Transferability of White-box Perturbations: Query-Efficient Adversarial Attacks against Commercial DNN Services

Meng Shen[1], Changyue Li[1], Qi Li[2], Hao Lu[3], Liehuang Zhu[1], and Ke Xu[4]

[1]*School of Cyberspace Science and Technology, Beijing Institute of Technology, China*
[2]*Institute for Network Sciences and Cyberspace, Tsinghua University, China*
[3]*School of Computer Science and Technology, Beijing Institute of Technology, China*
[4]*Department of Computer Science, Tsinghua University, China*
{shenmeng, licy, luhao1999, liehuangz}@bit.edu.cn, {qli01, xuke}@tsinghua.edu.cn

Abstract

Deep Neural Networks (DNNs) have been proven to be vulnerable to adversarial attacks. Existing decision-based adversarial attacks require large numbers of queries to find an effective adversarial example, resulting in a heavy query cost and also performance degradation under defenses. In this paper, we propose the Dispersed Sampling Attack (DSA), which is a query-efficient decision-based adversarial attack by exploiting the transferability of white-box perturbations. DSA can generate diverse examples with different locations in the embedding space, which provides more information about the adversarial region of substitute models and allows us to search for transferable perturbations. Specifically, DSA samples in a hypersphere centered on an original image, and progressively constrains the perturbation. Extensive experiments are conducted on public datasets to evaluate the performance of DSA in closed-set and open-set scenarios. DSA outperforms the state-of-the-art attacks in terms of both attack success rate (ASR) and average number of queries (AvgQ). Specifically, DSA achieves an ASR of about 90% with an AvgQ of 200 on 4 well-known commercial DNN services.

1 Introduction

Deep neural networks (DNNs) are known to be vulnerable to adversarial attacks [17, 55], where a subtle perturbation applied on an image can mislead DNN-based image classification models. This vulnerability poses a critical threat to the security of real-world DNN applications, ranging from face recognition [49] to traffic detection [47, 50]. When targeting commercial DNN services, potential attackers can get only access to the predicted label of a given input, without knowing the details of the DNN model such as the model structure and parameters [4]. For instance, the image recognition services from AWS [1] and Azure [40] only allow users to query and obtain the corresponding results via an API. It is still challenging to construct adversarial attacks against commercial DNN services in the decision-based settings, where the attacker can only access the Top-1 hard-label predictions (i.e., the label with the highest confidence [4]), due to a large number of required queries [27].

Recent years have witnessed a flurry of research and great progress in generating adversarial examples in the decision-based scenario. As shown in Table 1, the existing studies can be categorized into three main categories: transfer-based attacks, query-based attacks, and hybrid attacks.

Transfer-based attacks utilize white-box attacks against substitute models to craft adversarial perturbations. These perturbations are demonstrated to have transferability[1] and can potentially deceive the target model [17, 31, 60]. However, practical scenarios often introduce biases between the target and substitute models, primarily stemming from variations in model architecture and training data, which can usually result in a failure to deceive the target model.

Query-based attacks start from an image with another label (i.e., the target image), and gradually reduce its distance from the original image, while ensuring that the target model can be deceived [4, 9, 29]. Although effective, these attacks require a substantial number of queries to minimize the perturbation, making them costly for deployment in commercial services. These attacks are also susceptible to detection and mitigation by existing defenses [28]. Additionally, query-based attacks often introduce noise in specific regions, limiting their capacity to discover indistinguishable adversarial examples.

Hybrid attacks combine the strengths of both transfer-based and query-based attacks, aiming at reducing the number of queries by leveraging substitute models. For example, previous studies limit the sampling space using the gradient of substitute models [27, 56], or exploit substitute models to establish a favorable starting point for existing query-based attacks to reduce queries [53]. Attackers can easily obtain pre-trained models as substitute models from communities such as Hugging Face [21] and PyTorch Hub [43], making hybrid attacks more appealing. Nevertheless, the state-of-the-art (SOTA) hy-

[1]Transferability means that the same adversarial perturbation can mislead different models [55], which arises because different models often learn the same non-robust features for similar classification tasks [22].

Table 1: Summary of typical decision-based adversarial attacks.

Categories		Methods	Effective	Query Magnitude	Indistinguishable
Transfer-based	Gradient Calculation	FGSM [17], I-FGSM [26], MI-FGSM [12], NI-FGSM [30]	×	1	✓
	Gradient Combination	DIM [64], TIM [13], SIM [30]	×	1	✓
Query-based	Gradient Estimation	HSJA [8]	✓	1.E+5	×
	Boundary Sampling	BA [4]	✓	1.E+6	×
		TA [35]	✓	1.E+5	×
		AHA [29], SurFree [37]	✓	1.E+4	×
Hybrid	Mask Sampling	QEBA-I [27]	✓	1.E+4	×
		BiasedBA [5], BAODS [56]	✓	1.E+3	×
	Boundary Sampling	Prism [24], HybridAttack [53]	✓	1.E+3	×
	White-box Perturbation	**DSA** (This paper)	✓	**1.E+2**	✓

Three metrics are evaluated on ImageNet. Effectiveness means an ASR of over 90%; Query magnitude is measured by the AvgQ required for generating successful adversarial examples; Indistinguishability indicates whether an attack is effective under both ℓ_2 and ℓ_∞ norms. (See more details in Sec. 2.3).

brid attacks heavily rely on query-based attacks [48], resulting in the same limitations as the query-based attacks in terms of query efficiency and indistinguishability.

In this paper, we propose the Dispersed Sampling Attack (DSA), a query-efficient decision-based adversarial attack by exploiting the transferability of white-box perturbations. We design a dispersed sampling mechanism to generate diverse white-box perturbations based on substitute models. These perturbations have different transferability, allowing us to search for an effective one to deceive the target model while achieving indistinguishability.

We design four modules in DSA, utilizing image augmentation, various perturbation constraints, and substitute models to maximize the diversity of white-box perturbations. First, we propose an image augmentation strategy to generate a set of mutations from the original image. These mutated images serve as inputs for launching white-box attacks on substitute models to generate white-box perturbations with different transferability. Second, we investigate the effect of perturbation constraints on the transferability. By choosing an appropriate constraint, we can craft indistinguishable white-box perturbations while ensuring their transferability. Third, we design a substitute model selection algorithm, which leverages the feedback from the target model and tends to select the substitute model that is more likely to produce transferable perturbations. Finally, according to the results of candidate adversarial examples evaluated by the target model, we update the parameters in DSA to search for adversarial examples with smaller perturbations.

To comprehensively evaluate the performance of DSA, we conduct extensive experiments in both closed-set and open-set scenarios, as well as the 4 well-known commercial DNN services, including AWS [1], Azure [40], Baidu [2] and Tencent [57]. The experimental results demonstrate that DSA has higher attack success rates (ASR) while significantly reducing the average number of queries (AvgQ).

We summarize the main contributions as follows:

- We propose DSA, a query-efficient decision-based adversarial attack by exploiting the transferability of white-box perturbations. DSA is built on the key observation that varying the distribution or magnitude of perturbations results in different transferability.
- We conduct closed-set evaluation, where the attacker is assumed to have the same training dataset as that of the target model. DSA can achieve a higher ASR than the SOTA attacks while reducing the AvgQ by at least an order of magnitude. In particular, DSA also outperforms the other attacks when 5 typical defenses of adversarial attacks (e.g., Blacklight [28]) are deployed.
- We evaluate the performance of the attacks in the open-set scenario, where the attacker has no prior knowledge about the training dataset of the target model. The results demonstrate that DSA maintains high effectiveness and query efficiency in the open-set scenario.
- We conduct real-world evaluations on the attacks against 4 popular commercial DNN services. The results show that DSA has the ability to achieve about 90% ASR within 200 queries. Compared with the second-best attack (i.e., HybridAttack [53]), DSA increases the success rate by up to 45.0% and reduces the AvgQ by up to 92.8%.

2 Threat Model and Problem Formulation

In this section, we introduce the threat model of the adversarial attacks on commercial DNN services and formulate the problem with concrete design goals.

2.1 Threat Model

In this paper, we focus on decision-based adversarial attacks against commercial services, involving two primary parties: the victim and the attacker.

The victim refers to a commercial institution that deploys a well-trained model to provide public services. The victim trains the target model and provides Top-1 hard-label predictions for query examples. In addition, the victim may employ some defenses such as input processing [23] and adversarial training [36] to mitigate adversarial attacks.

The attacker is a service subscriber who attempts to find examples with indistinguishable perturbations to deceive the victim's model. The attacker has no knowledge about the parameters or structure of the target model, but can obtain predictions from the victim with a limited query budget. The attacker is assumed to have the capability to construct a shadow dataset and train substitute models using it. We consider two scenarios based on the knowledge about the shadow dataset: closed-set and open-set scenarios [14].

In the *closed-set* scenario, the attacker is assumed to have access to the training dataset of the target model, which is a common assumption in previous literatures [5, 8, 12] and used to evaluate the performance of the attack algorithm. The attacker can obtain the training dataset of the target model using several methods, such as database intrusion.

In the *open-set* scenario, the attacker is assumed to have no access to the training dataset of the target model, and have a shadow dataset that is entirely different from the victim's training dataset [14]. This assumption is more realistic, especially when targeting commercial services.

2.2 Problem Formulation

As stated in the previous subsection, we focus on the decision-based adversarial attacks on a target DNN model for image classification. The target model can be defined as $f : \mathbb{R}^d \rightarrow \{1, ..., K\}$, where d is the dimension of the input image and K is the number of classes. The input for the attacker is a pair of images with different labels, i.e., an original image $\boldsymbol{x}_o$ and a target image $\boldsymbol{x}_t$ ($f(\boldsymbol{x}_o) \neq f(\boldsymbol{x}_t)$). Then, we can define the corresponding adversarial region $\mathcal{O}$ [31] for both untargeted and targeted attacks in Eq. (1),

$$\mathcal{O} = \begin{cases} \boldsymbol{x} \in \mathbb{R}^d : f(\boldsymbol{x}) \neq f(\boldsymbol{x}_o) & (untargeted) \\ \boldsymbol{x} \in \mathbb{R}^d : f(\boldsymbol{x}) = f(\boldsymbol{x}_t) & (targeted) \end{cases} \tag{1}$$

For simplicity, we define an indicator function $\mathbb{I}(\boldsymbol{x})$ to indicate whether an image $\boldsymbol{x}$ is located in the adversarial region, i.e., $\mathbb{I}(\boldsymbol{x}) = 1$ if $\boldsymbol{x} \in \mathcal{O}$, and $\mathbb{I}(\boldsymbol{x}) = 0$ otherwise.

The goal of the attacker is to find an adversarial perturbation δ that is as small as possible, which can be formulated as minimizing the loss function $\mathcal{L}(\delta)$

$$\min_{\delta} \mathcal{L}(\delta) = \mathcal{D}(\delta) + \lambda \cdot (1 - \mathbb{I}(\boldsymbol{x}_o + \delta)) \tag{2}$$

where $\mathcal{D}(\delta) = ||\delta||_p$, and λ is a large number to make sure that $\mathcal{L}(\delta)$ is large enough when $\boldsymbol{x}_o + \delta \notin \mathcal{O}$. The success condition of the attacker is to find an adversarial perturbation within the query budget where the loss is not larger than a pre-defined threshold ε, i.e., $\mathcal{L}(\delta) \leq \varepsilon$.

