversion in real-time, we use the AdaIN framework described previously with the content image as the test image x and n randomly chosen images from the training dataset used for training the classifier. The style transfer network then produces n images stylized into the style of the source domain data, as illustrated in Fig. 1. The stylized images are then passed through the base classifier and the class that is predicted the most often (majority class) is returned as the prediction of the test image. This procedure of Test-Time Neural Style Smoothing ($\mathrm{TT-NSS}$) is detailed in Alg. 1.

To ascertain that the prediction returned by TT-NSS is reliable, we estimate the confidence of the style smoothed classifier on its prediction. In particular, we compute the proportion of the re-stylized test images that are classified as a particular class by the base classifier and obtain a vector containing counts of how often each class is predicted. Based on the entries in this vector, we compute the class which has the highest occurrence and if the proportion of the highest class exceeds a threshold α, TT-NSS classifies the test image as this class with the highest counts. However, if the proportion remains less than the threshold, then TT-NSS abstains due to a lack of consensus among the predictions. The abstained samples can then be sent for further processing to experts and save the system from returning a

Algorithm 1 Test-Time Neural Style Smoothing (TT-NSS)

```
Input: Test image $x$, base classifier $f$, VGG-19 encoder $g$, AdaIN decoder $h$, number of source style images $n$, $\mathcal{D}_{\text{styles}} = \{x_{\text{s}}^i\}_{i=1}^n$, threshold $\alpha$.
Output: Prediction for $x$ or ABSTAIN.
  Initialize class-wise counts class_counts to zeros

  # Generate $n$ stylized images from $x$ using $\mathcal{D}_{\text{styles}}$
  for $i = 1, \cdots, n$ do
    $t = \text{AdaIN}(g(x), g(x_{\text{s}}^i))$
    $x_{\text{stylized}} = h(t)$
    prediction $= f(x_{\text{stylized}})$
    class_counts[prediction]$+ = 1$
  end for

  # Get the top predicted class on stylized images
  $c_{\max}$ = index of class_counts with highest count
  $n_{\max} = \text{class\_counts}[c_{\max}]$

  # Predict or ABSTAIN
  if $\frac{n_{\max}}{n} < \alpha$ then
    return ABSTAIN
  else
    return $c_{\max}$
  end if
```

potentially incorrect prediction. A high value of α makes TT-NSS, produce better predictions i.e., accuracy on non-abstained samples increases but it also increases the number of abstained samples. On the other hand, a low value of α leads to decreased abstaining with an increased chance that the prediction is not confident and may lead to an incorrect decision. In our empirical analysis in Sec. 3, we use various values of α ranging from 0 to 1 and show how the accuracy on non-abstained samples and the proportion of abstained samples change as the value of α is increased.

2.3. Neural style smoothing-based training

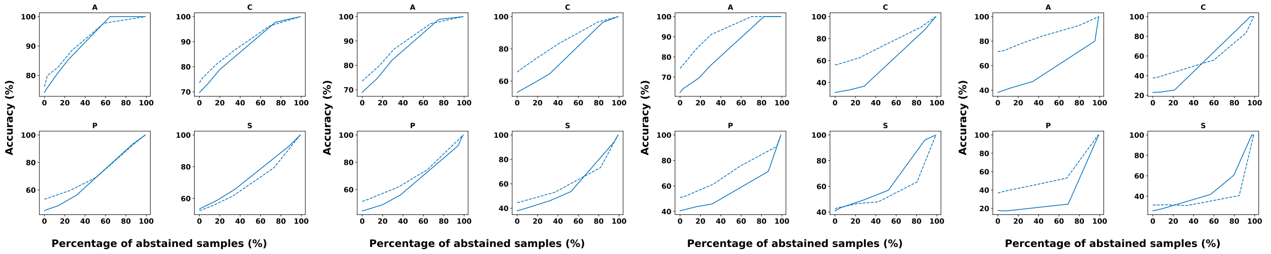
The performance of our inference procedure, TT-NSS, relies on the assumption that the base classifier, f, can classify the test image stylized into the source domain styles correctly and consistently. This requires that the base classifier be accurate on the images generated by the decoder used in the AdaIN-based neural style transfer network. However, our empirical evaluation of using TT-NSS on classifiers trained with ERM on benchmark datasets shows a relatively low accuracy on non-abstained samples at smaller abstaining rates. This suggests that the base classifier cannot accurately classify the stylized images generated through the AdaIN decoder. Thus, we propose a new training procedure based on neural style smoothing (NSS) that enables consistent and accurate predictions from the classifiers when evaluated using TT-NSS. The proposed loss functions can be combined with any training algorithm and can be used to improve the reliability of the predictions from classifiers when evaluated with TT-NSS. To achieve this, we propose to train the classifier f, by minimizing the sum of two loss functions. The first loss penalizes misclassification of the stylized images w.r.t. the label of the content image i.e., given a sample $(x, y) \sim \mathcal{D}_{\text{source}}$, the stylized misclassification loss is given by

