

# GDA: Generalized Diffusion for Robust Test-time Adaptation

Yun-Yun Tsai<sup>1\*</sup>, Fu-Chen Chen<sup>2</sup>, Albert Y. C. Chen<sup>2</sup>,  
Junfeng Yang<sup>1</sup>, Che-Chun Su<sup>2</sup>, Min Sun<sup>2</sup>, Cheng-Hao Kuo<sup>2</sup>

<sup>1</sup>Columbia University, <sup>2</sup>Amazon

<sup>1</sup>{yunyuntsai, junfeng}@cs.columbia.edu

<sup>2</sup>{cfchen, aycchen, ccusu, minnsun, chkuo}@amazon.com

## Abstract

Machine learning models face generalization challenges when exposed to out-of-distribution (OOD) samples with unforeseen distribution shifts. Recent research reveals that for vision tasks, test-time adaptation employing diffusion models can achieve state-of-the-art accuracy improvements on OOD samples by generating domain-aligned samples without altering the model’s weights. Unfortunately, those studies have primarily focused on pixel-level corruptions, thereby lacking the generalization to adapt to a broader range of OOD types. We introduce Generalized Diffusion Adaptation (GDA), a novel diffusion-based test-time adaptation method robust against diverse OOD types. Specifically, GDA iteratively guides the diffusion by applying a marginal entropy loss derived from the model, in conjunction with style and content preservation losses during the reverse sampling process. In other words, GDA considers the model’s output behavior and the samples’ semantic information as a whole, reducing ambiguity in downstream tasks. Evaluation across various model architectures and OOD benchmarks indicates that GDA consistently surpasses previous diffusion-based adaptation methods. Notably, it achieves the highest classification accuracy improvements, ranging from 4.4% to 5.02% on ImageNet-C and 2.5% to 7.4% on Rendition, Sketch, and Stylized benchmarks. This performance highlights GDA’s generalization to a broader range of OOD benchmarks.

## 1. Introduction

Deep networks have achieved unprecedented performance in many machine learning applications, yet unexpected corruptions and natural shifts at test time [9–11, 14, 27] still degrade their performance severely. This vulnerability hinders the deployment of machine learning models in the real world, especially in safety-critical, high-stake applications [32].

Test-time adaptation (TTA) [44, 50] emerges as a new

\*Work done in Amazon applied scientist internship

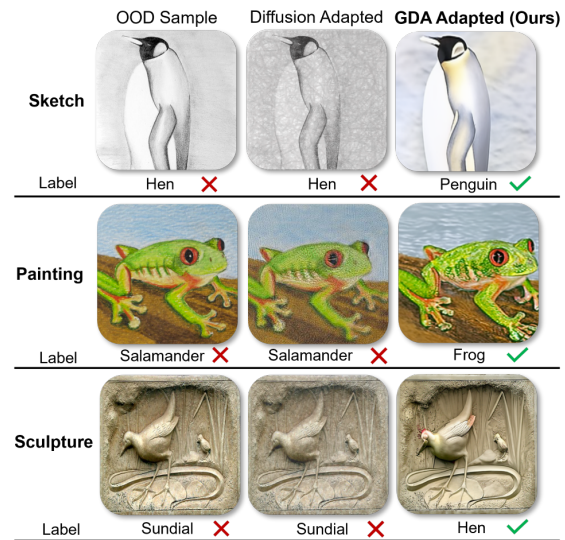


Figure 1. Sample OOD data and adaptations via existing diffusion method and our GDA method. The leftmost column shows OOD samples under different style changes, including sketch, painting, and sculpture. The middle column shows samples adapted by traditional diffusion. The rightmost column shows samples adapted with our GDA method. The visualization shows that GDA can generate samples with multiple visual effects, such as re-colorization for the sketch sample, texture enhancement for the painting sample, and object highlighting for the sculpture sample. All three GDA-adapted samples are correctly classified by ResNet50, whereas all others are misclassified.

branch to improve out-of-distribution robustness by adjusting either the model weights or the input data. The former assumes that the weights are not frozen, and can be modified iteratively during test time [40, 44, 50]. It thus requires edit access to the model and complicates model maintenance because all adapted model versions need to be tracked. The latter modifies the input with random noise vectors or structural visual prompts [25, 41–43] optimized for pre-defined objectives. The visual prompt design is, however, prone to overfitting due to the high dimensionality of the prompts.

