What Should They Look Like? Reinforcing Surrogate-based Black-box Attacks with Distribution Feedback

Raha Moraffah
Arizona State University
raha.moraffah@asu.edu

Chaowei Xiao
University of Wisconsin-Madison
cxiao34@wisc.edu

Huan Liu
Arizona State University
huanliu@asu.edu

Abstract

Surrogate-based black-box attacks train a surrogate network to imitate the black-box target and attack the trained surrogate with white-box attacks to craft adversarial examples. These attacks have gained tremendous attention because of their practical setup, where an attacker can attack any samples by only accessing the target for a limited number of samples during the surrogate training. However, surrogate-based attacks suffer from low success rates. We investigate the reason behind their low success rates and demonstrate that surrogate training methods fail to achieve high fidelity to the target, i.e., learn a functionally equivalent surrogate to the target that mimics the target outputs for every input, through empirical and theoretical analysis. Inspired by these results, we propose to rethink the surrogate-based attacks: instead of aiming to train surrogates with high fidelity to the target, we ask if characteristics of adversarial examples can be used as guiding signals to strengthen the attack. We then propose a framework to obtain the characteristics of adversarial examples and a novel plug-and-play adversarial objective that enforces the adversarial characteristics into the existing white-box attacks. Our approach results in attacks with remarkably higher attack success rates than the state-of-the-arts on various targets and datasets.

1. Introduction

The emerging deployment of Deep Neural Networks (DNNs) in safety- and security-sensitive applications [3] has been hindered by their vulnerability to adversarial examples, imperceptibly perturbed samples that lead to erroneous predictions [25]. Studying strong practical attacks is essential for understanding the vulnerability of these networks and robustifying them before real-world deployment.

Depending on the information available to the adversary, the adversarial attacks are white-box or black-box. In white-box attacks, the adversary has full access to the target, including its parameters and architecture, and uses the target's gradients to generate the attack [1, 7]. In black-box attacks, the target is only accessible through their feedback to queries, which is more practical and realistic. Black-box attacks are classified into query-based and surrogate-based attacks. Query-based attacks query the target for *each sample* numerous times during the attack [10, 16]. Despite achieving high success rates, these attacks have limited real-world practicality due to requiring a query budget to the number of attacked samples, which is not feasible in a real-world setting due to limited inference time or monetary limits [10, 12].

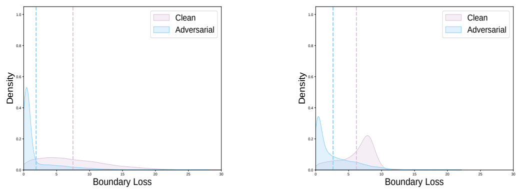


Figure 1. Kernel Density Estimation (KDE) plots for the distribution of boundary losses on CIFAR-10 (right) and CIFAR-100 (left) datasets. The dashed line represents the median boundary loss.

Surrogate-based attacks, on the other hand, provide a more practical attacking scenario. These attacks are designed to attack *any sample* with access to the target's feedback for only a limited number of samples. The idea behind these attacks is to train a surrogate on queried samples and their target feedback to imitate the black-box target. Once trained, the surrogate is used to attack any sample. Since these attacks only query the target during the surrogate training, they are more practical and desirable for real-world applications.

However, existing surrogate-based attacks suffer from low success rates [12]. We revisit the surrogate training of state-of-the-art surrogate-based attacks and demonstrate that the low fidelity of their trained surrogate to the target (i.e., their failure to mimic the target's output for every input) is the reason behind their low success rates. Spurred by this result and motivated by the theoretical evidence of the unfeasibility of learning high-fidelity surrogates (provided in Sec. 3.3), we propose a new perspective on surrogate-based attacks. Instead of aiming for the impossible improvement of the surrogate training to achieve high fidelity to the target as in existing methods [24, 28], we propose to reinforce the attacks applied on the trained surrogate with an additional signal. One possible guiding signal is the characteristics of adversarial examples, i.e., *how the adversarial examples should look like?* Characteristics of adversarial examples can be obtained from their distributions. However, learning the distributions of *all* adversarial examples requires an exponential number of adversarial examples, each requiring numerous queries to the target, which is impossible under the black-box setting. Luckily, a successful attack is only required to identify *some* of the adversarial examples, not all of them. We observe that a decent portion of adversarial examples resides extremely close to/on the target's decision boundaries (i.e., their boundary losses are close to zero), as shown in Figure 1(details of the analysis are described in Appendix). Hence, we propose to leverage the distribution of samples close to/on the target's decision boundaries as a proxy for adversarial examples distributions. Note that even though this distribution contains some real examples as well (as also shown in Figure 1), the synergy between the attack on the surrogate and the characteristics of potential adversarial examples is sufficient to effectively guide the adversarial attack. The potential adversarial example characteristics compensate for the inaccuracy of the surrogate while the feedback from the surrogate helps find the real adversarial examples from the distribution of potential adversarial examples. We henceforth, propose to use this distribution to obtain the guiding signal and call it the potential adversarial examples distribution.

