Provable Unrestricted Adversarial Training without Compromise with Generalizability

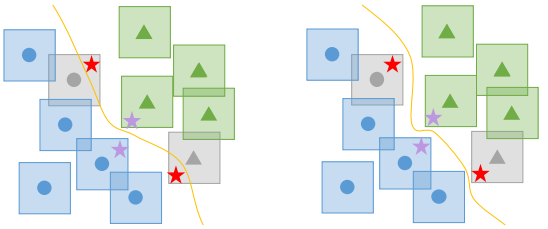
Lilin Zhang, Ning Yang, Yanchao Sun, Philip S. Yu, *Fellow, IEEE*

***Abstract*—Adversarial training (AT) is widely considered as the most promising strategy to defend against adversarial attacks and has drawn increasing interest from researchers. However, the existing AT methods still suffer from two challenges. First, they are unable to handle unrestricted adversarial examples (UAEs), which are built from scratch, as opposed to restricted adversarial examples (RAEs), which are created by adding perturbations bound by an l_p norm to observed examples. Second, the existing AT methods often achieve adversarial robustness at the expense of standard generalizability (i.e., the accuracy on natural examples) because they make a tradeoff between them. To overcome these challenges, we propose a unique viewpoint that understands UAEs as imperceptibly perturbed unobserved examples. Also, we find that the tradeoff results from the separation of the distributions of adversarial examples and natural examples. Based on these ideas, we propose a novel AT approach called Provable Unrestricted Adversarial Training (PUAT), which can provide a target classifier with comprehensive adversarial robustness against both UAE and RAE, and simultaneously improve its standard generalizability. Particularly, PUAT utilizes partially labeled data to achieve effective UAE generation by accurately capturing the natural data distribution through a novel augmented triple-GAN. At the same time, PUAT extends the traditional AT by introducing the supervised loss of the target classifier into the adversarial loss and achieves the alignment between the UAE distribution, the natural data distribution, and the distribution learned by the classifier, with the collaboration of the augmented triple-GAN. Finally, the solid theoretical analysis and extensive experiments conducted on widely-used benchmarks demonstrate the superiority of PUAT.**

***Index Terms*—Adversarial Robustness, Adversarial Training, Unrestricted Adversarial Examples, Standard Generalizability.**

I. Introduction

DEEP neural network (DNN) has achieved groundbreaking success in a number of artificial intelligence application areas, including computer vision, object identification, and natural language processing, among others. Despite the remarkable success of DNN, seminal researches reveal its vulnerability to **adversarial examples** (AEs), which are inputs with non-random perturbations unnoticeable to humans but intentionally meant to cause victim models to deliver false outputs [1]–[4]. The harms brought on by AEs have prompted ongoing efforts to increase DNN's **adversarial robustness** (i.e., the accuracy on AEs), among which **adversarial training** (AT) has been largely acknowledged as the most promising defensive strategy [5]–[7].



(a) Adversarial robustness against RAE (b) Adversarial robustness against UAE

Fig. 1. Illustration of adversarial robustness against RAE and UAE. The blue dots and green triangles represent two classes of observed examples, respectively, and the gray ones represent the unobserved examples of the two classes. Each example is surrounded by a box representing its neighborhood with small distance. The purple stars represent RAEs while the red stars UAEs. The yellow curves are decision boundaries obtained by adversarial training.

The fundamental idea of AT was first proposed by Szegedy *et al*. [2], which adds AEs to the training set and uses a min-max game to create a robust classifier that can withstand worst-case attacks. In particular, AT alternately maximizes an adversarial loss incurred by AEs that are purposefully crafted using adversarial attacking methods such as FGSM [8] and PGD [9], and minimizes it by adjusting the target classifier's parameters via a regular supervised training on the augmented training set. Numerous works have suggested strengthening the basic AT in order to increase adversarial robustness [10]–[15]. Although the existing AT approaches have advanced significantly, we argue that the problem of adversarial robustness is still far from being well solved partly due to the following two challenges.

The issue of *Unrestricted Adversarial Example* (UAE) is the first obstacle. As previously mentioned, AT is conducted on a training set that includes AEs. However, the conventional AT approaches are frequently limited to perturbation-based AEs, which are created by adding adversarial perturbations with magnitudes restricted by an l_p norm to observed natural examples. We refer to such kind of AEs as restricted AE (RAE). In fact, RAEs only account for a small portion of possible AEs in real world. Recent works reveal the existence of UAEs, which are dissimilar to any observable example since they are built from scratch rather than by perturbing existing

Lilin Zhang is with the School of Computer Science, Sichuan University, China. E-mail: zhanglilin@stu.scu.edu.cn

Ning Yang is the corresponding author and with the School of Computer Science, Sichuan University, China. E-mail: yangning@scu.edu.cn

Yanchao Sun is with the Department of Computer Science, University of Maryland, College Park, USA. E-mail: ycs@umd.edu

Philip S. Yu is with the Department of Computer Science, University of Illinois at Chicago, USA. E-mail: psyu@uic.edu

examples [16]–[20]. Figure 1 provides an illustration showing how UAEs (represented by the red stars) are not the neighbors of any observed example (blue circle or green triangle), in contrast to RAEs (shown by the purple stars). Despite the adversarial robustness against RAE provided by conventional AT approaches, as seen in Figure 1(a), the classifier (with the yellow curve as the decision boundary) can still be deceived by UAEs because they also appear natural [16]. UAEs are more numerous, more covert, and more dangerous than RAEs since UAEs are generated by perturbing the samples from the learned natural sample distribution, which may or may not be observable. However, the existing works on UAE mainly focus on the generation of UAEs, which suffer from two defects. First, no explanation for why UAEs exist and why they are imperceptible to humans is provided in the existing works. Second, adversarial robustness against UAE still remains largely unexplored. Therefore, we need a new AT method that is able to offer the robustness against both UAE and RAE, which we refer to as **comprehensive adversarial robustness**. As shown in Figure 1(b), the classifier with comprehensive adversarial robustness is able to correctly predict the labels of both the UAEs and the RAEs.

The second challenge faced by the current AT approaches is the problem of the *deterioration of standard generalizability* (i.e., the accuracy on natural/natural testing examples). Early researches show that the standard generalizability of a classifier is sacrificed in order for conventional AT approaches to obtain the adversarial robustness for the classifier [9]. Zhang *et al*. [21] further demonstrate the existence of an upper-bound of the standard generalizability given the adversarial robustness, and Attias *et al*. [22] suggest lowering the upper-bound by increasing sample complexity. However, recent studies argue that the tradeoff may not be inevitable, and adversarial robustness and standard generalizability can coexist without negatively affecting each other [15], [23]–[25]. For example, Stutz *et al*. [15] demonstrate that the adversarial robustness against the AEs existing on the manifold of natural examples will equivalently lead to better generalization. Song *et al*. [24] and Xing *et al*. [25] propose to improve the standard generalizability by introducing robust local features and l_1 penalty to AT, respectively. At the same time, some other works [26]–[28] show that the tradeoff bound can be improved by utilizing unlabeled data in AT. The existing efforts, however, only concentrate on RAEs. It is yet unknown that if and how the comprehensive adversarial robustness and the standard generalizability can both be improved simultaneously.

To address the first challenge, in this paper, we propose *a unique viewpoint that understands UAEs as imperceptibly perturbed unobserved examples* (gray dot or triangle in Figure 1), which logically explains why UAEs are dissimilar to observed examples and why UAEs can fool a classifier without confusing humans. Essentially, RAE can be thought of as a special UAE resulting from perturbing observed examples. If we generalize the concept of UAE to imperceptibly perturbed natural examples, regardless of whether they are observable, then *UAE can be regarded as a general form of AE*, which makes it feasible to provide comprehensive adversarial robustness.

For the second challenge, we find that the tradeoff exists because the target classifier can not generalize over two separated distributions, the AE distribution and the natural example distribution, and the distribution learned by the classifier is a mix of them. Therefore, if we can leverage partial labeled data to align the distributions of UAEs and natural examples with the distribution learned by the target classifier, then the classifier can generalize over UAEs and natural examples without conflict, and we can expect to achieve a better the tradeoff between comprehensive adversarial robustness and standard generalization.

Based on these ideas, we propose a novel AT method called Provable Unrestricted Adversarial Training (PUAT), which *integrates the generation of UAEs and an extended AT on UAEs with a GAN-based framework, so that the distributions of UAEs and natural examples can be aligned with the distribution learned by the target classifier for simultaneously improving its comprehensive adversarial robustness and standard generalizability*. At first, we design a UAE generation module consisting of an attacker A and a conditional generator G, treating a UAE as an adversarially perturbed natural example. Different from the existing generative models for AEs, which often result in suboptimal AEs due to the gradient conflicts during the single optimization for both imperceptibility and harmfulness of AEs, our UAE generation module decouples the optimizations of imperceptibility and harmfulness of UAEs. Particularly, A is in charge of producing an adversarial perturbation that can fool the classifier, based on which G will generate a UAE for a certain class, which is equivalent to perturbing a natural example. PUAT ensures the imperceptibility of UAEs by aligning the UAE distribution learned by G with the natural example distribution via a G-C-D GAN, in contrast to conventional AT approaches that accomplish imperceptibility by an explicit norm constraint on the perturbations. The G-C-D GAN is a novel augmented triple-GAN that combines two conditional GANs: the G-D GAN, which consists of G and a discriminator D, and the C-D GAN, which consists of the target classifier C and D. Different from the original triple-GAN proposed by Li *et al*. [29], in G-C-D GAN the generator G is shared with the UAE generation module, which augments the input of G with the output of the attacker A and consequently allows G to draw UAEs from the UAE distribution. To facilitate the distributional alignment, we also make use of unlabeled data to create pseudo-labeled data for the training of G-C-D GAN since labeled data are usually insufficient for capturing the real data distribution. By leveraging partially labeled data, G-D GAN and C-D GAN can cooperate to help G capture the natural data distribution and the UAE generation module understand how to build a legitimate AE for a certain class.

In order to simultaneously improve the target classifier's comprehensive adversarial robustness and standard generalizability, PUAT conducts an extended AT between the target classifier C and the attacker A, which incorporates a supervised loss of the target classifier with the adversarial loss. The extended AT together with the G-C-D GAN leads to the alignment between the UAE distribution, the natural example distribution, and the distribution learned by C, which

enables C to generalize without conflict on UAEs and natural examples. Finally, we theoretically and empirically demonstrate that PUAT realizes a consistent optimization for both the comprehensive adversarial robustness and the standard generalizability, eliminating the tradeoff between them and resulting in their mutual advantage. Our contributions can be summarized as follows:

- We propose a unique viewpoint that understands UAEs as imperceptibly perturbed unobserved examples, which offers a logical and demonstrable explanation for UAEs' discrepancies from observed examples, imperceptibility to humans, and the feasibility of comprehensive adversarial robustness against both RAE and UAE.
- We propose a novel AT approach called Provable Unrestricted Adversarial Training (PUAT), which, with theoretical guarantee, can simultaneously increase a classifier's comprehensive adversarial robustness and standard generalizability through a distributional alignment.
- We provide a solid theoretical analysis to demonstrate that PUAT can eliminate the tradeoff between the comprehensive adversarial robustness and the standard generalizability through the distributional alignment.
- The extensive experiments conducted on widely adopted benchmarks verify the superiority of PUAT.

II. Preliminaries and Problem Formulation

In this section, we first give the formal definition of UAE, then present our understanding of UAEs, and finally, formulate the AT on UAEs.

A. Definition of UAE

Let $x \in \mathcal{X}$ and $y \in \mathcal{Y}$ be an example and a label, respectively, where $\mathcal{X}$ represents the data space with distribution $P(x)$ while $\mathcal{Y}$ the label space with distribution $P(y)$. Let $o(\cdot) : \mathcal{O} \subseteq \mathcal{X} \to \mathcal{Y}$ be an oracle which is defined on its domain $\mathcal{O}$ and tells the true label of an example $x \in \mathcal{O}$. Usually, the oracle o corresponds to humans and $\mathcal{O}$ consists of all the samples that look natural to and can not confuse humans. Let $C(\cdot) : \mathcal{X} \to \mathcal{Y}$ be the target classifier. Following the idea of [16], we define UAE as follow:

Definition 1 (UAE)**.** *For a well-trained classifier C, an example $\tilde{x}$ with label $y = o(\tilde{x})$ is an unrestricted adversarial example if $\tilde{x} \sim P(x|y)$ and $C(\tilde{x}) \neq y$, where $P(x|y)$ is the class likelihood.*

B. Understanding of UAE

From Definition 1 we can see that similar to RAE, the harmfulness of a UAE is still defined by its ability to fool the classifier, i.e., $C(\tilde{x}) \neq y$. But in contrast to RAE, the imperceptibility of a UAE is due to its capability of being drawn from the realistic distribution ($\tilde{x} \sim P(x|y)$) rather than a norm-constraint. However, it is challenging how to enforce both imperceptibility and harmfulness for UAE. For this purpose, *we understand UAE from the viewpoint of imperceptibly perturbed observed and unobserved examples*. Inspired by [30], for a natural sample $x \sim P(x|y)$, there exists a mapping $T : \mathcal{X} \times \mathcal{Y} \to \mathcal{X}$ to generate a UAE $\tilde{x} = T(x, y)$ satisfying the imperceptibility $\tilde{x} \in supp(P(x|y))$, and the harmfulness $C(\tilde{x}) \neq y$, where $supp()$ represents the support set. Following this idea, we can generate a UAE through two steps: drawing natural example and generating appropriate perturbation for it. It is noteworthy that such understanding brings the advantage that RAE can be viewed as a special UAE and UAEs are a superset of RAEs, thus enabling the adversarial robustness against UAEs to cover RAEs.