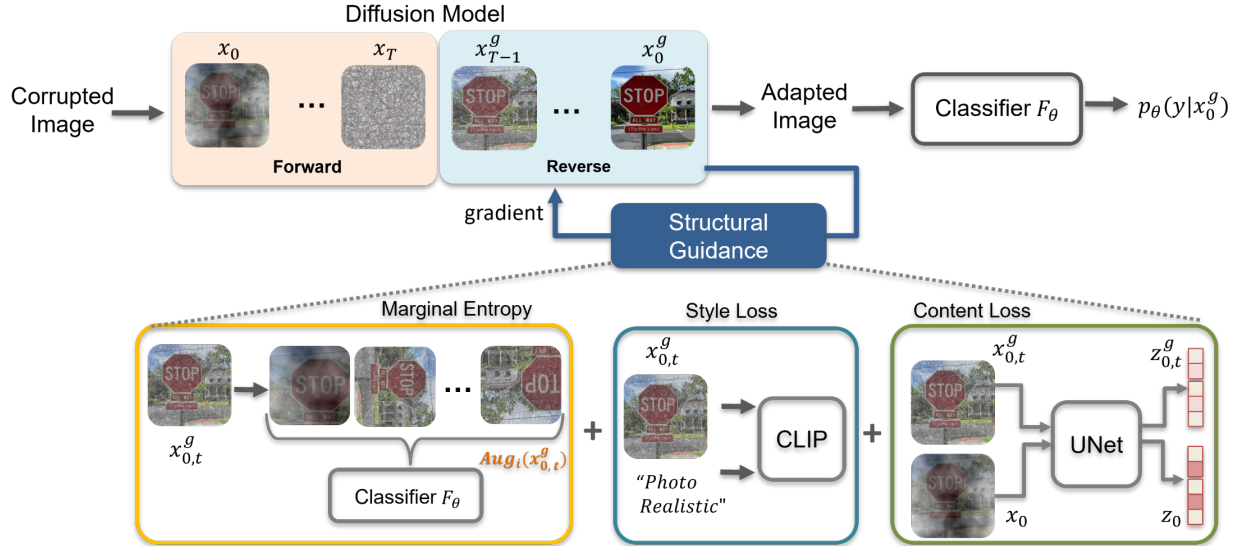


Figure 2. The flow of GDA. We guide the diffusion model with our novel structural guidance that includes marginal entropy, style loss, and content preservation loss. Given the corrupted samples  $x_0$ , when going through the reverse process at step  $t$ , our structural guidance will first (1) Generate the sample  $x_{t-1}^g$  for the next reverse time step  $t-1$ . (2) Update the  $x_{t-1}^g$  with the gradient calculated from the losses. Our loss is computed by the reference image  $x_0$  and its corresponding denoised image  $\hat{x}_{0,t}^g$  conditioned on  $x_t^g$  at reverse time step  $t$ .

Therefore, we focus on a new branch of test-time adaptation, diffusion-based adaptation, that does not need to modify model weights and provides more structured guidance. Prior work [2, 5] shows that diffusion is powerful for transferring style and countering natural corruptions by adding simple structural guidance, a latent refinement step conditioned on the input of the reverse process (e.g., a sequence of up-scaling and down-scaling processes). However, the key performance gain of prior work [5] is shown only in specific corruption types, such as the Gaussian noise or Impulse noise. The results imply two challenges that limit the generalizability of diffusion for adaptation: (1) The structural guidance in prior work can handle only high-frequency corruption and does not generalize well to other types of corruption. (2) The diffusion model is fully trained on the source domain data, which potentially causes learning biases and can fail to restore the distribution shift in OOD data.

To address these challenges and improve the generalizability of diffusion models, we propose *Generalized Diffusion Adaptation (GDA)*, an efficient diffusion-based adaptation method robust against diverse OOD shifts at test time, including style changes and multiple corruptions. Our key idea is a new structural guidance for unconditional diffusion models, consisting of three components: *style transfer*, *content preservation*, and *model output consistency*. We show sample OOD data adapted by GDA in Fig. 1 and demonstrate the schematic in Fig. 2. To let the corrupted sample shift back to the source domain, GDA incorporates the structural guidance into the reverse process, which has three components: (1) The style loss utilizes CLIP model to transfer the image style;

(2) The patch-wise contrastive loss calculated from samples' features aims to preserve the content information; (3) The marginal entropy loss calculated on samples and its augmenting version for ensuring the consistency of output behavior on the downstream task. During the reverse process, GDA iteratively updates the generated samples for every time step by calculating the gradient from three objectives.

The trade-off between style transfer and content preservation in the diffusion model has been studied by [48]. However, the output behavior of the downstream classifier on the generated samples is still unexplored in the diffusion-driven adaptation, which is crucial to the robustness. Our key insights are: (1) Marginal entropy can measure the ambiguity of the unlabeled data with respect to the target classifier [7, 50]. (2) The marginal entropy calculated from a sample without corruption (clean sample) and its augmented versions is usually lower than a corrupted sample; clean samples are typically less ambiguous to the target classifier. (3) The diffusion guided with marginal entropy will move the sample away from the decision boundary.

Our main contributions are as follows.

- We propose Generalized Diffusion Adaptation (GDA), a new diffusion-based adaptation method that generalizes to multiple local-texture and style-shifting OOD benchmarks, including ImageNet-C, Rendition, Sketch, and Stylized-ImageNet.
- Our key innovation is a new structural guidance towards minimizing marginal entropy, style, and content preservation loss. We demonstrate that our guidance is both effective and efficient as GDA reaches higher or on-par

accuracy with fewer reverse sampling steps.

- GDA outperforms state-of-the-art TTA methods, including DDA [5] and Diffpure [30] on four datasets with respect to target classifiers of different network backbones (ResNet50 [8], ConvNext [23], Swin [22], CLIP [34]).
- Ablation studies show that GDA indeed minimizes the entropy loss, enhances the corrupted samples, and recovers the correct attention of the target classifier.

## 2. Related Works

### 2.1. Domain Adaptation

Various types of out-of-distribution data (OOD) have been widely studied in recent works to show that OOD data can lead to a severe drop in performance for machine learning models [9, 14, 24, 26, 27, 35]. To improve the model robustness on OOD data, one can make the training robust by incorporating the potential corruptions or distribution shifts from the target domain into the source domain training data [14]. However, anticipating unforeseen corruption at training time is not realistic in practice. Domain generalization (DG) aims to adapt the model with OOD samples without knowing the target domain data during training time. Existing adaptation methods [4, 19, 24, 26, 37, 40, 44, 50–52] have shown significant improvement on model robustness for OOD datasets.

### 2.2. Test-time Adaptation

Test-time adaptation is a new paradigm for robustness to distribution shifting [25, 40, 50] by either updating the weights of deep models or updating the input. BN [20, 37] updates the model using batch normalization statistics. TENT [40] adapts the model weight by minimizing the conditional entropy on every batch. TTT [40] attempts to train the model with an auxiliary self-supervision model for rotation prediction and utilize the self-supervised loss to adapt the model. MEMO [50] augments a single sample and adapts the model with the marginal entropy of those augmented samples. Test-time transformation ensembling (TTE) [33] augments the image with a fixed set of transformations and aggregates the outputs through averaging. Input-based adaptation methods focus on efficient weight tuning [18, 25, 41–43] with prompting technique, which modify the pixels of input samples by minimizing the self-supervised loss. Tsai et al. [42] adapt the input by adding a learnable small convolutional kernel and optimizing the parameters during the test time. Mao et al. [25] add an additional vector to reverse the adversarial samples by minimizing the contrastive loss.