To materialize this idea, we propose a GAN-based architecture to model the distribution of potential adversarial examples. We propose a novel inter-class similarity loss to ensure the distribution is learned for the samples close to/on the target's decision boundaries for all classes. To prevent the model from mode collapse and ensure it learns the distribution effectively, we additionally propose an intra-class diversity loss, which promotes the diversity of samples generated by the model. Furthermore, we propose a novel plug-and-play adversarial objective that enforces the characteristics of the adversarial examples as a guiding signal by forcing the generated example in each attack iteration to resemble the characteristics of that distribution. This guiding signal and the misclassification constraint enforced via the surrogate synergistically guide the attack objective towards learning successful adversarial examples, while compensating for the inaccuracy in surrogate training. Our experiments demonstrate the effectiveness of our proposed method in identifying highly successful adversarial examples in both targeted and untargeted settings for various datasets and target models. Our contributions are summarized as follows:

- We explain the reason for low success rates of the surrogate-based attacks through empirical and theoretical analysis of the state-of-the-art (SOTA) surrogate-based attacks' failure.
- We propose a novel perspective on surrogate-based attacks. Instead of training a more accurate surrogate, we propose a novel plug-and-play adversarial objective. Our objective strengthens the attack by considering the potential characteristics of the adversarial examples.
- We identify the characteristics of the adversarial examples, and model them by learning the distribution of examples that reside close to the targets' decision boundary while possessing intra-class diversity.
- We design experiments to validate if our proposed attack achieves significantly higher attack success rates compared to the SOTAs.

2. Related Work

Adversarial Attacks. White-box adversarial attacks require full access to the target to craft the adversarial attacks, which significantly limits their real-world utility [1, 7, 13]. Black-box attacks adopt a more practical setting, allowing access only through queries. These attacks are divided into query-based and surrogate-based categories. Query-based attacks estimate target gradients dynamically for each sample during the attack [2, 8]. These attacks require querying the target multiple times for each sample, rendering them inefficient for real-world scenarios with query constraints. In contrast, surrogate-based attacks train a surrogate network for the black-box target, aiming to replicate its behavior. Once trained, this surrogate is subjected to attacks to generate adversarial examples, which are used for the black-box target. Surrogate-based attacks are the only viable option when queries are restricted during attacks. Our proposed framework falls under this category, aiming for practicality in real-world situations.

Surrogate-based Attacks. Existing surrogate-based attacks aim to train more accurate surrogate models via improved surrogate training methods. Papernot et al. [20] propose to train a surrogate that imitates the target's output for synthetic images. Orekondy et al. [19] propose to leverage the real and proxy images to steal the target behavior. Recently, a series of works have been developed to generate effective synthetic input data to exploit the decision boundaries of the targets and train more accurate surrogate networks [24, 28, 31]. Another approach involves training generative surrogates to capture input-output joint distributions [17]. Different from the prior work, we propose to strengthen the attack on the trained surrogate with a guiding signal, i.e., characteristics of adversarial examples to compensate for the low fidelity of the surrogates to the target.

Generative Models for Adversarial Attacks. Besides their application in generating effective data for surrogate-based attacks, generative models have been adopted for adversarial attacks under different settings. In the white-box setting, Xiao et al. [29] propose a GAN-based architecture to learn the distribution of the adversarial perturbations [29]. In the query-based setting, Dolatabadi et al. [14] propose to model the distribution of potential adversarial examples for each sample individually using normalizing flow. Generative models have also been used to improve the transferability of adversarial examples by crafting more transferable adversarial perturbations [18, 22]. Our method proposes a GAN-based architecture to model the characteristics of adversarial examples.

3. Methodology

3.1. Attack Setting

Our attack is performed under the surrogate-based black-box setting, where the attacker can access the target feedback for a limited number of samples. In terms of the target response, we will have two scenarios: (1) Label-only Scenario: denoted by “-L” suffix, in this setting the target returns the output labels of the classes for the queries samples; (2) Probability-only Scenario: , the target returns the output class probabilities. We denote this by “-P” suffix.

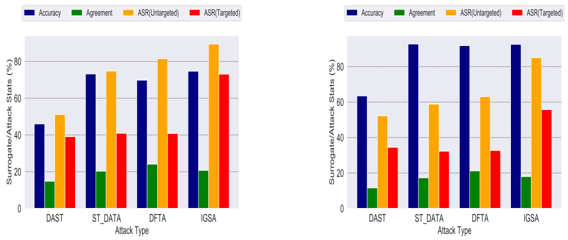
3.2. Surrogate-based Solution to Black-box Attacks

Let $\mathcal{C}(.)\colon x \in [0,1]^d \rightarrow y \in \mathbb{R}^c$ be a target classifier, and (x, y) denote a pair of input image and its true label. An adversarial attack imperceptibly perturbs x into the adversarial example x_{adv}, such that it is misclassified by the $\mathcal{C}$:

$$\begin{aligned} &\underset{x_{adv}}{\operatorname{argmin}} f(x_{adv}, t) \\ &\textbf{s.t. } \|x_{adv} - x\|_p \leq \delta, \end{aligned} \tag{1}$$

where f is the adversarial objective which measures the degree of uncertainty of $\mathcal{C}$ in assigning x_{adv} to class t. The l_p norm ($\|.\|_p$), $p \in \{2, \infty\}$ is used to measure the difference between the original and adversarial examples.

Surrogate-based black-box attacks train a surrogate on a dataset obtained by querying the target with the given set of data samples to imitate the target behavior. The trained surrogate will be attacked with existing white-box attacks for any given sample. The effectiveness of existing surrogate-based attacks depends on the fidelity of their trained surrogates to the target, i.e., how accurately the surrogate imitates the target's functionality. In the following, we provide empirical and theoretical evidence that surrogate training methods do not achieve high fidelity to the target.



(a) Resnet-20 on CIFAR-10 (b) VGG-19 on CIFAR-100

Figure 2. Demonstration of the Accuracy, Agreement (measure of agreement of the surrogate with target), untargeted and targeted attack success rates on different surrogates.