2.3 Design Goals

Effectiveness. It means that the attacker can successfully generate adversarial examples within a query budget to deceive the target model. It is usually measured by the attack success rate for a certain number of adversarial examples [34].

Query Efficiency. Querying the API of the target model in massive numbers can lead to a heavy overhead for the attacker. For example, typical commercial DNN services may charge \$0.3 to \$1.5 per thousand requests and limit 120 to 1800 requests per minute [1, 2, 18, 40, 57]. Therefore, a good attack algorithm should prioritize high query efficiency.

Indistinguishability. It implies that the modifications on the original image should be subtle and undetectable [16]. Given the widespread use of both the ℓ_2 and ℓ_∞ norms as approximations of indistinguishability, the perturbations generated by the attack should be less than the pre-defined thresholds for these norms.

Robustness. As a number of defenses have been developed to protect against adversarial attacks, such as ComDefend [23] and Blacklight [28], the attack should maintain effectiveness when a certain defense has been deployed on the target model in the closed-set scenario. In the open-set scenario, the attack should also be effective, where the shadow dataset is partially or totally different from the training dataset of the target model.

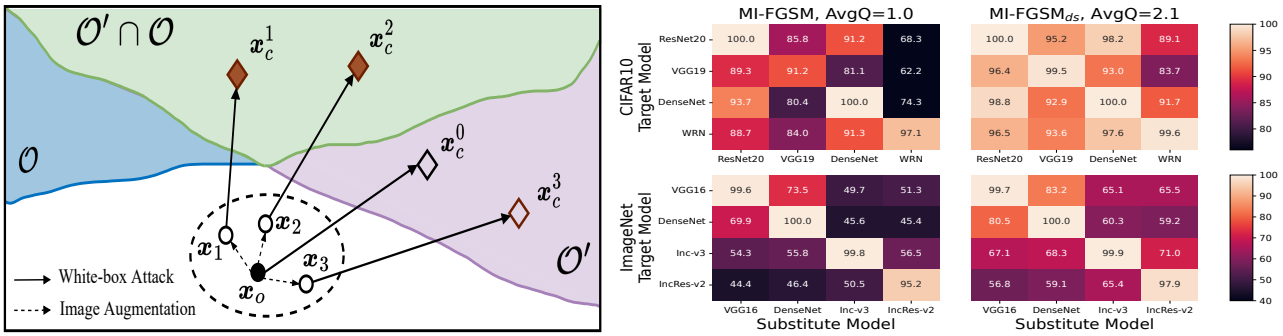
3 The Proposed DSA

In this section, we first present key observations on the transferability of white-box perturbations, and then describe the overview of the proposed method.

3.1 Observation on Transferability of White-box Perturbations

When launching adversarial attacks on a certain original image, the attacker aims to find a small perturbation to deceive the target model. We observe that *varying the distribution or magnitude of perturbations enables different transferability.*

We provide a detailed explanation of our observations through an example of altering the distribution of white-box perturbations by applying image augmentation to the original image. Given an original image $\boldsymbol{x}_o$, we can obtain multiple mutations by performing image augmentation (e.g., adding Gaussian noise on the original image). As illustrated in Fig. 1a, when these mutations are used as inputs for a white-box attack, we can generate perturbations with varying distributions.



(a) Illustration of dispersed sampling

(b) ASR without and with dispersed sampling

Figure 1: (a) Illustration of dispersed sampling. $\mathcal{O}$ and $\mathcal{O}'$ represent the adversarial regions of the target and substitute models, respectively. Given an original image $\boldsymbol{x}_o$, image augmentation can generate several augmented images, e.g., $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$. When applying white-box attacks on the substitute model using these images as input, diverse white-box perturbations can be generated, making the corresponding candidate adversarial examples dispersed in $\mathcal{O}'$. Only those located in $\mathcal{O}' \cap \mathcal{O}$ can deceive the target model. (b) Evaluation of ASR with dispersed sampling on ImageNet. MI-FGSM$_{ds}$ means MI-FGSM with dispersed sampling, which achieves a higher ASR with an average improvement of 9.6%.

These perturbations exhibit different transferability, and the perturbed images have diverse locations in the embedding space[2]. Therefore, we can probe the adversarial region of the target model with these examples.

The reason for this phenomenon is that the augmented image causes a change in the gradient of the neural network backpropagation, which changes the direction of perturbation optimization. The white-box attacks typically perform the above optimization multiple times, which leads to an accumulation of such changes, resulting in a dispersion of outputs (i.e., candidate adversarial examples) in the embedding space.

Existing transfer-based methods [56, 62] only use the original input $\boldsymbol{x}_o$ to generate the corresponding adversarial example $\boldsymbol{x}_c^0$, which may be not located in the adversarial region $\mathcal{O}$ and result in a failed attack. However, as illustrated in Fig. 1a, the candidate adversarial examples generated from augmented images have different transferability. For instance, $\boldsymbol{x}_c^1$ and $\boldsymbol{x}_c^2$ are located in the intersection area of the adversarial region of the target and substitute models (i.e., $\mathcal{O}' \cap \mathcal{O}$). Therefore, they can mislead the target model.

To justify this observation, we conduct experiments to evaluate the ASR of MI-FGSM with dispersed sampling. Specifically, we validate this observation using various substitute and target models with the full validation set of CIFAR-10 and ImageNet. Dispersed sampling generates five candidates for each image to search for a transferable example. As illustrated in Fig. 1b, dispersed sampling allows MI-FGSM to achieve a higher ASR with an average improvement of 9.6%.

[2]The embedding space refers to a lower dimensional representation of the input data that is learned by the DNN [3].

3.2 Overview of DSA

Based on the observation, we propose DSA, which is a query-efficient adversarial attack by exploiting the transferability of white-box perturbations. The basic idea of DSA is to gradually explore the transferability of white-box perturbations until obtaining an adversarial example that can deceive the target model while satisfying the requirement on indistinguishability. The overview of DSA is shown in Fig. 2, which mainly consists of four modules.

Image Augmentation. Based on the observation in Sec. 3.1, we can sample dispersed in the embedding space by performing a white-box attack on different augmented images. Thus, various white-box perturbations with different transferability can be generated, so that we can search for a transferable white-box perturbation by dispersed sampling.

Perturbation Constraint Generation. Previous studies [17, 26] generate white-box perturbations with a small and fixed constraint, which have been shown to be less effective and lead to a low ASR. To achieve a significant distance reduction using a few queries, we should set a reasonable perturbation constraint to balance the magnitude and transferability of perturbations. We generate the constraint of white-box perturbations according to the truncated Gaussian distribution [6], which generates a small-size perturbation while maintaining the transferability of the resulting adversarial example.

Substitute Model Selection. When multiple substitute models are available, the attacker prefers choosing appropriate substitute models to generate candidates with more transferability. Different from the existing studies [12, 31] that search for transferable candidates in the intersection of substitute models' adversarial regions, we generate white-box perturbations in the union of these regions. The reason for this choice

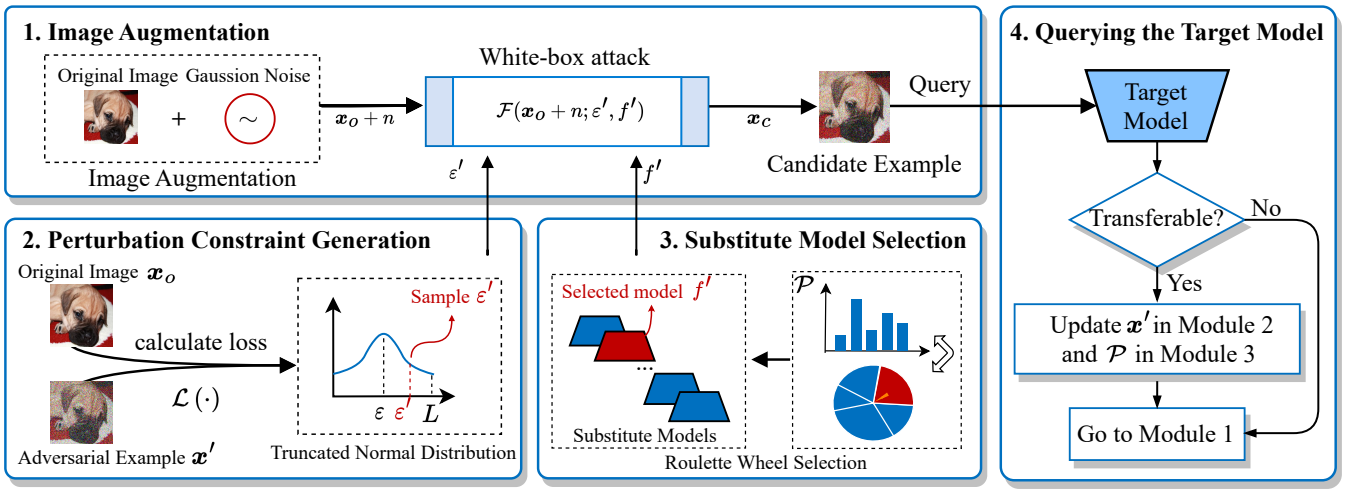


Figure 2: The overview of DSA. It samples from the embedding space and generates a single candidate example by launching a white-box attack on a selected substitute model under a certain perturbation constraint. If the adversarial example successfully deceives the target model, it will be used to update the parameters in substitute model selection and perturbation constraint generation.

will be explained in Sec. 4.3.

Specifically, we choose one substitute model to generate the white-box perturbation at each iteration. We then utilize the query results obtained from the black-box model to adjust the probability of selecting each substitute model. This adjustment increases the probability of selecting the substitute model that generates more transferable candidate examples.

Querying the Target Model. Based on the above modules, we can obtain a white-box perturbation and query the target model. A successful adversarial example is used to update the upper bound of the perturbation size in Module 2 and the selection probability of the substitute model in Module 3. Then, we go back to Module 1 and repeat the entire process, until the specified number of repetitions.

4 Design Details of DSA

In this section, we present the design details of the four modules in DSA, as shown in Fig. 2.

4.1 Image Augmentation

As illustrated in Fig. 1, the basic idea of DSA is to sample in the embedding space of substitute models and craft multiple white-box perturbations with different transferability. Then, it searches for a transferable perturbation to the target model while smaller than the pre-defined threshold. Following [11], we formalize the transferability of untargeted attacks. Given the original image $\boldsymbol{x}_o$ and its ground-truth label y, the transferability of the white-box perturbation $T(\delta)$ can be formulated in Eq. (3),

$$T(\delta) = \mathfrak{L}(y, \boldsymbol{x}_o + \delta) \approxeq \mathfrak{L}(y, \boldsymbol{x}_o) + \delta^\top \nabla_{\mathbf{x_o}} \mathfrak{L}(y, \mathbf{x_o}) \quad (3)$$

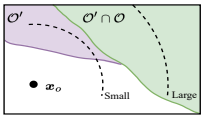
where $\mathfrak{L}$ is the classification loss of the target model. This equation indicates that we can craft perturbations with different transferability by varying the distribution or magnitude of the perturbations. Specifically, we generate multiple white-box perturbations by leveraging image augmentation (e.g., rotation, color jittering, and noise addition), perturbation constraint generation (in Sec. 4.2), and substitute model selection (in Sec. 4.3). Different from the random start in PGD [36], the specific goal of dispersed sampling is to launch black-box attacks by adjusting the distribution and magnitude of perturbations to enable different transferability. In contrast, PGD enhances the effectiveness of adversarial training by introducing noise into images.