$$\mathcal{L}_{stylized\_aug} = \mathbb{E}_{x_s \sim P_S}[\ell(f(h(t)), y)], \tag{5}$$

where $t = \text{AdaIN}(g(x), g(x_s))$ and ℓ is the cross entropy loss. Specifically, we first stylize a sample x from the source domain using multiple randomly sampled style images from the source domain and then penalize the misclassification loss of the classifier f on these stylized images. For a single source domain problem, even though all images from a domain may be considered as being in the same broad set of styles such as Art or Photos, individually the images have different non-semantic information such as textures, colors, patterns, etc., and thus stylizing an image into the styles of other source domain images is still effective and meaningful. The second loss which helps improve the trustworthiness of the predictions enforces consistency among the predictions of the stylized versions of the content image, generated using AdaIN. Previous works (Sohn et al., 2020; Jeong & Shin, 2020; Sun et al., 2021; Zhao et al., 2022b), have also demonstrated the effectiveness of enforcing consistency among the predictions of the classifier to be helpful for semi-supervised learning, randomized smoothing, and other settings. To define the style consistency loss, let $(x, y) \sim \mathcal{D}_{\text{source}}$, $F : \mathbb{R}^d \to \Delta^{K-1}$ be the softmax output of the classifier such that the prediction of the base classifier $f(x) = \arg\max_{k \in \mathcal{Y}} F(x)$, Δ^{K-1} be the probability simplex in $\mathbb{R}^K$, $\overline{F}(x) = \mathbb{E}_{x_s \sim P_S}[F(h(t))]$ with $t = \text{AdaIN}(g(x), g(x_s))$ be the average softmax output of the classifier on stylized images, $\text{KL}(\cdot \| \cdot)$ be the Kullback–Leibler divergence (KLD) (Joyce, 2011) and $\text{H}(\cdot)$ be the entropy. Then the style consistency loss is given by

$$\begin{aligned}\mathcal{L}_{consistency} = \mathbb{E}_{x_s \sim P_S}[\text{KL}(\overline{F}(x) \| F(h(t)))] \\ + \text{H}(\overline{F}(x), y).\end{aligned} \tag{6}$$

In practice, we minimize the empirical version of the two losses using multiple-style images sampled randomly from the available source domain data. These losses can be easily combined with losses of other training methods and enable training a classifier that can achieve high accuracy on non-abstained samples even at low abstaining rates. The trained classifier can then be evaluated using TT-NSS as in Alg. 1 to gauge the reliability of their predictions on domains unseen during training.



(a) Original style (b) Wikiart style (c) Gaussian noise with sev. 3 (d) Gaussian noise with sev. 5

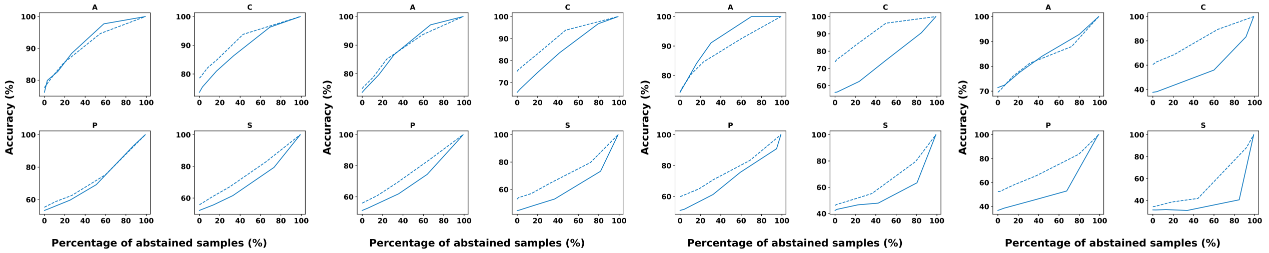
Figure 2. Comparison of TT-NSS (dashed lines) and confidence-based method (solid lines) in a single source domain setup. The graphs show accuracy vs abstained points for different datasets ((a) original, (b) wikiart, (c,d) corrupted), and different source/target domains. For most settings, the accuracy of the TT-NSS (dashed line) is higher than the corresponding accuracy of the confidence-based method (solid line) for most of the range of the percentage of abstained points. This demonstrates the superior performance of our style smoothing-based method as opposed to the conventional confidence-based method for producing risk-averse predictions. (Note: The source domain from PACS used for training is denoted in the title.)