### 2.3. Diffusion Model for Domain Adaptation

Recent works have shown diffusion models emerge as a powerful tool to generate synthetic samples [29, 36, 39]. A large body of work has studied high-quality image generation by diffusion models. Diffusion models can be widely applied

to various computer vision areas, such as super-resolution, segmentation, and video generation [16, 17, 21, 38, 47]. In particular, they learn how to reverse the sample from noisy to clean during the training process and the samples are usually drawn from a single source domain. Several works study using diffusion for image purification from out-of-domain data (e.g., corruption or adversarial attack) [5, 30]. Diffpure [30] purifies the adversarial samples by diffusion model by solving the stochastic differential equation (SDE) and calculating the gradient during the reverse process. DDA [5] applies diffusion to adapt the OOD samples with multiple corruption types and shows the diffusion-based adaptation is more robust than the model adaptation. However, this approach can only adapt well to noise-type corruption and requires large number of reverse sampling steps (e.g., 50). ILVR [2] attempts to generate diverse samples with image guidance using unconditional diffusion models, but the stochastic nature posed a challenge. In our work, we investigate how to enlarge the capability of diffusion with a more structured guidance. DSI [49] improves OOD robustness by linearly transforming the distribution from target to source and filtering samples with the confidence score. Different from prior works, GDA applies a new structural guidance conditioned on style, content information, and model’s output behavior during the sampling process in diffusion models. Our structural guidance is target domain-agnostic, meaning we do not access any ground-truth label or style information of input samples during test time.

## 3. Generalized Diffusion Adaptation

We now introduce our generalized diffusion-based adaptation method (GDA). Given an unconditional diffusion model pre-trained on the source domain  $\mathcal{X}_S$  and an input image  $x_0$  sampled from the target domain  $\mathcal{X}_T$ , the diffusion model should generate samples  $\hat{x}_0$  for  $x_0$ , and the generated samples  $\hat{x}_0$  should move closer to the source domain  $\mathcal{X}_S$ .

We apply the DDPM in our adaptation. Given an image  $x_0$  sampled from the target domain  $\mathcal{X}_T$ , DDPM first gradually adds Gaussian noise to the data point  $x_0$  through a fixed Markov chain during the forward process for  $T$  steps. Specifically, we sample data sequence  $[x_0, x_1, \dots, x_T]$  by adding Gaussian noise with variance  $\beta_t \in (0, 1)$  at timestep  $t \in [1, \dots, T]$  during the forward process, defined as:

$$q(x_t|x_0) = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon, \quad (1)$$

where  $\epsilon \sim \mathcal{N}(0, 1)$  is the noise we add,  $\alpha_t = 1 - \beta_t$ , and  $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$ . The reverse process then generates a sequence of denoised image  $[x_t^g, x_{t-1}^g, \dots, x_0^g]$  from timestep  $t \in [T, \dots, 1]$ . For timestep  $t$  in the reverse process, the denoised image can be defined as:

$$x_{t-1}^g = \frac{1}{\sqrt{\alpha_t}} \left( x_t^g - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \epsilon_\theta(x_t^g, t) \right) + \sigma_t \epsilon, \quad (2)$$

---

**Algorithm 1: Generalized Diffusion Adaptation**


---

**Input:** Pretrained classifier  $\mathcal{F}(\cdot)$ , Augment function set  $\mathcal{A}$ , OOD images  $x_0$ , Diffusion time step  $T$ , Objective function  $\ell_{style}(\cdot)$ ,  $\ell_{content}(\cdot)$ , and  $\ell_{marginal}(\cdot)$ , target prompt  $r$ , Uncertainty score function  $H(\cdot)$

**Output:** Class prediction  $\hat{y}$  for adapted sample of  $x$

**Inference**

```

 $x_T^g \leftarrow q(x_T|x_0)$ ,  $x_T^g \sim \mathcal{N}(1, 0)$  // forward process
for  $t \in \{T, \dots, 1\}$  do
   $\hat{x}_{t-1}^g = p_\theta(x_{t-1}^g|x_t^g)$  // reverse process
   $x_{0,t}^g = \sqrt{\frac{1}{\alpha_t}}x_t^g - \sqrt{\frac{1-\alpha_t}{\alpha_t}}\epsilon_\theta(x_t^g, t)$ 
   $\ell_{guided} = \ell_{content}(x_{0,t}^g, x_0)$  // structural guidance
   $\quad + \ell_{style}(x_{0,t}^g, r) + \ell_{marginal}(\mathcal{F}(\mathcal{A}(x_{0,t}^g)))$ 
   $x_{t-1}^g = \hat{x}_{t-1}^g + \nabla_x \ell_{guided}(x)|_{\{x=x_{0,t}^g, x_0\}}$ 
if  $H(x_0^g) < H(x_0)$  then
   $x_0^g \leftarrow x_0$  // confidence filtering
else
   $x^* \leftarrow x_0$ 
return  $\hat{y} \leftarrow \mathcal{F}(x^*)$ 

```

---

where  $\epsilon_\theta$  is a trainable noise predictor that generates a prediction for the noise at the current timestep and removes the noise.  $\sigma_t$  is the variance of noise. Ideally, the generated sample  $x_0^g$  should be moved forward to the distribution of the source domain trained for the diffusion model.

**Structural Guidance in Diffusion Reverse Process** The trade-off between preserving content while translating domains or style has been studied by DDA [5, 48]. When the noise variance  $\sigma$  is more extensive, it is challenging to preserve the content information. Therefore, the structural guidance allows the diffusion model to generate samples conditioned on the predefined objectives. In particular, the structural guidance iteratively refines the latent for the input images during the reverse process so that the content information in the sample can be preserved while translating the style or shifting the domain.