For image augmentation, we craft white-box perturbations from a set of mutations of the original image, and the default augmentation strategy is to add Gaussian noise to the original image. We evaluate the effect of different augmentation techniques on the performance of DSA in Appendix A.

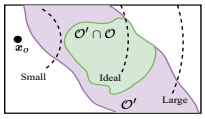
We denote the white-box attack as $\mathcal{F}$, which can generate a white-box perturbation δ for the substitute model f' with a perturbation constraint ε'. Then, the corresponding candidate adversarial example $\boldsymbol{x}_c$ can be obtained

$$\boldsymbol{x}_c = \boldsymbol{x}_o + \delta = \mathcal{F}(\boldsymbol{x}_o + n; \varepsilon', f') \quad (4)$$

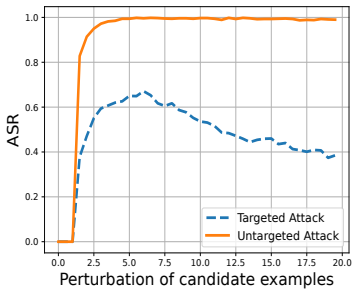
where n is Gaussian noise and the perturbation satisfies $\mathcal{D}(\delta) < \varepsilon'$. Then, the candidate adversarial example $\boldsymbol{x}_c$ is used as the input to query the target model.



(a) Untargeted attacks



(b) Targeted attacks



(c) ASR versus perturbation constraint

Figure 3: Illustration of the influence of perturbation constraint (dash line) on transferability. (a) For untargeted attacks, a large perturbation constraint makes candidate examples easier to transfer. (b) For targeted attacks, a large or small perturbation constraint makes the resulting candidate examples located outside the region $\mathcal{O} \cap \mathcal{O}'$. (c) ASR of MI-FGSM in both untargeted and targeted attacks.

4.2 Perturbation Constraint Generation

The perturbation constraint ε' is defined as the upper limit for the white-box perturbation to be generated. By modifying the value of ε', we can flexibly adjust the magnitude of white-box perturbations. In this section, we investigate the generation of perturbation constraints to obtain the transferable white-box perturbation with a small size.

In the following, we provide an in-depth analysis of the effect of perturbation constraints on the transferability of candidate adversarial examples. For untargeted attacks as shown in Fig. 3a, the adversarial region lies outside the decision boundary of the original image label, and larger perturbation constraints make candidates move beyond the decision boundary and fall into the adversarial region. This observation is consistent with the previous study [11]. Moreover, we systematically analyze the transferability of targeted attacks. For targeted attacks as illustrated in Fig. 3b, only an appropriate constraint can result in transferable candidates, as the adversary region is delimited by the target image labels. If the constraint is too small, candidates cannot reach the adversarial region $\mathcal{O}' \cap \mathcal{O}$, whereas if it is too large, candidates may also fall outside this region.

To justify the above claim, we evaluate the attack success rate of MI-FGSM under different perturbation sizes by generating adversarial examples for 1,000 images in CIFAR-10. ResNet20 and VGG19 are selected as the substitute and target models, respectively. As shown in Fig 3c, increasing the perturbation constraint can improve the ASR of untargeted attacks, until the perturbation constraint reaches 5.0. This demonstrates that a large perturbation constraint enables candidate examples to fall in the adversarial region. For targeted attacks, with the increase of perturbation constraint, the ASR first increases and then decreases, taking a maximum value when the perturbation constraint reaches around 6.0. This also demonstrates the observation illustrated in Fig. 3b.

The transferability is significantly impacted by the magnitude of perturbations. Thus, it is necessary to constrain the magnitude of perturbations with an appropriate upper bound. In order to achieve the goal, we design a mechanism to constrain the magnitude of perturbations.

First, we dynamically establish constraints for each white-box perturbation through a sampling process. It is important to note that both excessively large and small constraints are detrimental to transferability, and particularly large constraints are also not tolerated in adversarial attacks. To address this issue, we control the sampling probability distribution to avoid choosing extreme values. In particular, we derive the perturbation constraint ε' based on a truncated normal distribution $\varepsilon' \sim \mathcal{N}_{[0.5\varepsilon,l]}(\varepsilon, L)$ [6], which is defined as

$$\mathcal{N}_{[0.5\varepsilon,L]}(\varepsilon, L) = \begin{cases} \frac{1}{ZL} \cdot \exp\left(-\frac{(\varepsilon'-\varepsilon)^2}{2L^2}\right), & 0.5\varepsilon \le \varepsilon' \le L \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

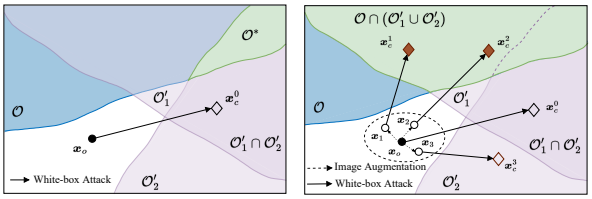
where L represents the upper bound of the sampling and Z refers to the normalization constant to ensure that the probability density function integrates to 1, which is the integral of the probability density function of the normal distribution over the interval $[0.5\varepsilon, L]$ [6]. This distribution ensures a lower probability of sampling values closer to the upper bound.

Second, the attacker cannot directly determine a suitable upper bound L to generate a small perturbation that can be successfully transferred. Therefore, L should be a gradually decreasing variable to ensure continuous generation of smaller transferable perturbations. Specifically, we initialize the upper bound as the distance between the target image and the original image. Once a new transferable perturbation δ is found, we narrow L to $\mathcal{L}(\delta)$ in Eq. (2).

4.3 Substitute Model Selection

The attacker can easily obtain multiple pre-trained models as substitute models from the community such as Hugging Face [21] and PyTorch Hub [43]. It is worth considering improving the performance of attacks using multiple substitute models. Previous transfer-based studies [12, 30] generate a candidate adversarial example in the *intersection* of adversarial regions of all substitute models. Different from these studies, we claim that the *union*, rather than the intersection, of adversarial regions of multiple substitute models should be used to obtain white-box perturbations.

As illustrated in Fig. 4, an effective candidate adversarial example can fall in the union of adversarial regions of multiple substitute models, as it has a large intersection area with the target model. Given m substitute models $\{f'_1, f'_2, ..., f'_m\}$, the adversarial region of the substitute model f'_i can be denoted



(a) Intersection (b) Union

Figure 4: Illustration of sampling in intersection or union of the adversarial regions of multiple substitute models. $\mathcal{O}'_1$ and $\mathcal{O}'_2$ represent adversarial regions of two substitute models. Existing transfer-based attacks ensure the generated candidate example $\boldsymbol{x}_c^0$ located in the intersection $\mathcal{O}'_1 \cap \mathcal{O}'_2$, however, it may not fall in $\mathcal{O}^* = \mathcal{O} \cap \mathcal{O}'_1 \cap \mathcal{O}'_2$. In contrast, as long as a candidate example (e.g., $\boldsymbol{x}_c^1$ and $\boldsymbol{x}_c^2$) is located in $\mathcal{O} \cap (\mathcal{O}'_1 \cup \mathcal{O}'_2)$, it will lead to a successful attack.

as $\mathcal{O}'_i$, and the feasible region can be formulated as

$$\begin{aligned} \mathcal{O}' \cap \mathcal{O} &= \left(\mathcal{O}'_1 \cup \mathcal{O}'_2 \cup \cdots \cup \mathcal{O}'_m\right) \cap \mathcal{O} \\ &= \left(\mathcal{O}'_1 \cap \mathcal{O}\right) \cup \left(\mathcal{O}'_2 \cap \mathcal{O}\right) \cup \cdots \cup \left(\mathcal{O}'_m \cap \mathcal{O}\right) \end{aligned} \tag{6}$$

The above equation indicates that we can sample in the adversarial region of each substitute model, i.e., one substitute model is selected to generate a candidate adversarial example by white-box attacks. Nevertheless, it raises a new challenge that the union of adversarial regions expands the sampling region, and thus the attacker has to consume more queries. The basic idea to address this issue is to select the substitute model f'_k with a larger intersection area in the adversarial region with the target model (i.e., $\mathcal{O}'_k \cap \mathcal{O}$).

During the black-box adversarial attack, the attacker usually generates a series of queries on the target model. We leverage the knowledge provided by the historical queries to select a substitute model. Aiming at improving the query efficiency of the attack, we assign substitute models with different probabilities of being selected to generate more white-box perturbations on parts of the union region $\mathcal{O}'$. At the start of the attack, each substitute model has the same probability of being selected. As the attack progresses and more transferable candidates are generated from a substitute model, the probability of selecting this substitute model increases. Specifically, we apply the roulette wheel algorithm to select the substitute model and dynamically adjust its probability of being selected during the attack. The probability of substitute model f'_i being selected is

$$\mathcal{P}\left(f'_i\right) = \frac{q\left(f'_i\right)}{\sum_{j=1}^{m} q\left(f'_j\right)} \tag{7}$$

where $q\left(f'_i\right)$ denote the fitness of i-th substitute model, which is defined as the number of transferable examples generated by the i-th substitute model.

$$q\left(f'_i\right) = \sum \mathbb{I}\left(\boldsymbol{x}'_{f'_i}\right) + b \tag{8}$$

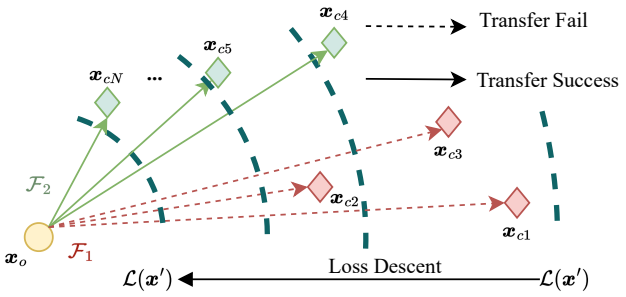


Figure 5: Geometrical configuration of DSA. DSA disperses the sampling of candidate adversarial examples. Once a transferable candidate is obtained, the upper bound of the perturbation constraint and the probability of each substitute model being selected are updated.

where $\boldsymbol{x}'_{f'_i}$ denotes historical candidate examples generated from the substitute model f'_i, b is an initial value and is set to 10 in the implementation, which can be used to adjust the influence of feedback from the target model. Note that the fitness is calculated using historical queries; no additional queries to the target model need to be introduced. The fitness of each substitute model reflects the intersection of its adversarial region with the target model, which is independent of the original images, so that the fitness can be shared when attacking different images.

4.4 Querying the Target Model

The algorithm of DSA is illustrated in Algorithm 1. There are N epochs in DSA, and for each one, we use the current loss as the upper bound of truncated Gaussian distribution and sample the perturbation constraint ε'. One substitute model f'_i is selected based on the probability distribution $\mathcal{P}$. Then, a candidate adversarial example $\boldsymbol{x}_c$ with the perturbation constraint is generated by attacking the substitute model. This candidate is used to query the target model.

DSA updates the parameters based on the query results to limit the sampling region. As shown in Fig. 5, if the candidate example can deceive the target model, we use it as the current adversarial example $\boldsymbol{x}'$, and its perturbation size as the upper bound of the constraint generation, which results in candidate examples being generated with less perturbation. At the same time, we update the fitness of each substitute model and the probability of roulette wheel selection, which makes the currently selected substitute model easier to select in the following. Eventually, we go to the first module to repeat the entire process. If the candidate example cannot deceive the target model, we move into the next epoch directly.

