Enhance the Visual Representation via Discrete Adversarial Training

Anonymous Author(s)
Affiliation
Address
email

Abstract

Adversarial Training (AT), which is commonly accepted as one of the most effective approaches defending against adversarial examples, can largely harm the standard performance, thus has limited usefulness on industrial-scale production and applications. Surprisingly, this phenomenon is totally opposite in Natural Language Processing (NLP) task, where AT can even benefit for generalization. We notice the merit of AT in NLP tasks could derive from the discrete and symbolic input space. For borrowing the advantage from NLP-style AT, we propose Discrete Adversarial Training (DAT). DAT leverages VQGAN to reform the image data to discrete text-like inputs, i.e. visual words. Then it minimizes the maximal risk on such discrete images with symbolic adversarial perturbations. We further give an explanation from the perspective of distribution to demonstrate the effectiveness of DAT. As a plug-and-play technique for enhancing the visual representation, DAT achieves significant improvement on multiple tasks including image classification, object detection and self-supervised learning. Especially, the model pre-trained with Masked Auto-Encoding (MAE) and fine-tuned by our DAT without extra data can get **31.40** mCE on ImageNet-C and **32.77%** top-1 accuracy on Stylized-ImageNet, building the new state-of-the-art.

1 Introduction

Nowadays, Deep Neural Networks (DNNs) has achieved excellent performance surpassing humans in most computer vision tasks. Although remarkable progress has been made, the success of DNNs is actually a false sense when i.i.d hypothesis is not satisfied in wild. Researchers have shown that deep models fail in most circumstances including adversarial perturbations [1], common corruptions [2], colors or textures changing [3, 4], etc. There is still a long way to make DNNs closer to the robust human perception.

A possible way towards robust machine perception can be Adversarial Training (AT) [5], which automatically finds failure input cases of DNNs and augment online with these cases for fixing "bugs". With online augmentation of adversarial examples, AT greatly enhances the adversarial robustness, and helps for learning perceptually-aligned representations [6] with good interpretability [7, 8] and transferability [9]. However, AT is double-edged, which meanwhile degrades the standard performance caused by problematic regularization [10]. Such problematic regularization makes the decision boundaries over-smoothed and enlarges indecisive regions.

Surprisingly, previous works [11, 12] observe a strange phenomenon that AT behaves conversely in Natural Language Processing (NLP) tasks. By automatically finding adversarial textual inputs,

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

AT will not hurt the accuracy and even benefit for both generalization and robustness of language models. This phenomenon motivates us considering whether the merit of NLP-style AT can be transferred to vision tasks. We notice such merit could derive from the unique data organizing form of language models. To be specific, an adversarial image perturbed in continuous pixel space actually differs with the truly "hard" examples appeared in real world. Contrarily, text space is discrete and symbolic, where adversarial text is practically existing when a typo is made by humans. Learning on such adversarial text will obviously improve the generalization on other more texts with confusing typos. Therefore, we borrow the symbolic nature of languages, and apply it on CV tasks by discretizing continuous images into a more meaningful symbolic space. Afterwards, AT is conducted for minimizing the maximal risk on such text-like inputs with symbolic adversarial perturbations.

In this paper, we propose Discrete Adversarial Training (DAT), a new type of adversarial training which aims to improve both robustness and generalization of vision models. DAT leverages VQ-GAN [13] to learn a vocabulary of visual words, also known as image codebook. For a continuous image input, each encoded patch embedding is replaced with its closest visual word in the codebook, and represented as a corresponding index. Then the image is transformed to a sequence of symbolic indices similar with language input. For generating adversarial examples based on such symbolic sequence, direct use of optimization methods in NLP like combinatorial optimization [14, 15, 16] or synonym substitutions [14, 16] can be challenging. The reason lies in: 1) the large search space of images and 2) the non-existence of synonym in visual codebook. To make it more efficient, DAT adopts a gradient-based method which assumes the backward adversarial gradient goes straight-through the complex discretization process, thus gradients on discretized image can be copied to original input. Then it use one-step search along the direction of estimated gradient such that the discrete representation will altered adversarially during the discretization process, resulting a discrete adversarial examples. Finally the discrete adversarial examples is fed into models for training. Different from AT which always adds l_p bound on augmented adversarial examples, DAT affects the discretization process to produce diverse adversarial inputs beyond l_p bound for training. We show in ablation experiment that DAT not only enhances the robustness on l_p bounded attacks, but also is partly beneficial in defense of unrestricted semantic attacks [17, 18]. The overall pipeline of discrete adversarial training is shown in Figure 1.

We further give an analysis to explain the effectiveness of our DAT from the perspective of distribution. By comparing the distributional difference of training examples in AT and DAT with clean images, we find the discrete adversarial examples in our DAT are much closer to the clean distribution. Such ability of generating "in-distribution" adversarial examples makes DAT can improve the visual representation learning on multiple vision models and tasks, with no sacrificing of clean accuracy.

Our contributions are summarized below:

- To the best of our knowledge, we appear to be the first to transfer the merit of NLP-style adversarial training to vision models, for improving robustness and generalization simultaneously.
- We propose Discrete Adversarial Training (DAT), where images are presented as discrete visual words, and the model is training on example which has the adversarially altered discrete visual representation.
- DAT achieves significant improvement on multiple tasks including image classification, object detection and self-supervised learning. Especially, it establishes a new state-of-the-art for robust image classification. By combining MAE [19] pre-training and DAT fine-tuning, our ViT-Huge [20] achieves 31.40 mCE on ImageNet-C [2] and 32.77% top-1 accuracy on Stylized-ImageNet [3].