Due to the sampling process of DDPM being a Markov chain, it requires all past denoising steps to obtain the next denoised image. The long stochastic operations can lead to huge distortion of the content information. To guide the diffusion more efficiently with structural guidance, we speed up the sampling process with DDIM [38] by skipping several reverse steps. The reverse process can be redefined as:

$$x_{t-1}^g = \sqrt{\bar{\alpha}_{t-1}}(x_t^g - x_{0,t}^g) + \sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2}\epsilon_\theta(x_t^g, t) + \sigma_t\epsilon \quad (3)$$

where  $x_{0,t}^g$  is the predicted denoised image for  $x_0$  conditioned on  $x_t^g$  at the time step  $t$  and is defined as:

$$x_{0,t}^g = \frac{x_t^g - \sqrt{1 - \bar{\alpha}_t}\epsilon_\theta(x_t^g, t)}{\sqrt{\bar{\alpha}_t}}, \quad (4)$$

Our structural guidance has two steps: (1) At time step  $t$ , generate the sample  $x_{t-1}^g$  for the next step  $t - 1$ . (2)

Update  $x_{t-1}^g$  with the gradient calculated from our structural guidance. To avoid the conflicting with the original reverse sampling step at each time step in the diffusion, our structural guidance is computed by the reference image  $x_0$  and its corresponding denoised image  $x_{0,t}^g$  at reverse time step  $t$ . The updated process is defined as:

$$\hat{x}_{t-1}^g \sim p_\theta(x_{t-1}^g|x_t^g) \\ x_{t-1}^g = \hat{x}_{t-1}^g + \nabla_x \ell_{guided}(x)|_{\{x=x_{0,t}^g, x_0\}} \quad (5)$$

where  $\ell_{guided}$  is our objective function for structural guidance, and the inputs of the objective are  $x_{0,t}^g$  and  $x_0$ .

**Sampling Strategy** In Algorithm 1, we present GDA. Our proposed structural guidance incorporates the marginal entropy loss into the objective function to ensure the output behavior of the model has consistent predictions on generated samples and their augmented version. Inspired by [48], we combine text-driven style transfer using CLIP and content preservation using zero-shot contrastive loss. Our objective function is:

$$\ell_{guided}(\cdot) = \ell_{marginal}(\cdot) + \ell_{style}(\cdot) + \ell_{content}(\cdot) \quad (6)$$

where  $\ell_{marginal}$  denotes the marginal entropy loss.  $\ell_{style}$  and  $\ell_{content}$  denote the style and content preservation loss. We further discuss the details for each loss component.

**Marginal Entropy Loss** We notice the stochastic nature of the diffusion model in the reverse process, where the noise  $\epsilon$  can lead to the distortion of content information in the input image and cannot correctly generate samples close to the source domain that diffusion has been trained on. Given a model  $f_\theta$  which is trained on the source domain, we add the marginal entropy loss for guiding the diffusion reverse process. In particular, the loss will force the whole diffusion process to generate samples that can decrease the model's uncertainty for  $f_\theta$ . At timestep  $t$ , given a generated sample  $x_t^g$  and a set of augmentation functions  $\mathcal{A} = \{A_1, A_2, \dots, A_n\}$ , we augment the sample  $x_t^g$  by choosing subset of augmentation functions from  $\mathcal{A}$ . We denote the image sequence of augmented data as  $A_1(x_t^g), A_2(x_t^g), \dots, A_k(x_t^g)$ , where  $k \leq n$ . The marginal output distribution for the given generated sample  $x_t^g$  is defined as:

$$\bar{p}_\theta(y|x_t^g) \approx \frac{1}{k} \sum_{i=1}^k p_\theta(y|A_i(x_t^g)), \quad (7)$$

where  $p_\theta$  is the output prediction of each augmented sample and  $\bar{p}_\theta$  is the average on all augmented samples. Our intuition lies in that  $f_\theta$  is trained on the source domain  $\mathcal{X}_S = [x_1, x_2, \dots, x_N]$  and should learn the invariance between the augmented samples  $\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N$  and  $\mathcal{X}_S$ . When generating a sample  $x_t^g$  at time step  $t \in [T, \dots, 1]$  from diffusion, if the sample is close to the source domain, the output

prediction of its augmented versions will be consistent, and the marginal entropy loss will become small. Thus, we can utilize this loss to ensure the diffusion process generates samples close to the source domain. Here, the entropy of marginal output distribution is defined as:

$$\ell_{marginal} = - \sum_{y \in \mathcal{Y}} \bar{p}_\theta(y|\mathcal{A}(x_t^g)) \log \bar{p}_\theta(\mathcal{A}(x_t^g)) \quad (8)$$

To better control the sample quality from the diffusion model, the uncertainty estimation on original and adapted samples is then applied to the sampling strategy. The uncertainty score function is  $H(x) = - \sum_{y \in \mathcal{Y}} p_\theta(y|x) \log p_\theta(x)$ , where the input can be the original sample  $x$  or adapted samples  $x_0^g$ .

**Style and Content Loss** To transfer samples from one style to another without content distortion, prior work proposed guided-loss for the diffusion model [48]. Inspired by them, We use the CLIP model to calculate the style loss. By injecting a text prompt related to the source domain (e.g., *photo-realistic, real*), the CLIP model calculates the similarity between the features extracted from the input image and the text prompt. Our style loss is defined as:

$$\ell_{style} = \frac{Enc_{img}(x_{0,t}^g) \cdot Enc_{txt}(t)}{\|x_{0,t}^g\| \cdot \|t\|}, \quad (9)$$

where  $Enc_{img}$  and  $Enc_{txt}$  are the image and text encoder in the CLIP model.

To avoid content distortion, we use patch-wise contrastive loss to ensure the generated sample’s content information is consistent with the original sample. In [31], they show contrastive unpaired image-to-image translation loss can preserve the content information by maximizing the mutual information between the input and output patches. To compute the content preservation loss, we extract the spatial features from the UNet component of the diffusion model. The content preservation loss is:

$$\ell_{content} = - \log y_{i,j} \frac{\exp(\hat{z}_i^T z_j) / \tau}{\sum_{k \neq i} \exp(\hat{z}_i^T z_k) / \tau}, \quad (10)$$

where  $\hat{z}$  and  $z$  are the corresponding patch-wise features of  $x_{0,t}^g$  and  $x_0$  extracted from UNet  $h(\cdot)$ .  $\tau$  is the temperature scaling value.  $y_{i,j}$  is a 0-1 vector for indicating the positive pairs and negative pairs. If  $y_{i,j}$  is 1, the  $i$ -th feature  $\hat{z}_i$  and  $j$ -th feature  $z_j$  are at the same location from the  $x_{0,t}^g$  and  $x_0$  samples. Otherwise, they are from different locations.