3. Experiments

In this section, we present the evaluation results of using our inference and training procedures for obtaining and improving the reliability of the predictions from classifiers. We present evaluations and comparisons with classifiers trained with Empirical Risk Minimization (ERM) which achieves competitive results in single and multiple source domain settings (Gulrajani & Lopez-Paz, 2020) on the PACS (Li et al., 2017) dataset consisting of images from Art, Cartoons, Photos, and Sketch domains. Along with this we also present evaluations on variations of the PACS dataset generated by stylizing the images into the styles of Wikiart (Nichol., 2016) and adding common corruptions (Hendrycks & Dietterich, 2018) such as Gaussian noise to the images. These novel variations allow us to evaluate the performance of the classifiers on realistic changes that do not affect the semantic content of the images. To stylize PACS images into the style of Wikiart, we use the AdaIN decoder pre-trained using MS-COCO (Lin et al., 2014) along with images from Wikiart (Nichol., 2016) as style images. To create corrupted versions, we follow (Hendrycks & Dietterich, 2018) and use corruption with severity levels 3 and 5. Following previous works (Gulrajani & Lopez-Paz, 2020), we use a ResNet50 pre-trained on the Imagenet dataset as our backbone network augmented with a fully connected layer with softmax activation to produce predictions. We use this network for training ERM and for neural style smoothing-based training. For all experiments in the single source domain setup, we train the classifiers with a single source domain and evaluate the performance of the remaining three domains. For multiple source domains setup, we train the classifiers with three domains and test on the fourth unseen domain. We compare the performance of TT-NSS (Alg. 1) with another abstaining mechanism, applicable in a black box setting, that relies on the classifier's confidence on the original test sample. For the confidence-based method, we abstain if the highest softmax score for a sample is below a set threshold. For TT-NSS we use 100 randomly sampled style images ($n = 100$) for the single source domain setup and 150 for the multiple source domain setup. We use a subsample of the test set to report our results (see Appendix B.2). Evaluating a single test sample with TT-NSS using 150 styles requires $\approx$ 1.3 seconds on our hardware. We present the results of our evaluation with multiple source domains in the Appendix along with other implementation details in Appendix B.

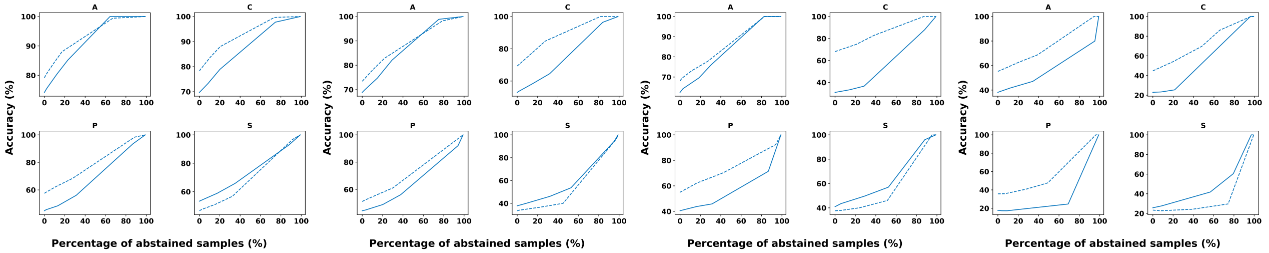
3.1. TT-NSS improves the reliability of the predictions from existing classifiers

In this section, we demonstrate the effectiveness of TT-NSS at producing reliable predictions from classifiers trained with ERM when evaluated on domains unseen during training. The results in Fig. 2 and 5 (in the Appendix) show the advantage of using the confidence of the style-smoothed classifier over the confidence of the original classifier to produce a risk-averse prediction on a test sample. Higher accuracy of the classifiers with TT-NSS at the same abstaining rates compared to the confidence-based strategy shows improved prediction reliability. This advantage of TT-NSS becomes more apparent on stylized and corrupted variants of the PACS dataset where the standard accuracy of the classifier drops significantly and necessitates abstaining for safeguarding against risky misclassifications. The classifier's high confidence incorrect predictions on unseen domains is the primary reason that prevents the confidence-based strategy from producing risk-averse predictions. This is in line with the findings from previous works which have shown that a classifier can produce high-confidence misclassification on samples from unseen domains (Hein et al., 2019; Mallick et al., 2020; Hendrycks & Gimpel, 2016; Zhang et al., 2017; 2020). On the other hand, using the



(a) Original style (b) Wikiart style (c) Gaussian noise with sev. 3 (d) Gaussian noise with sev. 5

Figure 3. Comparison of NSS training (dashed lines) vs ERM training (solid lines) in a single source domain setup. (See Fig. 2 for the explanation of settings.) NSS-trained classifiers evaluated with TT-NSS produce better accuracy on non-abstained samples at different abstaining rates compared to ERM-trained classifiers in the single source domain setup on different variants of the PACS dataset.



(a) Original style (b) Wikiart style (c) Gaussian noise with sev. 3 (d) Gaussian noise with sev. 5

Figure 4. Comparison of NSS training (dashed lines) vs ERM training (solid lines) in a single source domain setup. (See Fig. 2 for the explanation of settings.) NSS-trained classifiers when evaluated with the confidence-based method also produce better accuracy on non-abstained samples at different abstaining rates compared to ERM-trained classifiers in the single source domain setup on different variants of the PACS dataset.

confidence of the style-smoothed classifier, by stylizing the test sample into source domain styles, can mitigate the classifier's bias to non-semantic information in the test samples and produce better quality predictions.