The sampling process of DSA is repeated N epochs. In the end, we obtain N candidate examples and consume the same number of queries. The appropriate value of N is related to the number of substitute models, as more substitute models mean

Algorithm 1: Dispersed Sampling Attack

Input: Original image $\boldsymbol{x}_o$, target image $\boldsymbol{x}_t$, loss function $\mathcal{L}$, perturbation threshold ε, white-box attack function $\mathcal{F}$, probability distribution of substitute models $\mathcal{P}$, number of epochs N.

Output: Adversarial example $\boldsymbol{x}'$.

$\boldsymbol{x}' \leftarrow \boldsymbol{x}_t$;
for 1 **to** N **do**
  $L \leftarrow \mathcal{L}(\boldsymbol{x}')$;
  $\varepsilon' \sim \mathcal{N}_{[0.5\varepsilon, L]}(\varepsilon, L)$;
  $f' \sim \mathcal{P}$;
  $n \sim \mathcal{N}(0,1)$;
  $\boldsymbol{x}_c \leftarrow \mathcal{F}(\boldsymbol{x}_o + n; f', \varepsilon')$;
  if $\mathcal{L}(\boldsymbol{x}_c) < \mathcal{L}(\boldsymbol{x}')$ **then**
    $\boldsymbol{x}' \leftarrow \boldsymbol{x}_c$;
    Update $\mathcal{P}$ using Eq. (7) ;
  end
end
Optimizing $\boldsymbol{x}'$ with existing black-box attack ;
return $\boldsymbol{x}'$;

a larger area to sample, which requires more queries for exploration. Therefore, we set $N = m \times n$, where m is the number of substitute models, and n is a specified value. Specifically, we set n to 100, and evaluate the effects of different n values, which can be found in Appendix A.

After the last epoch, we obtain an adversarial example, which is the candidate with minimal perturbation. Then we refer to existing black-box attacks to search for adversarial examples with less loss. We argue that dispersed sampling plays a significant role, while the integrated black-box attack serves only as a complement. This can be demonstrated in experiments in Sec. 5.2, where perturbations decrease by over 90% in the dispersed sampling.

5 Performance Evaluation

In this section, we comprehensively evaluate the performance of DSA by comparing it with the SOTA attacks. We describe the experimental settings in Sec. 5.1. Then, we conduct experiments in the closed-set scenario in Sec. 5.2, which investigates the attack performance on both undefended and defended models. Next, we perform experiments in the open-set scenario in Sec. 5.3, and evaluate the attack performance against four commercial DNN services in Sec. 5.4. Finally, we conduct an ablation study in Sec. 5.5, assessing the contribution of each module and the scalability of DSA.

5.1 Experimental Settings

Datasets and Evaluation Metrics. We conduct the experiments using the CIFAR-10 [25] and ImageNet [46] datasets, which have been widely used in previous studies [4, 53]. CIFAR-10 consists of 10 classes and each class has 6K images with a size of $32 \times 32 \times 3$. ImageNet has 1,000 classes of images which are re-scaled to the size of $299 \times 299 \times 3$. All image pixel values are normalized in the range of $[0, 1]$. Following the setup in previous studies [5, 8], 1,000 original-target image pairs are randomly selected from the validation set of each dataset for evaluation. The selected images are correctly classified by all models used in the experiments, and the target image has a different label from the original image. In the closed-set scenario, the target and substitute models are trained on the same dataset. In the open-set scenario, we assemble two mutually exclusive datasets for training the substitute and target models. Specifically, we randomly select half of the images for each label and use them as one training set, while the remaining images constitute the other set.

The success condition for the attacker is to find an adversarial example $\boldsymbol{x}'$ with a loss less than the pre-defined threshold ε within the query budget. The query budget for each image pair is set to 4,000. During the attack process, any attack that exceeds the budget is considered failed and is terminated immediately; if the loss of an adversarial example is less than the threshold ε within the query budget, we consider the attack successful and stop querying the target model. We use the attack success rate (ASR) and the average number of queries (AvgQ) as evaluation criteria, which are defined as the ratio of original images that can be successfully attacked and the average number of queries on the target model consumed on each image, respectively. An attack is desired to achieve a high ASR at the cost of a low AvgQ.

We evaluate the performance of the attacks under two perturbation distance metrics, ℓ_2 and ℓ_∞ norm, which are widely used in previous studies [8, 29], and can fully benchmark the indistinguishability of the attacks. Follow the setting in previous study [10], the loss threshold is set at $\varepsilon = \sqrt{0.001 \cdot d}$ under ℓ_2 norm, (i.e. 1.75 on CIFAR-10 and 16.38 on ImageNet), and $\varepsilon = 16/255$ under ℓ_∞ norm, which are pretty small values compared with the existing decision-based attacks [5, 29].

Target and Substitute Models. For each dataset, we select 4 models that are widely used in existing studies [12, 30]. On CIFAR-10, we consider ResNet20 [19], VGG19 [52], DenseNet100 [20] and WRN28 [65], with the same structure and weights as the previous work [34]. The top-1 error rates of these four models are 8.23%, 6.72%, 4.73%, and 4.07%, respectively. On ImageNet, we select VGG16 [52], DenseNet121 [20], Inception v3 (Inc-v3) [54], Inception ResNet v2 (IncRes-v2) [59]. The implementation and pre-trained weights of models are based on GitHub repository[3], and the top-1 error rates of these models are 28.37%, 25.36%, 22.71% and 19.60%, respectively. The four models are used in turn as the target model unless otherwise specified.

Following the setting in the previous study [14], if the struc-

[3] https://github.com/Cadene/pretrained-models.pytorch

Table 2: The closed-set results (ASR% / AvgQ) of untargeted attacks on ImageNet.

	ℓ_2								ℓ_∞							
	VGG16		DenseNet121		Inc-v3		IncRes-v2		VGG16		DenseNet121		Inc-v3		IncRes-v2	
	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ
BiasedBA	99.6	1544.4	98.1	1703.8	96.6	1763.8	95.2	1838.3	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0
QEBA-I	41.2	2696.6	22.3	3307.7	15.0	3535.2	9.9	3704.3	14.6	3454.3	9.6	3651.8	6.5	3762.0	3.6	3866.1
BAODS	98.9	798.9	97.0	1109.7	94.9	1368.7	95.2	1402.6	7.2	3853.0	4.4	3904.5	2.5	3948.3	1.9	3962.4
Prism	87.3	611.5	58.6	1736.7	54.4	1903.7	47.6	2165.0	74.5	1139.1	45.7	2258.5	41.9	2401.8	32.6	2763.8
HybridAttack	96.5	520.1	91.8	723.5	88.6	837.8	85.2	1041.2	75.6	996.8	72.2	1118.2	70.4	1185.9	64.2	1435.2
DSA	**99.9**	**54.2**	**99.3**	**86.3**	**99.3**	**67.2**	**99.1**	**91.0**	**96.9**	**136.7**	**93.8**	**263.7**	**93.8**	**267.0**	**91.0**	**386.0**

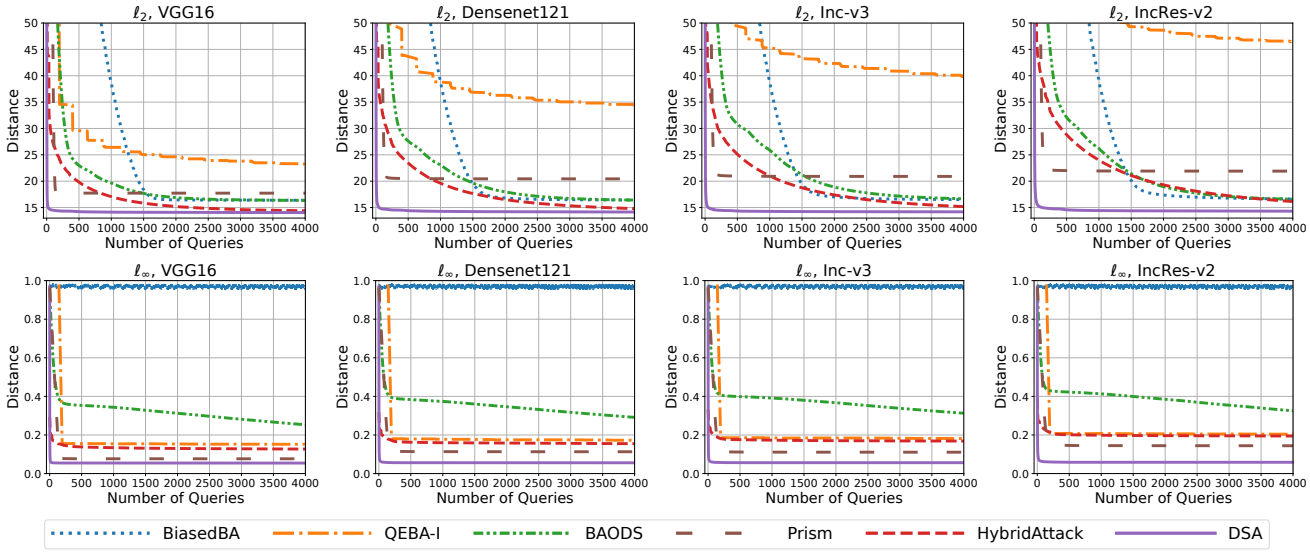


Figure 6: The ASR curves of untargeted attacks on ImageNet as the number of queries increases.

ture of the target model is the same as one of the four models, we select the rest three as substitute models; otherwise, all four models are used as substitute models. To fully demonstrate the superiority of DSA, we also evaluate the attacks with a single substitute model in Sec. 5.3.1.

Methods to Compare. Five SOTA hybrid attacks serve as baselines, including BiasedBA [5], QEBA-I [27], BAODS [56], Prism [24], HybridAttack [53]. They have shown a wide range of superiority [56]. All attacks are implemented based on the Foolbox library [45], which allows attacks to be well adapted to both ℓ_2 and ℓ_∞ norms. As DSA and HybridAttack integrate existing white- and black-box attacks, MI-FGSM [12] and SurFree [37] are selected as representative methods. For fair comparisons, DSA and other methods have the same attacker capabilities, with the same target model and substitute model settings in all scenarios.

5.2 Experiments in Closed-set Attack Scenario

In this section, we conduct closed-set experiments to evaluate the performance of adversarial attacks. The attacker is assumed to have the same training dataset as the target model for training substitute models, which is consistent with the experimental settings of most previous literatures [5, 12, 53]. Given that the victim may deploy defenses to mitigate adversarial attacks, we evaluate the attack performance in both undefended and defended scenarios.

5.2.1 Evaluation in Undefended Scenario

We conduct untargeted attacks on ImageNet[4], and we use the other three models as substitute models when evaluating one target model.

As summarized in Table 2, DSA achieves a significant improvement in attack effectiveness with different target models and distance metrics. Under the ℓ_2 norm, DSA achieves an ASR of more than 99% within an AvgQ of 100, reducing AvgQ by up to 92% over the second-best values. Under ℓ_∞ norm, DSA still achieves an ASR of over 90% using only a couple of hundred queries, with an improvement of ASR by 21.3% and an AvgQ reduction of more than 73.1% over the existing methods against all target models.

