Class-Conditioned Transformation for Enhanced Robust Image Classification

Tsachi Blau[†*] Roy Ganz[†] Chaim Baskin[‡] Michael Elad[†] Alex M. Bronstein[†]

2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) | 979-8-3315-1083-1/25/$31.00 ©2025 IEEE | DOI: 10.1109/WACV61041.2025.00637

Abstract

Robust classification methods predominantly concentrate on algorithms that address a specific threat model, resulting in ineffective defenses against other threat models. Real-world applications are exposed to this vulnerability, as malicious attackers might exploit alternative threat models. In this work, we propose a novel test-time threat model agnostic algorithm that enhances Adversarial-Trained (AT) models. Our method operates through COnditional image transformation and DIstance-based Prediction (CODIP) and includes two main steps: First, we transform the input image into each dataset class, where the input image might be either clean or attacked. Next, we make a prediction based on the shortest transformed distance. The conditional transformation utilizes the perceptually aligned gradients property possessed by AT models and, as a result, eliminates the need for additional models or additional training. Moreover, it allows users to choose the desired balance between clean and robust accuracy without training. The proposed method achieves state-of-the-art results demonstrated through extensive experiments on various models, AT methods, datasets, and attack types. Notably, applying CODIP leads to substantial robust accuracy improvement of up to $+23\%$, $+20\%$, $+26\%$, and $+22\%$ on CIFAR10, CIFAR100, ImageNet and Flowers datasets, respectively. For more details, visit the project page.

1. Introduction

Adversarial attacks are a significant concern in the field of computer vision, where maliciously designed perturbations are injected into an image to fool classifiers [2, 3, 6, 12, 17, 18, 21, 26, 27, 31, 40–42]. One common method of generating an adversarial example is through an iterative optimization process that searches for a norm-bounded perturbation $||\delta||_p \leq \epsilon$ that is added to the input image. The properties of the perturbation are determined by a threat model, characterized by the choice of the radius ϵ and the norm ℓ_p (typical choices for the latter are $p \in \{1, 2, \infty\}$). Following the discovery of adversarial attacks several defenses were introduced, such as Adversarial Training (AT) [6, 18, 19, 23, 29, 35, 48]. AT incorporates adversarial examples into the training process alongside the true labels, training the model to correctly classify malicious examples.

Test-time methods [7, 33, 34, 38, 45] aim to improve trained AT classifiers by changing their inference methodology. These methods change the standard prediction process where an instance is fed into a model that outputs a class prediction. Instead, they follow an inference algorithm that strengthens the prediction at the cost of additional inference time. While these methods are appealing, they have been used only to enhance the performance of a model using the same threat model used during training. Test-time methods have yet to be applied for the threat model-agnostic problem [4, 24, 28, 30], methods that defend a model against threat models that were not used during training. Threat model-agnostic property is desirable in real-world applications as the attacker is not limited to a single type of attack and will likely exploit the model's weaknesses, using threat models that were not used for training.

One property of adversarial attacks is that they do not substantially alter the input image x, as the perturbation δ is usually imperceptible. With this motivation in mind, we hypothesize that if we transform the attacked image towards the true class, the transformed image remains close to the input image, while, contrarily, a class-conditioned transformation toward a different class would lead to significant semantic changes to the input image. Hence, we conceive of a solution that operates through an optimal class-conditioned transformation with a regularization, which keeps the transformed image close to the input image. Unfortunately, such an optimal transformation does not exist and transformations usually require an additional model.

We show that such a transformation can be performed using the recently discovered perceptually aligned gradients (PAG) property possessed by AT models [15, 16, 36, 43]. Simply put, the gradients of a classifier with respect to the input image manifest an intelligible spatial structure perceptually similar to that of some class of the dataset. As a result, maximizing the conditional probability of such classifiers towards some target class, via a gradient-based method, results in class-related semantic visual features.

[*]tsachiblau@campus.technion.ac.il
[†]Department of Computer Science Technion - Israel Institute of Technology
[‡]School of Electrical and Computer Engineering - Ben-Gurion University of the Negev

2642-9381/25/$31.00 ©2025 IEEE
DOI 10.1109/WACV61041.2025.00637

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

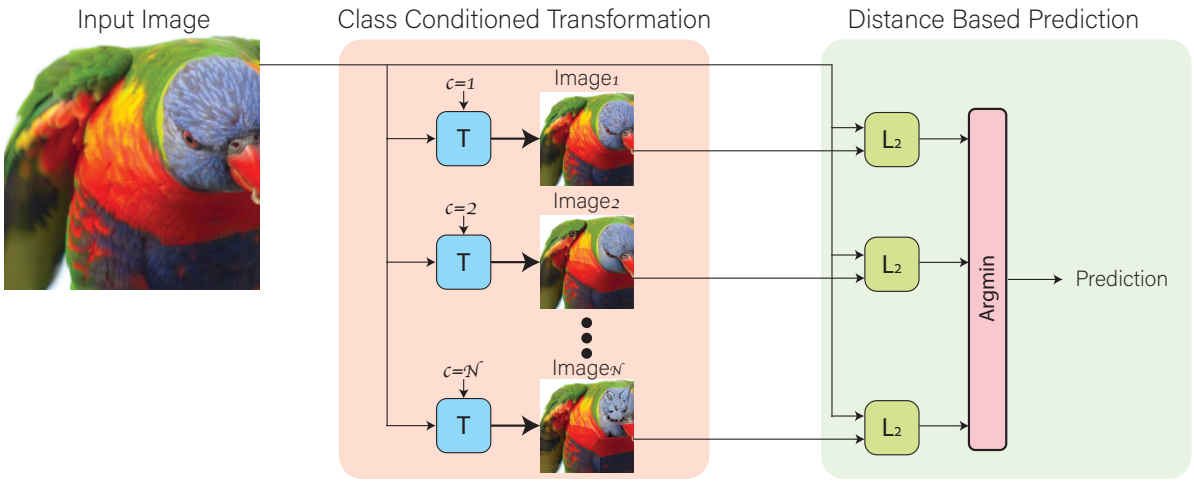


Figure 1. **An Overview of CODIP** At first, the input image (clean or attacked) is class conditioned transformed through $\{\mathrm{T}(\cdot|1), \ldots, \mathrm{T}(\cdot|N)\}$ to each one of the dataset classes, creating images $\{\mathrm{Image}_1, \ldots, \mathrm{Image}_N\}$. Next, the ℓ_2 distance is calculated between the input image and the transformed images $\{\mathrm{Image}_1, \ldots, \mathrm{Image}_N\}$, and prediction is made based on the shortest distance.