3.2. Effectiveness of NSS at producing reliable classifiers

Here we demonstrate the advantage of using the NSS training procedure for improving the reliability of the classifier's predictions. Our results in Fig. 3 and 6 (in the Appendix) shows that classifiers trained with NSS achieve significantly better accuracy on non-abstained samples than the classifiers trained with ERM on all domains of PACS in both single and multiple source domain settings. Models trained with NSS show a significant advantage when the accuracy of the base classifier deteriorates such as in the case when corrupted variants of the PACS dataset are used. While results in Fig. 4 and 7 (in the Appendix) show that NSS-trained models achieve better accuracy at different abstaining rates even when evaluated with the confidence-based strategy, evaluating models with TT-NSS achieves significantly better results across all settings. Moreover, NSS also leads to an improved performance without any abstaining (i.e. at 0% abstaining) which further highlights the improved generalization performance of the NSS-trained classifiers on unseen domains.

4. Discussion and Conclusion

Our work proposed and demonstrated the effectiveness of incorporating an abstaining mechanism based on neural style smoothing to improve the reliability of a classifier's predictions on unseen domains. Using advances in neural style transfer, our inference procedure uses the prediction consistency of the classifier on stylized images to predict or abstain on a test sample and requires only black-box access to the classifier. We also proposed a novel training procedure to improve the reliability of a classifier's prediction at different levels of abstaining. While neural style smoothing is an effective way to gauge the prediction consistency of the classifier on test samples, ascertaining robustness to arbitrary style changes is important to ensure that classifiers make trustworthy predictions when used in the real world and will be dealt in our future works.

References

Albuquerque, I., Monteiro, J., Darvishi, M., Falk, T. H., and Mitliagkas, I. Generalizing to unseen domains via distribution matching. *arXiv preprint arXiv:1911.00804*, 2019.

Alcorn, M. A., Li, Q., Gong, Z., Wang, C., Mai, L., Ku, W.-S., and Nguyen, A. Strike (with) a pose: Neural networks are easily fooled by strange poses of familiar objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4845–4854, 2019.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Athalye, A., Carlini, N., and Wagner, D. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International conference on machine learning*, pp. 274–283. PMLR, 2018.

Baker, N., Lu, H., Erlikhman, G., and Kellman, P. J. Deep convolutional networks do not classify based on global object shape. *PLoS computational biology*, 14(12): e1006613, 2018.

Bartlett, P. L. and Wegkamp, M. H. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(8), 2008.

Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473, 2018.

Ben-David, S., Blitzer, J., Crammer, K., Pereira, F., et al. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19:137, 2007.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Blanchet, J. and Murthy, K. Quantifying distributional model risk via optimal transport. *Mathematics of Operations Research*, 44(2):565–600, 2019.

Bulusu, S., Kailkhura, B., Li, B., Varshney, P. K., and Song, D. Anomalous example detection in deep learning: A survey. *IEEE Access*, 8:132330–132347, 2020.

Calian, D. A., Stimberg, F., Wiles, O., Rebuffi, S.-A., Gyorgy, A., Mann, T., and Gowal, S. Defending against image corruptions through adversarial augmentations. *arXiv preprint arXiv:2104.01086*, 2021.

Charoenphakdee, N., Cui, Z., Zhang, Y., and Sugiyama, M. Classification with rejection based on cost-sensitive classification. In *International Conference on Machine Learning*, pp. 1507–1517. PMLR, 2021.

Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *ACM Workshop on Artificial Intelligence and Security*, pp. 15–26, 2017.

Chen, P.-Y., Sharma, Y., Zhang, H., Yi, J., and Hsieh, C.-J. EAD: elastic-net attacks to deep neural networks via adversarial examples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 10–17, 2018.

Chow, C. On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, 16(1):41–46, 1970.

Cohen, J. M., Rosenfeld, E., and Kolter, J. Z. Certified adversarial robustness via randomized smoothing. *arXiv preprint arXiv:1902.02918*, 2019.

Cortes, C., DeSalvo, G., and Mohri, M. Learning with rejection. In *International Conference on Algorithmic Learning Theory*, pp. 67–82. Springer, 2016.

Dittadi, A., Träuble, F., Locatello, F., Wüthrich, M., Agrawal, V., Winther, O., Bauer, S., and Schölkopf, B. On the transfer of disentangled representations in realistic settings. *arXiv preprint arXiv:2010.14407*, 2020.

Dumoulin, V., Shlens, J., and Kudlur, M. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.

Fischer, M., Baader, M., and Vechev, M. Certified defense to image transformations via randomized smoothing. *Advances in Neural Information Processing Systems*, 33: 8404–8417, 2020.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Gatys, L. A., Ecker, A. S., and Bethge, M. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

Gatys, L. A., Ecker, A. S., Bethge, M., Hertzmann, A., and Shechtman, E. Controlling perceptual factors in neural style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3985–3993, 2017.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.