To further demonstrate the superiority of DSA, we plot the change curve of mean distance as the number of queries

[4]Targeted attacks are still a tough task for existing attacks due to the large number of classes in ImageNet. We leave it for future work.

Table 3: The closed-set results (ASR% / AvgQ) of untargeted attacks on defended models on CIFAR-10.

		Undefended		Blacklight		RND-GF		PCL		ComDefend		MagNet		AdvTrain		AdvTrain$^{\diamondsuit}$	
		ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ
ℓ_2	BiasedBA	100.0	1461.9	0.0	4000.0	93.9	1968.2	99.9	1518.2	56.2	3037.0	42.2	3431.8	33.4	3528.3	36.6	3465.4
	QEBA-I	100.0	585.2	10.6	3695.0	98.7	1102.5	99.9	610.4	19.3	3427.9	96.3	947.7	28.0	3364.0	60.0	2847.9
	BAODS	100.0	350.4	0.0	4000.0	92.6	1899.9	99.6	613.8	9.4	3713.8	94.5	1152.5	7.2	3848.4	27.8	3359.4
	Prism	99.6	121.1	0.0	4000.0	37.2	2667.0	80.1	888.6	22.8	3139.3	52.9	1955.4	3.4	3867.8	4.1	3841.8
	HybridAttack	100.0	60.1	89.7	419.9	99.9	234.1	100.0	108.3	89.4	746.5	99.3	201.7	57.6	2527.4	62.3	2096.6
	DSA	100.0	**3.7**	**100.0**	**3.6**	**100.0**	**31.0**	**100.0**	**6.5**	**93.8**	**499.6**	**99.6**	**29.6**	**61.2**	**2521.1**	**83.4**	**953.7**
ℓ_∞	BiasedBA	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0
	QEBA-I	66.0	2075.7	39.4	2740.7	22.3	3433.6	64.6	2123.8	28.8	3151.1	30.5	3261.8	5.8	3806.7	12.7	3544.2
	BAODS	14.8	3682.8	0.0	4000.0	2.4	3950.3	18.2	3622.0	2.1	3925.8	5.6	3879.9	2.3	3925.4	8.0	3742.6
	Prism	94.3	373.9	10.4	3599.2	47.7	2180.7	77.3	1056.8	19.2	3320.4	0.0	4000.0	2.3	3913.6	3.3	3876.4
	HybridAttack	98.6	65.2	90.8	369.5	91.8	344.8	94.6	255.2	59.4	1771.4	93.1	298.6	**9.2**	3686.0	38.8	2449.5
	DSA	100.0	**4.5**	**100.0**	**4.6**	**99.2**	**41.6**	**100.0**	**7.1**	**72.8**	**1136.0**	**98.6**	**69.6**	9.0	**3647.4**	**56.3**	**1759.3**

$\diamondsuit$ represents that the attacker has a substitute model obtained by adversarial training.

increases in Fig. 6. As the query increases, the distance of each method decreases to different degrees. Compared with other methods, DSA always achieves the lowest distance under all test cases when consuming the same number of queries.

We find that the perturbation distance metric ℓ_∞ has a negative impact on attack performance compared with the results under ℓ_2 norm. In particular, the distance of BiasedBA is consistently around 1 under the ℓ_∞ norm. This is because BiasedBA tends to add larger noise to local pixels on current adversarial examples and cannot generate examples with a smaller distance under the ℓ_∞ norm, resulting in optimization stagnation.

5.2.2 Evaluation in Defended Scenario

Here, we focus on the evaluation of attacks in the defended scenario. Following the settings in previous literature [34], we conduct untargeted attacks on CIFAR-10, using ResNet50 [19] integrated with several defenses as target models. The defenses are listed as follows.

- Blacklight [28]. It detects adversarial examples with an efficient similarity engine that detects similarities between queries on the input space.
- RND-GF [44]. It attempts to add large Gaussian noise on the queried image to disrupt the subtle structure of the adversarial perturbation, allowing the model to output the correct result.
- PCL [41]. It forces the features of each class to lie within a convex polygon that is maximally separated from the polygons of the other classes.
- ComDefend [23]. It consists of a compressed convolutional neural network and a reconstructed convolutional neural network, which can convert adversarial images into their clean version.
- MagNet [39]. It includes a detector network and a reformer network. The detector network aims to identify adversarial examples, while the reformer network moves adversarial examples toward normal examples for correct classification.
- AdvTrain [36]. It trains the model on a mixture of images with adversarial perturbations to make the model robust to adversarial attacks.
- AdvTrain$^{\diamondsuit}$. It is used to evaluate the effect of AdvTrain against a sophisticated attacker [12]. Under this setting, the attacker has access to an additional substitute model obtained through adversarial training. Specifically, we utilize ResNet20 as the additional substitute model and use PGD [36] to generate adversarial examples. These adversarial examples are then used to train ResNet20, enhancing its classification accuracy under adversarial examples. Note that the hyperparameters of the training process (i.e., learning rate and number of epochs) differ from those used in the target model. All attacks are conducted under the same assumption for a fair comparison.

The performance of untargeted attacks on CIFAR-10 in the defended scenario is exhibited in Table 3. DSA is robust in attacking the defended target models and achieves the highest ASR with the least AvgQ in most tests.

We classify existing defenses into four categories. The first category, including RND-GF and ComDefend, tries to defend against adversarial perturbations by disrupting their subtle structures so that the target model can still achieve correct classifications. For example, ComDefend reduces the ASR of BAODS from 100% to 9.4% under ℓ_2 norm. However, DSA can directly generate candidate adversarial examples deep in the adversarial region and achieves a 93.8% ASR with 499.6 AvgQ under ℓ_2 norm under ComDefend.

The second category, consisting of PCL, AdvTrain, and AdvTrain$^{\diamondsuit}$, increases the margin between normal images and adversarial examples for correct classifications. For instance, AdvTrain trains the target model with adversarial examples, making it difficult to generate transferable perturbations. The results of AdvTrain$^{\diamondsuit}$ reveal that different models trained

Table 4: The open-set results (ASR% / AvgQ) of untargeted attacks (U) and targeted attacks (T) on CIFAR-10. (* denotes the target and substitute models use the same DNN structure)

		ℓ_2								ℓ_∞							
		ResNet20*		VGG19		DenseNet		WRN		ResNet20*		VGG19		DenseNet		WRN	
		ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ
U	FIA	75.1	**1.0**	62.1	**1.0**	74.5	**1.0**	47.8	**1.0**	92.6	**1.0**	87.9	**1.0**	91.9	**1.0**	73.1	**1.0**
	DaST	6.8	4000.0	2.7	4000.0	7.3	4000.0	5.3	4000.0	30.8	4000.0	12.6	4000.0	23.8	4000.0	19.2	4000.0
	DST	11.3	4000.0	4.8	4000.0	8.1	4000.0	3.7	4000.0	29.4	4000.0	15.8	4000.0	25.1	4000.0	12.5	4000.0
	BiasedBA	99.9	1577.9	99.1	1698.2	99.9	1575.8	91.8	2065.3	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0
	QEBA-I	**100.0**	547.1	99.6	926.8	99.9	583.9	98.2	987.7	63.1	2189.1	33.6	3023.1	56.9	2342.3	32.2	3115.2
	BAODS	94.7	1166.5	86.2	1841.0	93.9	1171.5	78.5	2020.2	15.0	3664.5	8.2	3841.2	19.4	3564.5	8.7	3807.8
	Prism	69.8	1299.3	52.4	1982.1	71.9	1220.2	39.0	2494.3	67.4	1429.0	48.7	2152.0	67.8	1420.8	34.3	2691.8
	HybridAttack	**100.0**	106.1	**100.0**	244.7	**100.0**	143.8	**99.9**	334.3	**91.5**	438.9	85.5	676.1	94.1	336.1	76.1	1119.6
	DSA	**100.0**	22.3	**100.0**	65.8	**100.0**	30.3	**99.9**	154.4	**97.9**	92.9	**94.9**	214.6	**97.7**	99.9	**97.5**	514.7
T	FIA	16.4	**1.0**	13.2	**1.0**	16.3	**1.0**	12.5	**1.0**	25.8	**1.0**	25.1	**1.0**	22.5	**1.0**	21.8	**1.0**
	DaST	0.9	4000.0	0.1	4000.0	1.0	4000.0	0.6	4000.0	4.5	4000.0	1.6	4000.0	4.6	4000.0	1.7	4000.0
	DST	1.8	4000.0	0.7	4000.0	1.0	4000.0	0.5	4000.0	4.1	4000.0	2.2	4000.0	3.1	4000.0	1.2	4000.0
	BiasedBA	85.6	2944.1	71.7	3165.5	75.5	3142.2	58.0	3327.8	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0
	QEBA-I	96.0	1583.0	73.9	2424.0	93.6	1756.5	80.2	2258.6	14.2	3648.0	5.1	3865.4	8.4	3812.8	4.5	3875.0
	BAODS	59.8	2570.2	45.4	3087.5	59.3	2564.4	36.7	3284.4	3.3	3934.5	1.4	3976.6	3.0	3929.6	1.3	3967.3
	Prism	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.1	3996.2	0.1	3996.3	0.1	3996.1	0.2	3992.3
	HybridAttack	**100.0**	749.8	97.6	1138.9	**99.9**	774.7	97.6	1186.6	39.1	2637.7	28.2	3043.9	40.8	2629.3	24.1	3162.1
	DSA	**100.0**	495.1	**97.6**	877.1	**99.9**	633.6	**97.7**	1042.2	**54.4**	1877.0	**41.4**	2409.1	**48.7**	2189.5	**31.3**	2781.9

Table 5: The open-set results (ASR% / AvgQ) of untargeted attacks on ImageNet

	ℓ_2								ℓ_∞							
	VGG16		DenseNet121		Inc-v3		IncRes-v2		VGG16		DenseNet121		Inc-v3		IncRes-v2	
	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ
BiasedBA	88.4	1931.0	65.7	2632.3	88.9	1940.3	88.0	1960.1	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0
QEBA-I	51.8	2304.9	29.1	3067.9	44.8	2534.7	36.8	2864.9	15.2	3443.3	12.4	3547.0	10.6	3611.3	9.8	3639.6
BAODS	96.5	1248.9	86.0	1957.3	98.0	1094.5	97.5	1150.8	3.9	3930.2	2.4	3935.1	3.8	3919.4	2.6	3942.6
Prism	65.1	1486.8	42.8	2353.7	64.5	1506.7	71.9	1215.2	57.3	1815.1	41.1	2432.3	54.7	1913.3	54.7	1903.9
HybridAttack	96.1	584.8	89.9	999.4	95.2	646.2	93.3	673.5	68.7	1260.2	61.1	1555.1	71.4	1146.1	76.0	958.7
DSA	**98.7**	**152.0**	**96.1**	**355.5**	**99.2**	**90.1**	**99.1**	**94.2**	**89.3**	**447.5**	**81.6**	**760.9**	**93.1**	**294.4**	**95.1**	**212.7**

through adversarial training could share similar vulnerabilities against adversarial attacks, allowing evading AdvTrain. Compared with AdvTrain, DSA achieves superior results in terms of both ASR and AvgQ, with a maximum 47.3% improvement in ASR and a 62.2% decrease in AvgQ.

The third category is stateful. The defenses like Blacklight record historical queries and detect their similarity to suppress attacks that query around current examples. BiasedBA, BAODS, and Prism can only achieve an ASR of less than 10.4%. Nevertheless, DSA generates diverse adversarial examples in the embedding space. These examples have different distributions and magnitudes of perturbations, which are not similar and cannot be captured by Blacklight.