In this work, we propose COnditional transformation and DIstance-based Prediction (CODIP), a novel test-time threat model-agnostic defense that operates through two steps, as depicted in Fig. 1. First, we transform the input image into each one of the dataset classes, while regularizing the transformation, creating $\{\mathrm{Image}_1, \ldots, \mathrm{Image}_N\}$. The transformation is regularized since we want to change the class of the input image while remaining as close as possible to the input image. Moreover, the transformation utilizes the PAG property possessed by any AT model and spares the need for additional models or training. Next, we classify based on the shortest distance between the input image and the transformed images $\{\mathrm{Image}_1, \ldots, \mathrm{Image}_N\}$ with the assumption that the input image needs to cross a shorter distance to turn into the true class.

Our method can be applied to any differentiable classifier. Furthermore, it also equips the user with control over the clean-robust accuracy trade-off and enhances the model's performance for both seen and unseen threat models. Additionally, we propose $\mathrm{CODIP}_{\text{Top-}k}$, an efficient algorithm that filters the lowest probability classes, leading to speed up of inference time. We validate our method through extensive experiments on various AT methods, classifier architectures, and datasets. We empirically demonstrate that CODIP boosts the accuracy of any differentiable AT classifier, convolution or transformer based, across a variety of attacks, white-box or black-box, and generalizes well to unseen ones. Specifically, applying CODIP leads to substantial improvement of up to $+23\%$, $+20\%$, $+26\%$, and $+22\%$ on CIFAR10, CIFAR100, ImageNet and Flowers datasets, leading to state-of-the-art performance.

To summarize, the key contributions of our work are:

- CODIP , a novel test-time algorithm that operates through conditional transformation and distance-based prediction, utilizing the PAG property and does not require an additional transformation model or training.
- $\mathrm{CODIP}_{\text{Top-}k}$, an efficient algorithm that speeds up the inference time.
- A controllable clean-robust accuracy trade-off mechanism, that can be adapted by the user as needed, without training.
- State-of-the-art results for seen and unseen threat models, white-box or black-box. Demonstrated over multiple models, convolution or transformer based, AT methods, datasets, and provide extensive ablations to each component of our method.

2. Related work

Adversarial Training methods [6, 18, 29] were first introduced to robustify a classifier against a specific threat model. However, a group of works [5, 24] have shown that such adversarially trained classifiers are less effective against unseen attacks. This finding has led to research that focuses on defenses for unseen attacks by training a model on a group of threat models. The work done by Jordan *et al.* [22] suggests an attack that targets both texture and geometry, while Maini *et al.* [30] suggests a combination of PGD-based attacks. Another attempt was made by Laidlaw *et al.* [28], which provides a latent space norm-bounded AT. This line of work, however, has two major drawbacks. First, it leads to inferior performance for every threat model compared to training a designated model for each specific threat model. Second, even a group of threat models cannot cover all possible threats, leaving the model vulnerable to attacks not included in the training.

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

Generative Model-Based Defences are methods that aim at cleaning the attack from the image, shifting the image back to the image manifold, before feeding it into the classifier for classification. The work by Song *et al.* [39] purifies the image using pixelCNN and [14] suggest restoring the corrupt image with EBM. Hill *et al.* [20] suggest to apply long-run Langevin sampling and Yoon *et al.* [46] offer gradient ascent score based-model. The work done by Blau *et al.* [4] combines the two worlds of generative model and threat model agnostic defense, using a vanilla-trained diffusion model and classifier. While these methods are close in spirit to ours in the sense that they perform a transformation over the attacked image, they all require additional models to perform the transformation.

Test-Time Methods improve the performance of trained classifiers solely by using an inference phase of the neural network, while not requiring specialized training. However, they may result in longer inference times. Cohen *et al.* [7] and Raff *et al.* [34] suggest multiple realizations of the augmented image to be performed during test-time. Cohen *et al.* [7] suggest augmenting with Gaussian noise, while Raff *et al.* [34] advocate using other types of augmentation; both methods involve averaging the classification predictions from the augmented images. Perez *et al.* [33] perform test augmentation and predict the average. Schwinn *et al.* [38] suggest analyzing the robustness of a local decision region near the attacked image, and Wu *et al.* [45] suggest to attack towards every class. The methods proposed by [7,34] require fine-tuning the model, while [38,45] require neither fine-tuning nor access to the training data.

There are several distinctions between our approach and existing test-time methods. Firstly, while these methods focus on improving the performance of the threat model used for training the AT model, they fail to address the crucial issue of threat model-agnostic attacks. Secondly, unlike these methods, which rely on the classifier to make predictions, our predictions are based on measuring the distance between the input image and all the transformed images. Lastly, the works of [38, 45], which are most similar to our method, operate only in a local neighborhood around the input image, preventing the model from making significant semantic changes to the image. In contrast, our model is not limited to a specific neighborhood around the input image, as it transforms the image and alters its semantic appearance.

3. Our method

In this section, we supply an overview of our method. Next, we offer a detailed explanation that goes through the motivation and each part of our algorithm. Finally, we present an efficient alternative for our method.