2 Related Work

Adversarial Training Adversarial Training (AT) [5] is first proposed to improve robustness by training models with adversarial examples. As one of the most effective defense, existing

works [21, 22] have suggested a trade-off between adversarial and clean accuracy in AT. Despite great efforts [23, 24, 25] have been made for mitigating this trade-off, the bad generalization of AT still cannot be fully remedied till now. In opposite way, some other works [26, 27] use AT to improve the clean accuracy rather than adversarial robustness. The most close work to ours is the AdvProp [26], which splits batch norms to prevent the mixed statistics of clean and adversarial examples, thus learns better adversarial feature for generalization. Pyramid AT [28] makes AT specific to ViTs by crafting pyramid perturbation with Dropout enabling, yielding imporved performance. However, these methods are only applicable under specific models or tasks. VILLA [29] is also a representation enhancement technique using AT. But contrary to us, it applies vision-style AT only for vision-and-language representation learning. AGAT [30] is another kind of AT beyond pixel space, which perturbs images along attributes, however it has strict requirement of attributes annotation.

Adversarial Augmentation By borrowing the idea of AT, some previous works [31, 32, 33] propose to search augmentations adversarially for improving the hardness of training examples. Adversarial AutoAugment [31] uses augmentation policy network to produce hard augmentation policies on a pre-defined policy space. AugMax [32] mixes multiple randomly sampled augmentation operators like AugMix [34], by using adversarially learned mixing factors. AdA [33] optimizes the parameters of image-to-image models to generate adversarially corrupted augmented images. MaxUp [35] uses the worst augmented data of each data point in a set of random perturbations or transforms for training. However, these methods create adversarial inputs indirectly and rely on pre-defined augmentations or translation models. They cannot perturb the images locally. Instead, our DAT directly modifies the image with no need of pre-defined transforms, and can craft local perturbations on images, which is more elaborate.

Discrete Visual Representation Learning Early technique [36, 37] of Bag-Of-Visual-Words (BOVW) model has shown great power of discrete representation in visual understanding. VQ-VAE [38] uses DNNs to learn neural discrete representations, also known as visual codebook, by generative modeling the image distribution. Recently, the idea of discrete representations learning has been widely emerged in many vision tasks. In most Masked Image Modeling (MIM) methods [39, 40], visual codebook is needed for BERT-like self-supervised pretraining. For image classification, discrete representations strengthen the robustness by preserving the global structure of an object and ignoring local details [41]. For image synthesis, adversarial and perceptual objective can be added to VQ-VAE for learning perceptually-aligned visual codebook [13]. With growing power of generative models, now a well trained VQGAN can produce vivid images with 0.58 FID [42]. In this work, we use VQGAN for the discretization process in our DAT. As a powerful generative model, VQGAN can greatly reduce the information loss in reconstruction process and ensure the high-quality of the generated discrete adversarial examples.

3 Method

3.1 Traditional Adversarial Training

We take the typical image classification task as an example to show the formulation of Adversarial Training (AT). Consider the training image and label set $\mathcal{D} = \{x_i, y_i\}_{i=1}^n$, and a classifier F with learnable parameters θ, the classification objective is always a cross-entropy loss $\mathcal{L}(x, y, \theta)$. Adversarial Training (AT) finds the optimal θ by solving a minimax optimization problem:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta} \mathcal{L}(x+\delta, y, \theta) \right] \quad s.t. \|\delta\|_p < \epsilon, \tag{1}$$

where the inner optimization finds the perturbations δ on per-pixel values for maximizing the loss, and the outer minimization update θ to improve the worst-case performance of the network w.r.t. the perturbation. $\|\cdot\|_p$ constraints the p-norm of δ to a small value ϵ. A problem is that AT finds the failure case i.e. $x+\delta$ in continuous pixel space. However, human does not create or recognize images from complex pixel values, but from discrete semantic concepts. Although adversarial examples can successfully fool the models, they are still different from the real "hard" examples appeared in practice.

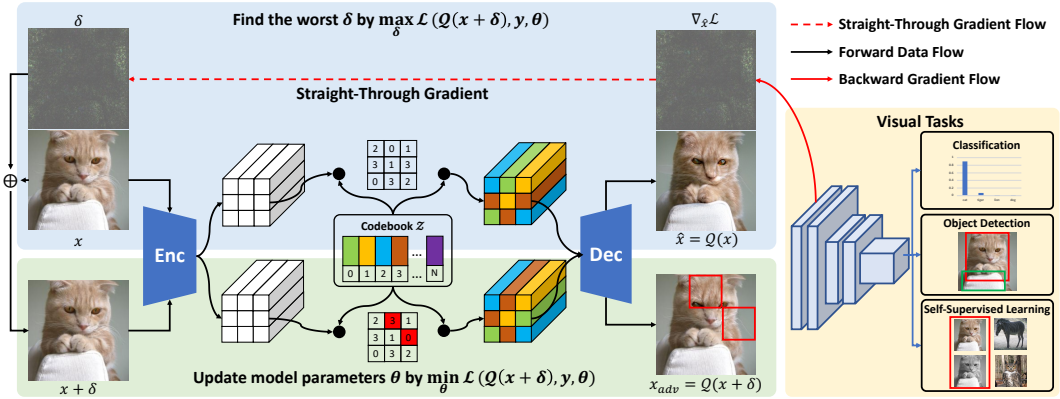


Figure 1: The overall pipeline of Discrete Adversarial Training (DAT).