Finally, the fourth category, including MagNet, aims to detect out-of-distribution samples, which can effectively detect images with obvious perturbations. Under MagNet, the ASR of BiasedBA decreases from 100.0% to 33.4% and the AvgQ increases from 1461.9 to 3431.8. Candidate adversarial examples generated by DSA may be located near the decision boundary and appear similar to normal images. Consequently, DSA achieves 99.6% ASR with 29.6 AvgQ under ℓ_2 norm and 98.6% ASR with 69.6 AvgQ under ℓ_∞ norm.

5.3 Experiments in Open-set Attack Scenario

In this section, we evaluate the robustness of attacks in open-set scenarios, where the attacker does not have access to the training dataset of the target model, as described in Sec. 2.1.

5.3.1 Evaluation with Single Substitute Models

In this subsection, we perform untargeted and targeted attacks on CIFAR10 in the open-set scenario with ResNet20 as the single substitute model. The training of the substitute model does not rely on the knowledge of the target model training dataset. Additionally, we also include the comparison with FIA [63], DST [61], and DaST [67]. Note that FIA is a sophisticated transfer-based attack (see Sec. 6), which requires only one query to generate an adversarial example and cannot adjust the settings based on feedback from the target model. As shown in the Table. 4, DSA can achieve the highest ASR of more than 97.6% with a relatively low AvgQ under ℓ_2 norm compared with the other attacks.

Prism relies on the gradient of substitute models to optimize the perturbation. However, the ASR is notably low when the gradient of substitute models deviates significantly from that

Table 6: Experimental results (ASR% / AvgQ) in four commercial APIs, i.e. AWS, Azure, Baidu, and Tencent.

	ℓ_2								ℓ_∞							
	AWS		Azure		Baidu		Tencent		AWS		Azure		Baidu		Tencent	
	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ
BiasedBA	0.0	1000.0	0.0	1000.0	0.0	1000.0	0.0	1000.0	0.0	1000.0	0.0	1000.0	0.0	1000.0	0.0	1000.0
QEBA-I	30.0	791.3	70.0	500.4	60.0	542.7	55.0	602.1	15.0	880.2	45.0	674.4	35.0	764.3	50.0	610.0
BAODS	20.0	855.3	10.0	934.1	40.0	735.9	60.0	660.3	5.0	983.7	0.0	1000.0	10.0	940.7	10.0	934.9
Prism	30.0	733.4	65.0	426.4	60.0	482.6	80.0	229.4	25.0	788.7	30.0	749.9	50.0	577.6	70.0	425.1
HybridAttack	45.0	564.4	60.0	441.4	65.0	387.4	60.0	415.2	50.0	519.1	70.0	308.5	65.0	362.0	60.0	403.0
DSA	**90.0**	**171.3**	**90.0**	**133.0**	**90.0**	**139.3**	**100.0**	**29.6**	**70.0**	**334.6**	**90.0**	**132.8**	**85.0**	**167.6**	**90.0**	**108.1**

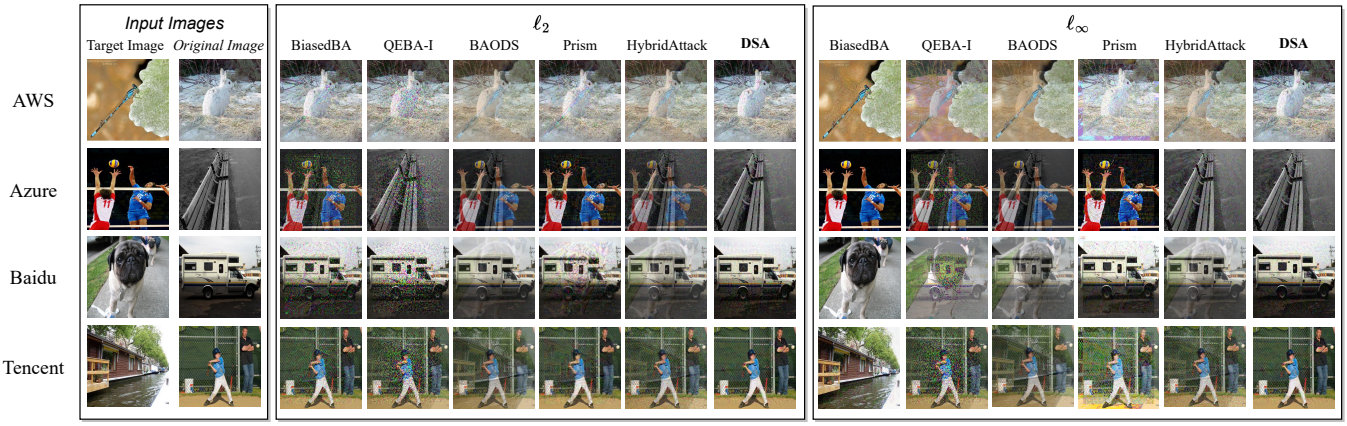


Figure 7: Adversarial examples of each attack. The adversarial examples generated by all existing methods consume the full 1,000 query budget, while those generated by DSA only consume an AvgQ of 546.

of the target model. For instance, Prism achieves an ASR of no more than 0.2% in targeted attacks.

Both DaST and DST show poor performance across all test cases, achieving an ASR of less than 30.8% and 29.4%, respectively. The reason is that the query budget is not sufficient for DaST and DST, which are considered extraction-based attacks, to train the substitute model.

FIA shows high transferability with one query. However, the ASR of FIA decreases when there is a greater disparity between the substitute and target models. For instance, FIA attains an ASR of 75.1% against ResNet20 but only 47.8% against WRN under the ℓ_2 norm for untargeted attacks.

5.3.2 Evaluation on ImageNet

In this subsection, we conduct the open-set evaluation on ImageNet, keeping the same experimental setting in Sec. 5.2.1.

The results are shown in Table 5, DSA can achieve the highest ASR and lowest AvgQ in all tests. Specifically, compared to the second-best value, DSA shows a maximum ASR improvement of 21.7%, from 71.4% to 93.1%, while the AvgQ drops by a maximum of 86.1%, from 646.2 to 90.1.

Compared with the closed-set evaluation on ImageNet (i.e., Table 2), we can observe that the difference in training datasets increases the challenge of obtaining effective adversarial examples, but some attacks still achieve comparable results in the open-set scenario. For example, QEBA-I is able to achieve a higher ASR under the open-set ℓ_2 norm, which increases from 9.9% to 36.8% when evaluated on IncRes-v2. This is because the training dataset of the current target model is only a subset of the closed-set scenario, and the reduction of training data makes the target model easier to attack.

5.4 Experiments on Commercial DNN Services

In this subsection, we carry out attacks on four well-known commercial DNN services, namely AWS [1], Azure [40], Baidu [2] and Tencent [57]. Query results returned by the APIs consist of major and minor categories. We use the major category as the final decision due to the large semantic differences, which raises the difficulty of adversarial attacks. Referring to the setting of the previous literature [14], we randomly select 20 image pairs from the validation set of ImageNet for evaluation and set the query budget as 1,000 for each image pair. Four models on ImageNet are selected as substitute models, as described in Sec. 5.1.

As shown in Table 6, DSA obtains more than 90% ASR using about 100 AvgQ under ℓ_2 norm. Compared with second-best value under both ℓ_2 and ℓ_∞ norms, DSA increases the ASR more than 20.0%, and decreases the AvgQ by at least 35.5%, which indicates that DSA can achieve effective and

Table 7: Results (ASR%/AvgQ) of the ablation study.

	Variations of DSA	ASR	AvgQ
Dispersed Sampling	DSA w/o Image Augmentation	98.9	97.8
	DSA w/o Perturbation Constraint Generation	97.3	150.2
	DSA w/o Substitute Model Selection	98.9	105.7
	DSA w/o Querying the Target Model	78.6	1491.7
	DSA w/ Proportion Fitness	98.6	110.5
Query-based Attack	DSA w/o Dispersed Sampling	78.0	2059.0
Default	**DSA**	**99.1**	**91.0**

query-efficient decision-based attacks against commercial DNN services.

We present real adversarial examples of each attack against the commercial DNNs, as shown in Fig. 7. We can intuitively observe that the adversarial examples generated by DSA are mostly approximate to the original image. In contrast, the adversarial examples generated by BiasedBA, BAODS, Prism, and HybridAttack have noticeable features of the target images, while those generated by QEBA-I also have noises that are easily perceived by human eyes. In addition, DSA requires the least number of queries on the target model.

5.5 Ablation Study

In this section, we conduct a comprehensive ablation study to investigate the key components of DSA, i.e., image augmentation, perturbation constraint generation, substitute model selection, and querying the target model, as well as their corresponding contributions. Furthermore, we measure the scalability of DSA by integrating various white-box attacks. We use IncRes-v2 as the target model and VGG16, DenseNet121, and Inc-v3 serving as substitute models, and construct untargeted adversarial attacks on the ImageNet dataset.

Contribution of Each Module. We generate different variants by changing the settings of each module. Specifically, for each module removal, we perform the following: 1) generating white-box perturbations based on the original image; 2) crafting perturbations under the pre-defined threshold ε; 3) launching a white-box attack against a random substitute model; and 4) generating only a single candidate adversarial example as a starting point for subsequent query-based attacks. Additionally, we devise a variant (DSA w/ Proportional Fitness), which uses the proportion of transferable examples rather than the number of examples, to calculate fitness (defined in Eq. (8)). As an extreme case, we also removed all four modules, simplifying DSA as a query-based attack. We find that each component contributes to the attack performance of DSA, i.e., increasing the ASR and reducing the AvgQ. The detailed results are illustrated in Table. 7.

First, we find that removing any of the three modules of DSA (i.e., image augmentation, perturbation constraint generation, or substitute model selection) limits the sampling range, resulting in a decrease in ASR and an increase in AvgQ. For instance, DSA without Perturbation Constraint Generation can only generate perturbations within the pre-defined upper bound ε, achieving lower ASR and increasing AvgQ by 65.1% compared with DSA.

Table 8: Results (ASR%/AvgQ) of the scalability of DSA

Method	ASR	AvgQ
FGSM [17]	53.4	1.0
DSA (FGSM)	98.7	166.4
I-FGSM [26]	58.3	1.0
DSA (I-FGSM)	98.3	142.7
MI-FGSM [12]	76.6	1.0
DSA (MI-FGSM, Default)	99.1	91.0
FIA [63]	67.3	1.0
DSA (FIA)	99.3	63.5
NAA [66]	74.4	1.0
DSA (NNA)	**99.5**	81.6

Second, dispersed sampling plays a critical role in DSA. DSA variants experience a significant decrease in ASR when the dispersed sampling is insufficient (DSA w/o Querying the Target Model) or completely removed (DSA w/o Dispersed Sampling). The ASR is reduced by at least 20.5%, while AvgQ increases over 16 times. This highlights the critical role of dispersed sampling.

Third, the number of transferable examples is more suitable than the proportion of the samples to compute fitness. DSA with Proportional Fitness achieves a lower ASR while increasing the AvgQ by 21.4%. The reason is that the substitute model selection in this setting devalues successfully transferred samples.