3.1. Method Overview

We propose a test-time classification method that enhances an AT classification model for seen and unseen attacks. Our method does not operate in the standard classification methodology $f : x \to y$, where $f(\cdot)$ is the classifier, $x \in R^{H\text{x}W\text{x}3}$ is the input image, $y \in [0,1]^N$ is the prediction probability vector, and N is the number of classes of the dataset.

Instead, our method operates through two phases, as illustrated in Fig. 2: First, we get an input image (which can be either clean or attacked) and transform it N times, each time to a different class of the dataset. The transformation utilizes the PAG property of the AT classifier and, hence, does not require additional architecture or additional training. The transformation is performed through an iterative class-conditioned regularized optimization process. For every class, we perform M such optimization steps, in each step we follow two objectives. First, maximizing the classifier probability towards the current class. Second, regularizing the transformation so the transformed image will stay close to the input image x.

3.2. CODIP

In what follows, we detail our method summarized in Algorithm 1: We perform N transformations towards each one of the dataset's classes, and we initialize the i^{th} class transformation to zero $\mathbf{T}_{i0}$, in Line 3. Next, we start the transformation, which operates through M gradient descent steps. In each iteration, we calculate the gradient of our objective and store it in $\mathbf{G}_{ij}$ in Line 5. Next, in Line 6, we perform a gradient descent step in the direction of $\mathbf{G}_{ij}$, the step of size α which is a hyper-parameter. It is essential to normalize the gradient step since we want to make an even progress in each step, regardless of the gradient size. Without normalization, the step size might be very small, which will prevent us from making progress during the transformation. Before finishing the update step, in Line 6, we perform a projection back to the image domain to $R^{H\times W\times 3} \in [0,1]^{H\times W\times 3}$. When finishing the transformation's iterations, we calculate the transformation distance using ℓ_2 over the M^{th} column of the transformation matrix, and we store it in the distance vector $\mathbf{d}$, in Line 9. Finally, when the process is over we predict according to the shortest transformation.

CODIP's objective, which appears in Line 5, weights two targets: The first term, $L_{CE}\left(f\left(x+\mathbf{T}_{ij}\right),i\right)$, is a cross entropy loss between the classifier's prediction $f(\cdot)$ over the j^{th} step of the i^{th} transformation $x+\mathbf{T}_{ij}$, and the current class i. The goal of this term is to measure the classifier's error towards class i and to transform the image so it semantically looks like class i. The goal of the second term, $\gamma\|\mathbf{T}_{ij}\|_2$, is to regularize the transformation to stay close to the input image x. This regularization is needed since we desire to change the input image toward class i with

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

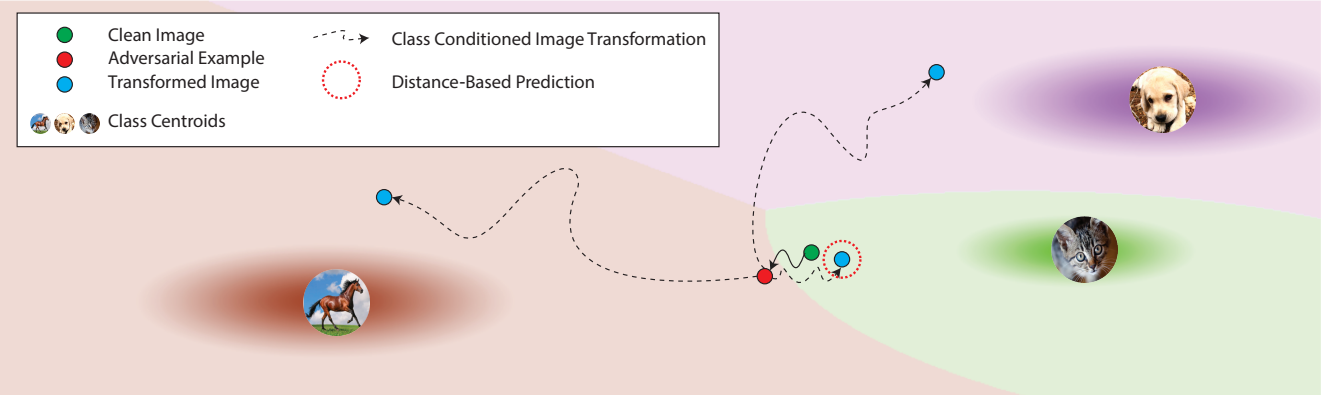


Figure 2. **A Comparison Between the Classifier and CODIP Decision Rules** The background color of the image describes the classifier's classification rules, and the intensity describes the classifier's certainty. The clean image (green dot) is attacked (red dot), leading to a wrong classification. In contrast, CODIP predicts based on the shortest transformation. It operates in two steps: First, it class conditioned transform (dotted arrow) the attacked image towards each one of the datasets's classes (blue dots). Next, prediction is made based on the shortest distance between the attacked image and the transformed images, highlighted by a red dotted circle.

Algorithm 1 CODIP

Input classifier $f(\cdot)$, input image x, step size α, regularization coefficient γ, number of iterations M, number of dataset's classes N

1: **procedure** CODIP
2: **for** $i \in C = \{1, \ldots, N\}$ **do** ▷ Iterate over # classes
3: $\mathbf{T}_{i0} \leftarrow 0$ ▷ Initialize the i^{th} transformation
4: **for** j `in` $0 : M-1$ **do** ▷ Iterate over # steps
5: $\mathbf{G}_{ij} \leftarrow \nabla_{\mathbf{T}_{ij}} \left[L_{CE} \left(f\left(x + \mathbf{T}_{ij}\right), i \right) + \gamma \left\| \mathbf{T}_{ij} \right\|_2^2 \right]$
6: $\mathbf{T}_{ij+1} \leftarrow \Pi \left(\mathbf{T}_{ij} - \alpha \frac{\mathbf{G}_{ij}}{\|\mathbf{G}_{ij}\|_2} \right)$
7: **end for**
8: **end for**
9: $\mathbf{d} \leftarrow \|\mathbf{T}_{1:NM}\|_2$ ▷ Update the transformation distances for each class
10: $\hat{y} = \arg\min_{i=1,\ldots,N} \mathbf{d}_i$ ▷ Classify based on the shortest distance
11: **return** $\hat{y}$
12: **end procedure**