3.3. Surrogate training methods fail to accurately mimic the black-box target

To investigate the reason behind the low success rates of surrogate-based attacks, we plot their surrogates' accuracy (fraction of samples predicted *correctly*), agreement accuracy (fraction of samples that are assigned to the same predictions as the targets' predictions by the surrogate), untargeted attack success rates (fraction of samples that are misclassified after the attack), and targeted success rates (fraction of samples that are misclassified to a pre-specified class after the attack) for the state-of-the-art surrogate-based models DAST [31], ST-Data [24], DFTA [30], and our framework (IGSA). From the results shown in Figure 2, we make the following observation: despite achieving high accuracy (from 60% to more than 90%), SOTA surrogate-based attacks' surrogate obtain extremely low agreement accuracy (less than 20%). This indicates that the surrogate training methods of these attacks fail to train surrogates that accurately mimic the target's outputs. This gives rise to an important question: *How difficult it is to train a surrogate with high fidelity to the target?* Drawing the ideas from model extraction, the following Proposition demonstrates extracting high-fidelity networks can require an exponential number of queries in the depth of the network.

Proposition 1 (Informal [5]) *Random deep network of depth h with domain $\{0,1\}^d$ (d is the input dimension) and range $\{0,1\}$ learned with any Statistical Query (SQ) algorithm such as (stochastic) gradient descent require* $\exp(O(h))$ *samples to learn.*

Proposition 1 implies that training a high-fidelity surrogate to the target is impossible without an exponential number of queries, which is not feasible under the practical black-box setting with only a limited query budget.

Theoretically proven to be infeasible to train high-fidelity surrogates to the target, we propose a new perspective on surrogate-based attacks to compensate for the lack of surrogate fidelity to the target: instead of aiming for im-

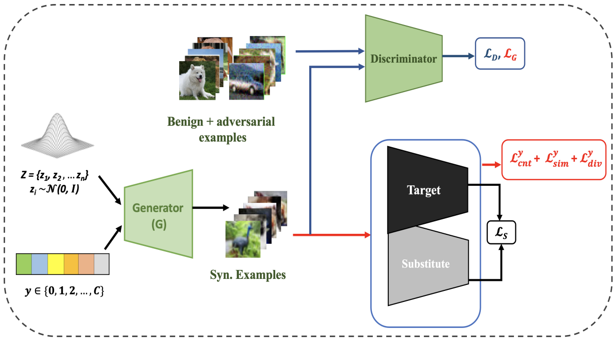

Discriminator
$\mathcal{L}_D, \mathcal{L}_G$
Benign + adversarial examples
$Z = \{z_1, z_2, \dots z_n\}$
$z_i \sim \mathcal{N}(0, I)$
Generator (G)
Syn. Examples
$y \in \{0, 1, 2, \dots, C\}$
Target
Substitute
$\mathcal{L}^y_{cnt} + \mathcal{L}^y_{sim} + \mathcal{L}^y_{div}$
$\mathcal{L}_S$

Figure 3. An overview of step 1 of the IGSA: modeling the distribution of potential adversarial examples and surrogate training. The generator is trained to learn the distribution of examples residing close to/on the target decision boundaries via optimizing Eq. (6). The surrogate is simultaneously trained via Eq. (7). Once trained, the surrogate is attacked with a novel attack strategy which is an augmentation of existing white-box attacks with external signal coming from the generator.

proved surrogate training methods, we propose to reinforce the attack on the surrogate with a guiding signal,i.e., the characteristics of adversarial examples. In the following, we propose a framework to obtain the characteristics of adversarial examples and then propose a novel plug-and-play attack that strengthens the attack by forcing the generated adversarial examples to acquire these characteristics.

3.4. Proposed Method

3.4.1 Framework Overview

We introduce our proposed surrogate-based black-box attack, termed IGSA, as illustrated in Figure 3. IGSA compensates for the lower-fidelity surrogate by taking the characteristics of adversarial examples into account during the attack. Particularly, IGSA consists of two steps: *(S1) Model the distribution of potentially adversarial examples and surrogate training:* building upon our analysis in Sec. 1, this step aims to model the distribution of potential adversarial examples, i.e., examples that are extremely close to/on the target's decision boundaries, to obtain the characteristics of adversarial examples. Moreover, it trains a surrogate on samples drawn from this distribution and queried from the target;*(S2) propose a novel attack strategy:* this step proposes a plug-and-play attack objective that forces the attack to craft samples with the characteristics of adversarial examples obtained from the previous step.

3.4.2 Modeling the distribution of Potential Adversarial Examples and Surrogate training

The goal of this step is two-fold: (1) characterize the distribution of potentially adversarial examples; and (2) train a surrogate network that mimics the behavior of the target around its decision boundaries as accurately as possible. Fortunately, these two goals are inline with each other. In other words, the distribution of potentially adversarial examples also possess characteristics of the useful data required for training surrogates to capture the behavior of the target. Therefore, we propose a holistic framework that performs these two tasks simultaneously: it models the distributions of potential adversarial examples and trains a surrogate on samples drawn from the distributions and their corresponding labels queried from the target.

In particular, based on our analysis in Sec. 1, potential adversarial examples possess two characteristics: (1) high inter-class similarity: they reside extremely close to/on the decision boundaries of the target model; (2) high intra-class diversity: the distribution of potential adversarial examples is required to generate samples that are diversely scattered across all intersections of decision boundaries for all classes. This is particularly important for the targeted attacks which require the adversarial examples to be misclassified as a specific target class, i.e., cross the decision boundary between the original and targeted class.