C. Adversarial Training on UAEs

To offer the comprehensive adversarial robustness to the target classifier, we want to conduct adversarial training on UAEs, which is defined as the following min-max game:

$$\min_{C} \max_{P_T} \mathbb{E}_{(\tilde{x},y)\sim P_T(x,y)} l\big(\tilde{x}, y; C\big), \tag{1}$$

where $l(x, y; C)$ is a loss function of the target classifier C, e.g., cross-entropy, $\tilde{x} = T(x, y)$ is a UAE with label y, and $P_T(x, y)$ is the labeled UAE distribution. In Equation (1), the inner maximization seeks the most harmful UAE set resulting in the poorest performance of the target classifier, while the outer minimization aims at adjusting the target classifier to reduce its sensitivity to UAEs. It is noteworthy that unlike traditional AT, where AEs are searched by explicitly perturbed the observed natural samples with respect a norm-constraint, the AT defined by Equation (1) is conducted on UAEs which are perturbed versions of not only observed natural samples but also unobserved ones, leading to superior comprehensive adversarial robustness to traditional AT methods as shown by later theoretical analysis and experimental verification. To fulfill the AT on UAEs, the key challenge is how to learn the adversarial distribution $P_T(x, y)$ that can reconcile the adversarial robustness and standard generalizability of the target classifier. To overcome this challenge, our general idea is to fulfill the learning of distribution $P_T(x, y)$ with two steps: (1) approximating the distribution $P(x, y)$ of the natural samples, and (2) learning the mapping T, which converts the inner maximization of Equation (1) to

$$\max_{T} \mathbb{E}_{(x,y)\sim P(x,y)} l\bigg(\big(\tilde{x} = T(x, y), y\big); C\bigg). \tag{2}$$

When T achieves the optimal solution of Equation (2), $P_T(x, y)$ converges to the AE distribution $P(x, y|C(x) \neq y)$.

III. Proposed Method

In this section, we first give an overview of our Provable Unrestricted Adversarial Training (PUAT) method, and then describe its components and learning procedure in detail.

A. Overview of PUAT

As shown in Figure 2, PUAT consists of four components, an attacker A, a conditional generator G, a discriminator D and the target classifier C. According to our idea that a UAE can be regarded as the perturbed version of an observed or unobserved natural sample, we first use G to produce a plausible natural sample $(x_g, y_g) \sim P_G(x, y)$, where $P_G(x, y)$

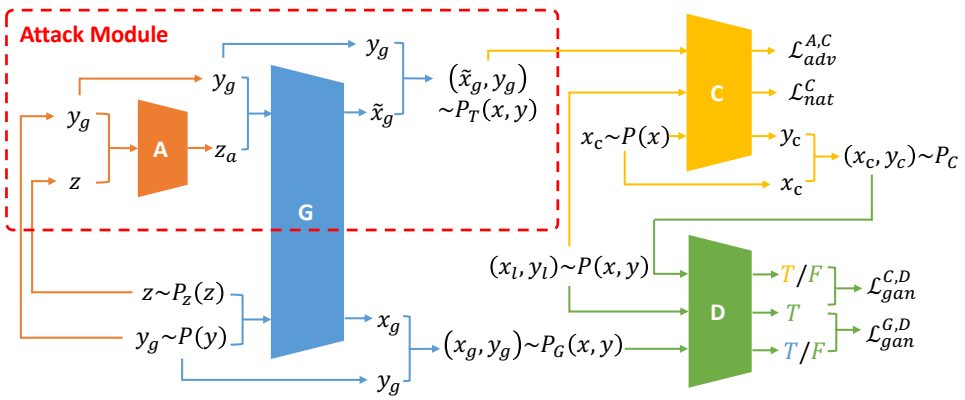


Fig. 2. Architecture of PUAT. The attacker A seeks the perturbation z_a, and the generator G synthesizes the UAEs $\{(\tilde{x}_g, y)\}$ based on the perturbation and the specified class y. In G-D GAN, the discriminator D aims to adversarially distinguish the unobserved natural examples $\{(x_g, y)\}$ generated by G from the true natural examples $\{(x, y)\}$ with loss $\mathcal{L}_{gan}^{G,D}$, while in C-D GAN, D aims to adversarially distinguish the pseudo labeled examples $\{(x, y_c)\}$ generated by the target classifier C from $\{(x, y)\}$ with loss $\mathcal{L}_{gan}^{C,D}$. The extended AT is conducted between A and C with the adversarial loss $\mathcal{L}_{adv}^{C,A}$ plus the supervised loss $\mathcal{L}_{nat}^{C}$.

is the distribution of the data-label pairs generated by G and the randomness stems from a white noise z. Then we invoke A and G to generate its perturbed version $(\tilde{x}_g, y_g)$ as a UAE, where the perturbation z_a is produced by A based on the same z. Note that conceptually, A and G together play the role of afore-mentioned mapping T to generate a UAE from the distribution $P_T(x, y)$, which we term as **attack module**, and in other words, $P_T(x, y)$ is exactly the distribution captured by A together with G.

To ensure the **harmfulness** of the UAEs, PUAT will try to adjust A via the AT between A and C so that the generated UAEs can fool the target classifier C, i.e. $C(\tilde{x}_g) \neq y_g$. At the same time, to ensure the **imperceptibility** of the generated UAEs, PUAT will align the distribution $P_G(x, y)$ with the true data distribution $P(x, y)$. For this purpose, we introduce a discriminator D to distinguish the examples $\{(x_g, y_g)\}$, which includes the UAEs $\{(\tilde{x}_g, y_g)\}$, from the true labeled examples $\{(x_l, y_l) \sim P(x, y)\}$, which together with G constitutes a class-conditional GAN, called G-D GAN. The optimization of the G-D GAN will adjust G so that the distribution $P_G(x, y)$ converges to the natural sample distribution $P(x, y)$. The alignment of $P_G(x, y)$ and $P(x, y)$ will make a UAE drawn from $P_G(x, y)$ look like a real natural sample, which results in the imperceptibility of UAEs. Finally, it is noteworthy that the harmfulness and imperceptibility offered by PUAT cause the UAEs essentially follow the conditional distribution $P(x, y|C(x) \neq y)$.

However, in traditional conditional GAN, the discriminator D also plays the role to predict the class label of x_g, for which the optimization objective conflicts with that for distinguishing $\{(x_g, y_g)\}$ from $\{(x_l, y_l)\}$. Such conflict will lead to a suboptimal D [29], [31], and consequently weaken the ability of the G-D GAN to accurately align $P_G(x, y)$ with $P(x, y)$. To alleviate this issue, we introduce another conditional GAN, called C-D GAN, which together with the G-D GAN constitutes the G-C-D GAN. The C-D GAN consists of the target classifier C and the discriminator D, where C aims at predicting the label y_c for unlabeled examples $\{x_c \sim P(x)\}$ to build pseudo-labeled examples $\{(x_c, y_c)\}$ that can fool D, while D tries to distinguish $\{(x_c, y_c)\}$ from true examples $\{(x_l, y_l)\}$. The adversarial game between C and D improves the ability of D to identify whether a label matches an example x. As D's role in C-D GAN is consistent with its role in G-D GAN, D strengthened by C-D GAN will ultimately help G capture the true data distribution $P(x, y)$ more accurately.

At last, to realize the **comprehensive adversarial robustness** of the target classifier C, we want to align the $P_G(x, y)$ with the distribution $P_C(x, y)$ learned by C, which equivalently improves the robust generalizability of C on UAEs. For this purpose, we conduct AT between the attacker A and C, with an adversarial loss $\mathcal{L}_{adv}^{A,C}$ over the generated UAEs. At the same time, to simultaneously improve the **standard generalizability** of C on natural examples, we further enforce the distribution $P_C(x, y)$ to be aligned with the true data distribution $P(x, y)$, by extending the AT with minimization of the supervised loss $\mathcal{L}_{sup}^{C}$ of C over the labeled data $\{(x_l, y_l)\}$. The overall logic of PUAT can be summarized as follows:

(1) For the imperceptibility of UAEs, $P_G(x, y)$ is aligned with the true data distribution $P(x, y)$, denoted by $P_G(x, y) = P(x, y)$, via the G-C-D GAN.
(2) The harmfulness of UAEs is offered by the maximization of $\mathcal{L}_{adv}^{A,C}$ during the AT.
(3) The adversarial robustness is offered by $P_C(x, y) = P_T(x, y)$, via the inner minimization of $\mathcal{L}_{adv}^{A,C}$ of the AT between A and C.
(4) The standard generalizability is offered by $P_C(x, y) = P(x, y)$, via the minimization of $\mathcal{L}_{sup}^{C}$.

It is noteworthy that traditional AT methods only realize (3) and (4), which causes $P_C(x, y)$ to be a mix of $P_T(x, y)$ and $P(x, y)$ and consequently incurs an inferior tradeoff between adversarial robustness and standard generalizability. In sharp contrast with the traditional methods, PUAT improves the tradeoff because (1) together with (3) and (4) results in the consistency between adversarial robustness and standard generalizability at $P_C(x, y) = P_G(x, y) = P(x, y)$, which

will be proved later.

B. UAE Generation

Our idea to generate a UAE $(\tilde{x}_g, y_g)$ is to perturb a generated natural example. Let $P(y)$ be the label distribution and $P_z(z)$ be a standard normal distribution. At first, we sample a label $y_g \sim P(y)$, and a seed noise $z \sim P_z(z)$. Then we invoke G to generate an example $(x_g, y_g) \sim P_G(x, y)$, where

$$x_g = G(z, y_g). \tag{3}$$

Note that since the label distribution $P(y)$ is unknown, we use the labels appearing in the labeled dataset $\mathcal{D}_l = \{(x_l, y_l) \sim P(x, y)\}$ to approximate it. In particular, the label y_l of each example in $\mathcal{D}_l$ will serve as a y_g once.

To generate the UAE corresponding to (x_g, y_g), we further feed (z, y_g) into the attacker A to produce the perturbation,

$$z_a = A(z, y_g). \tag{4}$$

Next, z_a together with y_g is fed into the generator G to produce perturbed data

$$\tilde{x}_g = G(z_a, y_g). \tag{5}$$

Finally, $\tilde{x}_g$ and y_g together form a labeled UAE $(\tilde{x}_g, y_g)$.

It is noteworthy that that (x_g, y_g) can be regarded as an unobserved natural example if $P_G(x, y)$ is aligned with the true data distribution $P(x, y)$. For each (x_g, y_g), we will invoke the attack module (A and G) generate a corresponding UAE $(\tilde{x}_g, y_g)$, which is actually the perturbed (x_g, y_g) as they are generated based on the same (z, y_g) with the only difference that whether z or its perturbed version $z_a = A(z, y_g)$ is fed into G. At the same time, as we will see later, during the AT between A and C, A will be adjusted to ensure $(\tilde{x}_g, y_g)$ be an adversarial example that is able to attack the classifier C, i.e., $C(\tilde{x}_g) \neq y_g$. Therefore, the UAEs $\{(\tilde{x}_g, y_g)\}$ are a subset of $\{(x_g, y_g)\}$ such that $(\tilde{x}_g, y_g) \sim P(x, y|C(x) \neq y)$, and essentially, A plays the role of finding out $\{(\tilde{x}_g, y_g)\}$ from $\{(x_g, y_g)\}$.

C. Distribution Alignment for Imperceptibility

As mentioned before, the imperceptibility of UAEs results from the alignment of the distribution $P_G(x, y)$ with the true data distribution $P(x, y)$, which is realized through the cooperation of G-D GAN and C-D GAN. Let $\mathcal{D}_l = \{(x_l, y_l) \sim P(x, y)\}$ be a labeled natural dataset, where $y_l = o(x_l)$, and $\mathcal{Z}$ be a set of sampled noises $z \sim P_z(z)$. Then the optimization objective of G-D GAN is defined as

$$\min_G \max_D \mathcal{L}_{gan}^{G,D} = \mathcal{L}_D + \mathcal{L}_G, \tag{6}$$

where

$$\begin{aligned} \mathcal{L}_D &= \mathbb{E}_{(x_l, y_l) \sim P(x,y)} D(x_l, y_l) \\ &\approx \hat{\mathcal{L}}_D = \frac{1}{|\mathcal{D}_l|} \sum_{(x_l, y_l) \in \mathcal{D}_l} D(x_l, y_l), \end{aligned} \tag{7}$$

$$\begin{aligned} \mathcal{L}_G &= \mathbb{E}_{(x_g, y_g) \sim P_G(x,y)} \left[-D(x_g, y_g)\right] \\ &\approx \hat{\mathcal{L}}_G = \frac{1}{|\mathcal{D}_l||\mathcal{Z}|} \sum_{y_l \in \mathcal{D}_l} \sum_{z \in \mathcal{Z}} \left[-D\Big(G\big(z, y_l\big), y_l\Big)\right]. \end{aligned} \tag{8}$$

Note that in Equations (7) and (8), the first line is the population loss defined on the overall distribution, while the second line is the empirical loss defined on training set which approximates the population loss. In Equation (8), the label y_l of each example $(x_l, y_l) \in \mathcal{D}_l$ serves as $y_g \sim P(y)$. For each y_l, we first sample a noise $z \sim P_z(z)$ and then invoke $G\big(z, y_l\big)$ to generate x_g, which together with y_l forms a data-label pair (x_g, y_g).

In G-D GAN, the generation of a data-label pair (x_g, y_g) can be understood as the procedure of first drawing $y_g \sim P(y)$ and then drawing $x_g|y_g \sim P_G(x|y)$, where $P_G(x|y)$ is the conditional distribution characterized by G. Therefore, $(x_g, y_g) \sim P_G(x, y) = P(y)P_G(x|y)$. As y_g is drawn from the true label distribution $P(y)$, to adversarially distinguish (x_g, y_g) from true examples, what D should do is to identify whether $x_g|y_g$ is true. There is one of two reasons causing a false $x_g|y_g$. One is x_g is false, which asks D to identify whether $x_g \sim P(x)$. The other is although x_g is true, it does not belong to class y_g, which asks D to identify whether $y_g|x_g \sim P(y|x)$. However, as we have mentioned before, these two tasks conflict with each other for D, which leads to a suboptimal D and consequently weakens G. To alleviate this issue, we introduce C-D GAN and can use an unlabeled natural dataset $\mathcal{D}_c = \{x_c \sim P(x)\}$ to train it with the following optimization objective:

$$\min_C \max_D \mathcal{L}_{gan}^{C,D} = \mathcal{L}_D + \mathcal{L}_C, \tag{9}$$

where

$$\begin{aligned} \mathcal{L}_C &= \mathbb{E}_{(x_c, y_c) \sim P_C(x,y)} \left[-D(x_c, y_c)\right] \\ &\approx \hat{\mathcal{L}}_C = \frac{1}{|\mathcal{D}_c|} \sum_{x_c \in \mathcal{D}_c} \left[-D\big(x_c, C(x_c)\big)\right]. \end{aligned} \tag{10}$$

To calculate $\mathcal{L}_C$, for each $x_c \in \mathcal{D}_c$, we invoke the target classifier C to label it with the softmax vector $y_c = C(x_c) \in \mathbb{R}^{|\mathcal{Y}|}$. This is equivalent to the procedure of first drawing $x_c \sim P(x)$ and then drawing $y_c|x_c \sim P_C(y|x)$. Therefore $(x_c, y_c) \sim P_C(x, y) = P(x)P_C(y|x)$. C-D GAN offers two benefits to D. The first is in C-D GAN, C provides the adversarial signal $y_c|x_c$ to help D learn $P(y|x)$, which ultimately promotes D's performance in G-D GAN for a better G. Thus in both C-D GAN and G-D GAN, D plays the same role of distinguishing the generated examples ($\{(x_g, y_g)\}$ or $\{(x_c, y_c)\}$) from true ones ($\{(x_l, y_l)\}$), which eliminates the conflict optimization of D in traditional conditional GAN. The second is by C-D GAN, unlabeled data can be leveraged to accurately approximate $P(x, y)$.