Hein, M., Andriushchenko, M., and Bitterwolf, J. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 41–50, 2019.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Hendrycks, D. and Dietterich, T. G. Benchmarking neural network robustness to common corruptions and surface variations. *arXiv preprint arXiv:1807.01697*, 2018.

Hendrycks, D. and Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., and Lakshminarayanan, B. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019.

Hermann, K., Chen, T., and Kornblith, S. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33:19000–19015, 2020.

Huang, X. and Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 1501–1510, 2017.

Ilyas, A., Engstrom, L., Athalye, A., and Lin, J. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, pp. 2137–2146. PMLR, 2018.

Iwasawa, Y. and Matsuo, Y. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 34: 2427–2440, 2021.

Jeong, J. and Shin, J. Consistency regularization for certified robustness of smoothed classifiers. *Advances in Neural Information Processing Systems*, 33:10558–10570, 2020.

Johansson, F. D., Sontag, D., and Ranganath, R. Support and invertibility in domain-invariant representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 527–536. PMLR, 2019.

Johnson, J., Alahi, A., and Fei-Fei, L. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pp. 694–711. Springer, 2016.

Joyce, J. M. Kullback-leibler divergence. In *International encyclopedia of statistical science*, pp. 720–722. Springer, 2011.

Kireev, K., Andriushchenko, M., and Flammarion, N. On the effectiveness of adversarial training against common corruptions. *arXiv preprint arXiv:2103.02325*, 2021.

Lecuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., and Jana, S. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy (SP)*, pp. 656–672. IEEE, 2019.

Li, B., Chen, C., Wang, W., and Carin, L. Second-order adversarial attack and certifiable robustness. ., 2018.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pp. 5542–5550, 2017.

Li, L., Weber, M., Xu, X., Rimanic, L., Kailkhura, B., Xie, T., Zhang, C., and Li, B. Tss: Transformation-specific smoothing for robustness certification. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 535–557, 2021.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Mallick, A., Dwivedi, C., Kailkhura, B., Joshi, G., and Han, T. Probabilistic neighbourhood component analysis: sample efficient uncertainty estimation in deep learning. *arXiv preprint arXiv:2007.10800*, 2020.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

Mehra, A., Kailkhura, B., Chen, P.-Y., and Hamm, J. How robust are randomized smoothing based defenses to data poisoning? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13244–13253, 2021a.

Mehra, A., Kailkhura, B., Chen, P.-Y., and Hamm, J. Understanding the limits of unsupervised domain adaptation via data poisoning. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021b.

Mehra, A., Kailkhura, B., Chen, P.-Y., and Hamm, J. Do domain generalization methods generalize well? In *NeurIPS ML Safety Workshop*, 2022. URL https://openreview.net/forum?id=SRWIQ0Yl53m.

Montero, M. L., Ludwig, C. J., Costa, R. P., Malhotra, G., and Bowers, J. The role of disentanglement in generalisation. In *International Conference on Learning Representations*, 2020.

Nam, H., Lee, H., Park, J., Yoon, W., and Yoo, D. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8690–8699, 2021.

Ni, C., Charoenphakdee, N., Honda, J., and Sugiyama, M. On the calibration of multiclass classification with rejection. *Advances in Neural Information Processing Systems*, 32, 2019.

Nichol., K. Painter by numbers. https://www.kaggle.com/competitions/painter-by-numbers, 2016.

Qiao, F., Zhao, L., and Peng, X. Learning to learn single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12556–12565, 2020.

Raghunathan, A., Steinhardt, J., and Liang, P. Certified defenses against adversarial examples. *arXiv preprint arXiv:1801.09344*, 2018.

Settles, B. Active learning literature survey. ., 2009.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. Wasserstein distance guided representation learning for domain adaptation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Shu, M., Wu, Z., Goldblum, M., and Goldstein, T. Encoding robustness to image style via adversarial feature perturbations. *Advances in Neural Information Processing Systems*, 34:28042–28053, 2021.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C. A., Cubuk, E. D., Kurakin, A., and Li, C.-L. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33:596–608, 2020.

Sun, J., Mehra, A., Kailkhura, B., Chen, P.-Y., Hendrycks, D., Hamm, J., and Mao, Z. M. Certified adversarial defenses meet out-of-distribution corruptions: Benchmarking robustness and simple baselines. *arXiv preprint arXiv:2112.00659*, 2021.

Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., and Hardt, M. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Tang, Z., Gao, Y., Zhu, Y., Zhang, Z., Li, M., and Metaxas, D. N. Selfnorm and crossnorm for out-of-distribution robustness. ., 2020.

Ulyanov, D., Lebedev, V., Vedaldi, A., and Lempitsky, V. Texture networks: Feed-forward synthesis of textures and stylized images. *arXiv preprint arXiv:1603.03417*, 2016.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6924–6932, 2017.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Wang, H., Xiao, C., Kossaifi, J., Yu, Z., Anandkumar, A., and Wang, Z. Augmax: Adversarial composition of random augmentations for robust training. *Advances in Neural Information Processing Systems*, 34, 2021.