Learning the distribution To simultaneously learn the distribution of potentially adversarial examples while training the surrogate with the inputs sampled from those distributions, we propose a triple-player Generative Adversarial Network (GAN)-based architecture. Our architecture consists of a generator (G), a discriminator (D), and a surrogate network (S). The generator (G) is responsible for modeling the distribution of potentially adversarial examples, as well as providing the data samples used to train the surrogate. It takes a set of random noise z and a vector of all possible labels $Y = \{0, 1, 2 \dots C\}$ (C is the number of classes) and outputs $X = G(z, Y)$, a set of input images for each class. The realisticness of the generated examples is ensured by feeding them to a discriminator D and optimizing the vanilla GAN adversarial objectives [6]:

$$\begin{aligned} \mathcal{L}_{\mathrm{D}} &= \mathop{\mathbb{E}}_{x \sim \mathbb{P}_d}[\log \mathrm{D}(x)] + \mathop{\mathbb{E}}_{\tilde{x} \sim \mathbb{P}_g}[\log(1 - \mathrm{D}(\mathrm{G}(z, Y)))], \\ \mathcal{L}_{\mathrm{GAN}} &= \mathop{\mathbb{E}}_{\tilde{x} \sim \mathbb{P}_g}[\log(1 - \mathrm{D}(\mathrm{G}(z, Y)))]. \end{aligned} \tag{2}$$

To ensure the distribution learned by the generator represents potential adversarial examples, which are realistic looking, we utilize a mixture of the real clean samples and their corresponding adversarial examples to optimize the objective. The adversarial examples are generated via DeepFool attack [15] (which is different from the white-box attacks used for the experiments) on the surrogate. Note that our framework requires a clean dataset that *has the same class labels as the target's training dataset*. We demonstrate in our experiment that the distribution of samples does not have a noticeable effect on the IGSA's performance (Sec. 4.3.1). To guarantee that generated samples will be assigned the same labels by the surrogate, we extend our adversarial loss with the following loss [24]:

$$\mathcal{L}_{cnt}^{y} = \mathrm{CE}(\mathrm{S}(\mathrm{G}(\mathrm{z}, \mathrm{Y})), Y), \tag{3}$$

where CE is the cross-entropy loss, S is the surrogate network and Y are the labels fed to the generator.

Optimizing Eq. (2) learns the distribution of samples scattered everywhere in the input space. To restrict the distribution to samples close to the target's decision boundaries and guarantee their diversity, we propose two novel Inter-class Similarity and Intra-class Diversity losses.

Inter-class Similarity Loss To restrict the learned distribution to samples close to/on the target's decision boundaries, we propose an inter-class similarity loss that measures the generated sample's distance to its decision boundary and how close it is to crossing it (i.e., be misclassifed to the closest class across the original decision boundary). We use the loss based on the C&W objective to measure the degree to which the samples are likely to be misclassified as in [24]:

$$\mathcal{L}_{sim}^{y} = \max_{j \neq y} \log \mathrm{S}(x)_j - \log \mathrm{S}(x)_y, \tag{4}$$

where $\mathrm{S}(x)_j$ is the probability of the j-th class assigned to x by S, and y is the original class label. Minimizing Eq. (4) decreases the sample's probability of being classified as the original class by pushing it towards the nearest class's decision boundary.

Intra-class Diversity Loss The distribution of potentially adversarial examples for each class is required to cover the entire decision boundary and its intersections with all possible classes. Eq. (2), Eq. (3), and Eq. (4) do not explicitly promote this diversity. On the other hand, GAN's training is prone to mode collapse [26], making it more likely to not learn the desired distribution. We propose a novel intra-class diversity loss to promote the diversity of samples. In particular, the inter-class similarity objective (Eq. (4)), which forces the samples to be close to decision boundaries, makes the samples have nearly-equal highest and second-highest class probabilities. The second-highest probabilities represent the class with which samples share the decision boundaries. If samples are evenly scattered across decision boundaries, their second-highest probability is also evenly distributed on average. This can be measured by the information-entropy of a vector of average probabilities of all classes except for the original one:

$$\mathcal{L}_{div}^{y} = \mathcal{H}\left(\frac{1}{N} \sum_{i}^{N} S_{c:0\ldots C \neq y}^{i}(x)\right), \tag{5}$$

where $\mathcal{H}(P) = -\frac{1}{K} \sum_{i}^{K} p_i$ is the information-entropy of probability vector $P = \{p_1, p_2, \ldots p_k\}$, and $S_{c:0\ldots C \neq y}^{i}(.)$ the probability vector of sample i except for the highest class probability.

Generator Optimization Our final objective is a linear combination of Eq. (2), Eq. (3), Eq. (4), and Eq. (5):

$$\mathcal{L}_G = \mathcal{L}_{\mathrm{GAN}} + \alpha_1 \mathcal{L}_{cnt}^{y} + \alpha_2 \mathcal{L}_{sim}^{y} + \alpha_2 \mathcal{L}_{div}^{y}. \tag{6}$$

The discriminator will be optimized using the Eq. (2).

Surrogate Training Despite infeasibility of learning functionally equivalent surrogate, we aim to learn a surrogate that mimics the target as accurately as possible. To this end, we adopt a knowledge distillation loss, which can be used to force the surrogate to imitate the target's output on a given set of inputs [28, 31]. Formally, given a set of samples X generated by the generator G, we minimize the distance between the surrogate and target's outputs:

$$\mathcal{L}_S = \mathrm{dist}(S(X), T(X)), \tag{7}$$

where $\mathrm{dist}(.)$ is a function to measure the distance between the Surrogate S and the target T's outputs on samples X. In the Label-only scenario, the distance is measured via the Cross Entropy (CE) loss between the class labels produces by the target and surrogate, while in the probability-only scenario it will be measured via the l_2-norm of the difference between the probability outputs $\|T(X) - S(X)\|_2^2$.