## 4. Experiment

This section presents the details of our experiment settings and evaluates the performance of our method. We comprehensively study multiple types of corruption and style-changed OOD benchmarks. More analyses are shown in Section 5 and Appendix, including sensitivity analysis on different adaptation methods and sample visualization.

### 4.1. Experimental Setting

**Dataset.** We evaluate our method on four kinds of OOD datasets: ImageNet-C [28], ImageNet-Rendition [13], ImageNet-Sketch [46], and ImageNet-Stylized [15]. The following describes the details of all datasets.

- **Natural OOD Data.** ImageNet-Rendition [14] contains 30,000 images collected from Flickr with specific types of ImageNet’s 200 object classes. ImageNet-Sketch [46] consists of 50000 sketch images that greatly degrade the performance on large-scale image classifiers.

- **Synthetic OOD Data.** The corruption data is synthesized with different types of transformations (e.g., snow, brightness, contrast) to simulate real-world corruption. ImageNet-C is the corrupted version of the original ImageNet dataset, including 15 corruption types and five severity levels. To evaluate our method, we generate the corruption samples with severity level 3 based on the official GitHub code [10] for each of the 15 corruption types. ImageNet-stylized [15] is another synthetic dataset with huge style change, including eight kinds of styles (e.g., oil painting, sculpture, watercolor, ... etc.). The local textures are heavily distorted, while global object shapes remain (more or less) intact during stylization. We generate the stylized-ImageNet based on the official code [6]

**Model.** We use an unconditional 256\*256 diffusion model trained with the original ImageNet dataset [3]. For the downstream classification models, we test on several architectures, including traditional CNNs, ResNet50 [8] and ConvNext [23]; and state-of-the-art transformer Swin [22].

**Baseline Details** We compare our method to several baselines, including standard models without adaption and diffusion-based adaption.

- **Standard:** This baseline uses the three pre-trained classification models without adaptation.

- **DDA [5]:** This diffusion-based adaptation method provides structural guidance by adding a linear low-pass filter  $\mathcal{D}$ , a sequence of downsampling and upsampling operations. We set the reverse step of DDA as 10. The samples will first go through the reverse process and the latent refinement step computes the difference between the output of  $\mathcal{D}$  on reference image  $x_0$  and the generated image.

- **Diffpure [30]:** This baseline uses the diffusion model to purify adversarial samples. It provides an ad-joint method to compute full gradients of the reverse generative process by solving the SDE. Diffpure and DDA rely on the same unconditional diffusion model but differ in their reverse steps and guidance.

- **w/o marginal:** To understand how every objective in our method contributes to the optimization, we remove marginal loss from our method and use only the style and content preservation loss.

	ResNet50	ConvNext-T	Swin-T
<b>Standard</b>	37.30	59.60	54.33
<b>Diffpure [30]</b>	15.83	47.23	35.69
<i>DDA</i> <sub>10</sub> [5]	38.90	63.26	49.65
<b>w/o marg.</b>	40.9	59.70	55.86
<b>GDA (ours)</b>	<b>41.70</b>	<b>65.24</b>	<b>59.35</b>

Table 1. Classification accuracy on the ImageNet-C under severity level 3 for three model architectures. We compare the result between GDA and the four baselines, including Standard, Diffpure [30], DDA [5], and w/o marginal. GDA consistently achieves the highest accuracy (numbers in bold).

	ResNet50	ConvNext-T	Swin-T	CLIP-B/16
<b>Rendition</b>				
<b>Standard</b>	37.0	49.8	43.6	72.7
<b>Diffpure</b>	29.8	49.4	43.5	71.4
<i>DDA</i> <sub>50</sub>	42.0	51.8	42.1	70.6
<b>w/o marg.</b>	39.4	50.5	44.2	73.4
<b>GDA (ours)</b>	<b>44.5</b>	<b>52.4</b>	<b>47.6</b>	<b>76.5</b>
<b>Sketch</b>				
<b>Standard</b>	23.0	35.4	29.0	50.7
<b>Diffpure</b>	13.9	37.4	27.2	48.9
<i>DDA</i> <sub>50</sub>	23.5	34.0	27.1	44.9
<b>w/o marg.</b>	23.9	35.7	31.1	51.2
<b>GDA (ours)</b>	<b>25.5</b>	<b>38.5</b>	<b>35.9</b>	<b>55.5</b>
<b>Stylized</b>				
<b>Standard</b>	16.5	35.3	27.3	22.4
<b>Diffpure</b>	6.1	19.8	16	22.4
<i>DDA</i> <sub>50</sub>	19.2	27.8	18.8	21.7
<b>w/o marg.</b>	20.1	36.6	30.9	22.6
<b>GDA (ours)</b>	<b>23.0</b>	<b>41.6</b>	<b>32.3</b>	<b>25.1</b>

Table 2. The classification accuracy on three OOD benchmarks, including Rendition, Sketch, and Stylized-ImageNet under four model architectures, including ResNet50, ConvNext-T, Swin-T, and CLIP-B/16. We set the timestep for DDA as 50. Numbers in bold show the best accuracy.

**Implementation Details** We adopt the DDPM strategy on the forward and reverse sampling process. The total time step  $t$  is set as 50. We replace the step size from  $T$  to  $t$ , where  $t \in [0, 50]$ . Given an input image  $x_0$ , we obtain the  $x_t$  at time step  $t$  from the forward diffusion process. We combine the three loss terms as a joint optimization, with their Lagrange multipliers as hyperparameters. The hyperparameter values for each benchmark are shown in Appendix Table 7. For the augmentation function  $\mathcal{A}$  in marginal entropy loss, we use AugMix [12], a data augmentation tool from Pytorch, which randomly select several augmentation functions (e.g., posterize, rotate, equalize) to augment the data.