Scalability with White-box Attacks. We evaluate the scalability of DSA by integrating with different white-box attacks. By default, DSA leverages MI-FGSM [12] as the white-box attack. However, the SOTA white-box attacks can be incorporated into DSA, which allows constructing black-box attacks with a higher ASR and a lower AvgQ, as illustrated in Table. 8. For example, as FIA [63] and NAA [66] are the latest transfer-based attacks, DSA achieves an ASR of 99.5% with NAA, and an AvgQ of 63.5 with FIA, which outperforms the default DSA.

5.6 Summary

Recalling the design goals in Sec. 2.3, DSA can achieve a higher ASR than the SOTA attacks while reducing the AvgQ by at least an order of magnitude in closed-set, which demonstrates DSA has high effectiveness and query efficiency. DSA can generate adversarial examples under both ℓ_2 and ℓ_∞ norms, which illustrates the indistinguishability of DSA. DSA can outperform other attacks in defended and open-set scenarios, which demonstrates the robustness of DSA.

6 Related Work

In this section, we review three main categories of black-box adversarial attacks in decision-based scenarios.

Transfer-based Attack. Transfer-based attacks are based on the simple fact that adversarial examples exhibit transferability across model structures, which can be used to launch attacks on the target model [60]. The attacker seeks to generate an adversarial example by applying a white-box attack on substitute models and then uses it to directly attack the target model [17, 42]. The first approach is gradient-based methods where the calculation of gradients is optimized, such as using momentum information (MI-FGSM [12]), Nesterov accelerated gradient (NI-FGSM [30]) and feature maps from intermediate layers (FIA [63], NAA [66]) in the iterative attacks. The second approach, namely input augmentation methods, is to integrate gradients of various augmented images as the final direction of optimization to enhance the attack transferability (e.g., DIM [64] and TIM [13]). Note that these two approaches can be naturally integrated by combining the gradients of the augmented images computed by a gradient-based method.

Query-based Attack. Query-based attacks start with images having different labels (i.e., target images), walk around the decision boundary of the target model, and search for images with smaller perturbations while keeping the images with different labels [4]. The walking process can be roughly divided into three components [15]: 1) generating a potential adversarial example according to a well-designed algorithm, 2) leveraging the potential adversarial example to query the target model, and 3) updating the algorithm based on the query results. Depending on the walking strategies, existing attacks can be divided into two types, namely randomly walking around the boundary (e.g., BA [4], AHA [29] and SurFree [37]) and estimating the gradient by sampling on the decision boundary (e.g., HSJA [8] and TA [35]).

Hybrid Attack. Hybrid attacks attempt to combine the advantages of the two preceding methods. The attacker tries to generate adversarial examples with minimal queries by exploiting the prior substitute models. Existing methods aim to enhance query-based attacks with the prior of substitute models. On the basis of BA, Brunner et al. [5] propose BiasedBA, where adversarial gradients from substitute models are projected orthogonally to the source direction and bias the perturbations toward the projected gradient. Based on HSJA, QEBA-I [27] attempts to sample in a representative subspace, such as the subspace constructed from the principal components of the gradient matrix of substitute models. Tashiro et al. [56] design a sampling strategy to maximize the diversity in the output space of substitute models, and integrate with BA to propose BAODS to reduce the overhead of queries. AMEBA [7] formulates substitute model training and adversarial attacks as an optimization problem to incorporate. Juuti et al. [24] leverage the gradient of ensemble substitute models to search for adversarial examples with progressively smaller perturbations. Differing from the previous studies, which tune the sampling strategy of query-based attacks using the prior substitute models, HybridAttack [53] exploits the candidate example generated by a transfer-based attack as the starting point for query-based attacks to save large queries in the earlier stage of the attack.

Extraction-based Attack. Extraction-based attacks aim to generate an effective substitute model to increase the success rate of transfer-based attacks. They train the substitute model using the query results of the target model to mimic the behaviors. DaST [67] constructs a generative model to produce training examples for the substitute model. Based on DaST, DST [61] utilizes the correlation between training examples to improve the effectiveness of training substitute models.

7 Discussion

In this section, we discuss the limitations of DSA and possible future extensions.

Indistinguishability. The experiments we conducted are based on the common distance metrics, i.e., the ℓ_2 and ℓ_∞ norms. Although DSA achieves better results than other attacks in both metrics, we observe that it still has limited performance in reducing distance under ℓ_∞ norm, which suggests that we need to work on improving the effectiveness of the attack under ℓ_∞ norm in future work.

Potential Countermeasures. DSA can be detected by recognizing adversarial examples with small perturbations, which may be disrupted by image manipulations, leading to significant changes in the output vectors. We provide a more detailed analysis of potential countermeasures in Appendix B.

8 Conclusion

In this paper, we proposed DSA, which can leverage the ability of white-box attacks to achieve query-efficient adversarial attacks in the black-box decision-based scenario. Based on the observation that varying the distribution or magnitude of perturbations enables different transferability, DSA resorted to sampling different perturbations and finding a small perturbation that can transfer successfully. In multiple scenarios and datasets, DSA significantly outperformed the SOTA methods with the same threat model, and achieved the best attack performance on commercial DNN services. In future work, we will investigate techniques to extend the application scenarios of DSA, and explore efficient and lightweight defenses.

Acknowledgments

This work is partially supported by National Key R&D Program of China with No. 2023YFB2703800, NSFC Projects with Nos. 62132011, 62222201, and U23A20304, Beijing

Nova Program with No. 20220484174, Beijing Natural Science Foundation with No. M23020.

Availability

Implementations and data for reproducing our results are available at https://github.com/lcycode/DSA.

References

[1] Amazon. Aws. https://aws.amazon.com/cn/rekognition/, 2023. Accessed 19 March 2023.

[2] Baidu. Baidu ai. https://ai.baidu.com/tech/imagerecognition/general, 2023. Accessed 13 January 2023.

[3] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[4] Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[5] Thomas Brunner, Frederik Diehl, Michael Truong Le, and Alois Knoll. Guessing smart: Biased sampling for efficient black-box adversarial attacks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4958–4966, 2019.

[6] John Burkardt. The truncated normal distribution. *Department of Scientific Computing Website, Florida State University*, 1:35, 2014.

[7] Stefano Calzavara, Lorenzo Cazzaro, and Claudio Lucchese. Ameba: An adaptive approach to the black-box evasion of machine learning models. In *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, pages 292–306, 2021.

[8] Jianbo Chen, Michael I Jordan, and Martin J Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In *2020 ieee symposium on security and privacy (sp)*, pages 1277–1294. IEEE, 2020.

[9] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pages 15–26, 2017.

[10] Shuyu Cheng, Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Improving black-box adversarial attacks with a transfer-based prior. *Advances in neural information processing systems*, 32, 2019.

[11] Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli. Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In *28th USENIX security symposium (USENIX security 19)*, pages 321–338, 2019.

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

[13] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4312–4321, 2019.

[14] Yan Feng, Baoyuan Wu, Yanbo Fan, Li Liu, Zhifeng Li, and Shu-Tao Xia. Boosting black-box attack with partially transferred conditional adversarial distribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15095–15104, 2022.

[15] Qi-An Fu, Yinpeng Dong, Hang Su, Jun Zhu, and Chao Zhang. {AutoDA}: Automated decision-based iterative adversarial attacks. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 3557–3574, 2022.

[16] Justin Gilmer, Ryan P Adams, Ian Goodfellow, David Andersen, and George E Dahl. Motivating the rules of the game for adversarial example research. *arXiv preprint arXiv:1807.06732*, 2018.

[17] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[18] Google. Google version. https://cloud.google.com/vision, 2023. Accessed 13 January 2023.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[20] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[21] HuggingFace. Hugging face. https://huggingface.co/, 2023. Accessed 13 January 2023.

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

[23] Xiaojun Jia, Xingxing Wei, Xiaochun Cao, and Hassan Foroosh. Comdefend: An efficient image compression model to defend adversarial examples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6084–6092, 2019.

[24] Mika Juuti, Buse Gul Atli, and N. Asokan. Making targeted black-box evasion attacks effective and efficient. In Lorenzo Cavallaro, Johannes Kinder, Sadia Afroz, Battista Biggio, Nicholas Carlini, Yuval Elovici, and Asaf Shabtai, editors, *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security, AISec@CCS 2019, London, UK, November 15, 2019*, pages 83–94. ACM, 2019.

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

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

[27] Huichen Li, Xiaojun Xu, Xiaolu Zhang, Shuang Yang, and Bo Li. Qeba: Query-efficient boundary-based blackbox attack. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1221–1230, 2020.

[28] Huiying Li, Shawn Shan, Emily Wenger, Jiayun Zhang, Haitao Zheng, and Ben Y. Zhao. Blacklight: Scalable defense for neural networks against query-based black-box attacks. In Kevin R. B. Butler and Kurt Thomas, editors, *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, pages 2117–2134. USENIX Association, 2022.

[29] Jie Li, Rongrong Ji, Peixian Chen, Baochang Zhang, Xiaopeng Hong, Ruixin Zhang, Shaoxin Li, Jilin Li, Feiyue Huang, and Yongjian Wu. Aha! adaptive history-driven attack for decision-based black-box models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16168–16177, 2021.

[30] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[31] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into transferable adversarial examples and black-box attacks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[32] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12009–12019, 2022.

[33] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[34] Chen Ma, Li Chen, and Jun-Hai Yong. Simulating unknown target models for query-efficient black-box attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11835–11844, 2021.

[35] Chen Ma, Xiangyu Guo, Li Chen, Jun-Hai Yong, and Yisen Wang. Finding optimal tangent points for reducing distortions of hard-label attacks. *Advances in Neural Information Processing Systems*, 34:19288–19300, 2021.

[36] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[37] Thibault Maho, Teddy Furon, and Erwan Le Merrer. Surfree: a fast surrogate-free black-box attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10430–10439, 2021.

[38] Sachin Mehta and Mohammad Rastegari. Separable self-attention for mobile vision transformers. *arXiv preprint arXiv:2206.02680*, 2022.

[39] Dongyu Meng and Hao Chen. Magnet: A two-pronged defense against adversarial examples. In Bhavani Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, pages 135–147. ACM, 2017.

[40] Microsoft. Azure. https://azure.microsoft.com/en-us/products/cognitive-services/vision-services/, 2023. Accessed 19 March 2023.

[41] Aamir Mustafa, Salman Khan, Munawar Hayat, Roland Goecke, Jianbing Shen, and Ling Shao. Adversarial defense by restricting the hidden space of deep neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3385–3394, 2019.

[42] Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016.

[43] PyTorch. Pytorch hub. https://pytorch.org/hub/, 2023. Accessed 13 January 2023.

[44] Zeyu Qin, Yanbo Fan, Hongyuan Zha, and Baoyuan Wu. Random noise defense against query-based black-box attacks. *Advances in Neural Information Processing Systems*, 34:7650–7663, 2021.

[45] Jonas Rauber, Roland Zimmermann, Matthias Bethge, and Wieland Brendel. Foolbox native: Fast adversarial attacks to benchmark the robustness of machine learning models in pytorch, tensorflow, and jax. *Journal of Open Source Software*, 5(53):2607, 2020.

[46] 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(3):211–252, 2015.

[47] Meng Shen, Kexin Ji, Zhenbo Gao, Qi Li, Liehuang Zhu, and Ke Xu. Subverting website fingerprinting defenses with robust traffic representation. In Joseph A. Calandrino and Carmela Troncoso, editors, *32nd USENIX Security Symposium, USENIX Security 2023, Anaheim, CA, USA, August 9-11, 2023*, pages 607–624. USENIX Association, 2023.