3.4.3 Proposed Attack

In this section, we explain how to obtain and incorporate the adversarial example characteristics from the potential adversarial example distributions (learned in the previous step) to reinforce the attack on the surrogate. To ensure a sample possesses the characteristics of a distribution, we need to maximize the probability of the sample belonging to that distribution. The GAN-based generator does not explicitly model the distribution density and only provides samples from the distribution. As a proxy to measure the likelihood of samples belonging to the distribution of generator G, we propose a reconstruction-based loss that measures the distance between the closest sample generated by the generator G and the generated adversarial example:

$$\underset{x_y^* = G(Z^*, y)}{\mathrm{argmin}} \quad \|G(Z^*, y) - x_{adv}\|, \tag{8}$$

In the untargeted setting, where the adversarial example is only required to be misclassified, we identify the closest sample x_y^* for all classes $y = 0, \ldots C$ to the example generated by the white-box attack (Eq. (1)) using Eq. (8) and force the generated adversarial example to be similar to the one with the closest distance:

$$\mathcal{L}_{data} = \|x_{adv} - x_y^*\|. \tag{9}$$

In the targeted setting, the adversarial example is required to be misclassified as a pre-selected target class t.

Table 1. Experimental results of untargeted and targeted attack on CIFAR-10, CIFAR-100, and Tiny-ImageNet datasets. Best results are in bold and second-best results are underlined.

		Untargeted						**Targeted**					
Type	Dataset	CIFAR-10			CIFAR-100		Tiny-ImageNet	CIFAR-10			CIFAR-100		Tiny-ImageNet
	Attack	AlexNet	ResNet-20	VGG-16	ResNet-50	VGG-19	ResNet-50	AlexNet	ResNet-20	VGG-16	ResNet-50	VGG-19	ResNet-50
Probability-Only	IGSA-P	**90.8%**	**89.5%**	**84.9%**	**93.9%**	**85.1%**	**91.9%**	**60.8%**	**73.1%**	**62.9%**	**57.9%**	**55.8%**	**58.3%**
	TDB-P [17]	75%	83.5%	80.1%	78.2%	74.1%	77.6%	42.3%	42.1%	47.8%	39.7%	38.8%	39.2%
	DFTA-P [30]	65.4%	81.5%	78.1%	73.6%	63.1%	69.1%	30.9%	40.8%	38.3%	33.3%	32.7%	33.4%
	ST-Data-P [24]	67.4%	74.8%	75.0%	66.1%	58.2%	64.9%	37.8%	40.9%	44.3%	35.4%	32.3%	30.1%
	DAST-P [31]	52.0%	51.1%	53.2%	69.7%	52.3%	70.8%	37.1%	39.1%	38.6%	34.4%	34.5%	33.8%
	Knock-off-P [19]	37.4%	28.5%	28.0%	27.3%	25.1%	25.6%	22.7%	17.1%	17.9%	19.0%	13.1%	13.8%
	JBDA-P [20]	35.1%	23.6%	19.9%	21.9%	20.1%	14.7%	16.4%	15.1%	15.3%	14.0%	9.2%	11.2%
Label-Only	IGSA-L	**82.8%**	**93.9%**	**88.7%**	**94.2%**	**94.2%**	**86.9%**	**59.9%**	**70.2%**	**64.1%**	**57.2%**	**56.7%**	**57.1%**
	TDB-L [17]	74.7%	80.5%	81.2%	76.1%	75.1%	75.4%	40.9%	41.2%	45.7%	39.1%	35.7%	38.1%
	DFTA-L [30]	63.8%	70.6%	77.5%	74.9%	57.1%	73.8%	34.1%	36.5%	39.3%	30.8%	31.4%	27.1%
	ST-Data-L [24]	63.9%	70.0%	71.3%	65.8%	58.2%	65.2%	35.1%	34.9%	39.1%	32.7%	28.9%	34.1%
	DAST-L [31]	54.6%	49.1%	53.2%	62.8%	49.1%	59.9%	38.1%	40.1%	38.8%	31.4%	32.7%	31.1%
	Knock-off-L [19]	33.2%	24.1%	29.7%	24.1%	26.0%	16.6%	21.9%	15.2%	18.3%	19.1%	12.4%	12.7%
	JBDA-L [20]	35.6%	23.9%	19.5%	22.1%	19.1%	11.8%	16.2%	14.7%	13.8%	14.9%	7.6%	4.8%

In each attack iteration, the generated adversarial example is forced to be closest to the most similar sample (x_t^* generated with Eq. (8)) generated from the distribution of potential adversarial examples for the class t.

The final objective of the attack is as follows:

$$\mathcal{L}_{att}^* = \mathcal{L}_{att} + \lambda \mathcal{L}_{data}, \tag{10}$$

where $\mathcal{L}_{att}$ can be any existing white-box attack.

4. Experiments

We examine three main aspects of IGSA: (1) IGSA's performance compared with the state-of-the-arts under the untargeted and targeted settings; (2) Ablation and parameter study of IGSA; and (3) Qualitative analysis of the IGSA.