3.2 Discrete Adversarial Training

3.2.1 Image Discretization by Visual Codebook

For discrete adversarial training, it is desirable first to learn an expressive visual codebook and represent the training image set in discrete space. We utilize VQGAN [13] for image discretization. More precisely, consider a continuous image $x \in \mathbb{R}^{H \times W \times 3}$, VQGAN learns an encoder $\mathsf{Enc}_\phi(\cdot)$, decoder $\mathsf{Dec}_\psi(\cdot)$ and quantization $\mathsf{q}_\mathcal{Z}(\cdot)$. Enc_ϕ is a convolutional model which maps x to intermediate latent vectors $v = \mathsf{Enc}_\phi(x) \in \mathbb{R}^{(h \times w) \times d}$, where h, w is the height, width of the intermediate feature map and d is the latent dimension. Subsequently, $\mathsf{q}_\mathcal{Z}(\cdot)$ learn a codebook $\mathcal{Z} = \{z_k | z_k \in \mathbb{R}^d\}_{k=1}^K$, such that each latent vector $v_{ij} \in \mathbb{R}^d$ can be quantized onto its closest codebook entry z_k as:

$$v_\mathsf{q} = \mathsf{q}_\mathcal{Z}(v) := \left(\underset{z_k \in \mathcal{Z}}{\arg\min} \|v_{ij} - z_k\| \right) \in \mathbb{R}^{h \times w \times d}, \tag{2}$$

where i, j present each location in feature map. Then the decoder $\mathsf{Dec}_\psi(\cdot)$ outputs the reconstruction image $\hat{x}$ from the quantized vectors v_q by:

$$\hat{x} = \mathsf{Dec}_\psi(v_\mathsf{q}). \tag{3}$$

VQGAN is trained by minimizing the reconstructed difference between $\hat{x}$ and x. More details can be referred to Appendix A. So far, given a continuous image x, we can get its corresponding discrete reconstruction $\hat{x}$. For simplicity, we use $\mathcal{Q}$ to stand for the above image discretization process, and then we have $\hat{x} = \mathcal{Q}(x)$.

3.2.2 Discrete Adversarial Training

Based on the definition in Sec 3.2.1, we can generate discrete adversarial examples in inner maximization step of AT. By slightly modifying the Eq 1, the objective of DAT is formulated as:

$$\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[\max_\delta \mathcal{L}(\mathcal{Q}(x+\delta), y, \theta) \right], \tag{4}$$

where $\mathcal{Q}$ transforms the continuous pixel space to discrete input space. We delete the constraint term since there is no need to bound the per-pixel values of δ. Suppose that $\mathcal{Q}$ is an ideal discretizer with no information loss in discretization process. The problem lies how to find the worst δ for maximizing the classification loss. Similar with traditional AT, we can use gradient-based methods to approximate δ by:

$$\delta \simeq \alpha \nabla_x \mathcal{L}(\mathcal{Q}(x), y, \theta), \tag{5}$$

where α determines the magnitude of the perturbations along the gradient direction. We set $\alpha = 0.1$ by default in DAT. To expand $\nabla_x \mathcal{L}(\mathcal{Q}(x), y, \theta)$ by chain rule, we have four partial derivative terms as follows:

$$\nabla_x \mathcal{L}(\mathcal{Q}(x), y, \theta) = \frac{\partial \mathcal{L}}{\partial \hat{x}} \cdot \frac{\partial \hat{x}}{\partial v_\mathsf{q}} \cdot \frac{\partial v_\mathsf{q}}{\partial v} \cdot \frac{\partial v}{\partial x} \tag{6}$$

Algorithm 1: Pseudo code of DAT

Input: Classifier F; Pre-trained discretizer $\mathcal{Q}$; A sampled mini-batch of clean images x with labels y; attack magnitude α.

Output: Learned network parameter θ of F

1: Fix the network parameters of $\mathcal{Q}$
2: **for** each training steps **do**
3: $\hat{x} \leftarrow \mathcal{Q}(x)$ //Get the discrete reconstruction $\hat{x}$
4: $\delta \leftarrow \alpha \nabla_{\hat{x}} \mathcal{L}(\hat{x}, y, \theta)$ //Estimate the adversarial perturbations
5: $x_{adv} \leftarrow \mathcal{Q}(x + \delta)$ //Generate discrete adversarial examples
6: Minimize the classification loss w.r.t. network parameter $\arg\min_\theta \mathcal{L}(x_{adv}, y, \theta)$
7: **end for**

Through analysing the feasibility of each term, we find only $\frac{\partial v_{\mathsf{q}}}{\partial v}$ is hard to solve as the non-differentiable nature of Eq 2. Fortunately, as proposed in previous work, a straight-through gradient estimator [43, 44] can be used by copying the gradients from v_{q} to v. By replacing $\frac{\partial \hat{x}}{\partial v_{\mathsf{q}}} \cdot \frac{\partial v_{\mathsf{q}}}{\partial v}$ with $\frac{\partial \hat{x}}{\partial v}$, we can simplify the Eq 6 to $\nabla_x \mathcal{L}(\mathcal{Q}(x), y, \theta) = \frac{\partial \mathcal{L}}{\partial \hat{x}} \cdot \frac{\partial \hat{x}}{\partial v} \cdot \frac{\partial v}{\partial x}$, which has derivative everywhere. Although the solution seems workable theoretically, the huge cost makes it impractical on large-scale vision tasks. The bottleneck mainly lies on that $\frac{\partial \hat{x}}{\partial v}$ and $\frac{\partial v}{\partial x}$ require the adversarial gradients backward through Enc and Dec. Actually, a generator capable of producing high-quality images always has a large amount of parameters. Compared with original adversarial training which only needs F for gradient calculation, it requires more than tripled GPU memory and computation cost.