The operator $\Pi(z)$ projects z onto the image domain $\mathbb{R}^{H \times W \times 3} \in [0,1]^{H \times W \times 3}$ by clipping its values.

minimal semantic changes. Without this term, the transformation could turn the image into any instance of class i. Even in the case where i is the true class, we can change the image into a completely different instance of the class, ultimately nullifying the efficacy of our method. To conclude, balancing between the two terms, which is determined by the hyper-parameter γ, influences the class-similarity transformation balance, hence, it is essential.We demonstrate the impact of changing this value in Fig. 3, and discuss it further in Appendix H.

3.3. CODIP$_{\text{Top-}k}$

While CODIP is an effective algorithm that significantly enhances the classifier robustness, its inference time grows with the number of datasets' classes. To mitigate this issue, we propose CODIP$_{\text{Top-}k}$, which reduces the inference time significantly by filtering out the classes with the lowest probability.

The classifier predicts the probability of the input image belonging to each class $f(x) \rightarrow \mathbb{R}^N \in [0,1]^N$. One way to speed up our algorithm is by limiting our prediction only to the most probable classes predicted by the classifier. More specifically, we choose a group of k classes $C_k \subset C = \{1, \ldots, N\}$ containing the top-k most probable prediction made by the classifier. Next, we modify our method, which is presented in Algorithm 1, by adapting Line 2 to transform only towards the most probable classes C_k.

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

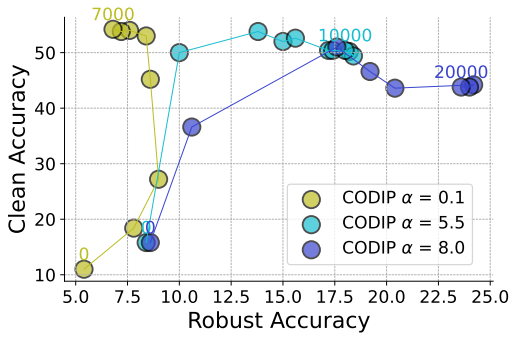


Figure 3. **Impact of γ on Clean-Robust Accuracy Trade-off** We present three α working points on the ImageNet dataset using an AT model $L_2, \epsilon = 3.0$.

4. Experiments

In this section, we provide implementation details, along with qualitative and quantitative evidence supporting our method. We present the datasets, baselines, and implementation details, followed by the results on CIFAR-10, CIFAR-100 [25], ImageNet [11], and Flowers [32]. We compare our method using common practice attacks, including black-box attacks. Next, we introduce a method to control the clean-robust accuracy trade-off and conclude with a thorough runtime comparison.

In the Appendix, we include an ablation study of each component of our method in Appendix A, qualitative results in Appendix B, white-box attack in Appendix C, the influence of a designated Top-k attack in Appendix D, further implementation details in Appendix E and further explanations in Appendix F, Appendix G, Appendix H.

4.1. Experimental Settings

Datasets The dataset CIFAR10 and CIFAR100 [25] both containing 50K samples for training and 10K samples for test, each of size $(3 \times 32 \times 32)$. They differ in the number of classes, where CIFAR10 contains 10 classes and CI-

Table 1. **CIFAR10 and CIFAR100 Results** The performance of test-time methods over two datasets: CIFAR10 and CIFAR100. The results are grouped per AT model, evaluating different test-time methods: 'Base' in which we do not use any test-time method, followed by other test-time methods.

Dataset	AT Method	Trained Threat Model	Architecture	Test-Time Method	Clean	Attack L_∞ 8/255	Attack L_∞ 16/255	Attack L_2 0.5	Attack L_2 1.0
CIFAR10	PAT [28]		RN50		71.60%	28.70%	–	33.30%	–
	AT [29]	$L_2, \epsilon = 0.5$	RN50	Base	90.83%	29.04%	00.93%	69.24%	36.21%
				RSmooth [7]	89.43%	30.84%	01.23%	68.94%	38.53%
				TTE [33]	**90.99%**	36.41%	02.40%	71.90%	41.18%
				DRQ [38]	88.79%	45.37%	07.09%	77.56%	51.28%
				CODIP	87.40%	**51.66%**	**14.96%**	**78.66%**	**59.82%**
	Rebuffi *et al.* [35]	$L_2, \epsilon = 0.5$	WRN28-10	Base	**91.79%**	47.85%	05.00%	78.80%	54.73%
				TTE [33]	91.59%	50.49%	06.79%	79.18%	55.38%
				DRQ [38]	90.99%	58.66%	**13.69%**	84.12%	64.69%
				CODIP	88.23%	**59.99%**	11.45%	**85.56%**	**66.15%**
	Rebuffi *et al.* [35]	$L_\infty, \epsilon = 8/255$	WRN28-10	Base	**87.33%**	60.77%	25.44%	66.72%	35.01%
				TTE [33]	87.30%	61.52%	27.50%	66.88%	36.07%
				DRQ [38]	87.17%	66.23%	33.62%	72.24%	44.56%
				CODIP	85.00%	**66.86%**	**34.88%**	**74.24%**	**53.02%**
	Gowal *et al.* [19]	$L_\infty, \epsilon = 8/255$	WRN70-16	Base	**91.09%**	65.88%	25.95%	66.43%	27.21%
				DRQ [38]	90.77%	71.00%	35.89%	72.87%	39.51%
				CODIP	88.18%	**72.02%**	**40.30%**	**75.90%**	**49.21%**
CIFAR100	Rebuffi *et al.* [35]	$L_\infty, \epsilon = 8/255$	WRN28-10	Base	**62.40%**	32.06%	12.47%	38.32%	18.86%
				TTE [33]	62.35%	33.25%	13.84%	39.14%	20.22%
				DRQ [38]	61.32%	38.22%	19.41%	44.58%	26.78%
				CODIP	55.18%	37.61%	19.72%	45.86%	32.79%
				CODIP$_{\text{Top-}k}$	57.95%	**38.73%**	**19.87%**	**47.56%**	**33.01%**
	Gowal *et al.* [19]	$L_\infty, \epsilon = 8/255$	WRN70-16	Base	**69.15%**	36.90%	13.64%	40.86%	17.20%
				DRQ [38]	69.12%	43.96%	20.25%	48.95%	25.43%
				CODIP	59.83%	44.66%	**23.81%**	51.32%	**37.02%**
				CODIP$_{\text{Top-}k}$	62.47%	**46.09%**	23.48%	**53.06%**	36.19%