## 4.2. Experimental Results

Table 1 shows the results on ImageNet-C. Compared with the three standard models without adaptation, including ResNet50, ConvNext-Tiny, and Swin-Tiny, GDA improves the performance by 4.4% ~ 5.64%. Compared with DDA [5] and Diffpure [30], GDA outperforms them by 2 ~ 4% on average. Besides, to study the effect of marginal entropy, the without marginal shows the baseline without guiding with the marginal entropy loss. Our results show that the diffusion

	GDA (ours)					
# of Aug.	0	2	4	8	16	32
<b>Rendition</b>	39.4	39.7	40.5	44.2	<b>44.5</b>	44.7
<b>Sketch</b>	23.9	22.8	23.2	24.3	<b>25.5</b>	25.3
<b>Stylized</b>	20.1	19.4	19.6	21.7	<b>23.0</b>	23.5

Table 3. The classification accuracy of GDA with different augmentation numbers on Rendition, Sketch, and Stylized-ImageNet OOD benchmarks using ResNet50 model architecture. When number of augmentation is 0, we show the results of GDA w/o marginal guidance. The accuracy values start to saturate when the number of augmentations exceeds 16.

model can effectively guide the sample back to the source domain with marginal entropy guidance when compared with no marginal guidance and can improve the accuracy by 5.2%. Fig. 3 shows the details of the performance for every 15 corruption types under three model architectures compared with four baselines. In Table 2, we further demonstrate the performance on Rendition, Sketch, and Stylized-ImageNet, which are more challenging datasets with massive style changes. For the Rendition, our method can improve by 2.6~7.4% robust accuracy compared with three standard model and outperform state-of-the-art by 0.6%~5.5%. For the Sketch, GDA can improve the accuracy by 2.5%~6.9%. We show the state-of-the-art DDA and Diffpure do not have any improvement on the performance for Sketch dataset. For the Stylized-ImageNet, we improve the accuracy by 6.4% on average and outperform the state-of-the-art DDA by 2.7~5%. In Appendix 8, we show more experimental results of GDA on ImageNet-C severity 5, and the comparison with other model adaptation baselines.

**Number of augmentation in marginal guidance** In Table 3, we show the performance of guiding with marginal entropy loss under different numbers of augmentation on three OOD benchmarks, including Rendition, Sketch, and Stylized-ImageNet. For every step, the marginal entropy loss is computed based on all augmented samples. We set the number of augmentations from 2 to 32. Our result shows that when increasing the number of augmentations to 8 and 16, the performance significantly increases on every benchmark. To be more efficient, in our experiment, we set up the number of augmentation for marginal entropy loss as 16.

## 5. Ablation Studies

**Entropy Loss Measurement** We do the quantitative measurement of our method by showing the entropy loss distribution for different corruptions. Our entropy is defined as  $H(x) = -\sum_{y \in \mathcal{Y}} p_{\theta}(y|x) \log p_{\theta}(x)$ , where it measures the ambiguity of the data with respect to the given target classifier. The lower entropy loss means the model has the higher confidence in the samples. As Fig. 4 shows, the different colors represent different adaptation methods. The dark

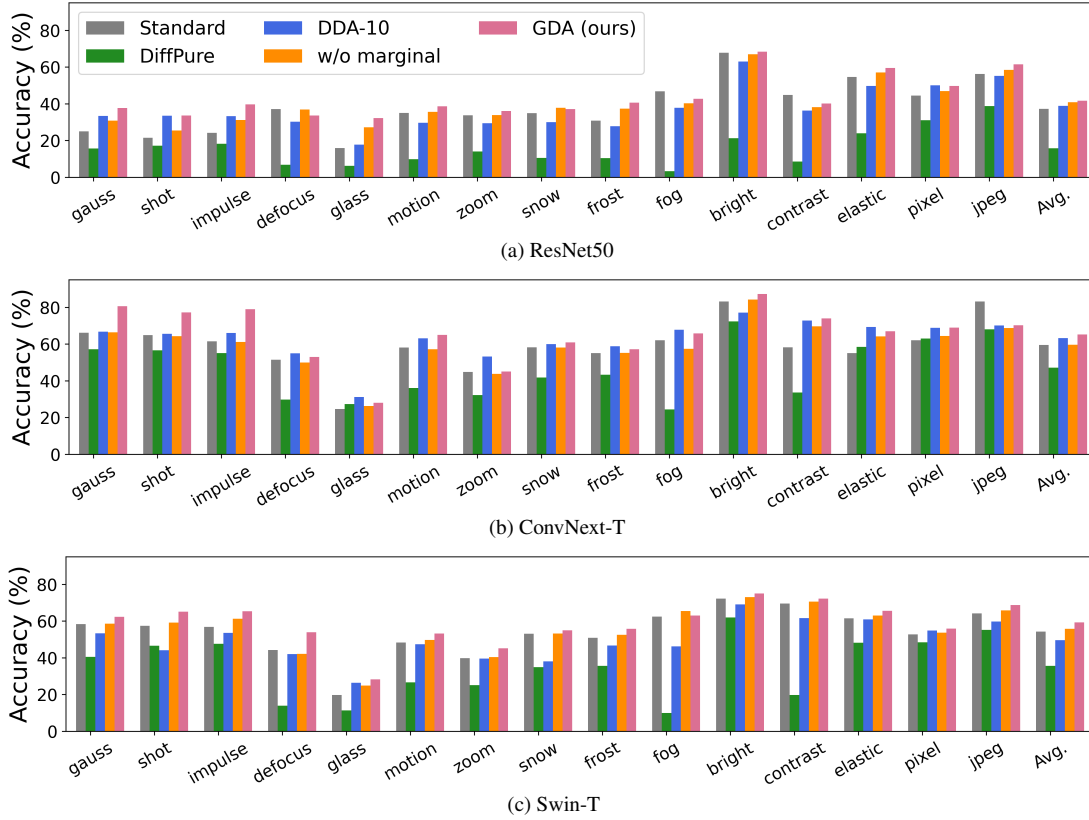


Figure 3. Comparison of the performance for our method with baselines under 15 types of corruption in ImageNet-C for three model architectures, including ResNet50, ConvNext-T, and Swin-T. GDA shows better improvement on all corruption types for ImageNet-C.