To solve this problem, we propose an efficient alternative solution. Since $\hat{x} \simeq x$ established for an ideal discretizer $\mathcal{Q}$, we can also use a straight-through estimator between $\hat{x}$ and x, which is given by

$$\nabla_x \mathcal{L}(\mathcal{Q}(x), y, \theta) = \frac{\partial \mathcal{L}}{\partial \hat{x}} \cdot \frac{\partial \hat{x}}{\partial x} \simeq \frac{\partial \mathcal{L}}{\partial \hat{x}}. \quad (7)$$

Finally, we can solve the worst δ by $\nabla_{\hat{x}} \mathcal{L}(\hat{x}, y, \theta)$. By this way, the computation cost of DAT has been largely reduced. Compared with original adversarial training, it only has extra computation cost on VQGAN forward, which is relatively controllable.

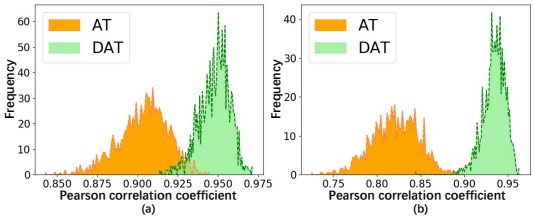


Figure 2: The frequency histogram of the Pearson correlation coefficient (PCC) between BN statistics on clean and adversarial images. Larger PCC value means smaller distributional difference with clean images. (a), (b) present the difference on mean and variance statistics respectively.

For clarity, let us restate the pipeline of our DAT. For each training image x, DAT first use VQGAN to get discrete reconstruction $\hat{x}$. By feeding $\hat{x}$ to classifier F, a worst-case perturbation δ can be estimated by computing the gradient of $\hat{x}$ towards maximizing the classification loss. The perturbed image thus can be created by adding δ on original x. Finally, $x + \delta$ is discretized by VQGAN again and acts as the adversarial input, on which F is trained by minimizing the classificaiton loss. The details of our DAT is summarized in Algorithm 1.

Explaining the Effectiveness of DAT from the Perspective of Distribution We give an empirical analysis to explain why DAT can improve the robustness and generalization without sacrificing clean accuracy. Previous work [26] has pointed out that the underlying distributions of adversarial examples are different from clean images. Training on both clean and adversarial images will force the Batch Normalization (BN) [45] to estimate an inaccurate mixture statistics of feature distribution, and thus impact the standard performance. We study this effect by sampling 1000 mini-batches in ImageNet validation set, and generate corresponding adversarial images for AT and our DAT. For each batch we calculate the mean and variance statistics of last BN of ResNet50. Then the Pearson correlation coefficient (PCC) between the statistics on clean and adversarial examples is computed for showing the distributional difference. Figure 2 shows the frequency histogram of the distributional difference on 1000 mini-batches. For training samples of DAT, the peak of the histogram is at 0.95, which is greater than AT. It suggests that DAT generates discrete adversarial examples much closer with the

clean distributions. Therefore, training on these examples will reduce the shift of clean distribution in AT, yielding both the robustness and generalization improvement.

4 Experiments

To demonstrate the versatility of our method, we experiment Discrete Adversarial Training (DAT) on multiple tasks including image classification, object detection and self-supervised learning.

4.1 Image Classification

Implementation We implement DAT on two representative architectures: ResNet50 [46] and ViTs [20]. For ResNet50, we first experiment DAT with vanilla training recipes using "robustness" library [1]. Then we combine DAT with other orthogonal robust training techniques: DeepAugment [47] and AugMix [34]. Only cross entropy loss is used for generating discrete adversarial examples. The JSD loss in AugMix is optimized merely on clean samples. For ViTs, we adopt ViT-B/16 as baseline models, which is trained by the recipes in AugReg [48]. Besides, we use DAT to conduct supervised finetuning on downstream ImageNet classification task based on a self-supervised ViT-Huge pretrained by MAE [19]. By default, we refer ViT to ViT-B/16 in all tables and figures.

Benchmarks The trained model is evaluated in three aspects: 1) in-distribution performance on ImageNet-Validation set; 2) adversarial robustness on white-box FGSM [49] and transfer-based black-box attack dataset DamageNet [50]; 3) out-of-distribution robustness on ImageNet(IN)-A, IN-C, IN-V2, IN-R, IN-Sketch and Stylized IN. Each of them represents a type of out-of-distribution scenario where the classifier is prone to make mistakes. IN-A [51] places the ImageNet objects in hard contexts; IN-C [52] applies a series of noise, blur, digital and weather corruptions; IN-R [47] collects online images with artificial creation, e.g., cartoons, graphics, video game renditions, etc; IN-Sketch [53] contains images described by sketches; Stylized IN [3] destroys the texture but maintains the shape feature by conducting style transfer on ImageNet images. Except for IN-C which is measured by mCE, we report the top-1 accuracy on all above datasets.

Results We report all results in Table 1. For fair comparison, we add DAT on base methods without modifying the original training hyper-parameters, such that improvement is entirely attributed to the DAT. For ResNet50, DAT achieves significant improvement on both clean accuracy, adversarial and out-of-distribution robustness. The improvement seems greater when combining DAT with DeepAugment and AugMix. For ViTs, we find DAT is compatible with other complex augmentations such as MixUp [54], CutMix [55] or RandAugment [56] used in AugReg, yielding greater improvement. Compared with plain ViT, AugReg-ViT and ViT with discrete representation called DrViT [41], DAT with AugReg achieves better performance. The best result is from ViT-Huge pretrained by MAE [19] and finetuned by DAT, which suggests DAT is also effective in downstream fine-tuning tasks.