4.1. Experimental Setting

4.1.1 Dataset, Target Models, and Whitebox attacks

We utilize three widely-used datasets, namely CIFAR-10 [11], CIFAR-100 [11], and Tiny-ImageNet [21]. Microsoft Azure experiments are provided in the Appendix. Following previous research on black-box attacks [28, 31], we select correctly classified samples in the untargeted setting, and samples not classified as the target class in the targeted setting from the test set of the datasets to attack. For the target architectures, we adopt AlexNet [11], ResNet-20 [9], VGG-16 [23] for CIFAR-10, ResNet-50 [9] and VGG-19 [23] for CIFAR-100, and ResNet-50 [9] for Tiny-ImageNet as used by the state-of-the-arts [24, 28, 30]. To demonstrate the performance of the proposed plug-and-play attack, we adopt three commonly-used white-box attack methods as our $\mathcal{L}_{att}$, namely FGSM [7], C&W [1], and PGD [13]. We use C&W as our default attack method unless otherwise mentioned.

4.1.2 Implementation Details

Our experiments are conducted for targeted and untargeted scenarios under l_2 norm. In the targeted setting, we select the target adversarial class as $y_{adv} = (y_{orig} + 1)$ mode c, where y_{adv} is the target adversarial class, y_{orig} is the original class and "c" is the total number of classes in the dataset.

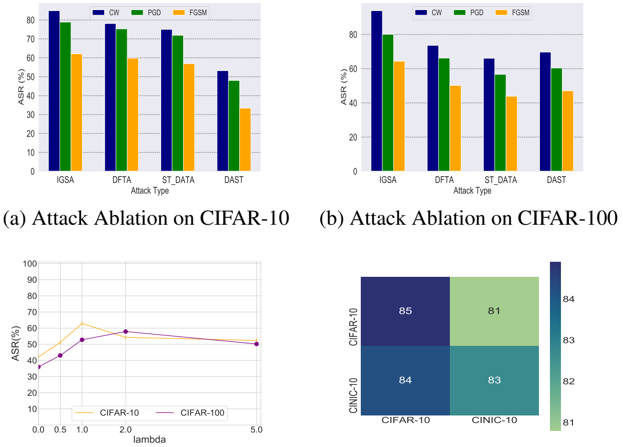
IGSA's architecture consists of a generator, a discriminator, and a surrogate. For the generator and discriminator, we use the same architecture used by [31]. For the surrogate architecture, we use VGG-13 for the CIFAR-10 and Resnet-18 for CIFAR-100 and Tiny-ImageNet. Note that our surrogates are not initialized with pre-trained weights and are trained from scratch. This is to ensure that no prior knowledge of the target is used to conduct the attack. The training set of the attacked dataset is used to train the IGSA, which is completely disjoint from the test sets used to conduct the attack. We use ADAM optimizer to train all of our networks. We use mini-batch size of 500 for CIFAR-10. For CIFAR-100 and Tiny-ImageNet datasets, which have more class categories, we use a bigger mini-batch size of 1000 to achieve a higher diversity of generated samples. We limit the query budget during the surrogate training for all methods to 4M (million) for all methods. The training hyperparameters in Eq.(6), $\alpha_1 = 1$, $\alpha_2 = 1$, and $\alpha_3 = 1$. λ , the attack hyper-parameter in Eq.(10), is selected by varying the parameter in range $\{0.5, 1.0, 2.0, 5.0\}$. For all baselines, we use publicly available implementations and strictly follow their default experimental setups.

4.1.3 Compared State-of-the-art Methods

We compare our performance with three types of state-of-the-art methods: (i) *Attacks with generative surrogate:* **TDB [17]** trains a generative surrogate to mimic the joint distribution of the target on (input, output) pairs; (ii) *At-*

Table 2. Experimental results of Ablation Study on CIFAR-10 and CIFAR-100 datasets on Resnet-20 and VGG-19, respectively.

	Probability-Only										Label-Only									
Data / Type	CIFAR-10					CIFAR-100					CIFAR-10					CIFAR-100				
Base		✓	✓	✓	✓		✓	✓	✓	✓		✓	✓	✓	✓		✓	✓	✓	✓
$+\mathcal{L}^y_{sim}$			✓		✓			✓		✓			✓		✓			✓		✓
$+\mathcal{L}^y_{div}$				✓	✓				✓	✓				✓	✓				✓	✓
Untargeted	69.1%	71.2%	81.5%	75.0%	89.5%	67.6%	70.1%	78.7%	74.3%	85.1%	72.2%	75.1%	83.9%	79.1%	93.9%	68.1%	75.6%	82.9%	77.4%	94.2%
Targeted	46.7%	50.2%	62.1%	55.8%	73.1%	32.4%	35.1%	47.2%	41.0%	55.8%	49.2%	50.1%	59.5%	55.6%	70.2%	33.2%	34.9%	45.3%	40.1%	56.7%



(a) Attack Ablation on CIFAR-10

(b) Attack Ablation on CIFAR-100

(c) Parameter Analysis

(d) Data Ablation Study

Figure 4. Attack ablation and parameter study on CIFAR-10 and CIFAR-100 on VGG-16 and Resnet-50, respectively.

tacks with discriminative surrogates trained on the generator data: Different from our IGSA, they use the generator to generate the input data to query the target and craft a dataset to train the discriminative surrogate. The surrogate aims to generate the same output as the target for the given dataset. The difference between these methods originates from the quality of the data generated by their generator. **DFTA** [30], **ST-Data** [24], and **DAST** [31] are the state-of-the-arts in this category; and (iii) *Attacks with discriminative surrogates trained on traditionally selected data:* these attacks augment or select data from a given dataset to train a discriminative surrogate. **JBDA** [20] and **Knock-off** [19] are two representative methods in this category.