Wang, X., Oxholm, G., Zhang, D., and Wang, Y.-F. Multimodal transfer: A hierarchical deep convolutional neural network for fast artistic style transfer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5239–5247, 2017.

Xiao, C., Li, B., Zhu, J.-Y., He, W., Liu, M., and Song, D. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*, 2018.

Zhai, R., Dan, C., He, D., Zhang, H., Gong, B., Ravikumar, P., Hsieh, C.-J., and Wang, L. Macer: Attack-free and scalable robust training via maximizing certified radius. *arXiv preprint arXiv:2001.02378*, 2020.

Zhang, G., Zhao, H., Yu, Y., and Poupart, P. Quantifying and improving transferability in domain generalization. *arXiv preprint arXiv:2106.03632*, 2021a.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Zhang, H., Weng, T.-W., Chen, P.-Y., Hsieh, C.-J., and Daniel, L. Efficient neural network robustness certification with general activation functions. In *Advances in*

Neural Information Processing Systems, pp. 4944–4953, 2018.

Zhang, H., Zhang, Y.-F., Liu, W., Weller, A., Schölkopf, B., and Xing, E. P. Towards principled disentanglement for domain generalization. *arXiv preprint arXiv:2111.13839*, 2021b.

Zhang, J., Kailkhura, B., and Han, T. Leveraging uncertainty from deep learning for trustworthy materials discovery workflows. *arXiv preprint arXiv:2012.01478*, 2020.

Zhao, H., Zhang, S., Wu, G., Moura, J. M., Costeira, J. P., and Gordon, G. J. Adversarial multiple source domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Zhao, H., Des Combes, R. T., Zhang, K., and Gordon, G. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pp. 7523–7532. PMLR, 2019.

Zhao, Y., Zhong, Z., Luo, Z., Lee, G. H., and Sebe, N. Source-free open compound domain adaptation in semantic segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(10):7019–7032, 2022a.

Zhao, Y., Zhong, Z., Zhao, N., Sebe, N., and Lee, G. H. Style-hallucinated dual consistency learning for domain generalized semantic segmentation. *arXiv preprint arXiv:2204.02548*, 2022b.

Zhong, Z., Zhao, Y., Lee, G. H., and Sebe, N. Adversarial style augmentation for domain generalized urban-scene segmentation. *arXiv preprint arXiv:2207.04892*, 2022.

Zhou, K., Yang, Y., Qiao, Y., and Xiang, T. Domain generalization with mixstyle. *arXiv preprint arXiv:2104.02008*, 2021.

Appendix

A. Related work

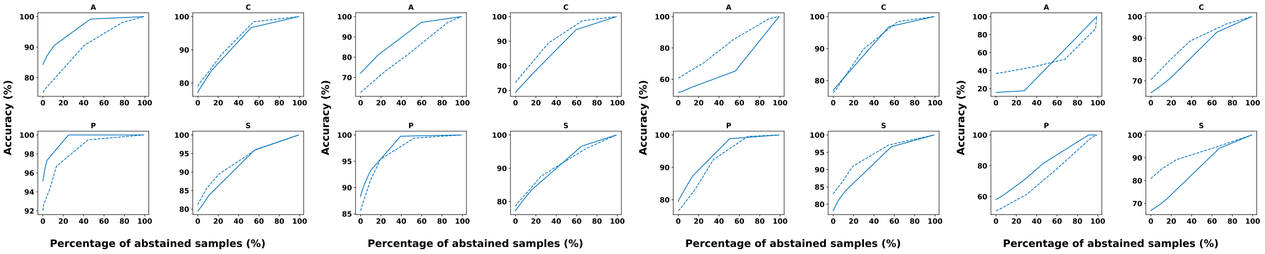
Domain generalization: The goal of domain generalization () is to produce classifiers whose accuracy remains high when faced with data from domains unseen during training. Many works have proposed to address this problem by capturing invariances in the data by learning a representation space that reduces the divergence between multiple source domains thereby promoting the use of only domain invariant features for prediction (Albuquerque et al., 2019; Zhang et al., 2021a; Ganin et al., 2016; Zhao et al., 2018; Qiao et al., 2020; Gulrajani & Lopez-Paz, 2020). Another line of work learns to disentangle the style and content information from the source domains and trains the classifier to be agnostic to the styles of the source domains (Arjovsky et al., 2019; Zhang et al., 2021b; Dittadi et al., 2020; Montero et al., 2020). Yet another line of research focuses on diversifying the source domain data to encompass possible variations that may be encountered at test time (Hendrycks et al., 2019; Wang et al., 2021; Kireev et al., 2021; Calian et al., 2021; Sun et al., 2021). Unlike previous works which focus on improving classifier accuracy on unseen domains, we focus on making risk-averse and improving the reliability of classifier predictions on unseen domains.

