# Decision-based Query Efficient Adversarial Attacks via Adaptive Boundary Learning

Meng Shen, *Member, IEEE*, Changyue Li, Hao Yu, Qi Li, *Senior Member, IEEE*, Liehuang Zhu, *Senior Member, IEEE*, Ke Xu, *Senior Member, IEEE*

***Abstract*—Decision-based adversarial attacks pose a severe threat to real-world applications of Deep Neural Networks (DNNs), as the attackers are assumed to have no prior knowledge about target DNNs except hare labels of model outputs. Existing decision-based attacks require a large number of queries on the target model for a successful attack. In this paper, we propose DEAL, a decision-based query-efficient attack based on adaptive boundary learning. DEAL relies on a local model initialized with meta learning mechanism to gain the ability to fit the new model boundaries. We conduct extensive experiments to evaluate the effectiveness of DEAL, which demonstrates that it outperforms 8 state-of-the-art attacks. Specifically, DEAL achieves similar attack success rates with a maximum query reduction of 51% in untargeted attacks and 14% in targeted attacks.**

***Index Terms*—Adversarial attack, query efficiency, meta learning.**

## I. Introduction

DEEP Neural Networks (DNNs) have been extensively deployed in many applications, e.g., the image recognition on cloud platforms [1], face recognition [2] and network traffic analysis [3]–[5]. However, DNNs are known to be vulnerable to adversarial attacks [6]–[8], which carefully craft examples with small magnitude of perturbations added to mislead DNNs into making incorrect decisions.

We focus the task of DNN-enpowered image classification in decision-based scenario where the adversary can only query the target model and obtain the predicted label (i.e., the decision). This is a more realistic assumption due to the concern of model privacy and self-interest. In practice, model owners often deploy the well-trained classification models in the cloud platform and only release query APIs for public to access services, and the adversary is restricted from interacting with the target DNN via queries and the corresponding classification decisions [9]. Usually, the model owners will charge for each query which means that a large amount of queries for adversarial attack is not tolerated. Meanwhile, the attacker's frequent queries on the target model also increase its risk of being detected. Thus, it is a crucial issue to launch adversarial attack with high query efficiency in such scenarios.

Many recent studies focus on decision-based attacks (Table I) [10], [11]. The basic idea of decision-based attacks is generating perturbed examples (e.g., images) with large magnitude of adversarial perturbations, and then minimizing the distance between the adversarial and original examples while ensuring the effectiveness of the adversarial examples in deceiving target DNNs. Existing decision-based attacks are divided into substitute-free and substitute-based methods depending on whether the attacker has substitute models locally. Substitute-free attacks minimize the distance between original and perturbed images by randomly sample directions (e.g., BA [12], TA [13], AHA [14]) or estimate gradients at the decision boundary of DNNs (e.g. HSJA [15], QEBA [16], qFool [17] and GeoDA [18]). However, above attacks all require high query overhead to search for adversarial examples satisfying the distance constraint. The substitute-based methods attempt to leverage local models to guide the process of searching for adversarial examples. BiasedBA [19] and BAODS [20] are both extensions of BA, they incorporate the adversarial perturbations generated by the substitute model into the sampling phase of BA, making it toward a more practicable direction. Hybrid Attack (HA) [21] locally generates adversarial examples as start images of subsequent black-box attack. However, the improvement of these attacks is very limited, and they may suffer failure when there is a large gap between the substitute models and the target model.

In this paper, we propose DEAL, a decision-based query efficient attack. Typical decision-based attacks inevitably consume a large target model queries in sample walking or gradient estimation, our basic idea is to sink the black-box search process on the target model to local for query reduction. Intuitively, if the decision boundary of local substitute model perfectly matches that of target model better, more valid adversarial examples are prone to be generated on the local model. To eliminate the impact of model gaps, we introduce a boundary learner, which is a binary classifier for predicting the decision boundary corresponding to a specified label. Compared to building local substitute models, the learning task of the boundary classifier is more specialized and can effectively and adaptively adjust the learned boundaries through small interactions with the target model during attack.

There are two processes in the DEAL, the initialization and optimization phases. During initialization, we build multiple substitute DNNs that have the similar functionality as the target model. Then, we leverage the meta learning mechanism [22] to obtain the initial parameters of boundary learner by learning decision boundaries of these substitute DNNs, which provides the boundary learner the ability to quickly adapt its decision boundary to a new DNNs (e.g., the target

M. Shen, C. Li, and L. Zhu are with the School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081, China (e-mail: {shenmeng, licy, liehuangz}@bit.edu.cn).