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

FAR100 contains 100. The ImageNet dataset [11] contains 1M images for training and 50K for validation, the images are resized to $(3\times256\times256)$ and cropped to $(3\times224\times224)$. The Flowers dataset [32] contains 103 classes, 1030 images for training and 6129 for test, the images are resized to $(3 \times 256 \times 256)$ and cropped to $(3 \times 224 \times 224)$.

Evaluation For evaluation we use the test set of CIFAR10, CIFAR100 and Flowers, and the validation set of ImageNet as we follow the convention of ImageNet evaluation, and we report the average accuracy. We employ AutoAttack [9] as a robust and widely accepted benchmark for adversarial attack. AutoAttack is an ensemble of four diverse attacks, where all four attacks are applied to perform a successful attack and allow a reliable robustness evaluation. It includes the attacks: APGD-CE and APGD-DLR which were proposed in [9], FAB [8] and Square Attack [1]. For CIFAR10 and CIFAR100 we evaluate the performance using the following thereat models $(\ell_\infty, \epsilon = 8/255)$, $(\ell_\infty, \epsilon = 16/255)$, $(\ell_2, \epsilon = 0.5)$, $(\ell_2, \epsilon = 1.0)$, and for ImageNet and Flowers we use $(\ell_\infty, \epsilon = 4/255)$, $(\ell_\infty, \epsilon = 8/255)$, $(\ell_2, \epsilon = 3.0)$, $(\ell_2, \epsilon = 6.0)$. Following previous works [20, 28, 46], we attack on the classifier, as it is computationally impossible to attack these methods.

We follow the common data split and portion, proposed by Trustworthy-AI-Group*, for all of the transformer based models and for the black-box attacks.

Implementation Details We check our method performance on SOTA AT trained classifiers. For CIFAR10 and CIFAR100 we use Rebuffi *et al.* [35], Gowal *et al.* [19] and Madry *et al.* [29], for ImageNet we use Madry *et al.*, Salman *et al.* [37] and Debenedetti et al. [10], and for Flowers we use Debenedetti et al. [10]. For each AT classifier, we specify the chosen hyper-parameters in Appendix E, and for each transformation we perform 30 iterations. We use the following architectures: ResNet50, WideResNet50-2, WideResNet28-10, WideResNet70-16, XCiT-S/M.

Baselines To evaluate our method we compare it to a few baselines. We start with 'Base' which is the AT classifier and PAT [28] which suggests an AT threat model-agnostic defense. PAT [28] and Randomized Smoothing [7] were evaluated only on the CIFAR10 dataset. Additionally, we compare our method to two other test-time methods: TTE [33] and DRQ [38]. We evaluate both of these methods on different threat models, allowing a fair comparison. TTE was evaluated on CIFAR10, CIFAR100, and ImageNet, however, DRQ can not be applied to ImageNet, as

*https://github.com/Trustworthy-AI-Group/TransferAttack

Table 2. **ImageNet and Flowers Results** The performance of test-time methods over ImageNet and Flowers. The results are grouped per AT model, evaluating different test-time methods: 'Base' in which we do not use any test-time method, followed by other methods.

Dataset	AT Method	Trained Threat Model	Architecture	Test-Time Method	Clean	Attack L_∞ 4/255	Attack L_∞ 8/255	Attack L_2 3.0	Attack L_2 6.0
Imagenet	Madry et al. [29]	$L_2, \epsilon = 3.0$	RN50	Base	57.90%	24.82%	05.26%	30.85%	10.35%
				TTE [33]	**57.97%**	26.56%	06.56%	32.33%	11.71%
				CODIP$_{\text{Top-}k}$	51.51%	**40.64%**	**22.10%**	**45.75%**	**34.0%**
	Salman et al. [37]	$L_2, \epsilon = 3.0$	WRN50-2	Base	66.91%	30.86%	06.19%	38.27%	12.85%
				TTE [33]	**66.94%**	-	07.61%	-	14.57%
				CODIP$_{\text{Top-}k}$	63.27%	**47.68%**	**25.23%**	**53.35%**	**38.36%**
	Madry et al. [29]	$L_\infty, \epsilon = 4/255$	RN50	Base	62.42%	28.96%	08.12%	07.06%	00.39%
				TTE [33]	**62.58%**	31.02%	09.69%	12.14%	01.25%
				CODIP$_{\text{Top-}k}$	55.75%	**40.45%**	**17.97%**	**18.03%**	**04.42%**
	Salman et al. [37]	$L_\infty, \epsilon = 4/255$	WRN50-2	Base	68.41%	37.80%	12.84%	07.03%	00.21%
				TTE [33]	**68.61%**	-	13.32%	-	01.78%
				CODIP$_{\text{Top-}k}$	60.44%	**48.44%**	**25.44%**	**20.18%**	**04.24%**
	Debenedetti et al. [10]	$L_\infty, \epsilon = 8/255$	XCiT-S	Base	**84.70%**	58.70%	32.60%	45.20%	14.50%
				CODIP$_{\text{Top-}k}$	81.00%	**67.30%**	**44.00%**	**60.00%**	**27.50%**
	Debenedetti et al. [10]	$L_\infty, \epsilon = 8/255$	XCiT-M	Base	**89.55%**	63.80%	33.70%	45.70%	11.20%
				CODIP$_{\text{Top-}k}$	87.70%	**72.80%**	**45.70%**	**61.40%**	**22.10%**
Flowers	Debenedetti et al. [10]	$L_\infty, \epsilon = 8/255$	XCiT-S	Base	**79.30%**	62.40%	39.60%	55.60%	23.20%
				CODIP$_{\text{Top-}k}$	78.30%	**69.80%**	**50.40%**	**68.40%**	**45.70%**

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

stated in their work, since it is computationally infeasible.