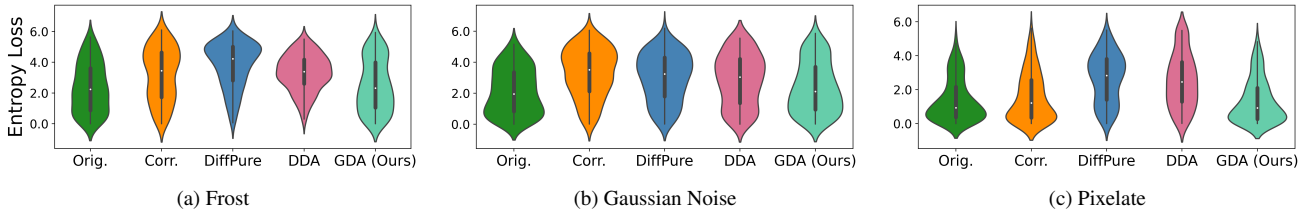


Figure 4. Entropy loss measurement for different corruptions on ImageNet-C. From left to right, the x-axis shows different adaptation methods. The y-axis shows the entropy loss values. The lower value means the model has higher confidence on the sample. In each subfigure, from left to right, we show the loss distribution for original sample (green), corrupted samples (orange), samples adapted by Diffpure [30] (blue), samples adapted by DDA [5] (pink), and samples adapted by our method (light green).

green color represents the original sample and the orange color represents the loss distribution of corrupted samples. We show that the entropy loss distribution has a massive shift between corrupted and original samples, which means the model has lower confidence in most of the corrupted samples than the original samples. We then show the entropy loss of samples after adapting with three diffusion-driven adaptation methods, including Diffpure (blue), DDA (red), GDA (light green). As every subfigure in Fig. 4 shows, for every corruption type, the loss distribution of samples generated from GDA moves toward the entropy loss distribution

of original samples, which means that our method indeed shifts the OOD samples back to the source domain. However, DDA and Diffpure do not have excessive shifting on the entropy loss distribution.

**Sensitivity Analysis on Sampling Steps** In Figure 5, we show the effect of different reverse steps on the performance of the diffusion model. In our experimental results in Section 4, we fix the reverse step number as 10 for every baseline. Here, we compare different reverse sampling steps for DDA and ours from small to large (1 to 50). As Fig. 5 shows, GDA

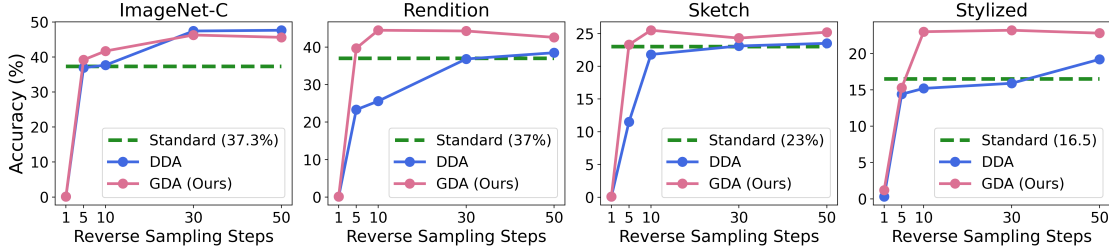


Figure 5. Sensitivity analysis on the reverse sampling steps. We compare our method with DDA under different sampling steps from 1 to 50. We evaluate on the ResNet50 model and show the standard accuracy with green color line.

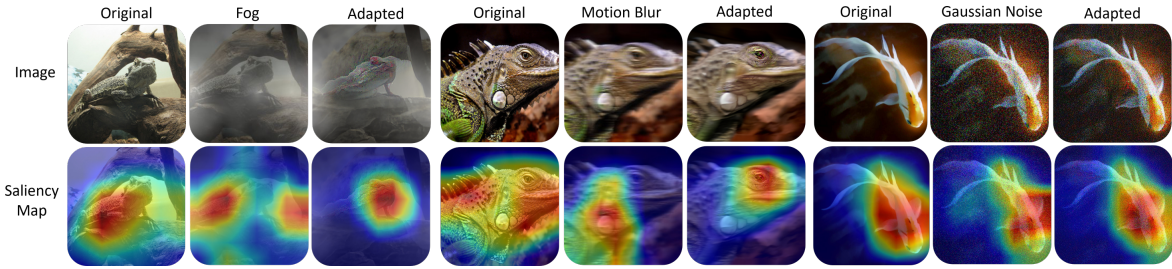


Figure 6. GradCam Visualization on ImageNet-Corruption. For every subfigure, from left to right, we show the original, corrupted, and the samples after using GDA to adapt at the first row. The second row shows their corresponding GradCAM.

has a more significant improvement under a small number of reverse steps (e.g., 10) and is more effective compared to the DDA baseline. When increasing the reverse sampling steps to 50, GDA slightly improves but still outperforms the DDA baseline on every OOD benchmark.

**Analysis on Structural Guidance** To show how our structural guidance can guide the diffusion model, we visualize the samples generated from GDA and their corresponding gradient classification activation maps (GradCAM). In Fig. 6, the corrupted images after adaptation are visually de-corrupted, and the saliency map from GradCAM demonstrates how our objective function can guide the model during the adaptation. In Appendix Fig. 7 and 8, we show the samples from Rendition and Stylized with wrong predictions before adaptation and their corresponding adapted models with correct predictions.

**Adaptation Cost v.s. Robustness** In Table 4, we show the adaptation cost under different adaptation methods, including DDA, Diffpure, without marginal guidance, and GDA. For GDA, the run time depends on the number of augmented samples. Thus, we select the number with the best accuracy (16) for comparison. Compared to DDA and Diffpure, our method outperforms them by  $\sim 7\%$  on ImageNet-Rendition and reduces 3.85x run time.

	Diffpure	$DDA_{10}$	$DDA_{50}$	w/o marg.	GDA (Ours)
Run time	31.7 s	2.1s	13.5 s	2.65 s	<b>3.49 s</b>
Acc. (%)	29.8	24.2	42	39.4	<b>44.5</b>

Table 4. Adaptation run time v.s. Robustness. We show the robust accuracy of Rendition on ResNet50 for every baseline and their corresponding run time for adapting per sample. Compared to DDA and Diffpure, our method outperforms them in smaller run time.