We refer to the combination of G-D GAN and C-D GAN as G-C-D GAN. By combining Equations (6) and (9), we obtain the following optimization objective for G-C-D GAN:

$$\min_{C,G} \max_D \mathcal{L}_{gan}^{G,C,D} = \mathcal{L}_D + \frac{1}{2}\mathcal{L}_G + \frac{1}{2}\mathcal{L}_C. \tag{11}$$

D. Extended Adversarial Training

Since the harmfulness of the generated UAEs is caused by the attacker A, we conduct the AT between A and the target classifier C. Meanwhile, to improve both the adversarial robustness and standard generalizability, we extend the AT

defined in Equation (1) by incorporating the loss of C on natural data, which leads to the following min-max game:

$$\min_{C} \big[\mathcal{L}_{nat}^{C} + \lambda \max_{A} \mathcal{L}_{adv}^{C,A}\big], \tag{12}$$

where λ is a non-negative constant controlling the weight of adversarial robustness and the standard generalizability. $\mathcal{L}_{nat}^{C}$ is the loss of target classifier C over natural samples. To benefit from unlabeled natural samples, here we introduce a consistent loss term offered by a weight-averaged classifier, following the idea of the semi-supervised algorithm Mean Teacher [32], which leads to the following definition:

$$\begin{aligned}\mathcal{L}_{nat}^{C} =& \mathbb{E}_{(x,y)\sim P(x,y)}\big[-\log P_C(y|x)\big] \\ &+ \alpha \mathbb{E}_{x\sim P(x)}\big[\|P_C(y|x) - P_{C'}(y|x)\|^2\big] \\ \approx& \hat{\mathcal{L}}_{nat}^{C} = \frac{1}{|\mathcal{D}_l|}\sum_{(x_l,y_l)\in\mathcal{D}_l}\big[-\log P_C(y_l|x_l)\big] \\ &+ \alpha\frac{1}{|\mathcal{D}_c|}\big[\|P_C(y|x_c) - P_{C'}(y|x_c)\|^2\big],\end{aligned} \tag{13}$$

where the conditional probability $P_C(y|x)$ is the output of C, C' is the weight-averaged classifier, and α is the weight of consistency cost.

The adversarial loss is defined as

$$\begin{aligned}\mathcal{L}_{adv}^{C,A} &= \mathbb{E}_{(\tilde{x},y)\sim P_T(x,y)}[-\log P_C(y|\tilde{x})] \\ &\approx \hat{\mathcal{L}}_{adv}^{C,A} = \frac{1}{|\mathcal{D}_l||\mathcal{Z}|}\sum_{y_l\in\mathcal{D}_l}\sum_{z\in\mathcal{Z}}\big[-\log P_C(y_l|\tilde{x})\big],\end{aligned} \tag{14}$$

where $P_T(x,y)$ is the UAE distribution captured by the attack module, and $\tilde{x}$ is the UAE generated by $\tilde{x} = G(A(z,y_l),y_l)$. Note that for each $(x,y)\sim P_G(x,y)$, we generate its corresponding UAE $(\tilde{x},y)$ by invoking Equations (4) and (5) (see Section III-B), i.e., $\tilde{x} = G(A(z,y),y)$.

In Equation (12), the inner maximization of $\mathcal{L}_{adv}^{C,A}$ enforces A to be adjusted so that a generated UAE $(\tilde{x},y)$ is harmful enough, i.e., $C(\tilde{x}) \neq y$, which causes the attack module (consisting of A and G) to converge to the underlying UAE distribution $P(x,y|C(x)\neq y)$ which maximizes the adversarial loss. The outer minimization realizes the alignment $P_C(x,y) = P_T(x,y)$ by adjusting C, which results in C's robust generalizability over UAEs (i.e. the adversarial robustness). As proved later, the KL-divergence of $P_C(x,y)$ and $P_T(x,y)$ is the upper bound of the KL-divergence of $P_C(x,y)$ and $P_G(x,y)$. Therefore, the minimization of $\mathcal{L}_{adv}^{C,A}$ will approximately lead to $P_C(x,y) = P_G(x,y)$. At the same time, one can note that the minimization of $\mathcal{L}_{nat}^{C}$ will be achieved when $P_C(x,y) = P(x,y)$, which offers C the standard generalizability on natural examples. Later we will also prove that with the help of the G-C-D GAN (Equation(11)), the extended AT defined by Equation(12) will achieve the optimal solution at $P_C(x,y) = P_G(x,y) = P(x,y)$, which means the adversarial robustness and the standard generalizability are satisfied simultaneously.

E. Joint Training

By combining Equations (11) and (12), we can obtain the following overall optimization objective for joint learning of A, G, C and D:

$$\min_{C,G}\max_{A,D}\big[\mathcal{L}_{nat}^{C} + \lambda\mathcal{L}_{adv}^{C,A} + \gamma\mathcal{L}_{gan}^{G,C,D}\big], \tag{15}$$

where λ and γ are weight factors. Basically, Equation (15) integrates the adversarial generation of UAEs and the AT over UAEs within a unified min-max game. Algorithm 1 gives the sketch of the joint training, where the Lines 6 and 7 solve the inner maximization with stochastic gradient ascent, while the Lines 8 and 9 solve the outer minimization with stochastic gradient descent.

Algorithm 1 Training Algorithm of PUAT

Input: Semi-supervised datasets $\mathcal{D}$;
Output: Well-trained target classifier C;
1: Random initialize A, C, G, and D;
2: Pre-trained C, G, and D for T_{pre} epochs;
3: **for** T epochs **do**
4: Generate UAEs $\{(\tilde{x}_g,y)\}$ using Equations (4) and (5).
5: Generate pseudo-data pairs $\{(x_g,y)\}$ w.r.t. Equation (8).
6: Generate pseudo-label pairs $\{(x,y_c)\}$ w.r.t. Equation (10).
7: Update D by **ascending** stochastic gradient $\nabla_D(\mathcal{L}_{gan}^{C,D} + \mathcal{L}_{gan}^{G,D})$;
8: Update A by **ascending** stochastic gradient $\nabla_A(\mathcal{L}_{adv}^{C,A})$;
9: Update C by **descending** stochastic gradient $\nabla_C(\mathcal{L}_{nat}^{C} + \lambda\mathcal{L}_{adv}^{C,A} + \gamma\mathcal{L}_{gan}^{C,D})$ and use C to update C';
10: Update G by **descending** stochastic gradient $\nabla_G(\mathcal{L}_{gan}^{G,D})$;
11: **end for**

IV. Theoretical Justification

In this section, we will theoretically justify: 1) PUAT's ability to simultaneously improve adversarial robustness and standard generalizability of the target classifier without compromising either of them; 2) superiority of UAE to RAE. We first discuss the ideal setting with infinite labeled data available for training, and then the practical setting with finite labeled data.

A. Infinite Data

In this section, we discuss the population loss which is defined over the overall distribution $P(x,y)$, i.e., dataset $\mathcal{D}_l = \mathcal{O}$. For the brevity of later derivations, in the following text $P(x,y)$, $P_G(x,y)$, $P_T(x,y)$ and $P_C(x,y)$ will be abbreviated to P, P_G, P_T and P_C, respectively, when the context is unambiguous.

Recall that PUAT offers the adversarial robustness by $P_C = P_T$, while the standard generalizability by $P_C = P$. As Equation (15) defines a unified framework integrating the UAE generation with the AT against UAEs, proving PUAT's ability to eliminate the tradeoff is equivalent to proving that the optimal solution of Equation (12) will be achieved without any conflict, which means the adversarial robustness and the standard generalizability agree with each other under the

framework of Equation (15). We first give the equilibrium of the G-C-D GAN by the following lemma:

Lemma 1. *The optimal solution of* $\min_{C,G}\max_{D}\mathcal{L}_{gan}^{G,C,D}$ *(Equation (11)) is achieved at* $P(x,y) = (P_G(x,y) + P_C(x,y))/2$.

The proof of Lemma 1 can be found in Appendix A. Now we show that $\min_C \mathcal{L}_{nat}^C$ and $\min_C\max_A \mathcal{L}_{adv}^{C,A}$ are two consistent optimization problems with the same optimal solution C^*.

Theorem 1. *In Equation (12),* $\min_C \mathcal{L}_{nat}^C$ *and* $\min_C\max_A \mathcal{L}_{adv}^{C,A}$ *can be achieved at the same optimal* C^* *subjected to* $P_{G^*} = P_{C^*} = P$.

Proof. (1) By the definition of $\mathcal{L}_{nat}^C$ (Equation (13)), the optimal C for $\min_C \mathcal{L}_{nat}^C$ is

$$\begin{aligned} C_{nat}^* &= \operatorname*{argmin}_C \mathcal{L}_{nat}^C \\ &= \operatorname*{argmin}_C \iint P(x,y)\log\frac{P(x)P(x,y)}{P_C(x,y)P(x,y)}\,\text{dxdy} \\ &= \operatorname*{argmin}_C \iint P(x,y)\log\frac{P(x,y)}{P_C(x,y)}\,\text{dxdy} + H(y|x) \\ &= \operatorname*{argmin}_C \mathrm{KL}\big(P\|P_C\big), \end{aligned}$$

where KL is Kullback–Leibler divergence, $H(y|x)$ is the conditional entropy that independent of C, and the second equality holds because $P_C(x,y) = P(x)P_C(y|x)$. Since many works have proved that consistency loss can help classification, we simply ignore the consistency term here, i.e., we take $\alpha = 0$. Therefore, $C_{sup}^* = \operatorname{argmin}_C \mathcal{L}_{nat}^C$ is achieved at the KL divergence equals zero,

$$P_{C_{nat}^*} = P.$$

(2) Let $A^* = \operatorname{argmax}_A \mathcal{L}_{adv}^{C,A}$ be the optimal A and then fix it, we first analysis C according to Equation (14),

$$\begin{aligned} C_{adv}^* &= \operatorname*{argmin}_C \mathcal{L}_{adv}^{C,A^*} \\ &= \operatorname*{argmin}_C \iint P_T(x,y)\log\frac{P(x)P_T(x,y)}{P_C(x,y)P_T(x,y)}\,\text{dxdy} \\ &= \operatorname*{argmin}_C \iint P_T(x,y)\log\frac{P_T(x,y)}{P_C(x,y)}\,\text{dxdy} \\ &= \operatorname*{argmin}_C \mathrm{KL}\big(P_T\|P_C\big), \end{aligned}$$

in which P_T is the perturbed sample distribution. Obviously, $\mathrm{KL}\big(P_T\|P_C\big) \geq \mathrm{KL}\big(P_G\|P_C\big) \geq 0$, which indicates that $\min_C \mathcal{L}_{adv}^{C,A^*}$ essentially minimizes the upper-bound of the KL-divergence of P_G and P_C. Therefore, $C_{adv}^* = \operatorname{argmin}_C \mathcal{L}_{adv}^{C,A^*}$ can be achieved at

$$P_{C_{adv}^*} = P_G.$$

(3) The part (1) and part (2) together tell us that if P and P_G are two different distributions, the standard generalizability and adversarial robustness of the target classifier C will be at odd, i.e., $C_{sup}^* \neq C_{adv}^*$, and C will converge to a solution that depends on the tradeoff. But conversely, if we simultaneously adjust G by G-C-D GAN (Equation (11)), the tradeoff will be eliminated. Combining Lemma 1 leads to the same result regardless of C_{nat}^* or C_{adv}^*,

$$P_{G^*} = 2P - P_{C^*} = P.$$

equationTherefore, minimizing $\mathcal{L}_{nat}^C$ and $\max_A \mathcal{L}_{adv}^{C,A}$ for C are consistent optimization problems with the same optimal solution C^* subjected to $P_{C^*} = P = P_{G^*}$. □

Remark 1. *Theorem 1 shows that* $\mathcal{L}_{adv}^{C,A}$ *has no conflict with* $\mathcal{L}_{nat}^C$*, which further implies adversarial robustness is not necessarily at odd with standard generalization, and our PUAT can achieve adversarial robustness without compromising standard accuracy.*

Theorem 1 tells us why PUAT can consistently optimize the standard generalization by applying UAEs to the AT. Now we further compare the adversary of UAE and RAE. When the labeled data is infinite, the adversary of RAEs is defined by the target classifier's loss they incurred over P, i.e.,

$$\mathbb{E}_{(x,y)\sim P}\max_{\hat{x}:\|\hat{x}-x\|_p\leq\epsilon} l(\hat{x},y;C),$$

where ϵ is perturbation budget, and $\|\cdot\|_p$ is p-norm. According to Equation (2), the adversary of the UAEs satisfying the same perturbation budget is

$$\max_{T:\|T(x)-x\|_p\leq\epsilon}\mathbb{E}_{(x,y)\sim P} l(T(x),y;C).$$

We have the following theorem:

Theorem 2. *For a target classifier* C *with loss function* l*, the UAE adversary is equivalent to the RAE adversary under the same perturbation budget, i.e.,*

$$\begin{aligned} &\max_{T:\|T(x)-x\|_p\leq\epsilon}\mathbb{E}_{(x,y)\sim P} l(T(x),y;C) \\ =&\mathbb{E}_{(x,y)\sim P}\max_{\hat{x}:\|\hat{x}-x\|_p\leq\epsilon} l(\hat{x},y;C). \end{aligned}$$

Proof. For simplicity, the constrains $\|\tilde{x}-x\|_p \leq \epsilon$ and $\|\ T(x) -x\|_p \leq \epsilon$ are omitted in this proof hereinafter. At first, one can note that proving Theorem 2 is equivalent to proving the proposition that if $T^* = \operatorname{argmax}_T \mathbb{E}_{(x,y)\sim P} l(T(x),y;C)$, then $\forall(x,y)\sim P$,

$$l(T^*(x),y;C) = \max_{\hat{x}} l(\hat{x},y;C).$$

Equivalently, we prove its contrapositive, i.e., if there exists a sample $(x_k,y_k) \sim P$ making $l(T_1(x_k),y_k;C)<\max_{\hat{x}_k} l(\hat{x}_k,y_k;C)$, then $T_1\neq\operatorname{argmax}_T \mathbb{E}_{(x,y)\sim P} l(T(x),y;C)$, i.e., $\mathbb{E}_{(x,y)\sim P} l(T_1(x),y;C)<\mathbb{E}_{(x,y)\sim P} l(T^*(x),y;C)$. Note that it is impossible that $l(T_1(x_k),y_k;C)>\max_{\hat{x}_k} l(\hat{x}_k,y_k;C)$ because $T_1(x_k)$ and $\hat{x}_k$ are AEs in the same neighborhood of x_k and $\max_{\hat{x}_k} l(\hat{x}_k,y_k;C)$ is the maximal loss incurred by the adversarial examples in this neighborhood.

$$\begin{aligned} &\mathbb{E}_{(x,y)\sim P} l(T_1(x),y;C) \\ =&\sum_{i\neq k} P(x_i,y_i) l(T_1(x_i),y_i;C) + P(x_k,y_k) l(T_1(x_k),y_k;C). \end{aligned}$$

Since for $\forall i \neq k$, $l(T_1(x_i), y_i; C) \leq \max_{\hat{x}_i} l(\hat{x}_i, y; C)$, and $l(T_1(x_k), y; C) < \max_{\hat{x}_k} l(\hat{x}_k, y; C)$, we have

$$\begin{aligned}\mathbb{E}_{(x,y)\sim P} l(T_1(x), y; C) &< \sum_i P(x_i, y_i) \max_{\hat{x}_i} l(\hat{x}_i, y_i; C) \\ &= \mathbb{E}_{(x,y)\sim P} \max_{\hat{x}} l(\hat{x}, y; C).\end{aligned}$$

Let another mapping $T_2(x) = \operatorname{argmax}_{\hat{x}} l(\hat{x}, y; C)$, and then $\mathbb{E}_{(x,y)\sim P} \max_{\hat{x}} l(\hat{x}, y; C) = \mathbb{E}_{(x,y)\sim P} l(T_2(x), y; C)$. Hence

$$\mathbb{E}_{(x,y)\sim P} l(T_1(x), y; C) < \mathbb{E}_{(x,y)\sim P} l(T_2(x), y; C),$$

and thus $T_1 \neq \operatorname{argmax}_T \mathbb{E}_{(x,y)\sim P} l(T(x), y; C)$. □

Remark 2. *Theorem 2 demonstrates that why the AT over UAEs can offer the adversarial robustness against RAEs. In summary, Theorem 1 and Theorem 2 explain why PUAT can consistently optimize the standard generalization and adversarial robustness. These together form the theoretical foundation of the tradeoff improvement by PUAT.*

B. Finite Data

Now we give the theoretical results of PUAT in the practical case where data is finite. Specifically, we will establish the quantitative relationship between the generalizability of PUAT and the amount of the semi-supervised training dataset $|\mathcal{D}_c \cup \mathcal{D}_l|$. At first, we give the generalization error bound of G-C-D GAN when it is optimized over finite data.

1) The Generalization Error Bound of G-C-D GAN: The following lemma shows when the G-C-D GAN is optimized over finite data (i.e., using the empirical loss $\hat{\mathcal{L}}_{gan}^{G,C,D} = \hat{\mathcal{L}}_D + \frac{1}{2}\hat{\mathcal{L}}_C + \frac{1}{2}\hat{\mathcal{L}}_G$), the relationship between the generalization error of G-C-D GAN, i.e., how close the distribution P_{GC} learned by G and C is to P (which is measured by TV (P, P_{GC})), and the amount of data.

Lemma 2. *Let $\mathcal{Z}$ be the set of sampled noise $z \sim P_z(z)$, $m = |\mathcal{D}_l|$, $n = |\mathcal{D}_c|$, and $b = \| \log \frac{P}{P_{GC}} + 1 \|_\infty = \max_{(x,y)} \log \frac{P}{P_{GC}} + 1$. For any G, C and $0 < \delta < 1$, if $|\mathcal{Z}| \to \infty$, TV$(P, P_{GC}) \leq B$ holds with probability at least $(1-\delta)^2$, where $B = \left(\frac{1}{2} b + \frac{1}{4} \max_D \hat{\mathcal{L}}_{gan}^{G,C,D} + \sqrt{\frac{\log \frac{1}{\delta}}{8m}} + \sqrt{\frac{\log \frac{1}{\delta}}{32n}} \right)^{\frac{1}{2}}$.*

The proof of Lemma 2 can be found in Appendix A. It is noteworthy that in Lemma 2 the condition $|\mathcal{Z}| \to \infty$ is reasonable since we can always generate arbitrarily many $z \sim P_z(z)$. This means that when $|\mathcal{Z}|$ is large enough, its impact on the generalization error TV(P, P_{GC}) can be ignored. Specifically, if $|\mathcal{Z}| \to \infty$, then $\hat{\mathcal{L}}_G \to \mathcal{L}_G$, allowing us to focus on the impact of $m + n$, the amount of the semi-supervised training data. Therefore, Lemma 2 establishes the bound of TV(P, P_{GC}) on finite data, and shows that as m and n increase, the optimization over finite data with $\min_{G,C} \max_D \hat{\mathcal{L}}_{gan}^{G,C,D}$ as objective will lead to smaller TV(P, P_{GC}). Consequently P_{GC} gradually approaches P, which is consistent with the optimization result over infinite data claimed by Lemma 1.

2) The Generalization Error Bound of PUAT: Based on Lemma 2, now we can give the generalization error bound of PUAT when it is optimized on finite data, which is claimed by the following theorem. Recall that the generalization of PUAT includes two parts, the generalization over natural samples (i.e., the standard generalizability), which is achieved by minimizing $\hat{\mathcal{L}}_{nat}^C$, and the generalization over adversarial examples (i.e., the adversarial robustness), which is achieved by minimizing $\max_A \hat{\mathcal{L}}_{adv}^{C,A}$. The following theorem confirms that on finite data, these two generalizations are also consistent, and gives their generalization error bounds. Again, in the following text we use P, P_G, and P_C to represent $P(x, y)$, $P_G(x, y)$, and $P_C(x, y)$, respectively, when the context is unambiguous.

Theorem 3. *Let $m = |\mathcal{D}_l|$, $n = |\mathcal{D}_c|$. The following results hold:*
(1) TV$(P, P_C) \leq \frac{1}{\sqrt{2}} \left(\hat{\mathcal{L}}_{nat}^C + b_1 \sqrt{\frac{\log \frac{1}{\delta}}{2m}} - R_1^* \right)^{\frac{1}{2}}$ with probability at least $1 - \delta$ for any $0 < \delta < 1$, where $b_1 = \| -\log P_C(y|x) \|_\infty$, and $R_1^* = -\mathbb{E}_{P(x,y)} \log P(y|x)$ is Bayes error.
(2) TV$(P, Q) \leq \frac{1}{2\sqrt{2}} \left(\max_A \hat{\mathcal{L}}_{adv}^{C,A} - R_2^* \right)^{\frac{1}{2}} + B$ with probability at least $(1-\delta)^2$ for any $0 < \delta < 1$ if $|\mathcal{Z}| \to \infty$, where $Q \in \{P_C, P_G\}$, and $R_2^* = -\mathbb{E}_{P_G} \log \frac{P_G(x,y)}{P(x)}$ is Bayes error.

Proof. (1) Note that for the population loss $\mathcal{L}_{nat}^C$ defined by the first line of Equation (13), we have

$$\begin{aligned}\mathcal{L}_{nat}^C &= \iint P(x, y) \log \left(\frac{P(x, y)}{P_C(y|x) P(x)} \cdot \frac{1}{P(y|x)} \right) \text{ dxdy} \\ &= \iint P(x, y) \log \frac{P(x, y)}{P_C(x, y)} \text{ dxdy} - \mathbb{E}_{P(x,y)} \log P(y|x) \\ &= \mathrm{KL}\big(P \| P_C\big) + R_1^*,\end{aligned}$$

According to Hoeffding's Inequality [35], with probability at least $1 - \delta$ the following inequality holds:

$$\mathcal{L}_{nat}^C \leq \hat{\mathcal{L}}_{nat}^C + b_1 \sqrt{\frac{\log \frac{1}{\delta}}{2m}},$$

where $b_1 = \| -\log P_C(y|x) \|_\infty$ is the upper bound of the loss on a single sample $(x, y) \sim P$. Then by applying Pinsker's Inequality [34] as we do in the proof of Lemma 2, it is easy to show that the following inequality holds with probability at least $1 - \delta$:

$$\begin{aligned}\mathrm{TV}(P, P_C) &\leq \sqrt{\frac{1}{2} \mathrm{KL}\big(P \| P_C\big)} = \sqrt{\frac{1}{2} (\mathcal{L}_{nat}^C - R_1^*)} \\ &\leq \frac{1}{\sqrt{2}} \left(\hat{\mathcal{L}}_{nat}^C + b_1 \sqrt{\frac{\log \frac{1}{\delta}}{2m}} - R_1^* \right)^{\frac{1}{2}}.\end{aligned}$$

(2) In this part, we first (a) obtain the upper bound of $\mathrm{KL}(P_G \| P_C)$ through $\mathcal{L}_{adv}^{C,A}$, and then (b) obtain the upper bound of TV(P, P_C) and TV(P, P_G) using $\mathrm{KL}(P_G \| P_C)$ and TV(P, P_{GC}).

(a) According to Equation (14),

$$\begin{aligned}\mathcal{L}_{adv}^{C,A} &= \mathbb{E}_{P_T(x,y)} \big[-\log P_C(y|x) \big] \\ &\geq \mathbb{E}_{P_G(x,y)} \big[-\log P_C(y|x) \big] \\ &= \iint P_G(x, y) \log \left(\frac{P_G(x, y)}{P_C(y|x) P(x)} \cdot \frac{P(x)}{P_G(x, y)} \right) \text{ dxdy} \\ &= \mathrm{KL}(P_G \| P_C) + R_2^*.\end{aligned}$$

Hence

$$\mathrm{KL}(P_G\|P_C) \leq \mathcal{L}_{adv}^{C,A} - R_2^*.$$

According to PAC learning theory [36], with probability at least $1-\delta$ the following inequality holds:

$$\mathcal{L}_{adv}^{C,A} \leq \hat{\mathcal{L}}_{adv}^{C,A} + b_2\sqrt{\frac{log\frac{1}{\delta}}{2m|\mathcal{Z}|}},$$

where $b_2 = \|-\log P_C(y|\tilde{x})\|_\infty$ is the upper bound of the loss on a single sample from P_T. Since $\lim_{|\mathcal{Z}|\to\infty} \mathcal{L}_{adv}^{C,A} = \hat{\mathcal{L}}_{adv}^{C,A}$, taking $A^* = \operatorname{argmax} \hat{\mathcal{L}}_{adv}^{C,A}$, we can get the following inequality:

$$\mathrm{KL}\big(P_G\|P_C\big) \leq \max_A \hat{\mathcal{L}}_{adv}^{C,A} - R_2^*.$$

(b) Since

$$\begin{aligned}
\mathrm{TV}(P,P_C) =& \frac{1}{2}\iint |P_C - P| \text{ dxdy} \\
=& \frac{1}{2}\iint \left|\frac{1}{2}(P_C - P_G) + \frac{1}{2}(P_C + P_G) - P\right| \text{ dxdy} \\
\leq& \frac{1}{4}\iint |P_C - P_G| \text{ dxdy} + \frac{1}{2}\iint |P_{GC} - P| \text{ dxdy} \\
=& \frac{1}{2}\mathrm{TV}(P_G,P_C) + \mathrm{TV}(P,P_{GC})
\end{aligned}$$

and

$$\begin{aligned}
\mathrm{TV}(P,P_G) =& \frac{1}{2}\iint |P_G - P| \text{ dxdy} \\
=& \frac{1}{2}\iint \left|\frac{1}{2}(P_G - P_C) + \frac{1}{2}(P_G + P_C) - P\right| \text{ dxdy} \\
\leq& \frac{1}{4}\iint |P_G - P_C| \text{ dxdy} + \frac{1}{2}\iint |P_{GC} - P| \text{ dxdy} \\
=& \frac{1}{2}\mathrm{TV}(P_G,P_C) + \mathrm{TV}(P,P_{GC}),
\end{aligned}$$