We propose two variants of our method: method CODIP and CODIP$_{\text{Top-}k}$. Both are presented in Sec. 3 and while CODIP performs conditioned transformation of the input image towards all of the dataset's classes, CODIP$_{\text{Top-}k}$ transforms only toward the group of classes that is more likely to contain the true class. CODIP$_{\text{Top-}k}$ is essential when employing our method on datasets with a large number of classes. To ensure a fair comparison, we explore a designated Top-k attack which was proposed by Zhang *et al.* [47] and we discuss our findings in Appendix D.

4.2. Comparison

We present the performance over four datasets: CIFAR10 and CIFAR100 in Tab. 1, and ImageNet and Flowers in Tab. 2. We present AT models that were trained on different Trained Threat Model and with different architecture, and we employ different test-time method: Base, TTE DRQ, CODIP and CODIP$_{\text{Top-}k}$. For CIFAR10 and CIFAR100 datasets we compare our method to PAT [28], Randomized Smoothing [7], TTE [33] and to DRQ [38], and for ImageNet we use TTE [33].

CODIP enhances the base AT methods (represented by 'Base' under 'Test-Time Method') for both seen and unseen attacks. In particular, we enhance the performance for seen attacks by up to 9%, 10%, 15%, 11%, and for unseen attacks by up to 10%, 20%, 26%, 22% over CIFAR10, CIFAR100, Imagenet, and Flowers respectively.

When comparing to other test-time methods, we enhance the performance for seen attacks by up to 1%, 3% and 12%, and for unseen attacks by up to 10%, 20% and 24% over CIFAR10, CIFAR100 and Imagenet respectively. Additionally, for CIFAR10 we compare CODIP to the PAT [28], a method that addresses the issue of robustness to unseen attacks, and we report up to 52% higher accuracy. We further compare our method to Randomized Smoothing [7], a test-time method requiring the robust classifier to be trained on Gaussian noise. Typically, robust classifiers are not trained using this augmentation, which often leads to ineffective outcomes with Randomized Smoothing, particularly with the models we employ, such as AT [29] RN50. Moreover, for CIFAR100 we use both CODIP and CODIP$_{\text{Top-}k}$ ($k = 10$) demonstrating that not only do we benefit from much faster inference time but also that the performances are slightly better. For ImageNet we use CODIP$_{\text{Top-}k}$ ($k = 20$), as CODIP is computationally not feasible, and for Flowers we use CODIP$_{\text{Top-}k}$ ($k = 10$).

Top-k The Top-k version of CODIP sometimes outperforms the full CODIP because it focuses on the most likely classes, reducing the influence of less relevant or noisy class transformations. By concentrating on a smaller set of high-confidence predictions, the Top-k version can enhance accuracy by avoiding the potential noise introduced when evaluating all possible classes. This selective attention allows it to maintain or even improve performance in scenarios where the correct class is consistently within the top k predictions. Additional information in Appendix F.

Table 3. **Black-Box Attacks** Two different Trained Threat Models (TTM) are evaluated. Both RN50-2 models were adversarially trained by Salman et al. [37]. Each model is evaluated using Imagenet [11] dataset, under clean images, three black-box attacks, and compared to the base model.

TTM	Test Time Method	Clean	MI-FGSM [13]	Admix [44]	L2T [49]
$L_2, \epsilon = 3.0$	Base	**89.40**%	79.70%	74.20%	73.10%
	CODIP$_{\text{Top-}k}$	87.80%	**80.60**%	**76.40**%	**74.30**%
$L_\infty, \epsilon = 4/255$	Base	**91.10**%	83.40%	78.30%	76.30%
	CODIP$_{\text{Top-}k}$	91.00%	**84.70**%	**80.00**%	**78.30**%

4.3. Black Box Attack

We present the effectiveness of our method against advanced black box attacks in Tab. 3. including MI-FGSM [13] and Admix [44], and L2T [49] which is currently considered the strongest one. We utilize the official implementation Trustworthy-AI-Group[†], maintain the same data split, and use an ensemble of surrogate models under the $L_\infty = 16/255$ attack. As demonstrated, our defense enhances the performance under black-box attacks as well.

4.4. Clean-Robust Accuracy Trade-off

The clean-robust accuracy trade-off is a well-known phenomenon [43, 48]. Training a classifier with adversarially perturbed images can significantly improve the robust accuracy, but it often comes at the expense of clean accuracy. When robustifying a model using AT, this trade-off is determined during training and cannot be controlled afterwards. Similarly, test-time methods also operate with a fixed clean-robust accuracy trade-off and do not offer a way to control it. Further details are provided in Appendix G.