Additionally, we also compare DAT with other robust training strategies in Table 2. AugReg-ViT is adopted as the baseline model. Most strategies, e.g., AdvProp [26] and Debiased [57] are proposed for ResNet with auxiliary BatchNorm. We show these methods cannot work properly on ViTs with only LayerNorm. Compared with Pyramid AT [28], our DAT has lower clean accuracy but yields stronger robustness. More results of strategies comparison on ResNet50 refer to Appendix B.

4.2 Self-Supervised Learning

Implementation We experiment DAT on three self-supervised methods: SimCLR [58], SimSiam [59] and recently proposed MoCov3 [60]. The discrete adversarial training is only conducted during pre-training stage, and the learned representation is evaluated on downstream tasks by standard pipeline [61]. We craft adversarial examples based on RoCL [62], which attacks the pre-training objective by maximizing the contrastive loss. For preventing the cost explosion, we pre-train 200

[1] https://github.com/MadryLab/robustness

Methods	ImageNet	Adversarial Robustness		Out of Distribution Robustness					
		FGSM	DamageNet	A	C↓	V2	R	Sketch	Stylized
ResNet50 [46]	76.13	12.19	5.94	0.0	76.70	63.20	36.17	24.09	7.38
+ DAT (Ours)	**76.52**	**30.66**	**14.42**	**4.38**	**74.16**	**65.02**	**41.90**	**27.27**	**10.8**
DeepAugment [47]	76.66	21.61	11.94	3.46	60.37	**65.24**	42.17	29.50	14.68
+ Augmix [34]	75.82	27.05	19.60	3.86	53.55	63.63	46.77	32.62	21.23
+ DAT (Ours)	**77.10**	**35.32**	**22.86**	**6.86**	**50.82**	65.14	**47.88**	**34.98**	**21.89**
ViT [20]	72.00	23.30	28.99	6.44	77.61	57.34	25.69	15.56	5.82
DrViT [41]	79.48	45.76	44.91	17.20	46.22	68.05	44.77	34.59	19.38
AugReg-ViT [48]	79.91	44.32	45.24	19.03	54.50	67.90	39.46	29.16	16.62
+ DAT (Ours)	**81.46**	**51.82**	**45.70**	**30.15**	**44.65**	**70.83**	**47.34**	**34.77**	**23.13**
MAE-H [19]	86.90	60.16	64.36	68.18	33.92	78.47	64.12	49.08	26.36
+ DAT (Ours)	**87.02**	**63.77**	**70.42**	**68.92**	**31.40**	**78.82**	**65.61**	**50.03**	**32.77**

Table 1: The results of DAT on image classification. Bold number indicates the better performance.

Training Strategies	ImageNet	Adversarial Robustness		Out of Distribution Robustness					
		FGSM	DamageNet	A	C↓	V2	R	Sketch	Stylized
Normal [48]	79.91	44.32	45.24	19.03	54.50	67.90	39.46	29.16	16.62
Advprop [26]	79.54	**72.38**	45.48	18.53	51.46	68.74	43.51	31.68	19.24
Fast Advprop [27]	79.02	70.52	44.87	17.86	53.31	67.09	41.84	29.42	18.39
Pyramid AT [28]	**81.68**	50.36	45.53	23.18	44.95	70.32	47.30	**36.87**	20.02
Debiased [57]	79.33	46.85	44.99	18.32	49.82	67.55	40.32	29.43	22.37
DAT (Ours)	81.46	51.82	**45.70**	**30.15**	**44.65**	**70.83**	**47.34**	34.77	**23.13**

Table 2: Comparison of DAT with other training strategies. We use AugReg-ViT as the base model.

epochs for SimCLR, and 100 epochs for both SimSiam and MoCov3. We use the implementations on Github[2], and the only modified argument is batch size, which is set as 1024, 512, 2048 for SimCLR, SimSiam, MoCov3 respectively.

Benchmarks For self-supervised learning, we adopt ImageNet-1K for both training and in-distribution testing. Beyond that, in order to give more comprehensive assessment, we build three downstream tasks to metric the transferability of the learned visual representation. Specifically, the linear evaluation reports the top-1 accuracy on five classification datasets: Flowers, CIFAR10, Caltech101, Cars and DTD. For downstream evaluation of object detection, we present mAP, AP50, AP75 on Pascal VOC2007 [63]. ADE20K [64] is used for semantic segmentation task, and both the mean intersection over union (mIoU) and accuracy are reported.

Results As shown in Table 3, DAT can enhance the learned representation on ImageNet and get 0.97%, 0.58% and 0.25% improvement on MoCov3, SimCLR, SimSiam respectively. For excluding that DAT is not just over-fitting on ImageNet, we transfer the representations on downstream recognition, object detection, semantic segmentation tasks. The results suggest that DAT enhances the self-supervised representations with better transferability. In particular, MoCov3 with DAT achieves significant improvement in all downstream tasks. This conclusion echoed with previous works [9], which find adversarially robust models often perform better on transfer learning.

4.3 Object Detection

Implementation We implement DAT on two popular detectors: EfficientDet [65] and YOLOv3 [66]. Object detection generally has two sub-tasks: classification and localization. Det-AdvProp [67] proposes to select more vulnerable sub-task to generate adversarial images, and use auxiliary BN for training. Instead, DAT regards the two sub-tasks as a whole, and attacks the overall detection loss to produce adversarial images. Besides, DAT does not modify BN in detectors. As the memory and computation cost of VQGAN increased quadratic with the input resolution, DAT become unaffordable when input size is large. Therefore, we only experiment DAT on lightweight version of EfficientDet and YOLOv3 with input size smaller than 512.