$\mathrm{TV}(P,P_C)$ and $\mathrm{TV}(P,P_G)$ share the same upper bound. For any $Q \in \{P_C, P_G\}$, by Pinsker's inequality and Lemma 2,

$$\begin{aligned}
\mathrm{TV}(P,Q) &\leq \frac{1}{2}\mathrm{TV}(P_G,P_C) + \mathrm{TV}(P,P_{GC}) \\
&\leq \frac{1}{2}\sqrt{\frac{1}{2}\mathrm{KL}(P_G\|P_C)} + \mathrm{TV}(P,P_{GC}) \\
&\leq \frac{1}{2\sqrt{2}}\Big(\max_A \hat{\mathcal{L}}_{adv}^{C,A} - R_2^*\Big)^{\frac{1}{2}} + B
\end{aligned}$$

holds with probability at least $(1-\delta)^2$. □

Remark 3. *The conclusion (1) of Theorem 3 provides the generalization error gap incurred by optimizing over natural examples (i.e., minimizing $\hat{\mathcal{L}}_{nat}^{C}$), while the conclusion (2) gives the generalization error gap incurred by optimizing over adversarial examples (i.e., minimizing $\max_A \hat{\mathcal{L}}_{adv}^{C,A}$). We can see that minimizing $\hat{\mathcal{L}}_{nat}^{C}$ and $\max_A \hat{\mathcal{L}}_{adv}^{C,A}$ can consistently leads to smaller TV(P,P_C). This shows that even if the data is finite, PUAT can improve standard generalizability and adversarial robustness without conflict, which is consistent with the case of infinite data claimed by Theorem 1. Additionally, Theorem 3 indicates that as the amount of data increases (i.e., larger values of m or n), PUAT can achieve better generalization (i.e., smaller values of TV(P,P_C))).*

3) The Bound of the UAE Adversary Gap: As demonstrated by Theorem 3, when data is finite, there exists generalization error between the learned distribution P_G from which UAEs are drawn and the overall distribution P. Essentially, the difference between P_G and P is caused by the samples (x,y) whose probabilities are different in P_G and P, i.e., $P(x,y) \neq P_G(x,y)$. This discrepancy results in the UAE adversary gap, which is the difference between the adversary offered by the empirical UAEs drawn from P_G, and the ideal adversary offered by the UAEs drawn from P. For simplicity, we will omit the constrains $\|\tilde{x}-x\|_p \leq \epsilon$ and $\|T(x)-x\|_p \leq \epsilon$ in this subsection. Following the idea of Theorem 2, the adversary offered by empirical UAEs is measured by $\mathbb{E}_{(x,y)\sim P_G}\max_{\tilde{x}} l(\tilde{x},y;C)$, while the ideal adversary of UAEs is measured by $\mathbb{E}_{(x,y)\sim P}\max_{\tilde{x}} l(\tilde{x},y;C)$. Then the adversary gap is

$$G_U = \left|\mathbb{E}_{(x,y)\sim P_G}\max_{\tilde{x}} l(\tilde{x},y;C) - \mathbb{E}_{(x,y)\sim P}\max_{\tilde{x}} l(\tilde{x},y;C)\right|.$$

Theorem 4 states that this gap is bounded by the number of the semi-supervised data.

Theorem 4. *Let $m = |\mathcal{D}_l|$, $n = |\mathcal{D}_c|$. For any $0 < \delta < 1$,*

$$G_U \leq 2B_1\Big(B + \frac{1}{2\sqrt{2}}\big(\max_A \hat{\mathcal{L}}_{adv}^{C,A} - R_2^*\big)^{\frac{1}{2}}\Big)$$

holds with probability at least $(1-\delta)^2$, where B and R_2^* are defined in Lemma 2 and Theorem 3, respectively. Furthermore, $B_1 = \max_{P(x,y)\neq P_G(x,y)} \max_{\tilde{x}} l(\tilde{x},y;C)$ measures the adversary of the AEs generated by the examples (x,y) such that their probabilities differ in distributions P and P_G.

Proof. According to the concept of expectation and the property of integral,

$$\begin{aligned}
G_U =& \left|\iint (P_G(x,y) - P(x,y)) \max_{\tilde{x}} l(\tilde{x},y;C) \text{ dxdy}\right| \\
\leq& \iint |P_G(x,y) - P(x,y)| \max_{\tilde{x}} l(\tilde{x},y;C) \text{ dxdy}.
\end{aligned}$$

Noting that only (x,y) such that $P_G(x,y) \neq P(x,y)$ contributes to the integral,

$$G_U \leq 2B_1\mathrm{TV}(P,P_G),$$

where $\mathrm{TV}(P,P_G) = \frac{1}{2}\iint |P_G(x,y) - P(x,y)| \text{ dxdy}$. According to the conclusion (2) of Theorem 3,

$$G_U \leq 2B_1\left(B + \frac{1}{2\sqrt{2}}\Big(\max_A \hat{\mathcal{L}}_{adv}^{C,A} - R_2^*\Big)^{\frac{1}{2}}\right)$$

holds with probability at least $(1-\delta)^2$. □

Remark 4. *It is easy to see that the upper bound of G_U is a monotonically decreasing function of m and n. Therefore, Theorem 4 shows that an increase in the number of labeled or unlabeled data points reduces G_U and results in a stronger adversary of empirical UAEs. This allows PUAT to offer better comprehensive robustness.*

When $P_G \neq P$, there may exist natural samples $x \in supp(P)$ but $x \notin supp(P_G)$, which leads to some RAEs that are not covered by the learned UAE distribution. To rectify

this and further minimize the adversary gap of the empirical UAE, one solution is to train PUAT using both UAEs and RAEs with the following loss function

$$\min_{C,G}\max_{A,D}\big[\mathcal{L}_{nat}^{C}+\lambda\mathcal{L}_{adv}^{C,A}+\gamma\mathcal{L}_{gan}^{G,C,D}+\beta\mathcal{L}_{r}^{C}\big], \tag{16}$$

where $\mathcal{L}_r^C$ is the adversarial loss on RAE and β controls its weight. Specifically, $\mathcal{L}_r^C$ is defined as

$$\mathcal{L}_{r}^{C}=\mathbb{E}_{(x,y)\sim P}\big[-\log P_C(y|\hat{x})\big], \tag{17}$$

where $\hat{x}$ is an RAE satisfying a norm-constraint, which is the result by perturbing a natural sample x, i.e. $\hat{x} = \text{argmax}_{\hat{x}:\|\hat{x}-x\|_p\leq\epsilon} l(\hat{x}, y; C)$.

V. Experiments

The goal of experiments is to answer the following research questions:

- **RQ1** How does PUAT perform as compared to the state-of-the-art baselines in terms of the standard generalization and comprehensive adversarial robustness?
- **RQ2** Does PUAT improve the tradeoff between standard generalization and adversarial robustness?
- **RQ3** How do UAEs and unlabeled data influence the performance of PUAT?
- **RQ4** How to visually show the superiority of PUAT?
- **RQ5** How long is the training of PUAT?

A. Experimental Setting

1) Datasets: We conduct the experiments on Tiny ImageNet [37], ImageNet32 [38], SVHN [39], CIFAR10 [40], and CIFAR100 [40], which are widely used benchmarks for evaluating adversarial training. Tiny ImageNet has 200 classes each of which consists of 500 images for training and 50 ones for testing. Following [31], for Tiny ImageNet we select 10 classes from 200 classes and downscale their resolution to 32×32. And for ImageNet32 which has 1,000 classes and 1,281,167 images, we use the subset consisting of its first 10 classes. SVHN is a set of 73,257 and 26,032 digit images for training and testing respectively. CIFAR10 contains 50,000 training images and 10,000 testing images distributed across 10 classes, while CIFAR100 has 50,000 training images and 10,000 testing images over 100 classes.

Each training set is randomly divided into two parts, labeled data and unlabeled data, where unlabeled data are built by removing the labels of the images. We will repeat the random division of the dataset three times, and report the average results with standard deviation. Table I shows the dataset configurations following the widely used semi-supervised settings [31], [32]. Besides, on each dataset, we randomly select 20 percent of the labeled training data as validation set for the tuning of hyper-parameters.

2) Baseline Methods: To verify the superiority of PUAT, we compare it with the following baseline methods, whose characteristics are summarized in Table II.

- **Regular** Regular method trains the target classifier using labaled natural data with cross-entropy loss, which plays the role of performance benchmark.

TABLE I
CONFIGURATION OF THE DATASETS.

Datasets	training		testing	classes
	labeled	unlabeled		
Tiny ImageNet [37]	1,000	4,000	500	10
ImageNet32 [37]	1,000	11,850	500	10
SVHN [39]	1,000	72,257	26,032	10
CIFAR10 [40]	4,000	46,000	10,000	10
CIFAR100 [40]	10,000	40,000	10,000	100

TABLE II
CHARACTERISTICS OF THE BASELINE METHODS AND PUAT.

Methods	RAE	UAE	unlabeled data	generated data
Regular				
TRADES [41]	✓			
DMAT [42]	✓			✓
RST [27]	✓		✓	
PUAT	✓	✓	✓	✓

- **TRADES** [41] TRADES is an AT method based on RAEs that are generated by PGD based on labeled data, which can trade adversarial robustness off against standard generalizability by optimizing a regularized surrogate loss.
- **DMAT** [42] DMAT performs an RAE-based AT on a training dataset augmented by samples generated by a class conditional elucidating diffusion model (EDM) [43].
- **RST** [27] RST is also an AT method based on RAEs that are generated by PGD. Different from TRADES, RST first trains a classifier to predict labels for unlabeled data and then feeds these pseudo-labeled data together with labeled data into the sequel adversarial training algorithm.

3) Evaluation Protocol: We will use PUAT and the baseline methods to train the target classifier and compare their performances from the perspectives of the target classifier's standard generalizability and adversarial robustness. The generalizability is evaluated in terms of the target classifier's **natural accuracy** on testing natural examples, while the adversarial robustness is evaluated in terms of the target classifier's **robust accuracy** on the testing adversarial examples which are generated by various attack methods. In particular, for the robustness against RAE, we choose PGD [9] and Auto Attack (AA) [44] to generate testing RAEs, which are the attacking methods widely adopted by the existing works. For the robustness against UAE which beyond RAE, we also choose GPGD [45] and USong [16] as the attacking methods, both of which are GAN-based methods and dedicated to the generation of UAE. The hyper-parameters of the attacking methods are shown in Appendix B, where ϵ is the perturbation budget and the bigger it, the stronger the attack. To evaluate the performance of PUAT and the baseline methods, we first invoke the attacking methods to generate testing AEs that can cause worst-case loss of the target classifier well trained by PUAT or a baseline method, and then check the classifier's accuracy on those testing AEs.

4) Hyper-parameter Setting and Implement Details: All the hyper-parameters and the architecture of PUAT are shown in Appendix B, where the generator G, discriminator D are

implemented with respect to those adopted in [29], and A is implemented as an MLP plus two residual blocks. During the training and testing, we employ the Wide ResNet (WRN-28-10) [46] as the target classifier C. We use SGD as the optimizer with Nesterov momentum [47] and cosine annealed cyclic learning rate schedule [48], where learning-rate, weight-decay, and momentum are set to 0.2, 5E-4, and 0.9, respectively. During training, a batch consists of 256 labeled images and 256 unlabeled ones, and early stopping is adopted as default option.

To avoid gradient disappearance and mode collapse for the training of the G-C-D GAN, spectral norm [49] together with hinge GAN loss [50], [51] is used for D to stabilize the training. When labeled data is severely sparse, it is difficult to train G to precisely capture the real distribution $P(x, y)$. To overcome this issue, following the idea of [32], during the training we augment $\mathcal{D}_l$ with the pseudo-labeled pairs $\{(x_c, \hat{y}_c)\}$, where $x_c \in \mathcal{D}_c$ and $\hat{y}_c = \text{argmax}_{y_c \in \mathcal{Y}} P_C(y = y_c|x_c)$. Note that here the pseudo label $\hat{y}_c$ is a hard label represented by a one-hot vector, while in Equation (10) the pseudo label y_c is a soft label represented by a softmax vector. This will not introduce conflict to the training of C-D GAN because (1) if $\hat{y}_c = y_c$, the loss incurred by $\{(x_c, \hat{y}_c)\}$ and the loss incurred by $\{(x_c, y_c)\}$ will cancel each other out; (2) if $\hat{y}_c \neq y_c$, D will force C to assign $\hat{y}_c$ a higher probability, which is similar to self-training on C.

B. Performance (RQ1)

Tables III, IV, V, VI, and VII show the performances of PUAT and the baseline methods on all the datasets, from which we can make the following observations:

- On all datasets, PUAT surpasses the baseline AT methods in robust accuracy under all attacks regardless of the attacking strength (ϵ), with the exception of GPGD-0.1 on CIFAR10 and CIFAR100, and the exception of USong-0.01 on CIFAR100. At the same time, PUAT achieves the highest natural accuracy compared with the baseline AT methods. Therefore, PUAT is the only AT approach that simultaneously increases standard generalization and adversarial robustness against all attacks on all datasets, due to its ability to align the AE distribution and natural data distribution, which results in a consistent generalization on AEs and natural samples. This result demonstrates PUAT's capacity to improve the tradeoff between the standard generalization and the adversarial robustness.
- PUAT beats all all defensive methods against RAE attacks in all cases and UAE attacks in most cases. This result shows how effective PUAT is at providing comprehensive adversarial robustness against both RAE and UAE attacks. We contend that the comprehensive adversarial robustness is made possible by our innovative notion to generalize the UAE concept to imperceptibly perturbed natural examples, whether observable or unobservable, which makes PUAT able to cover RAEs in a unified way.
- The robustness of all methods decreases as the attack strength (ϵ) grows up. However, under all the attack strengths, PUAT consistently outperforms the baseline methods in the robustness against RAEs, and the robustness against UAEs in most cases. This suggests that AT on UAEs provides better adversarial robustness because the adversary of UAEs can be improved by utilizing semi-supervised data, as told by Theorem 4.
- We also note that PUAT outperforms DMAT which is an AT method also based on generative model. This is because DMAT only uses labeled data, while PUAT can benefit from both labeled and unlabeled data. In fact, in most cases on CIFAR100 where labeled data per each class is extemely sparse, DMAT performs poorly than the other baselines, which shows DMAT is unsuitable for sparse labeled data.
- We observe that PUAT is second to RST in robust accuracy under attacks USong-0.01 and GPGD-0.1 on CIFAR100. We argue that this is partly due to the fact that it is hard to train a triple-GAN on a dataset with too many classes like CIFAR100 which contains 100 classes. One possible solution is to improve the model complexity of the G-C-D GAN, which is worth exploring in future.