Our method, on the other hand, allows the user to control the trade-off by adjusting α, which is the transformation's step size, presented in Algorithm 1. By increasing α, the user can prioritize robust accuracy, while decreasing α prioritize clean accuracy. In Fig. 4, we present the clean-robust accuracy tradeoff for AT model trained on CIFAR10 by Madry et al.'s [29] on threat model $L_2, \epsilon = 0.5$ and architecture ResNet-50. We compare the performance of the robust classifier under the attack threat model ($L_\infty, \epsilon = 16/255$) using the AT model and three other test-time methods: Base which is the performance of the AT classifier, TTE [33], DRQ [38], and our method, CODIP . As depicted in the

[†]https://github.com/Trustworthy-AI-Group/TransferAttack

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

Table 4. **Runtime and Compute** Comparing Base, TTE [33], DRQ [38], CODIP , and CODIP$_{\text{Top-}k}$. For both CODIP and CODIP$_{\text{Top-5}}$ we provide the runtime alongside Maximum Batch-Size (MBS).

Dataset	Method	Architecture	Base	TTE [33]	DRQ [38]	CODIP Times / MBS	CODIP$_{\text{Top-20}}$	CODIP$_{\text{Top-10}}$	CODIP$_{\text{Top-5}}$ Times / MBS
CIFAR10	AT [29]	RN50	$\times 1$	$\times 10$	$\times 1,192$	$\times 144/150$	–	–	$\times 144/300$
	Rebuffi et al. [35]	WRN28-10	$\times 1$	$\times 10$	$\times 1,441$	$\times 126/128$	–	–	$\times 119/256$
	Gowal et al. [19]	WRN70-16	$\times 1$	$\times 11$	$\times 1,510$	$\times 240/36$	–	–	$\times 143/72$
CIFAR100	Rebuffi et al. [35]	WRN28-10	$\times 1$	$\times 10$	$\times 14,128$	$\times 1,017/12$	$\times 205$	$\times 123$	$\times 112/280$
	Gowal et al. [19]	WRN28-10	$\times 1$	$\times 11$	$\times 15,193$	$\times 1,776/3$	$\times 420$	$\times 246$	$\times 146/65$
ImageNet	AT [29]	RN50	$\times 1$	$\times 7$	–	–	$\times 133$	$\times 90$	$\times 90/80$

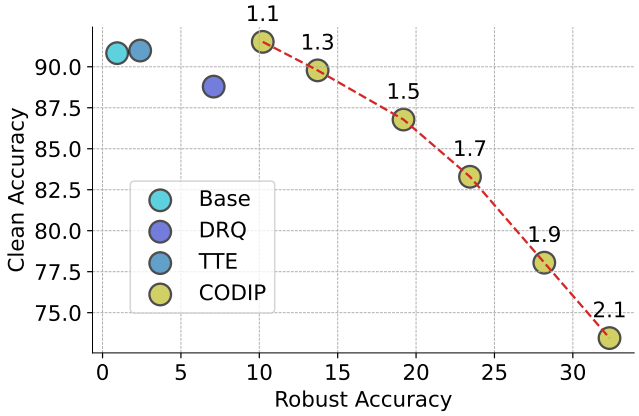


Figure 4. **Clean-Robust Accuracy Trade-off** A demonstration of our proposed controlled clean-robust accuracy tradeoff. The tradeoff is controlled by adapting the step size value α, specified beside each of CODIP workpoints. The used test-time methods: 'Base' which is the base AT model, TTE [33], DRQ [38], and CODIP.

figure, CODIP we can control the trade-off by adjusting α, prioritizing clean or robust accuracy as needed. We show that we can improve the clean accuracy or enhance robust accuracy by up to 25%.

4.5. Runtime Analysis

This section provides a runtime comparison across different architectures and datasets, detailed in Tab. 4. For a fair comparison, all experiments were conducted using one RTX A6000 with a batch size of 1.

We compare our method to the base classifier, TTE [33] and to DRQ [38], reporting the relative inference time compared to the base classifier. We can see that TTE [33] increase the inference time due to the predetermined number of augmentations performed for each image. This extra time is added since we perform a predetermined number of augmentation for each image. The other methods, DRQ [38] and our approach, also result in increased inference time compared to the base classifier. However, both CODIP and, in particular, CODIP$_{\text{Top-}k}$ demonstrate significantly faster performance. Specifically, CODIP$_{\text{Top-}k}$ is up to 100 times faster than DRQ [38].

Another crucial consideration is memory consumption, detailed under both CODIP and CODIP$_{\text{Top-5}}$ columns. We specify the Maximum Batch Size (MBS) that fits into memory. It is evident that CODIP$_{\text{Top-5}}$ enables an increase in batch size of up to 23 times.

5. Conclusion

In this work, we introduce CODIP – a novel test-time threat model-agnostic method that enhances AT models through conditional image transformation and distance-based prediction. Our method leverages AT models specialized in a specific threat model, boosts their performance on the trained threat model, and equips them with threat model-agnostic capabilities without further training or access to the training data. Additionally, we provide a controllable, clean-robust accuracy trade-off mechanism that can be adapted by the user as required. Moreover, we propose CODIP$_{\text{Top-}k}$, an efficient method that speeds up the inference time. Finally, we conducted an extensive evaluation to demonstrate state-of-the-art results on various architectures, convolution or transformer based, AT training methods, white-box and black-box attacks, and four widely used datasets: CIFAR10, CIFAR100, ImageNet, and Flowers.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 863839).