4.2. Comparison with State-of-the-Art Attacks

We evaluate the performance of IGSA and the state-of-the-art surrogate-based attacks under the untargeted and targeted settings and report the results in Table 1. Our results indicate that overall the IGSA significantly outperforms all state-of-the-arts with over 20% of improvement on the average. This validates the effectiveness of utilizing the potential adversarial example distribution's feedback in improving the attack's success rate. Moreover, the improvement is more evident in the targeted setting, where the high fidelity of surrogates is necessary but impossible to achieve. In this case, the role of reinforcing the attack with a guiding signal, i.e., characteristics of adversarial examples, is more prominent. In the following, we elaborate on our in-depth observations: (1) Discriminative surrogate-based attacks trained on traditionally selected data, i.e., JDBA and Knock-off demonstrate the worst performance in all cases. This is because traditionally selected data are not representative of the data distribution near the target's decision boundaries, resulting in surrogates that fail to mimic the target's prediction, thus have low agreement; (2) Attacks with discriminative surrogates trained on the generator data, namely, DFTA, ST-Data, and DAST outperform the discriminative attacks trained on the traditional data, i.e., JDBA and Knock-off. This is because their corresponding generators generate more useful data to train the surrogates with high agreement with the black-box target; (3) TDB, the generative surrogate-based attack, achieves the second-best performance, due to learning the joint distributions of joint distribution of inputs and outputs rather than imitating the target's output for a given set of inputs; and (4) Finally, our proposed IGSA achieves the highest success rates in all cases. This is because IGSA is strengthened with an additional signal, i.e., characteristics of adversarial examples, which compensates for the inevitable unfaithfulness of the learned surrogate to the target.

4.3. Ablation and Parameter Studies

We conduct two types of studies: (1) generator and data ablation study: analyzes the effect of the quality of the potential adversarial example distribution (enforced by $\mathcal{L}^y_{sim}$ and $\mathcal{L}^y_{div}$in Eq. (6)) on the success of the attack and the sensitivity of the IGSA to the training data; and (2) attack and parameter ablation study: examines the effect of guiding signal enforces through $\mathcal{L}_{data}$ in Eq. (10) and the plug-and-play nature of the attack.

4.3.1 Generator and Data Ablation Study

Generator ablation study. We examine the effect of the quality of the distribution learned by the generator and the feedback it provides, on the attack success rate in both untargeted and targeted settings. We fix the surrogate across all attacks (the surrogate trained with the IGSA's generator) and list the variants of our IGSA's attack which uses different generators feedback, by adding the generator constraints, i.e., "Base" ($\mathcal{L}_{\text{GAN}} + \mathcal{L}^{y}_{cnt}$), $\mathcal{L}^{y}_{sim}$, and $\mathcal{L}^{y}_{div}$ one by one. Our results reported in Table 2 indicate the following: (1) "Base" has the lowest success rates among all variants; (2) adding $\mathcal{L}^{y}_{sim}$ and $\mathcal{L}^{y}_{div}$ both improve the success rates, while $\mathcal{L}^{y}_{sim}$ has a higher impact. This is because it forces the generator to learn the distribution close to/on the target's decision boundaries which is where adversarial examples reside in general; (3) adding all constraints (IGSA's generator) leads to the best performance with approximately 20% of improvement over the "Base". This highlights the effectiveness of incorporating characteristics of adversarial examples in the attack when the surrogate (shared across all attacks) is not able to faithfully learn the target.

Data ablation study. As explained in Sec. 3.4.2. IGSA's generator requires a clean dataset to generate realistic-looking samples. To examine the sensitivity of IGSA to the dataset used for training, we train IGSA on CIFAR-10 and a subset of CINIC-10, a dataset of downsampled samples from ImageNet with same labels as CIFAR-10 [4], and report the results in Figure 4d. While in our experiments we mostly utilize the generator training data, our results in Figure 4d illustrate that as long as the data used to train IGSA has the same class labels as the dataset used to train the black-box target, IGSA's attack success rate does not show noticeable difference.

4.3.2 Attack Ablation Study and Parameter Analysis

We demonstrate the effect of employing the characteristics of adversarial examples by varying the parameter λ in Eq. (10) and show the results in Figure 4c. To further demonstrate the plug-and-play nature of our proposed attack, we combine it with different white-box attacks (FGSM, C&W, and PGD) and report the results in Figure 4a and 4b. Our results demonstrate that incorporating the characteristics of adversarial examples improves all white-box attacks.

4.4. Further Analysis

Qualitative Analysis. To further analyze the quality of the data generated by the IGSA's generator, we visualize the t-SNE [27] of all classes and individual classes and show the results in Figure 5. For the sake of comparison, we also visualize the IGSA-Base in which the generator is trained with $\mathcal{L}_{\text{GAN}} + \mathcal{L}^{y}_{cnt}$. Our visualization of all classes illustrates that IGSA achieved higher inter-class similarity and intra-class diversity, as classes are lying closer to each other while being completely distinguishable, and each class shares more boundaries with other classes. Our one-class visualization further shows the diversity of the samples generated by the IGSA compared to the IGSA-Base.