C. Improvement of The Tradeoff (RQ2)

Now we examine if PUAT can improve the tradeoff between standard generalization and adversarial robustness. For this purpose, in Figure 3 we show the Pareto front curves of natural accuracy vs. robust accuracy under adversarial attacks PGD-8/255, AA, GPGD-0.1 and USong. We can see that under all adversarial attacks, the points of PUAT's curve always lies to the upper right of the curves of the baseline methods, which implies that PUAT can achieve a better natural accuracy as well as a better robust accuracy than the baseline methods, or in other words, PUAT can improve the tradeoff between standard generalization and adversarial robustness. This is because (1) By using G-C-D GAN to align the generator distribution $P_G(x, y)$, the target classifier distribution $P_C(x, y)$ and the natural data distribution $P(x, y)$, PUAT is able to reconcile the robust generalization on AEs and the standard generalization on natural data, which is asserted by Theorem 3; and (2) PUAT conducts AT on UAEs which results in greater robustness since UAE adversary will be improved by both labeled and unlabeled data, which is asserted by Theorem 4. Additionally, we also observe that RST achieves a better tradeoff than TRADES and DMAT, which is partly because RST also utilizes unlabeled samples to augment the data used for AT.

D. Ablation Study (RQ3)

Now we investigate how UAEs and (un)labeled data influence the performance of PUAT.

1) Effect of UAE: To investigate the UAE's effect on PUAT, we first train PUAT with different $\lambda \in \{0.0, 1.0, 10.0, 20.0\}$ in Equation (16), where $\lambda = 0.0$ means the training without UAEs, and then check its testing natural accuracy and robust accuracy. The result is shown in Figure 4(a), from which we can see as the importance of UAEs (indicated by λ) increases, both natural accuracy and robust accuracy increase, which verifies that UAEs can benefit PUAT by boosting standard generalization and adversarial robustness simultaneously. We also note that when λ exceeds 10, the performance of PUAT

TABLE III
PERFORMANCE ON **TINY IMAGENET**.

Methods	Natural Accuracy	Robust Accuracy to RAE				Robust Accuracy to UAE		
		PGD-2/255	PGD-4/255	PGD-8/255	AA	GPGD-0.01	GPGD-0.1	USong
Regular	65.93 ± 0.25	22.13 ± 2.76	2.33 ± 0.52	0.0 ± 0.0	0.0 ± 0.0	33.27 ± 0.84	14.60 ± 0.75	14.40 ± 1.56
TRADES	58.33 ± 1.32	51.20 ± 0.28	44.53 ± 0.77	31.47 ± 2.31	29.47 ± 1.61	40.47 ± 0.66	20.27 ± 0.90	43.53 ± 2.29
DMAT	50.90 ± 3.10	45.00 ± 2.80	37.90 ± 2.90	26.60 ± 1.60	22.00 ± 1.60	37.70 ± 3.30	22.70 ± 0.90	43.30 ± 2.30
RST	43.87 ± 3.17	38.80 ± 2.14	33.20 ± 1.34	24.00 ± 1.51	21.60 ± 2.73	30.93 ± 3.47	18.13 ± 2.32	35.60 ± 2.26
PUAT	$\mathbf{69.00 \pm 0.40}$	$\mathbf{62.40 \pm 0.20}$	$\mathbf{55.60 \pm 0.20}$	$\mathbf{38.60 \pm 0.40}$	$\mathbf{37.40 \pm 1.20}$	$\mathbf{44.90 \pm 2.10}$	$\mathbf{27.10 \pm 0.50}$	$\mathbf{48.80 \pm 2.20}$

TABLE IV
PERFORMANCE ON **IMAGENET32**.

Methods	Natural Accuracy	Robust Accuracy to RAE				Robust Accuracy to UAE		
		PGD-2/255	PGD-4/255	PGD-8/255	AA	GPGD-0.01	GPGD-0.1	USong
Regular	$\mathbf{53.13 \pm 1.33}$	18.87 ± 1.65	6.87 ± 0.77	1.40 ± 0.28	1.33 ± 0.38	34.53 ± 2.45	22.20 ± 1.14	24.13 ± 2.31
TRADES	32.80 ± 1.28	28.53 ± 1.24	23.53 ± 2.23	17.47 ± 2.78	15.60 ± 2.27	32.13 ± 0.69	20.53 ± 1.05	30.47 ± 0.98
RST	48.07 ± 2.81	41.27 ± 1.75	35.40 ± 1.47	26.47 ± 0.66	22.60 ± 1.28	39.80 ± 1.25	23.07 ± 0.68	43.80 ± 1.89
PUAT	51.20 ± 0.71	$\mathbf{45.33 \pm 0.50}$	$\mathbf{39.87 \pm 0.25}$	$\mathbf{27.40 \pm 0.33}$	$\mathbf{25.93 \pm 0.41}$	$\mathbf{41.47 \pm 1.11}$	$\mathbf{25.33 \pm 1.79}$	$\mathbf{45.00 \pm 1.86}$

TABLE V
PERFORMANCE ON **SVHN**.

Methods	Natural Accuracy	Robust Accuracy to RAE				Robust Accuracy to UAE		
		PGD-2/255	PGD-4/255	PGD-8/255	AA	GPGD-0.01	GPGD-0.1	USong
Regular	73.76 ± 0.63	4.73 ± 2.35	0.15 ± 0.11	0.0 ± 0.0	0.0 ± 0.0	78.27 ± 1.11	48.42 ± 2.40	70.10 ± 3.01
TRADES	58.35 ± 2.14	44.67 ± 1.71	32.44 ± 1.12	15.71 ± 0.66	13.31 ± 0.58	59.27 ± 2.88	32.64 ± 1.24	61.73 ± 1.63
DMAT	83.65 ± 0.40	80.71 ± 1.41	74.49 ± 1.57	55.41 ± 1.71	47.26 ± 1.62	81.40 ± 1.27	58.90 ± 2.39	93.91 ± 2.11
RST	85.18 ± 0.11	78.25 ± 0.19	69.83 ± 0.11	50.60 ± 0.12	43.74 ± 0.23	86.03 ± 1.76	66.25 ± 1.49	97.27 ± 0.21
PUAT	$\mathbf{92.27 \pm 1.04}$	$\mathbf{87.79 \pm 1.63}$	$\mathbf{80.31 \pm 1.56}$	$\mathbf{59.19 \pm 0.12}$	$\mathbf{53.37 \pm 0.00}$	$\mathbf{88.22 \pm 0.54}$	$\mathbf{68.03 \pm 0.48}$	$\mathbf{98.75 \pm 0.08}$

TABLE VI
PERFORMANCE ON **CIFAR10**.

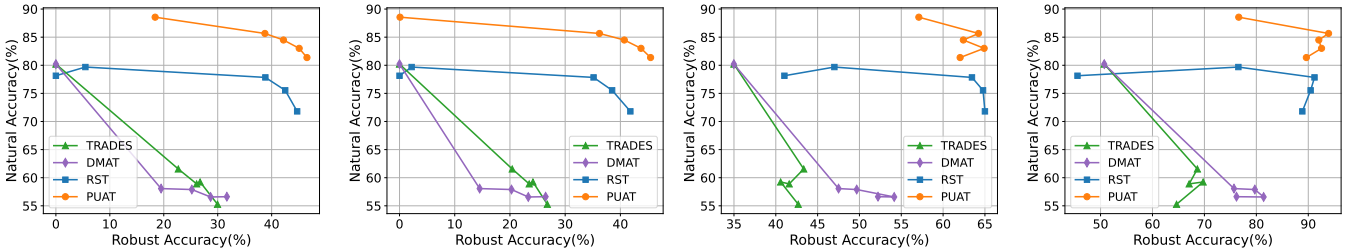
Methods	Natural Accuracy	Robust Accuracy to RAE				Robust Accuracy to UAE		
		PGD-2/255	PGD-4/255	PGD-8/255	AA	GPGD-0.01	GPGD-0.1	USong
Regular	80.21 ± 0.50	2.41 ± 0.43	0.02 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	65.87 ± 1.54	34.99 ± 2.14	50.71 ± 2.16
TRADES	55.27 ± 0.70	48.65 ± 0.49	42.27 ± 0.44	29.97 ± 0.27	26.77 ± 0.64	63.36 ± 1.87	42.70 ± 1.24	64.64 ± 0.91
DMAT	56.57 ± 0.90	49.21 ± 0.28	42.20 ± 0.26	28.69 ± 0.25	23.32 ± 0.29	74.28 ± 1.86	54.16 ± 2.10	81.40 ± 0.20
RST	71.80 ± 0.58	65.82 ± 0.55	58.93 ± 0.37	44.73 ± 0.17	41.81 ± 1.94	83.63 ± 0.48	$\mathbf{64.99 \pm 0.43}$	89.01 ± 0.64
PUAT	$\mathbf{83.02 \pm 0.36}$	$\mathbf{76.09 \pm 0.26}$	$\mathbf{66.76 \pm 0.21}$	$\mathbf{45.10 \pm 0.24}$	$\mathbf{43.74 \pm 0.12}$	$\mathbf{86.78 \pm 0.29}$	64.91 ± 0.64	$\mathbf{92.55 \pm 0.31}$

TABLE VII
PERFORMANCE ON **CIFAR100**.

Methods	Natural Accuracy	Robust Accuracy to RAE				Robust Accuracy to UAE		
		PGD-2/255	PGD-4/255	PGD-8/255	AA	GPGD-0.01	GPGD-0.1	USong
Regular	$\mathbf{59.02 \pm 0.31}$	12.14 ± 0.44	1.04 ± 0.02	0.02 ± 0.0	0.0 ± 0.0	25.28 ± 0.29	10.59 ± 0.31	26.59 ± 0.62
TRADES	34.23 ± 0.15	28.10 ± 0.12	22.46 ± 0.14	14.40 ± 0.01	11.99 ± 0.18	20.45 ± 0.19	8.43 ± 0.30	24.56 ± 0.27
DMAT	30.81 ± 0.77	22.74 ± 0.43	15.20 ± 0.84	5.88 ± 0.75	2.68 ± 0.73	21.92 ± 1.50	10.88 ± 0.54	31.81 ± 2.19
RST	50.34 ± 0.52	40.30 ± 0.52	31.32 ± 0.50	17.36 ± 0.35	13.36 ± 0.48	32.00 ± 0.15	$\mathbf{16.26 \pm 0.18}$	$\mathbf{41.82 \pm 0.21}$
PUAT	51.95 ± 0.45	$\mathbf{42.70 \pm 0.43}$	$\mathbf{33.54 \pm 0.01}$	$\mathbf{18.45 \pm 0.38}$	$\mathbf{17.16 \pm 0.46}$	$\mathbf{33.00 \pm 0.10}$	16.21 ± 0.16	38.32 ± 0.43

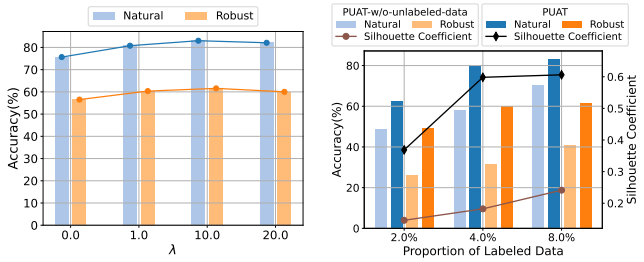
begins to drop, which shows that excessively large λ may lead to overfitting of both natural sample distribution and UAE distribution.

To gain a deeper understanding of the impact of UAE on the learning of PUAT, we also plot the learning curves of PUAT both with and without UAEs in Figure 5. From Figure 5(a) we can see that regardless of whether UAEs are used or not, the training robust accuracy curves both rise with the number of training epochs, while their corresponding testing robust accuracy curves both first rise and then fall gradually.



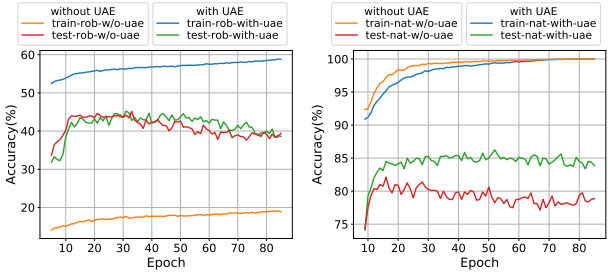
(a) PGD-8/255 (b) AA (c) GPGD-0.1 (d) USong

Fig. 3. Pareto front curves of natural accuracy vs. robust accuracy on CIFAR10 under different adversarial attacks including (a) PGD-8/255, (b) AA, (c) GPGD-0.1, and (d) USong. Each curve consists of points generated by setting β with various values.



(a) effect of UAE (b) effect of (un)labeled data

Fig. 4. Effect of UAE and (un)labeled data. Subfigure (a) shows the natural accuracy and robust accuracy of PUAT under different λ on CIFAR10. Subfigure (b) shows the natural accuracy and robust accuracy of PUAT and PUAT-w/o-unlabeled-data, and the Silhouette Coefficients, over different amount of labeled data on CIFAR10, where PUAT-w/o-unlabeled-data is a variant of PUAT without using unlabeled data during the training. The robust accuracy in both (a) and (b) is the average of the robust accuracies on PGD-8/255, AA, GPGD-0.1 and USong, which reflects the comprehensive adversarial robustness.



(a) Robust Accuracy (b) Natural Accuracy

Fig. 5. Learning curves of PUAT with UAEs ($\lambda = 10.0$) and without UAEs ($\lambda = 0.0$). In both Subfigures (a) and (b), blue and green curves are the robust/natural accuracy curve of training with UAEs and its corresponding testing robust/natural accuracy curve, respectively, while orange and red curves are the robust/natural accuracy curve of training withouth UAEs and its corresponding testing robust/natural accuracy curve, respectively.

This results indicate that there exist robust overfitting [52], i.e., the overfitting on AEs, whether or not UAEs are used. To address this issue, following the idea proposed in [52], we adopt validation-based early stopping to regularize the training of PUAT. On the other hand, Figure 5(b) shows that whether UAEs are used or not, the training natural accuracy curves (blue and yellow curves) both first rise and then gradually stabilize. However, when UAEs are not used for training, the testing natural accuracy curve (red curve) first rises and then drops, which indicates overfitting occurs when training without UAEs. In sharp contrast, when training with UAEs, the testing natural accuracy curve (green curve) first rises and then converges gradually, which suggests that using UAEs helps mitigate the overfitting on natural samples.