Benchmarks We train the detectors using the COCO 2017 object detection dataset [68] and evaluate them on COCO's validation set. COCO-C [69] is used for testing the robustness to natural corruptions.

[2] we use the implementation of MoCov3, SimSiam and SimCLR respectively.

Method	ImageNet	Linear Evaluation					VOC Object Detection			ADE20K	
		Flowers	CIFAR10	Caltech101	Cars	DTD	mAP	AP50	AP75	mIoU	Acc.
MoCov3	68.63	91.54	93.40	90.38	49.01	73.03	50.42	80.53	53.93	0.3508	75.64
+ DAT	**69.60**	**93.15**	**95.16**	**91.42**	**53.09**	**73.55**	**51.92**	**80.97**	**56.04**	**0.3585**	**76.33**
SimCLR	64.89	89.28	88.47	83.20	38.84	**73.14**	48.50	78.75	51.35	0.3396	**75.61**
+ DAT	**65.47**	**90.14**	**89.97**	**85.09**	**39.42**	72.93	**48.83**	**79.27**	**51.95**	**0.3412**	75.46
SimSiam	68.16	**87.67**	89.45	85.94	34.95	71.70	48.92	77.22	52.69	**0.2212**	68.57
+ DAT	**68.41**	86.93	**91.70**	**87.03**	**35.10**	**73.03**	**51.73**	**79.69**	**55.74**	0.2203	**68.92**

Table 3: The results of DAT on self-supervised learning.

Models	Datasets	f	K	d	FID	ImageNet	FGSM	A	C↓	R	Stylized
VQGAN [42]	OpenImages [70]	8	16384	4	1.14	76.52	30.66	4.38	74.16	41.90	10.80
VQGAN [42]	OpenImages [70]	16	16384	8	5.15	75.65	30.48	3.17	74.71	40.64	10.76
VQGAN [42]	OpenImages [70]	8	256	4	1.49	76.52	30.46	3.11	74.06	40.97	10.01
VQGAN [13]	ImageNet [71]	16	16384	256	4.98	75.88	30.28	2.65	75.38	40.66	9.58
DALL-E [72]	Private	8	8192	128	32.01	75.23	30.17	3.97	74.26	40.13	10.65

Table 5: Results of DAT based on different pretrained discretizer $\mathcal{Q}$. f presents the downsampling factors. We use ResNet50 as base model and the subset of benchmarks in Sec 4.1 for evaluation.

We report mAP on COCO and COCO-C as the clean and robust performance respectively. rPC [69] is used to to measure relative performance degradation under corruption.

Results We compare the detectors with DAT, Det-AdvProp [67] and vanilla training in Table 4. Although Det-AdvProp has been shown effective on EfficientDet with size larger than D0, we find on smaller detectors, the promotion of Det-AdvProp is subtle. On EfficientDet-Lite0 and YOLOv3-320, it even gets worse on COCO mAP than vanilla training. While our DAT achieves better result on both clean and corrupted input. EfficientDet-Lite0 with DAT get 27.3 and 17.89 mAP on COCO and COCO-C, resulting in 65.53% Relative rPC. By comparison, vanilla YOLOv3 models have lower Relative rPC, showing it is more vulnerable than EfficientDet at same clean mAP. After equipped with DAT, YOLOv3-320 can get 20.55 mAP on COCO-C, leading to 7.75% improvement on rPC.

Models	Training Strategy	COCO mAP	COCO-C mAP	Relative rPC (%)
EffDet-Lite0-320 [65]	Normal	26.41	16.11	61.00
	Det-Advprop	26.34	16.38	62.19
	DAT (Ours)	**27.32**	**17.89**	**65.48**
EffDet-Lite1-384 [65]	Normal	31.50	19.43	61.68
	Det-Advprop	31.82	20.21	63.51
	DAT (Ours)	**32.31**	**21.32**	**65.99**
YOLOv3-320 [66]	Normal	35.91	18.39	51.21
	Det-Advprop	35.73	19.34	54.13
	DAT (Ours)	**36.02**	**20.55**	**57.05**
YOLOv3-416 [66]	Normal	40.30	21.19	52.58
	Det-Advprop	**40.69**	22.55	55.42
	DAT (Ours)	40.41	**23.38**	**57.86**

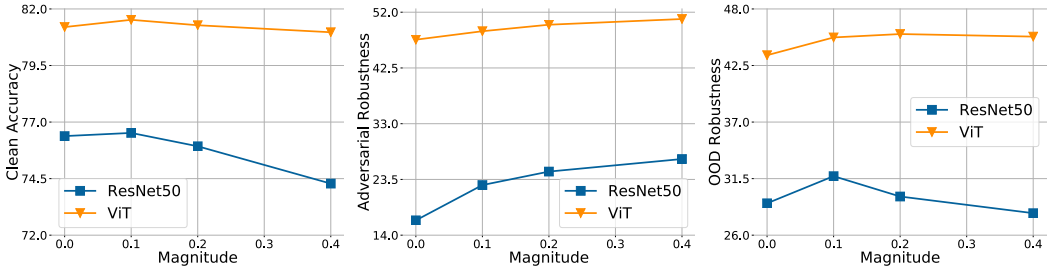
Table 4: The results of DAT on object detection.