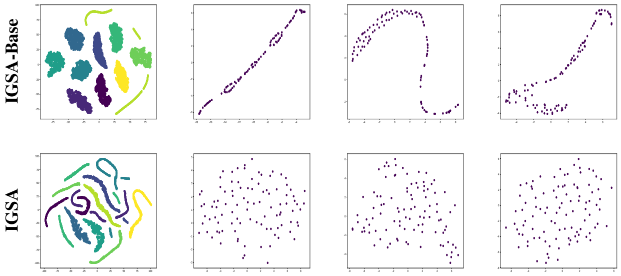
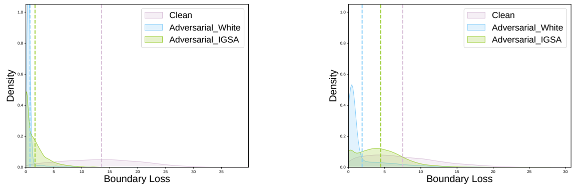


Figure 5. The t-SNE visualization of images for CIFAR-10 on VGG-16. The first column illustrates the visualization for CIFAR-10 all 10 classes the last three are the visualization of 3 randomly selected individual classes.



(a) ResNet-20 on CIFAR-10

(b) VGG-19 on CIFAR-100

Figure 6. Kernel Density Estimation (KDE) plots for the distribution of boundary losses for real and adversarial examples. The dashed line represents the median boundary loss.

Statistical Analysis. We plot the Kernel Density Estimation (KDE) curves of the distribution of boundary loss for the real (clean) samples, adversarial examples generated on the white-box target using its gradients, and adversarial examples generated by our IGSA in Figure 6. The Figure shows that IGSA successfully identifies a decent number of adversarial examples close to/on the target's decision boundary generated using the white-box attacks.

5. Conclusion

In this paper, we investigate the surrogate-based attacks' low success rates through theoretical and empirical analysis and demonstrate that their surrogates fail to achieve high fidelity to the target. We propose a new perspective on surrogate-based attacks, which involves utilizing the adversarial examples characteristics to strengthen the attack on the surrogate. We propose a method to learn the distribution of adversarial examples, while training a surrogate and use that distribution to strengthen the attack. Our framework results in remarkable improvement of 20% over the SOTA. Note that, even though IGSA results in significant improvement, it increases the attack's running time. We plan to explore explicit generative models to improve IGSA's efficiency.

References

[1] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. Ieee, 2017. 1, 2, 6

[2] Pin-Yu Chen et al. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. 2017. 2

[3] Kashyap Chitta, Aditya Prakash, and Andreas Geiger. Neat: Neural attention fields for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15793–15803, 2021. 1

[4] Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. Cinic-10 is not imagenet or cifar-10. *arXiv preprint arXiv:1810.03505*, 2018. 8

[5] Abhimanyu Das, Sreenivas Gollapudi, Ravi Kumar, and Rina Panigrahy. On the learnability of deep random networks. *arXiv preprint arXiv:1904.03866*, 2019. 3

[6] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 4

[7] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2014. 1, 2, 6

[8] Chuan Guo, Jacob Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Weinberger. Simple black-box adversarial attacks. In *ICML*, 2019. 2

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[10] Andrew Ilyas, Logan Engstrom, and Aleksander Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *arXiv preprint arXiv:1807.07978*, 2018. 1

[11] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

[12] Chen Ma, Li Chen, and Jun-Hai Yong. Simulating unknown target models for query-efficient black-box attacks. In *CVPR*, 2021. 1

[13] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *ICLR*, 2018. 2, 6

[14] Hadi Mohaghegh Dolatabadi, Sarah Erfani, and Christopher Leckie. Advflow: Inconspicuous black-box adversarial attacks using normalizing flows. *Advances in Neural Information Processing Systems*, 33:15871–15884, 2020. 3

[15] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *CVPR*, 2016. 4

[16] Raha Moraffah and Huan Liu. Query-efficient target-agnostic black-box attack. In *2022 IEEE International Conference on Data Mining (ICDM)*, pages 368–377. IEEE, 2022. 1

[17] Raha Moraffah, Paras Sheth, and Huan Liu. Exploring the target distribution for surrogate-based black-box attacks. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1310–1315. IEEE, 2022. 2, 6

[18] Muzammal Naseer, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Fatih Porikli. On generating transferable targeted perturbations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7708–7717, 2021. 3

[19] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Knockoff nets: Stealing functionality of black-box models. In *CVPR*, 2019. 2, 6, 7

[20] Nicolas Papernot et al. Practical black-box attacks against machine learning. In *ASIACSS*, 2017. 2, 6, 7

[21] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015. 6

[22] Mathieu Salzmann et al. Learning transferable adversarial perturbations. *Advances in Neural Information Processing Systems*, 34:13950–13962, 2021. 3

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. 6

[24] Xuxiang Sun, Gong Cheng, Hongda Li, Lei Pei, and Junwei Han. Exploring effective data for surrogate training towards black-box attack. In *CVPR*, 2022. 2, 3, 5, 6, 7

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

[26] Hoang Thanh-Tung and Truyen Tran. Catastrophic forgetting and mode collapse in gans. In *2020 international joint conference on neural networks (ijcnn)*, pages 1–10. IEEE, 2020. 5

[27] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 8

[28] Wenxuan Wang et al. Delving into data: Effectively substitute training for black-box attack. In *CVPR*, 2021. 2, 5, 6

[29] Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. Generating adversarial examples with adversarial networks. *IJCAI*, 2018. 3

[30] Jie Zhang, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, Lei Zhang, and Chao Wu. Towards efficient data free black-box adversarial attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15115–15125, 2022. 3, 6, 7

[31] Mingyi Zhou, Jing Wu, Yipeng Liu, Shuaicheng Liu, and Ce Zhu. Dast: Data-free substitute training for adversarial attacks. In *CVPR*, 2020. 2, 3, 5, 6, 7