[48] Meng Shen, Changyue Li, Hao Yu, Qi Li, Liehuang Zhu, and Ke Xu. Decision-based query efficient adversarial attack via adaptive boundary learning. *IEEE Transactions on Dependable and Secure Computing*, 2023.

[49] Meng Shen, Hao Yu, Liehuang Zhu, Ke Xu, Qi Li, and Jiankun Hu. Effective and robust physical-world attacks on deep learning face recognition systems. *IEEE Trans. Inf. Forensics Secur.*, 16:4063–4077, 2021.

[50] Meng Shen, Jinpeng Zhang, Liehuang Zhu, Ke Xu, and Xiaojiang Du. Accurate decentralized application identification via encrypted traffic analysis using graph neural networks. *IEEE Trans. Inf. Forensics Secur.*, 16:2367–2380, 2021.

[51] Zhiqiang Shen, Mingyang Huang, Jianping Shi, Zechun Liu, Harsh Maheshwari, Yutong Zheng, Xiangyang Xue, Marios Savvides, and Thomas S Huang. Cdtd: A large-scale cross-domain benchmark for instance-level image-to-image translation and domain adaptive object detection. *International Journal of Computer Vision*, 129:761–780, 2021.

[52] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[53] Fnu Suya, Jianfeng Chi, David Evans, and Yuan Tian. Hybrid batch attacks: Finding black-box adversarial examples with limited queries. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 1327–1344, 2020.

[54] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[55] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[56] Yusuke Tashiro, Yang Song, and Stefano Ermon. Diversity can be transferred: Output diversification for white-and black-box attacks. *Advances in Neural Information Processing Systems*, 33:4536–4548, 2020.

[57] Tencent. Tencent cloud. https://cloud.tencent.com/product/imagetagging, 2023. Accessed 19 March 2023.

[58] Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIV*, pages 516–533. Springer, 2022.

[59] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian J. Goodfellow, Dan Boneh, and Patrick D. McDaniel. Ensemble adversarial training: Attacks and defenses. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[60] Florian Tramèr, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017.

[61] Wenxuan Wang, Xuelin Qian, Yanwei Fu, and Xiangyang Xue. DST: dynamic substitute training for data-free black-box attack. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 14341–14350. IEEE, 2022.

[62] Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1924–1933, 2021.

[63] Zhibo Wang, Hengchang Guo, Zhifei Zhang, Wenxin Liu, Zhan Qin, and Kui Ren. Feature importance-aware transferable adversarial attacks. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 7619–7628. IEEE, 2021.

[64] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2730–2739, 2019.

[65] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In Richard C. Wilson, Edwin R. Hancock, and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*. BMVA Press, 2016.

[66] Jianping Zhang, Weibin Wu, Jen-tse Huang, Yizhan Huang, Wenxuan Wang, Yuxin Su, and Michael R Lyu. Improving adversarial transferability via neuron attribution-based attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14993–15002, 2022.

[67] Mingyi Zhou, Jing Wu, Yipeng Liu, Shuaicheng Liu, and Ce Zhu. Dast: Data-free substitute training for adversarial attacks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 231–240. Computer Vision Foundation / IEEE, 2020.

A Parameter Tuning

In this section, we evaluate the attack performance of DSA under different parameters on ImageNet, using VGG16 as the substitute model and IncRes-v2 as the target model.

A.1 Number of Attack Epochs

We evaluate how the ASR changes with the number of epochs of dispersed sampling. As illustrated in Fig. 8, the three curves under different query budgets all show a parabolic-like nature. A limited number of epochs restricts the effective utilization of substitute model transferability. In this case, the attacker

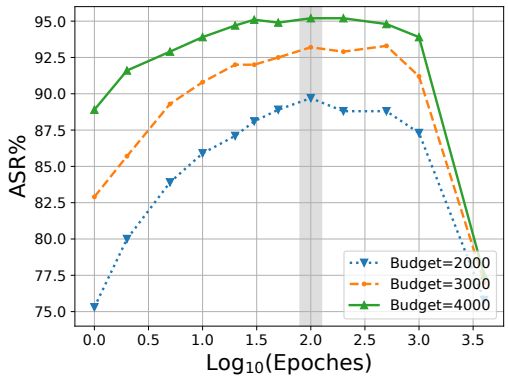


Figure 8: ASR versus number of attack epochs.

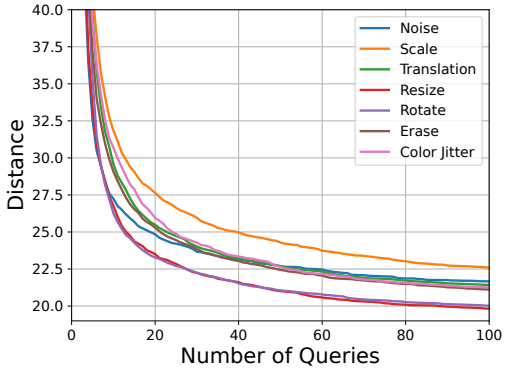


Figure 9: Mean distance versus number of queries under different input augmentations.

cannot fully explore the adversarial region and obtain a transferable example with a small perturbation, leading to a lower ASR. On the contrary, although a larger number of epochs can fully leverage substitute models, it also consumes excessive and unnecessary queries, causing a declining trend in ASR as the number of epochs increases.

A.2 Different Input Augmentations

In this subsection, we evaluate the impact of different input augmentations. Specifically, we test the following input augmentations: adding Gaussian noise, scale transformation, translation, resizing, rotation, random erasing, and color jitter. As illustrated in Fig. 9, we observe that under the same query conditions, the distance gap between different input augmentations is within 5. Rotation and resizing consistently achieve the lowest distances, while scaling exhibits the highest distance. For example, the distance of rotation and resizing is approximately 20, whereas scaling reaches as high as 22.5 with 100 queries. Due to varied sample regions for different augmentations, images that undergo rotation and resizing exhibit greater ℓ_2 norm distances than images subjected to scaling. Thus, candidate adversarial examples from rotation and resizing are more dispersed in the embedding space, which allows DSA to discover smaller transferable perturbations.

B Insights for Potential Countermeasures

It is possible to detect DSA by recognizing adversarial examples with small perturbations. These adversarial examples closely resemble the original images and are located near the decision boundary. Consequently, even minor image manipulations can disrupt the structure of the perturbation, causing significant variations in the output vector. In contrast, normal images exhibit higher confidence and trigger minimal variation during image manipulation, due to their robust features and the spatial invariance of deep neural networks (DNNs).

Table 9: The results (ASR% / AvgQ) of untargeted attacks on ImageNet21k

	ℓ_2								ℓ_∞							
	ConvNext		MobileVitv2		Deit3		Swinv2		ConvNext		MobileVitv2		Deit3		Swinv2	
	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ	ASR	AvgQ
BiasedBA	86.7	2175.2	93.7	1840.1	67.5	2763.4	48.4	3308.7	0.0	4000.0	0.0	4000.0	0.0	4000.0	0.0	4000.0
QEBA-I	6.5	3808.0	14.9	3578.2	19.6	3410.4	3.0	3936.9	1.9	3929.6	2.2	3921.6	2.8	3894.8	0.7	4000.0
BAODS	92.8	1536.2	96.8	1174.1	78.2	2286.9	29.4	3415.7	1.1	3987.7	0.7	3984.9	0.9	3977.0	0.0	4000.0
Prism	32.0	2784.9	62.8	1572.5	14.4	3446.7	3.4	3874.0	18.3	3317.8	35.1	2662.5	13.6	3483.5	0.6	3978.6
HybridAttack	82.2	1265.6	89.3	890.6	85.9	1533.0	43.3	2712.2	56.6	1742.8	70.2	1194.4	33.3	2685.7	22.3	3115.6
DSA	**98.5**	**218.5**	**99.6**	**107.7**	**94.6**	**694.3**	**81.0**	**1307.8**	**85.4**	**617.0**	**90.8**	**393.0**	**57.7**	**1733.0**	**35.6**	**2611.1**

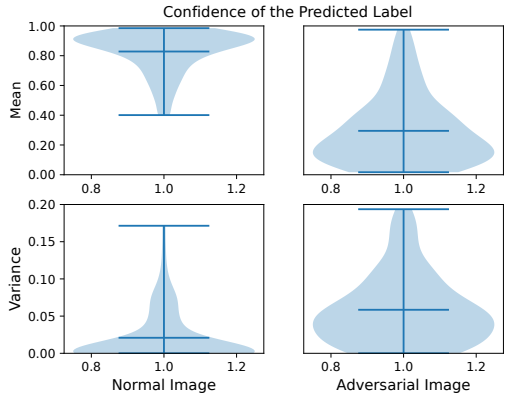


Figure 10: Mean and variance of the confidence for the predicted labels.

We conduct experiments to validate our previous claims. We randomly select 100 original images from ImageNet and collect 100 adversarial examples generated by DSA. For each query, we create several variants of the query image using image manipulations such as rotation, adding noises, and Gaussian blur. We calculate the mean and variance of the confidence for the predicted labels of the query variants, as shown in Fig. 10. We observe that the query variants of normal images have a higher mean confidence score and lower variance than adversarial examples. Therefore, we can utilize the difference in mean and variance of confidence scores to detect the perturbations generated by DSA.

C Experiments on ImageNet-21k.

Due to the applications of large models and advanced model architectures [51], we evaluate the performance of existing attacks on ImageNet-21k [46].

ImageNet-21k is a superset of ImageNet, with about 15 million images. We select ConvNext [33], MobileVitv2 [38], Deit3 [58], Swinv2 [32] as the target model, respectively. These models are able to achieve excellent performance on multiple vision tasks and have been widely used in applications such as mobile devices [38]. All of them are trained on ImageNet-21k with the same classification categories as ImageNet-1k, and can be accessed in the repository[5].

Table 10: Average time consumption on attacking IncRes-v2

Methods	Time Consumption(s)
QEBA-I	117.7
BAODS	106.3
BiasedBA	64.1
Prism	44.3
HybridAttack	26.1
DSA	**13.5**

The results are summarized in Table 9, and it can be seen that DSA is able to achieve the highest ASR and the lowest AvgQ in all tests. Under ℓ_2 norm, compared to the second best values, we see a maximum increase of 37.7% in ASR (i.e., from 43.3% to 81.0%) and a maximum decrease of 7 times in AvgQ (i.e., from 890.6 to 107.7).

In addition, we observe a large difference in the effectiveness of the attack when using the model with different structures from the target model. For example, for the evaluation of the ℓ_∞ norm, DSA can achieve an ASR of 85.4% on ConvNext, while only an ASR of 35.6% on Swinv2.

D Time Consumption

To comprehensively evaluate the attack performance, we record the average time consumption of each attack under the experimental setting in Sec. 5.2 with IncRes-v2 as the target model on ImageNet. The results are shown in Table 10. Compared with the second-best value, DSA takes almost half of the time to complete the attack. We believe that the advantage of DSA lies in two aspects. First, only a white-box attack against one substitute model is needed to generate a candidate, which results in little preparation for each query, requiring neither additional training of auxiliary models [14, 27, 34] nor the calculation of the sampling direction based on query results [27, 56]. Second, benefiting from the high query efficiency of DSA, we can find an adversarial sample with a few queries, allowing the attack to be stopped in a short time.

[5] https://github.com/huggingface/pytorch-image-models