4.4 Ablations

Impact of the Codebook in DAT A general understanding is that the codebook with larger K should have stronger representation power. As suggested in Table 5, when the K is reduced from 16384 to 256, the FID of generated images increases 0.35. We show DAT on larger codebook size can achieve better generalization. However, the clean accuracy seems no improvement with increasing of K. We think the reason is that the reconstruction quality has already met the demand of DAT. And the improvement on clean accuracy is saturated when K further increases.

Different Types of Discretizer $\mathcal{Q}$ Our work is based on the hypothesis of an ideal $\mathcal{Q}$ with no loss in discretization process. However, such perfect $\mathcal{Q}$ is not existing in practice. So we study if and how the ability of $\mathcal{Q}$ affects the performance of our DAT in Table 5. The results show DAT performs better by using discretizer with higher FID. We find clean accuracy declines 1.3% on DALL-E, showing it is important for choosing a powerful discretizer. Types of pre-training datasets have little impact. We find the discretizer pre-trained on OpenImages [70] and transferred to discretize ImageNet images, can also achieve comparable results.

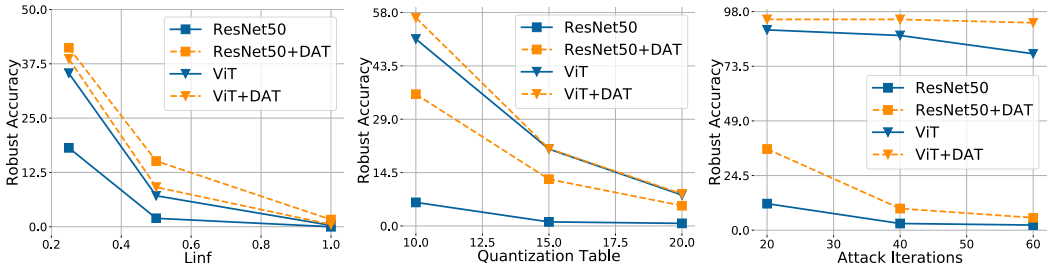
The Performance on Different Magnitude α We present the DAT results on ResNet50 and ViT with different magnitude α in Figure 3. $\alpha = 0$ means the model is only trained on images augmented by VQGAN with no adversarial training process. We show DAT with $\alpha = 0$ makes the models have high clean accuracy but the lowest robustness. With the increase of α, there is still a robustness and accuracy trade-off. DAT with $\alpha = 0.4$ on ResNet50 achieves the best adversarial robustness but generalization is getting worse. By contrast, we surprisingly find ViT has the lower sensibility on α. Even if ViT is trained on DAT with $\alpha = 0.4$, there is still no great drop of clean and OOD accuracy. We suspect that ViT is more suitable for training on discrete examples, and the strong modeling ability makes ViT greater than ResNet50.



(a) Clean Accuracy (b) Adversarial Robustness (c) OOD Robustness

Figure 3: The performance of DAT with different magnitude α in Eq 5.

Results on Stronger Attacks In the main experiment, only one-step FGSM is used for examining the robustness under white-box adversarial attacks. To give more comprehensive evaluation, we additionally test our DAT under stronger AutoAttack [73], and two unrestricted attacks named Adv-Drop [17] and PerC-Adversarial [18]. The result is shown in Figure 4. DAT brings the improvement of robustness under all three attackers. ViT trained with DAT achieved extremely high robust accuracy on PerC-Adversarial, which suggests DAT also effects well in defense against unrestricted adversarial attacks. More details can be referred to Appendix C.



(a) AutoAttack [73] (b) AdvDrop [17] (c) PerC-Adversarial [18]

Figure 4: The adversarial robustness test under other stronger attackers.

5 Limitations and Conclusions

In this paper, we find transferring NLP-style adversarial training to vision models can enhance the learned visual representation effectively. We propose Discrete Adversarial Training (DAT), where images are presented as discrete visual words by VQGAN, and the model is training on examples which have the adversarially altered discrete visual representation. DAT needs not to modify the model architecture and works for both CNNs and ViTs across multiple tasks. DAT reports the state-of-the-art robustness on ImageNet-C and Stylized-ImageNet, exhibiting strong generalization. However, DAT still costs increased training time, this limitation also holds for any adversarial training. The strict assumption of an ideal discretizer is another potential limitation, despite DAT has used powerful VQGAN model to approach this assumption. Overall, this work is an empirical exploration of visual adversarial training in discrete space. All the above limitations are remained as the future optimization direction, and we will further seek for more deeper theoretical explanation of the effectiveness of discrete representation in the next work.

References

[1] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, D. Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2019.

[2] Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

[3] 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*, 2018.

[4] Ali Shahin Shamsabadi, Ricardo Sanchez-Matilla, and Andrea Cavallaro. Colorfool: Semantic adversarial colorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1151–1160, 2020.

[5] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[6] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Learning perceptually-aligned representations via adversarial robustness. *arXiv preprint arXiv:1906.00945*, 2(3):5, 2019.

[7] Andrew Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[8] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32, 2019.

[9] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? *Advances in Neural Information Processing Systems*, 33:3533–3545, 2020.

[10] Yuxin Wen, Shuai Li, and Kui Jia. Towards understanding the regularization of adversarial robustness on neural networks. In *International Conference on Machine Learning*, pages 10225–10235. PMLR, 2020.

[11] Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. Freelb: Enhanced adversarial training for natural language understanding. In *International Conference on Learning Representations*, 2019.

[12] Maor Ivgi and Jonathan Berant. Achieving model robustness through discrete adversarial training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1529–1544, 2021.