## 6. Conclusion

We propose Generalized Diffusion Adaptation (GDA), a novel approach for robust test-time adaptation on OOD samples. As opposed to existing methods that require adjusting model weights or inputs with additional vectors, GDA utilizes a diffusion model to shift the OOD samples back to the source domain directly. With our proposed structural guidance based on marginal entropy, style, and content preservation losses, GDA achieves a more generalized adaptation. Our evaluation results indicate that GDA offers greater robustness across a variety of OOD benchmarks when compared to other diffusion-driven baselines, achieving the best accuracy gain on multiple OOD benchmarks. Our work offers fresh perspectives on OOD robustness by employing the emerging techniques of diffusion models. For the continued extension of GDA’s applications, future research directions include: (1) adapting GDA for tasks such as object detection; (2) investigating a broader range of structural guidance mechanisms, such as incorporating text prompt guidance for the diffusion model; and (3) examining alternative guidance processes to enhance the efficiency of GDA.



## References

- [1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Visual prompting: Modifying pixel space to adapt pre-trained models. *arXiv preprint arXiv:2203.17274*, 2022. 14
- [2] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models, 2021. 2, 3
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5
- [4] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *Advances in Neural Information Processing Systems*, 32, 2019. 3
- [5] Jin Gao, Jialing Zhang, Xihui Liu, Trevor Darrell, Evan Shelhamer, and Dequan Wang. Back to the source: Diffusion-driven test-time adaptation. *arXiv preprint arXiv:2207.03442*, 2022. 2, 3, 4, 5, 6, 7, 13
- [6] 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 *International Conference on Learning Representations*, 2019. 5
- [7] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004. 2
- [8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3, 5, 13
- [9] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019. 1, 3
- [10] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019. 5
- [11] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. *Advances in Neural Information Processing Systems*, 32, 2019. 1
- [12] Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty, 2020. 6, 11
- [13] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 5
- [14] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8349, 2021. 1, 3, 5
- [15] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. *CVPR*, 2021. 5
- [16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3
- [17] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv:2204.03458*, 2022. 3
- [18] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. *arXiv preprint arXiv:2203.12119*, 2022. 3
- [19] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 3
- [20] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation, 2016. 3
- [21] Gongye Liu, Haoze Sun, Jiayi Li, Fei Yin, and Yujiu Yang. Accelerating diffusion models for inverse problems through shortcut sampling. *arXiv preprint arXiv:2305.16965*, 2023. 3
- [22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 3, 5, 13
- [23] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 3, 5, 13
- [24] Chengzhi Mao, Augustine Cha, Amogh Gupta, Hao Wang, Junfeng Yang, and Carl Vondrick. Generative interventions for causal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3947–3956, 2021. 3
- [25] Chengzhi Mao, Mia Chiquier, Hao Wang, Junfeng Yang, and Carl Vondrick. Adversarial attacks are reversible with natural supervision. *arXiv preprint arXiv:2103.14222*, 2021. 1, 3, 14
- [26] Chengzhi Mao, Lu Jiang, Mostafa Dehghani, Carl Vondrick, Rahul Sukthankar, and Irfan Essa. Discrete representations strengthen vision transformer robustness. *arXiv preprint arXiv:2111.10493*, 2021. 3
- [27] Chengzhi Mao, Kevin Xia, James Wang, Hao Wang, Junfeng Yang, Elias Bareinboim, and Carl Vondrick. Causal transportability for visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7521–7531, 2022. 1, 3
- [28] Claudio Michaelis, Benjamin Mitzkus, Robert Geirhos, Evgenia Rusak, Oliver Bringmann, Alexander S. Ecker, Matthias Bethge, and Wieland Brendel. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019. 5

- [29] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 3
- [30] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *International Conference on Machine Learning (ICML)*, 2022. 3, 5, 6, 7, 13
- [31] Taesung Park, Alexei A Efros, Richard Zhang, and Jun-Yan Zhu. Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*, pages 319–345. Springer, 2020. 5
- [32] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems. In *proceedings of the 26th Symposium on Operating Systems Principles*, pages 1–18, 2017. 1
- [33] Juan C Pérez, Motasem Alfarra, Guillaume Jeanneret, Laura Rueda, Ali Thabet, Bernard Ghanem, and Pablo Arbeláez. Enhancing adversarial robustness via test-time transformation ensembling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 81–91, 2021. 3
- [34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 3
- [35] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pages 5389–5400. PMLR, 2019. 3
- [36] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3
- [37] Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization, 2019. 3, 15
- [38] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3, 4
- [39] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020. 3
- [40] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts, 2019. 1, 3
- [41] Yun-Yun Tsai, Pin-Yu Chen, and Tsung-Yi Ho. Transfer learning without knowing: Reprogramming black-box machine learning models with scarce data and limited resources. In *International Conference on Machine Learning*, pages 9614–9624. PMLR, 2020. 1, 3
- [42] Yun-Yun Tsai, Chengzhi Mao, Yow-Kuan Lin, and Junfeng Yang. Self-supervised convolutional visual prompts. *arXiv preprint arXiv:2303.00198*, 2023. 3, 14
- [43] Hsi-Ai Tsao, Lei Hsiung, Pin-Yu Chen, Sijia Liu, and Tsung-Yi Ho. AutoVP: An automated visual prompting framework and benchmark. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 3
- [44] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization, 2020. 1, 3, 15
- [45] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021. 15
- [46] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019. 5
- [47] Weilun Wang, Jianmin Bao, Wengang Zhou, Dongdong Chen, Dong Chen, Lu Yuan, and Houqiang Li. Semantic image synthesis via diffusion models. *arXiv preprint arXiv:2207.00050*, 2022. 3
- [48] Serin Yang, Hyunmin Hwang, and Jong Chul Ye. Zero-shot contrastive loss for text-guided diffusion image style transfer, 2023. 2, 4, 5, 11
- [49] Rumpeng Yu, Songhua Liu, Xingyi Yang, and Xinchao Wang. Distribution shift inversion for out-of-distribution prediction. *The IEEE / CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2023. 3
- [50] M. Zhang, S. Levine, and C. Finn. MEMO: Test time robustness via adaptation and augmentation. 2021. 1, 2, 3, 15
- [51] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13025–13032, 2020.
- [52] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. *arXiv preprint arXiv:2104.02008*, 2021. 3