Certified robustness via randomized smoothing: Many works have demonstrated the failure of SOTA machine learning classifiers on adversarial examples which are crafted by adding imperceptible perturbations to test samples (Szegedy et al., 2013; Chen et al., 2018; Xiao et al., 2018; Chen et al., 2017; Ilyas et al., 2018). In response, many works proposed to provide empirical (Athalye et al., 2018) and provable (Li et al., 2018; Lecuyer et al., 2019; Cohen et al., 2019; Raghunathan et al., 2018; Zhang et al., 2018) robustness to these examples. Among them, Randomized Smoothing (RS) (Li et al., 2018; Lecuyer et al., 2019; Cohen et al., 2019) is one of the popular methods which provides provable robustness to adversarial examples by considering a smoothed version of the original classifier and certifying that no adversarial perturbation exists within a certified radius (in ℓ_2 norm) that can change the prediction of the classifier. RS uses Gaussian noise to produce a smoothed version of the base classifier. For a test sample, it then assigns the label which is most likely to be predicted by the base classifier on Gaussian perturbations of the test sample. While RS was proposed to certify the robustness to additive noise, the idea has been extended to certify robustness to parameterized transformations of the data such as geometric transformation (Fischer et al., 2020; Li et al., 2021) where the noise is added to the parameters of the transformations. Our neural style smoothed classifier is in a similar spirit to RS with crucial differences. Firstly, we use neural styles for smoothing (which cannot be parameterized) instead of adding Gaussian noise to the input or parameters of specific transformations. Secondly, our goal is not to provide certified robustness guarantees against style changes but to provide a practical method to produce reliable predictions on test samples and an abstaining mechanism to curb incorrect predictions.

Neural style transfer: Following the work of (Gatys et al., 2016), which for the first time demonstrated the effectiveness of using the convolutional layers of CNN for style transfer, several ways have been proposed to achieve better and faster neural style transfer (Gatys et al., 2017; Johnson et al., 2016; Ulyanov et al., 2016; Wang et al., 2017; Ulyanov et al., 2017; Dumoulin et al., 2016). AdaIN (Huang & Belongie, 2017) is a popular approach that allows arbitrary style transfer in real time by changing only the mean and variance of the convolutional feature maps. Other ways of generating stylized images include mixing styles (Zhou et al., 2021), exchanging (Tang et al., 2020; Zhao et al., 2022a) styles, or using adversarial learning (Zhong et al., 2022; Shu et al., 2021).

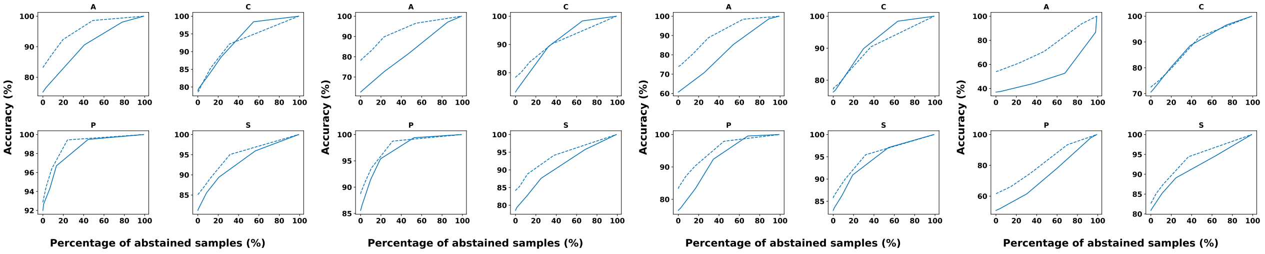
Test-time adaptation: Recent works have demonstrated the effectiveness of using test-time adaptation for improving generalization to unseen domains, where the classifier is updated on the incoming batch of test samples (Wang et al., 2020; Sun et al., 2020). This approach has also been shown to be effective in the setup (Iwasawa & Matsuo, 2021). Our approach is different from these methods since we do not update the classifier but rather only assume black-box access to it and produce the prediction of the smoothed classifier. Moreover, we use a single test sample, unlike previous methods which assume that the data from various unseen domains arrives in batches at test time.

Classification with abstaining: A learning framework allowing a classifier to abstain on samples has been studied extensively (Chow, 1970; Bartlett & Wegkamp, 2008; Ni et al., 2019; Charoenphakdee et al., 2021; Cortes et al., 2016). Two main approaches in these works include a confidence-based rejection where the classifier's confidence is used to abstain based on a predefined threshold and a classifier-rejector approach where the classifier and rejector are trained together. Our work is closer to the former since we do not train a rejector and abstain when the top class is not much more likely than other classes.