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

References

[1] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European conference on computer vision*, pages 484–501. Springer, 2020. 6

[2] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International conference on machine learning*, pages 284–293. PMLR, 2018. 1

[3] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 387–402. Springer, 2013. 1

[4] Tsachi Blau, Roy Ganz, Bahjat Kawar, Alex Bronstein, and Michael Elad. Threat model-agnostic adversarial defense using diffusion models. *arXiv preprint arXiv:2207.08089*, 2022. 1, 3

[5] Tom B Brown, Nicholas Carlini, Chiyuan Zhang, Catherine Olsson, Paul Christiano, and Ian Goodfellow. Unrestricted adversarial examples. *arXiv preprint arXiv:1809.08352*, 2018. 2

[6] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 3–14, 2017. 1, 2

[7] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR, 2019. 1, 3, 5, 6, 7

[8] Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In *International Conference on Machine Learning*, pages 2196–2205. PMLR, 2020. 6

[9] 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. 6

[10] Edoardo Debenedetti, Vikash Sehwag, and Prateek Mittal. A light recipe to train robust vision transformers. In *2023 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 225–253. IEEE, 2023. 6

[11] 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, 6, 7

[12] Samuel Dodge and Lina Karam. A study and comparison of human and deep learning recognition performance under visual distortions. In *2017 26th international conference on computer communication and networks (ICCCN)*, pages 1–7. IEEE, 2017. 1

[13] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9185–9193, 2018. 7

[14] Yilun Du and Igor Mordatch. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2019. 3

[15] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. *arXiv preprint arXiv:1906.00945*, 2019. 1

[16] Christian Etmann, Sebastian Lunz, Peter Maass, and Carola-Bibiane Schönlieb. On the connection between adversarial robustness and saliency map interpretability. *arXiv preprint arXiv:1905.04172*, 2019. 1

[17] Robert Geirhos, David HJ Janssen, Heiko H Schütt, Jonas Rauber, Matthias Bethge, and Felix A Wichmann. Comparing deep neural networks against humans: object recognition when the signal gets weaker. *arXiv preprint arXiv:1706.06969*, 2017. 1

[18] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 2

[19] Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. *arXiv preprint arXiv:2010.03593*, 2020. 1, 5, 6, 8

[20] Mitch Hill, Jonathan Mitchell, and Song-Chun Zhu. Stochastic security: Adversarial defense using long-run dynamics of energy-based models. *arXiv preprint arXiv:2005.13525*, 2020. 3, 6

[21] Hossein Hosseini, Baicen Xiao, and Radha Poovendran. Google's cloud vision api is not robust to noise. In *2017 16th IEEE international conference on machine learning and applications (ICMLA)*, pages 101–105. IEEE, 2017. 1

[22] Matt Jordan, Naren Manoj, Surbhi Goel, and Alexandros G Dimakis. Quantifying perceptual distortion of adversarial examples. *arXiv preprint arXiv:1902.08265*, 2019. 2

[23] Harini Kannan, Alexey Kurakin, and Ian Goodfellow. Adversarial logit pairing. *arXiv preprint arXiv:1803.06373*, 2018. 1

[24] Max Kaufmann, Daniel Kang, Yi Sun, Steven Basart, Xuwang Yin, Mantas Mazeika, Akul Arora, Adam Dziedzic, Franziska Boenisch, Tom Brown, et al. Testing robustness against unforeseen adversaries. *arXiv preprint arXiv:1908.08016*, 2019. 1, 2

[25] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009. 5

[26] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016. 1

[27] Alexey Kurakin, Ian J Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *Artificial intelligence safety and security*, pages 99–112. Chapman and Hall/CRC, 2018. 1

[28] Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. *arXiv preprint arXiv:2006.12655*, 2020. 1, 2, 5, 6, 7

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.

[29] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 1, 2, 5, 6, 7, 8

[30] Pratyush Maini, Eric Wong, and Zico Kolter. Adversarial robustness against the union of multiple perturbation models. In *International Conference on Machine Learning*, pages 6640–6650. PMLR, 2020. 1, 2

[31] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 427–436, 2015. 1

[32] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008. 5, 6

[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. 1, 3, 5, 6, 7, 8

[34] Edward Raff, Jared Sylvester, Steven Forsyth, and Mark McLean. Barrage of random transforms for adversarially robust defense. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6528–6537, 2019. 1, 3

[35] Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness. *arXiv preprint arXiv:2103.01946*, 2021. 1, 5, 6, 8

[36] 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. 1

[37] 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. 6, 7

[38] Leo Schwinn, Leon Bungert, An Nguyen, René Raab, Falk Pulsmeyer, Doina Precup, Björn Eskofier, and Dario Zanca. Improving robustness against real-world and worst-case distribution shifts through decision region quantification. *arXiv preprint arXiv:2205.09619*, 2022. 1, 3, 5, 6, 7, 8

[39] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *arXiv preprint arXiv:1710.10766*, 2017. 3

[40] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

[41] Dogancan Temel and Ghassan AlRegib. Traffic signs in the wild: Highlights from the ieee video and image processing cup 2017 student competition [sp competitions]. *arXiv preprint arXiv:1810.06169*, 2018. 1

[42] Dogancan Temel, Jinsol Lee, and Ghassan AlRegib. Cure-or: Challenging unreal and real environments for object recognition. In *2018 17th IEEE international conference on machine learning and applications (ICMLA)*, pages 137–144. IEEE, 2018. 1

[43] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018. 1, 7

[44] Xiaosen Wang, Xuanran He, Jingdong Wang, and Kun He. Admix: Enhancing the transferability of adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16158–16167, 2021. 7

[45] Boxi Wu, Heng Pan, Li Shen, Jindong Gu, Shuai Zhao, Zhifeng Li, Deng Cai, Xiaofei He, and Wei Liu. Attacking adversarial attacks as a defense. *arXiv preprint arXiv:2106.04938*, 2021. 1, 3

[46] Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score-based generative models. In *International Conference on Machine Learning*, pages 12062–12072. PMLR, 2021. 3, 6

[47] Chaoning Zhang, Philipp Benz, Adil Karjauv, Jae Won Cho, Kang Zhang, and In So Kweon. Investigating top-k white-box and transferable black-box attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15085–15094, 2022. 7

[48] 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. 1, 7

[49] Rongyi Zhu, Zeliang Zhang, Susan Liang, Zhuo Liu, and Chenliang Xu. Learning to transform dynamically for better adversarial transferability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24273–24283, 2024. 7

Authorized licensed use limited to: Institute of Science and Technology Austria (IST). Downloaded on September 22,2026 at 09:42:20 UTC from IEEE Xplore. Restrictions apply.