[13] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12873–12883, 2021.

[14] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8018–8025, 2020.

[15] Yuan Zang, Fanchao Qi, Chenghao Yang, Zhiyuan Liu, Meng Zhang, Qun Liu, and Maosong Sun. Word-level textual adversarial attacking as combinatorial optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6066–6080, 2020.

[16] Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, 2018.

[17] Ranjie Duan, Yuefeng Chen, Dantong Niu, Yun Yang, A Kai Qin, and Yuan He. Advdrop: Adversarial attack to dnns by dropping information. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7506–7515, 2021.

[18] Zhengyu Zhao, Zhuoran Liu, and Martha Larson. Towards large yet imperceptible adversarial image perturbations with perceptual color distance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1039–1048, 2020.

[19] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2022.

[20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[21] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations*, 2018.

[22] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.

[23] Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. In *International Conference on Machine Learning*, pages 7909–7919. PMLR, 2020.

[24] Rahul Rade and Seyed-Mohsen Moosavi-Dezfooli. Reducing excessive margin to achieve a better accuracy vs. robustness trade-off. In *International Conference on Learning Representations*, 2021.

[25] Alex Lamb, Vikas Verma, Juho Kannala, and Yoshua Bengio. Interpolated adversarial training: Achieving robust neural networks without sacrificing too much accuracy. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pages 95–103, 2019.

[26] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 819–828, 2020.

[27] Jieru Mei, Yucheng Han, Yutong Bai, Yixiao Zhang, Yingwei Li, Xianhang Li, Alan Yuille, and Cihang Xie. Fast advprop. In *International Conference on Learning Representations*, 2021.

[28] Charles Herrmann, Kyle Sargent, Lu Jiang, Ramin Zabih, Huiwen Chang, Ce Liu, Dilip Krishnan, and Deqing Sun. Pyramid adversarial training improves vit performance. *arXiv preprint arXiv:2111.15121*, 2021.

[29] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. In *NeurIPS*, 2020.

[30] Tejas Gokhale, Rushil Anirudh, Bhavya Kailkhura, Jayaraman J Thiagarajan, Chitta Baral, and Yezhou Yang. Attribute-guided adversarial training for robustness to natural perturbations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7574–7582, 2021.

[31] Xinyu Zhang, Qiang Wang, Jian Zhang, and Zhao Zhong. Adversarial autoaugment. In *International Conference on Learning Representations*, 2019.

[32] Haotao Wang, Chaowei Xiao, Jean Kossaifi, Zhiding Yu, Anima Anandkumar, and Zhangyang Wang. Augmax: Adversarial composition of random augmentations for robust training. *Advances in Neural Information Processing Systems*, 34, 2021.

[33] Dan Andrei Calian, Florian Stimberg, Olivia Wiles, Sylvestre-Alvise Rebuffi, András György, Timothy A Mann, and Sven Gowal. Defending against image corruptions through adversarial augmentations. In *International Conference on Learning Representations*, 2021.

[34] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. In *International Conference on Learning Representations*, 2019.

[35] Chengyue Gong, Tongzheng Ren, Mao Ye, and Qiang Liu. Maxup: Lightweight adversarial training with data augmentation improves neural network training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2474–2483, 2021.

[36] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Computer Vision, IEEE International Conference on*, volume 3, pages 1470–1470. IEEE Computer Society, 2003.

[37] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004.

[38] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[39] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. In *International Conference on Learning Representations*, 2022.

[40] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

[41] Chengzhi Mao, Lu Jiang, Mostafa Dehghani, Carl Vondrick, Rahul Sukthankar, and Irfan Essa. Discrete representations strengthen vision transformer robustness. In *International Conference on Learning Representations*, 2022.

[42] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *arXiv preprint arXiv:2112.10752*, 2021.

[43] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[44] Penghang Yin, Jiancheng Lyu, Shuai Zhang, Stanley Osher, Yingyong Qi, and Jack Xin. Understanding straight-through estimator in training activation quantized neural nets. *arXiv preprint arXiv:1903.05662*, 2019.

[45] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[46] 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.

[47] 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.

[48] Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your vit? data, augmentation, and regularization in vision transformers. *arXiv preprint arXiv:2106.10270*, 2021.

[49] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[50] Sizhe Chen, Zhengbao He, Chengjin Sun, Jie Yang, and Xiaolin Huang. Universal adversarial attack on attention and the resulting dataset damagenet. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[51] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15262–15271, 2021.

[52] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2018.

[53] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. *Advances in Neural Information Processing Systems*, 32, 2019.

[54] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[55] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.

[56] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[57] Yingwei Li, Qihang Yu, Mingxing Tan, Jieru Mei, Peng Tang, Wei Shen, Alan Yuille, et al. Shape-texture debiased neural network training. In *International Conference on Learning Representations*, 2020.

[58] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[59] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[60] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021.

[61] Linus Ericsson, Henry Gouk, and Timothy M Hospedales. How well do self-supervised models transfer? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5414–5423, 2021.

[62] Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial self-supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:2983–2994, 2020.

[63] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[64] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

[65] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020.

[66] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[67] Xiangning Chen, Cihang Xie, Mingxing Tan, Li Zhang, Cho-Jui Hsieh, and Boqing Gong. Robust and accurate object detection via adversarial learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16622–16631, 2021.

[68] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[69] 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.

[70] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, 2020.

[71] 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.

[72] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[73] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.

[74] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [No] There are no negative societal impacts needed to discuss.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] The experiment is expensive and need long time, so we couldn't run multiple times
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See in Appendix
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No] We do not release new assets
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]