(a) Original style (b) Wikiart style (c) Gaussian noise with sev. 3 (d) Gaussian noise with sev. 5

Figure 5. Comparison of TT-NSS (dashed lines) and confidence-based method (solid lines) in a multiple source domain setup. The graphs show accuracy vs abstained points for different datasets ((a) original, (b) wikiart, (c,d) corrupted), and different source/target domains. For most settings, the accuracy of the TT-NSS (dashed line) is higher than the corresponding accuracy of the confidence-based method (solid line) for most of the range of the percentage of abstained points. This demonstrates the superior performance of our style smoothing-based method as opposed to the conventional confidence-based method for producing risk-averse predictions. (Note: The target domain from PACS used for evaluation is denoted in the title.)



(a) Original style (b) Wikiart style (c) Gaussian noise with sev. 3 (d) Gaussian noise with sev. 5

Figure 6. Comparison of NSS training (dashed lines) vs ERM training (solid lines) in the multiple source domain setup. (See Fig. 5 for the explanation of settings.) NSS-trained classifiers evaluated with TT-NSS produce better accuracy on non-abstained samples at different abstaining rates compared to ERM-trained classifiers in the multiple source domain setup on different variants of the PACS dataset.

B. Dataset and experimental details

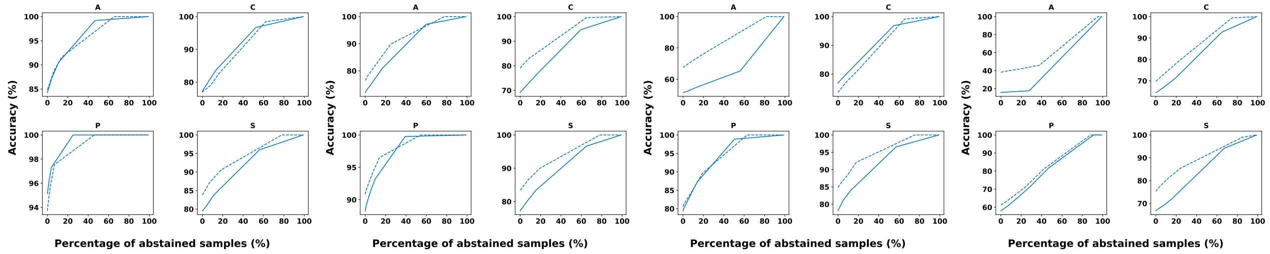
All codes are written in Python using Tensorflow/Pytorch and were run on an AMD EPYC 7J13 CPU with 200 GB of RAM and an Nvidia A100 GPU. Implementation and hyperparameters are described below.

B.1. Dataset description

In this work, we use the PACS dataset comprising of 9991 images belonging to 7 categories from four domains Art, Cartoons, Photos, and Sketches along with its stylized and corrupted version to evaluate the performance of various methods. For single source domain setting, we use 90% of the data for training and 10% for hyperparameter tuning, and for multiple source domains setting, we use 80% of the data for training and 20% for hyperparameter tuning.

B.2. Details of the subsample used for reporting the evaluation results

As mentioned in Sec. 3, to speed up the evaluation process when using TT-NSS, we present results on a subsample of the target domain. This approach has been used to report the results in previous works related to randomized smoothing (Cohen et al., 2019; Sun et al., 2021; Zhai et al., 2020; Mehra et al., 2021a). For the single source domain setting, we report the results on a balanced subsample of the dataset containing 50 images from each class and each target domain for PACS. For the multiple source domains setting, we use 100 images for each class of the target domain for PACS. For classes with fewer samples, we use all the samples from that class. This subsample is used to report the results for the dataset in the original style and the Wikiart style. For reporting results on the corrupted version of the dataset, we create a balanced subsample of roughly one-fifth of the samples chosen for other styles (e.g. we used 10 images per class for each target domain in a single source domain setting for PACS) and report the results by averaging over all ten corruption types.



(a) Original style (b) Wikiart style (c) Gaussian noise with sev. 3 (d) Gaussian noise with sev. 5

Figure 7. Comparison of NSS training (dashed lines) vs ERM training (solid lines) in multiple source domain setup. (See Fig. 5 for the explanation of settings.) NSS-trained classifiers when evaluated with the confidence-based method also produce better accuracy on non-abstained samples at different abstaining rates compared to ERM-trained classifiers in the multiple source domains setup on different variants of the PACS dataset.

B.3. Experimental details

To train the classifiers with NSS, we incorporate style augmentation and style consistency losses computed on stylized versions of the source domain images generated through the AdaIN decoder. We additionally incorporate the ERM training loss which minimizes the misclassification on original source domain samples. As mentioned in Sec. 2 other losses used in specific algorithms can also be incorporated to improve the quality of risk-averse predictions from classifiers trained with those methods. To compute the style consistency loss we use four different styles for every sample in the batch and use a batch size of 16. These losses are then used to fine-tune the ResNet50 backbone augmented with a fully connected layer used for classification. For the multiple source domains setting, the classifier that achieves the highest accuracy on the validation set is used for final evaluation whereas for the single source domain setting, the classifier at the last step is used for final evaluation.