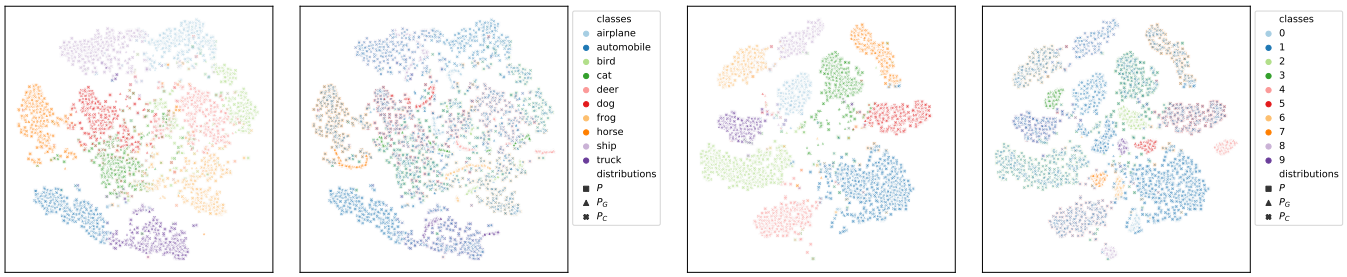
2) Effect of natural samples: Recall that Theorem 3 states that the more the labeled natural samples are used for training, the better the distribution alignment, and the better the tradeoff between standard generalization and comprehensive adversarial robustness. Now we empirically verify this conclusion by checking the testing natural accuracy, robust accuracy and the Silhouette Coefficient [53] after training with various amount of natural samples. The results are shown in Figure 4(b), where we compare PUAT with PUAT-w/o-unlabeled-data (the variant of PUAT without using unlabeled natural samples for training).

At first, from Figure 4(b) we can see that regardless of whether or not unlabeled samples are used, the standard generalization (measured by the natural accuracy), the comprehensive adversarial robustness (measured by the average robust accuracy), and the quality of the clustering of the samples from different distributions (P, P_G, P_C) of each class (measured by Silhouette Coefficient), are all improved as the amount of labeled samples used for training increases. These results validate that both PUAT and PUAT-w/o-unlabeled-data can achieve better distribution alignment and better tradeoff with more labeled samples for training due to G-C-D GAN's ability to align the UAE distribution with natural sample distribution, which is consistent with the conclusions of Theorem 3.

At the same time, it is noteworthy that PUAT performs significantly better than PUAT-w/o-unlabeled-data even with the labeled data of only 2%, which shows that PUAT can improve the performance by leveraging unlabeled data even though the labeled samples are extremely sparse. This is because unlabeled data helps G-C-D GAN more accurately capture real distribution $P(x, y)$, and consequently, enables the target classifier C to generalize consistently over the aligned distributions of natural samples and AEs.

E. Visual Study (RQ4)

Now, we visually demonstrate PUAT's capability to align the natural data distribution $P(x, y)$, the distribution $P_G(x, y)$ learned by the generator, and the distribution $P_C(x, y)$ learned by target classifier, using the testing sets of CIFAR10 and



(a) Alignment on CIFAR10 (b) Misalignment on CIFAR10 (c) Alignment on SVHN (d) Misalignment on SVHN

Fig. 6. Visualized distribution alignment or misalignment, where different colors denote classes and different patterns denote distributions.

TABLE VIII
THE SAMPLE NUMBER OF DIFFERENT CLASSES FROM DIFFERENT DISTRIBUTIONS.

Distribution	CIFAR10										SVHN									
	"airplane"	"automobile"	"bird"	"cat"	"deer"	"dog"	"frog"	"horse"	"ship"	"truck"	"0"	"1"	"2"	"3"	"4"	"5"	"6"	"7"	"8"	"9"
$P(y)$	200	200	200	200	200	200	200	200	200	200	134	392	319	221	194	183	152	155	128	122
$P_G(y)$	200	200	200	200	200	200	200	200	200	200	134	392	319	221	194	183	152	155	128	122
$P_C(y)$	208	201	180	165	201	210	227	187	217	204	143	421	306	199	203	183	152	152	111	131
misaligned $P_C(y)$	0	2000	0	0	0	0	0	0	0	0	0	2000	0	0	0	0	0	0	0	0

SVHN which both contain 10 classes. In particular, for each class y_i $(1 \leq i \leq 10)$ of each testing set, we invoke G to generate a set of $\{(x_g, y_i)\}$ such that $\left|\{(x_g, y_i) \sim P_G(x, y)\}\right| = \left|\{(x_l, y_i) \sim P(x, y)\}\right|$. At the same time, all the testing examples are also fed into C, and those classified as y_i make up $\{(x_c, y_i) \sim P_C\}$. At last, we plot the sample points $\{(x, y_i)\}$, $\{(x_g, y_i)\}$, and $\{(x_c, y_i)\}$ in Figure 6 by using t-SNE algorithm, where different colors denote classes and different patterns denote distributions.

Figures 6(a) and 6(c) show the results of the PUAT on CIFAR10 and SVHN, respectively. From Figures 6(a) and 6(c) we can see that the data points are distributed in different colored clusters with clear boundaries, where the examples from different distributions (with different patterns) but of the same class (with the same color) are grouped in the same cluster, and there are essentially no data points in any cluster that have a different color from the majority. However, once we remove the extended AT from PUAT so that Theorem 1 does not hold, the cluster boundaries become blurred and the colors in the same cluster become impure, as shown in Figures 6(b) and 6(d).

Besides, Figures 6(a) and 6(c) also show that when the distributions are aligned, the samples of every class from P_G distributed evenly in the cluster. On the contrary, Figures 6(b) and 6(d) show that when the extended AT is removed from PUAT, the samples in clusters from P_G concentrate to a small area, which indicates the mode collapse of G-C-D GAN. These results confirm that the distribution alignment brought by the extended AT of PUAT can avoid mode collapse of G-C-D GAN.

Therefore, we can contend that PUAT aligns the three conditional distributions P, P_C and P_G. Furthermore, as shown in Table VIII, when the distributions are aligned, the number of examples belonging to y_i is approximately the same in different distributions, which indicates the marginal distributions $P(y) = P_G(y) = P_C(y)$, so we can confirm that $P(x, y)$, $P_C(x, y)$ and $P_G(x, y)$ are aligned by PUAT.

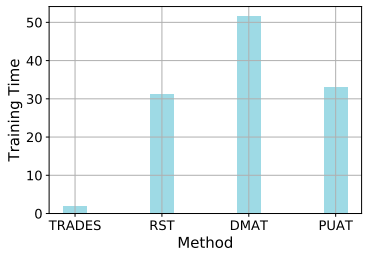


Fig. 7. Training time comparison between AT baselines and PUAT. The training time of each method is measured by its ratio relative to the time of regular training. Only the time taken until early stopping is factored into the calculations. The average values across Tiny ImageNet, SVHN, CIFAR10, and CIFAR100 datasets are reported.

F. Training Time

The training time of each method is measured by its ratio relative to the time of regular training. As we can see from Figure 7, the training time of PUAT is significantly shorter than that of DMAT and slightly longer than that of RST. This is attributable to the rapid convergence exhibited by PUAT, as consistently demonstrated by its learning curves in Figure 5. We also see that the training of TRADES takes the shortest time, which is because it only uses labeled data. In summary, Figure 7 together with Figure 3 shows that PUAT can achieve a superior balance between standard generalization and adversarial robustness at an acceptable additional time cost.

VI. Related Work

In this section, we briefly introduce the related works from three domains, adversarial attacks, adversarial training, and

standard generalizability, and comment their differences from our work.

A. Adversarial Attacks

Generally, adversarial attacks aim to fool a target model well-trained on natural data with AEs that are imperceptible to humans [4], [54]. The existence of AEs is first discovered by the seminal works of Szegedy *et al*. [2] and Goodfellow *et al*. [8]. In [2], Szegedy *et al*. propose an attack method called L-BFGS which can imperceptibly perturb a natural example with respect to its second-order gradient to cause misclassifications. In [8], Goodfellow *et al*. further propose the fast gradient sign method (FGSM), which searches adversarial perturbations just based on a single first-order gradient step. Following the idea of [8], various adversarial attack methods have been proposed by enhancing the first-order method. For example, Madry *et al*. [9] propose a projected gradient descent (PGD) method which can be regarded as an iterative FGSM projecting AEs into a ball constrained by L_∞ norm. Instead of restricting the perturbation norms to a fixed value, Moosavi-Dezfooli *et al*. [55] propose an optimization based attack method Deepfool to seek adversarial perturbations for differentiable classifiers by minimizing L_2 norm. Similarly, Carlini and Wagner [56] propose another optimization based method called C&W attack, which can generate stronger AEs with perturbations minimizing L_0, L_2 and L_∞ norms at the expense of higher computation complexity. In contrast to gradient based methods, another line of works employs GAN to generate AEs for better imperceptibility. For example, Xiao *et al*. [57] propose AdvGAN to generate perturbations for observed examples, which extends standard GAN with explicit perturbation norm loss to ensure the imperceptibility. Additionally, instead of generating an AE by perturbing a single example, Moosavi-Dezfooli *et al*. [58] also further propose universal adversarial perturbations which are computed in an example-agnostic fashion and can fool the target classifier on any examples.

The above-mentioned adversarial attacks focus on the AEs which are generated by adding imperceptible perturbations restricted by L_p norm on observed examples, which we call restricted AE (RAE). Recently, some researchers have discovered that there exist unrestricted AEs (UAEs) that are not norm-constrained. To ensure the imperceptibility of UAEs, one line of UAE generation methods manipulates example features by perturbations within perceptual distances that are aligned with human perception [18], [19], [59], [60]. At the same time, some researchers find that in a broader sense, UAEs can be completely new examples built from scratch and dissimilar to any observed example but can still fool classifiers without confusing humans [16]. Following this idea, a few of UAE generation methods based on generative models are proposed [16], [17], [20], [45], [61], [62]. For example, Song *et al*. [16] and Xiao *et al*. [45] propose to first generate perturbations based on initial random noise and then feed them into a pre-trained conditional GAN (e.g. AC-GAN [63]) to produce UAEs, while Poursaeed *et al*. [61] propose to use a Style-GAN [64] to synthesize UAEs with stylistic perturbations.

The existing works for UAE generation, however, do not explain why UAEs pose threats, or rather, where the imperceptibility of UAEs stems. By contrast, our work proposes a novel viewpoint to view UAEs as perturbed observed and unobserved examples, which provides a logical and provable explanation of UAE's dissimilarity to observed examples, imperceptibility to humans, and the feasibility of comprehensive adversarial robustness. Based on this understanding, we further propose a novel G-C-D GAN for UAE generation, which together with the attacker A can synthesize UAEs that are more adversarial by aligning UAE distribution with natural data distribution.

B. Adversarial Training

AT has proved to be the strongest principled defensive technique against adversarial attacks [5]–[7]. The basic idea of AT is initially proposed by Szegedy *et al*. [2], which uses a min-max game to incorporate deliberately crafted AEs to the training process of a DNN model to obtain immunity against AEs. Madry *et al*. [9] are the first time to theoretically justify the adversarial robustness offered by the min-max optimization used in AT.

Over the past few years, many enhancements of AT have been proposed to employ various perturbation generation methods to improve adversarial robustness [10]–[15]. For example, Wang *et al*. [10] propose a bilateral adversarial training method that perturbs both normal images and real labels with respect to a smaller gradient magnitude. Wang *et al*. [11] propose a misclassification-aware adversarial training (MART) method, which defines a novel adversarial loss with a regularization on misclassified examples. Cheng *et al*. [12] argue that large perturbations may be more adversarial for the robustness of a target model and propose a customized adversarial training (CAT) method which can offer adaptive perturbations for natural examples. Shafahi *et al*. [13] propose a universal adversarial training method for the robustness against universal attacks [58]. Among others, generative model-based AT methods are verified as a promising approach due to the GAN's capability of distributional alignment. For example, to strengthen the adversaries, Lee *et al*. [14] propose a generative adversarial trainer (GAT) which employs a GAN to produce adversarial perturbations with respect to the gradients of the natural examples, while Stutz *et al*. [15] and Xiao *et al*. [45] propose to use VAE-GAN to produce AEs whose distribution is aligned with that of the natural examples. As discussed in VI-A, GAN-based methods strengthen AEs with stronger imperceptibility because of the alignment between the distributions of AEs and the natural examples, which is equivalent to drawing AEs on the manifold (distribution) of the natural examples. Another line of works proposed to improve adversarial robustness with data augmentation for AT [42], [65]–[67]. Among them, some studies [42], [65], [66] utilize generative model to densify the underlying manifold of natural data, and then conduct an RAE-based AT (e.g. TRADES [41]) on the augmented dataset. At the same time, Xing *et al*. [67] theoretically show that the generated data can help improve adversarial robustness.

The above-mentioned AT methods are basically based on supervised learning, which often suffer from the sparsity of labeled data. Recently, some researchers have discovered that introducing richer unlabeled data can increase the performance of AT, and hence proposed semi-supervised AT methods leveraging partially labeled data [26]–[28]. For example, Zhang *et al*. [26] argue that the performance of AT will be impaired by the perturbations generated only based on labeled data since they are unable to reflect the underlying distribution of all potential examples, and propose a GAN-based semi-supervised AT method where AEs can preserve the inner structure of unlabeled data. Carmon *et al*. [27] propose a robust self-training (RST) model which first uses an intermediate classifier trained on labeled examples to infer the labels for unlabeled examples, and then generate AEs for AT from the labeled and pseudo-labeled examples. Stanforth *et al*. [28] propose an unsupervised adversarial training (UAT) method which estimates the worst-case AEs with KL-divergence between the distributions of the perturbed unlabeled examples and the natural unlabeled examples.

Different from the existing AT methods providing adversarial robustness against either RAE or UAE, our PUAT focuses on the concept of generalized UAE which makes it feasible to offer comprehensive adversarial robustness against both UAE and RAE. At the same time, PUAT is both GAN-based and semi-supervised.

C. Standard Generalizability

A model's standard generalizability and adversarial robustness are measured by its accuracy on testing natural examples and testing AEs, respectively. Traditional AT methods are based on AEs with norm-constraints, which will lead to manifold-shift between AEs and natural examples [45], [68]. Therefore, there is a belief that AT forces the target model to generalize on two separated manifolds and hence the tradeoff between standard generalizability and adversarial robustness inevitably comes into being, or in other words, adversarial robustness comes at the expense of standard generalizability [9], [21], [41], [68].

However, some researchers have opposing opinions that the tradeoff may be not inevitable, with support of recent empirical evidences and theoretical analyses [15], [23]–[25], [30]. One line of works tries to extend the existing AT methods with regularizations to mitigate the tradeoff. For example, Song *et al*. [24] and Xing *et al*. [25] propose to improve the generalizability by introducing robust local features and l_1 penalty to AT, respectively. The other line aims at breaking the tradeoff by exploiting the distributional property of AEs and natural examples. For example, Yang *et al*. [23] discover the r-separation phenomenon in widely-used image datasets, namely the distributions of different classes are at least distance $2r$ apart from each other in pixel space. Based on this discovery, they theoretically prove that a classifier with sufficient local Lipschitz smoothness can achieve both high accuracy on natural examples and acceptable robustness on AEs with perturbations of size up to r. Differently, Stutz *et al*. [15] find that there exist AEs on the manifold of natural examples, and hence the adversarial robustness against the on-manifold AEs will equivalently improve the standard generalizability. Similarly, Staib *et al*. [30] propose Distributionally Robust Optimization (DRO), which offers a more general perspective that the tradeoff can be eliminated by the alignment between the distributions of AEs and natural examples. Instead of generating point-wise perturbations with norm-constrained magnitude, DRO aims to seek worst-case AE distribution by restricting the distance between the AE distribution $\tilde{p}$ and the natural data distribution p through the following min-max game:

$$\min_{C} \max_{P_W : W_p(P, P_W) \leq \epsilon, (\tilde{x}, y) \sim P_W} \mathbb{E}_{(x,y)\sim)P} l(\tilde{x}, y; C), \quad (18)$$

where W_p is a distributional distance measure, e.g., Wasserstein distance. As ϵ approaching 0, P_W is gradually aligning with P, and consequently, the tradeoff tends to disappear. Indeed, on-manifold AEs can be seen as a special case of DRO when $\epsilon = 0$.

Our work provides new arguments for the opinion that the tradeoff is eliminable. In particular, we make two contributions which distinguish our work from the existing works addressing the tradeoff: (1) We extend the research scope to UAEs by proposing a novel AT method PUAT; (2) By solid theoretical analysis and extensive empirical investigation, we show that the standard generalizability and comprehensive adversarial robustness are both achievable for PUAT.

VII. Conclusion

In this paper, we propose a unique viewpoint that understands UAEs as imperceptibly perturbed unobserved examples, which rationalizes why UAEs exist and why they are able to deceive a well-trained classifier. This understanding allows us to view RAE as a special UAE, thus providing the feasibility of achieving comprehensive adversarial robustness against both UAE and RAE. Also, we find that if the UAE distribution and the natural data distribution can be aligned, the conflict between robust generalization and standard generalization in traditional AT methods can be eliminated, thus improving both the adversarial robustness and standard generalizability of a target classifier. Based on these ideas, we propose a novel AT method called Provable Unrestricted Adversarial Training (PUAT). PUAT utilizes partially labeled data to achieve efficient UAE generation by accurately capturing the real data distribution through a well-designed G-C-D GAN. At the same time, PUAT extends the traditional AT by introducing the supervised loss of the target classifier into the adversarial training and achieves alignment between the UAE distribution, the natural data distribution, and the distribution learned by the classifier, with the collaboration of the G-C-D GAN. The solid theoretical analysis and the extensive experiments demonstrate that PUAT can improve tradeoffs between adversarial robustness and standard generalizability through distributional alignment, and compared to the baseline methods, PUAT enables a target classifier to better defend against both UAE attacks and RAE attacks, as well as significantly increases its standard generalizability.

ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China under grant 61972270, and in part by NSF under grants III-1763325, III-1909323, III-2106758, and SaTC-1930941.

REFERENCES

[1] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2013, pp. 387–402.

[2] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *International Conference on Learning Representations*, 2014.

[3] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[4] D. J. Miller, Z. Xiang, and G. Kesidis, "Adversarial learning targeting deep neural network classification: A comprehensive review of defenses against attacks," *Proceedings of the IEEE*, vol. 108, no. 3, pp. 402–433, 2020.

[5] T. Bai, J. Luo, J. Zhao, B. Wen, and Q. Wang, "Recent advances in adversarial training for adversarial robustness," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)*, 2021.

[6] N. Akhtar, A. Mian, N. Kardan, and M. Shah, "Advances in adversarial attacks and defenses in computer vision: A survey," *IEEE Access*, vol. 9, pp. 155 161–155 196, 2021.

[7] W. Zhao, S. Alwidian, and Q. H. Mahmoud, "Adversarial training methods for deep learning: A systematic review," *Algorithms*, vol. 15, no. 8, p. 283, 2022.

[8] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *International Conference on Learning Representations*, 2014.

[9] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *International Conference on Learning Representations*, 2018.

[10] J. Wang and H. Zhang, "Bilateral adversarial training: Towards fast training of more robust models against adversarial attacks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[11] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, "Improving adversarial robustness requires revisiting misclassified examples," in *International Conference on Learning Representations*, 2020.

[12] M. Cheng, Q. Lei, P.-Y. Chen, I. Dhillon, and C.-J. Hsieh, "Cat: Customized adversarial training for improved robustness," *CoRR*, vol. abs/2002.06789, 2020.

[13] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, and T. Goldstein, "Universal adversarial training," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 5636–5643.

[14] H. Lee, S. Han, and J. Lee, "Generative adversarial trainer: Defense to adversarial perturbations with gan," *CoRR*, vol. abs/1705.03387, 2017.

[15] D. Stutz, M. Hein, and B. Schiele, "Disentangling adversarial robustness and generalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6976–6987.

[16] Y. Song, R. Shu, N. Kushman, and S. Ermon, "Constructing unrestricted adversarial examples with generative models," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[17] I. Dunn, H. Pouget, T. Melham, and D. Kroening, "Adaptive generation of unrestricted adversarial inputs," *CoRR*, vol. abs/1905.02463, 2019.

[18] A. Bhattad, M. J. Chong, K. Liang, B. Li, and D. A. Forsyth, "Unrestricted adversarial examples via semantic manipulation," in *International Conference on Learning Representations*, 2020.

[19] H. Naderi, L. Goli, and S. Kasaei, "Generating unrestricted adversarial examples via three parameteres," *Multimedia Tools and Applications*, pp. 1–20, 2022.

[20] T. Xiang, H. Liu, S. Guo, Y. Gan, and X. Liao, "Egm: An efficient generative model for unrestricted adversarial examples," *ACM Transactions on Sensor Networks*, 2022.

[21] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry, "Robustness may be at odds with accuracy," in *International Conference on Learning Representations*, 2019.

[22] I. Attias, A. Kontorovich, and Y. Mansour, "Improved generalization bounds for adversarially robust learning," *Journal of Machine Learning Research*, vol. 23, no. 175, pp. 1–31, 2022.

[23] Y.-Y. Yang, C. Rashtchian, H. Zhang, R. R. Salakhutdinov, and K. Chaudhuri, "A closer look at accuracy vs. robustness," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[24] C. Song, K. He, J. Lin, L. Wang, and J. E. Hopcroft, "Robust local features for improving the generalization of adversarial training," in *International Conference on Learning Representations*, 2020.

[25] Y. Xing, Q. Song, and G. Cheng, "On the generalization properties of adversarial training," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 505–513.

[26] H. Zhang and J. Wang, "Defense against adversarial attacks using feature scattering-based adversarial training," in *Advances in Neural Information Processing Systems*, 2019.

[27] Y. Carmon, A. Raghunathan, L. Schmidt, J. C. Duchi, and P. S. Liang, "Unlabeled data improves adversarial robustness," in *Advances in Neural Information Processing Systems*, 2019.

[28] J.-B. Alayrac, J. Uesato, P.-S. Huang, A. Fawzi, R. Stanforth, and P. Kohli, "Are labels required for improving adversarial robustness?" in *Advances in Neural Information Processing Systems*, 2019.

[29] C. Li, T. Xu, J. Zhu, and B. Zhang, "Triple generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[30] M. Staib and S. Jegelka, "Distributionally robust deep learning as a generalization of adversarial training," in *NIPS workshop on Machine Learning and Computer Security*, 2017.

[31] C. Li, K. Xu, J. Zhu, J. Liu, and B. Zhang, "Triple generative adversarial networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 12, pp. 9629–9640, 2022.

[32] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," *Advances in neural information processing systems*, vol. 30, 2017.

[33] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, 2014.

[34] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[35] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *The collected works of Wassily Hoeffding*, pp. 409–426, 1994.

[36] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[37] Y. Le and X. Yang, "Tiny imagenet visual recognition challenge," *CS 231N*, vol. 7, no. 7, p. 3, 2015.

[38] P. Chrabaszcz, I. Loshchilov, and F. Hutter, "A downsampled variant of imagenet as an alternative to the cifar datasets," *arXiv preprint arXiv:1707.08819*, 2017.

[39] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Advances in Neural Information Processing Systems Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[40] A. Krizhevsky, "Learning multiple layers of features from tiny images," *Citeseer*, 2009.

[41] H. Zhang, Y. Yu, J. Jiao, E. Xing, L. El Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7472–7482.

[42] Z. Wang, T. Pang, C. Du, M. Lin, W. Liu, and S. Yan, "Better diffusion models further improve adversarial training," *arXiv preprint arXiv:2302.04638*, 2023.

[43] T. Karras, M. Aittala, T. Aila, and S. Laine, "Elucidating the design space of diffusion-based generative models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 26 565–26 577, 2022.

[44] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," in *International conference on machine learning*. PMLR, 2020, pp. 2206–2216.

[45] J. Xiao, L. Yang, Y. Fan, J. Wang, and Z.-Q. Luo, "Understanding adversarial robustness against on-manifold adversarial examples," *CoRR*, vol. abs/2210.00430, 2022.

[46] S. Zagoruyko and N. Komodakis, "Wide residual networks. british machine vision conference (bmvc)," 2016.

[47] Y. E. Nesterov, "A method for solving the convex programming problem with convergence rate $o(1/k^2)$," in *Dokl. akad. nauk Sssr*, vol. 269, no. 3, 1983, pp. 543–547.

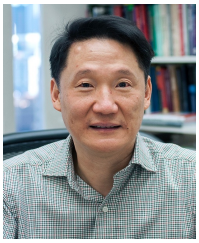
[48] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," in *Artificial intelligence and machine learning for multi-domain operations applications*, vol. 11006. SPIE, 2019, pp. 369–386.

[49] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida, "Spectral normalization for generative adversarial networks," *arXiv preprint arXiv:1802.05957*, 2018.
[50] J. H. Lim and J. C. Ye, "Geometric gan," *arXiv preprint arXiv:1705.02894*, 2017.
[51] I. Kavalerov, W. Czaja, and R. Chellappa, "A multi-class hinge loss for conditional gans," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 1290–1299.
[52] L. Rice, E. Wong, and Z. Kolter, "Overfitting in adversarially robust deep learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 8093–8104.
[53] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.
[54] X. Zhang, X. Zheng, and W. Mao, "Adversarial perturbation defense on deep neural networks," *ACM Computing Surveys (CSUR)*, vol. 54, no. 8, pp. 1–36, 2021.
[55] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.
[56] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*, 2017, pp. 39–57.
[57] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," *CoRR*, vol. abs/1801.02610, 2018.
[58] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017.
[59] A. S. Shamsabadi, R. Sanchez-Matilla, and A. Cavallaro, "Colorfool: Semantic adversarial colorization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1151–1160.
[60] Z. Zhao, Z. Liu, and M. 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*, 2020.
[61] O. Poursaeed, T. Jiang, Y. Goshu, H. Yang, S. Belongie, and S.-N. Lim, "Fine-grained synthesis of unrestricted adversarial examples," *CoRR*, vol. abs/1911.09058, 2019.
[62] X. Wang, K. He, and J. E. Hopcroft, "At-gan: A generative attack model for adversarial transferring on generative adversarial nets," *CoRR*, vol. abs/1904.07793, 2019.
[63] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International Conference on Machine Learning*. PMLR, 2017, pp. 2642–2651.
[64] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019.
[65] S.-A. Rebuffi, S. Gowal, D. A. Calian, F. Stimberg, O. Wiles, and T. Mann, "Fixing data augmentation to improve adversarial robustness," *arXiv preprint arXiv:2103.01946*, 2021.
[66] S. Gowal, S.-A. Rebuffi, O. Wiles, F. Stimberg, D. A. Calian, and T. A. Mann, "Improving robustness using generated data," *Advances in Neural Information Processing Systems*, vol. 34, pp. 4218–4233, 2021.
[67] Y. Xing, Q. Song, and G. Cheng, "Why do artificially generated data help adversarial robustness," *Advances in Neural Information Processing Systems*, vol. 35, pp. 954–966, 2022.
[68] S. Yang, T. Guo, Y. Wang, and C. Xu, "Adversarial robustness through disentangled representations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

Lilin Zhang obtained her bachelor's degree from the School of Information and Security Engineering, Zhongnan University of Economics and Law, in 2021. She is now pursuing the master's degree in the School of Computer Science, Sichuan University, China. Her research interest focuses on adversarial machine learning and its applications.

Ning Yang is an associate professor at Sichuan University, China. He obtained his PhD degree in Computer Science from Sichuan University in 2010. His research interests include adversarial machine learning, graph learning, and recommender systems.

Yanchao Sun is a Ph.D. candidate in University of Maryland, College Park, USA, where she is advised by Prof. Furong Huang. Her research mainly focuses on reinforcement learning (RL), including adversarial RL, multi-task RL, sample-efficient RL, etc.



Philip S. Yu received the PhD degree in electrical engineering from Stanford University. He is a distinguished professor in computer science at the University of Illinois at Chicago and is also the Wexler chair in information technology. His research interests include big data, data mining, and social computing. He is a fellow of the ACM and the IEEE.