H. Yu is with Ant Group, Beijing, China (e-mail: csyuhao@gmail.com).

Q. Li is with the Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing, China (e-mail: qli01@tsinghua.edu.cn).

K. Xu is with the Department of Computer Science, Tsinghua University, Beijing, China (e-mail: xuke@tsinghua.edu.cn).

TABLE I: Summary of Decision-based Black-box Attacks.

| Methods | Existing Approaches | Untargeted | Targeted | Query Magnitude[1] | Success Rate[2] |
|---|---|---|---|---|---|
| Substitute-free | BA [12] | ✓ | ✓ | 2800 | $\sim 60\%$ |
| | qFool [17] | ✓ | ✓ | 1700 | $\sim 70\%$ |
| | TA [13], GeoDA [18] | ✓ | ✗ | 2000 | $\sim 90\%$ |
| | AHA [14] | ✓ | ✓ | 1000 | $\sim 100\%$ |
| | HSJA [15], QEBA [16] | ✓ | ✓ | 800 | $\sim 100\%$ |
| Substitute-based | BiasedBA [19] | ✓ | ✓ | 1000 | $\sim 100\%$ |
| | BAODS [21] | ✓ | ✓ | 800 | $\sim 100\%$ |
| | HA [21] | ✓ | ✓ | 500 | $\sim 100\%$ |
| | DEAL (**ours**) | ✓ | ✓ | 300 | $\sim 100\%$ |

[1] It means the average number of queries against target DNNs required to generate per perturbed image.
[2] We list attack success rates of targeted attacks by default with a budget of 5000.

model) with limited queries. Note that initialization process is conducted with no need to query the target model. Optimization process is developed to make the boundary learner adapt to the target model and generate adversarial examples, consisting of three steps. First, DEAL employs an existing decision-based attack (e.g., HSJA [15]) to explore the decision boundary of target DNNs, and re-train the boundary learner with the query-result pairs. Second, DEAL generates the adversarial example based on boundary learner. Third, we leverage recent query-result pairs to fine-tune the boundary learner to further narrow down the gap between boundary learner and target DNNs.

We conduct extensive experiments to evaluate the performance of DEAL. Three well-known image datasets, i.e., CIFAR-10, CIFAR-100 and Tiny-ImageNet, are utilized to investigate the attack success rate and the average number of queries on target DNNs. We compare DEAL with 8 state-of-the-art methods enabling both untargeted and targeted attacks, including BA [12], HSJA [15], QEBA [16], qFool [17], AHA [14], BiasedBA [19], BAODS [20] and HA [21]. Experimental results with the CIFAR-10 dataset show that DEAL can achieve an attack success rate of 100% in all attacks with a maximum query reduction of 51% in untargeted attacks and 14% in targeted attacks, compared with the most query-efficient existing method. We also employ 5 typical defenses to demonstrate the robustness of DEAL against defended DNNs.

We summarize the main contributions as follows:

- We propose DEAL, a decision-based query-efficient attack based on adaptive boundary learning, which reduces the number of queries while maintaining a high attack success rate.
- We conduct an ablation study to separately demonstrate the contribution of main modules of the proposed adaptive boundary learning, in terms of success rate improvement and query reduction.
- We conduct extensive experiments to demonstrate that DEAL outperforms the state-of-the-art decision-based attacks in both untargeted and targeted scenarios.
- We leverage 5 typical defenses and 3 public datasets to demonstrate the effectiveness and query efficiency of DEAL against the defended target models.

## II. Related Work

Although DNNs have achieved the striking performance on many applications, they easily misclassifies inputs with imperceptible perturbations [6]. In this section, we will introduce the taxonomy of *black-box adversarial attacks*, where the adversary has no knowledge of target DNNs and only obtains outputs, and summarize existing approaches. Depending on outputs of target DNNs, there are *score-based attacks* and *decision-based attacks*.

**Score-based attacks**. These attacks assume that outputs of target DNNs are probabilities. Some approaches utilize the gradient estimation (e.g., ZOO [23] and Bandits [24]), genetic algorithms [25] and Bayesian optimization [26] to optimize adversarial perturbations. Some score-based attacks (e.g., Liu et al. [27] and Wang et al. [28]) exploit the prior of substitute models [6] to employ the probability produced by target DNNs to train substitute DNNs. Meta Attack [29] and Simulator Attack [30] use the meta learning mechanism to train substitute DNNs, which enables new tasks to be learned using a small number of training dataset. However, for many DNNs-based applications, service providers will not expose probabilities to users, which hinders the practicality of score-based attacks.

**Decision-based attacks**. These attacks consider that the attacker only has predicted labels (i.e., the top-1 class). As shown in Table I, existing decision-based attacks can be divided into substitute-free and substitute-based attacks depending on whether the attacker can construct substitute models.

The substitute-free methods randomly sample optimization directions (e.g., vectors [12], [14] and angles [13], [31]) to minimize the distance between perturbed and original samples. Brendel et al. propose BA [12] to generate optimization directions. TA [13] optimizes perturbation perturbations using the geometric information, and AHA [14] utilizes the state variable to maintain historical directions and a coefficient to balance the state variable and the random direction at each iteration. Since randomly sampled optimization directions, perturbed images always cannot deceive target DNNs with deceasing distance to original images, resulting in low attack success rates. Some other methods find examples near the decision boundary of target DNNs. Chen et al. introduce HSJA [15] to estimate the gradient direction using classification results at the decision boundary between adversarial

TABLE II: List of Notations.

| Signs | Description | Signs | Description |
|---|---|---|---|
| $\mathcal{F}$ | Target DNN | $\mathcal{S}$ | A set of substitute DNNs |
| $\mathcal{C}$ | Target classifier | $\mathcal{B}$ | Boundary learner |
| $\theta$ | Parameters of target DNN $\mathcal{F}$ | $\phi$ | Parameters of boundary learner $\mathcal{B}$ |
| $\mathbf{x}$ | Benign image | $y$ | Label of the image $\mathbf{x}$ |
| $\mathbf{x}'$ | Adversarial image | $y^\dagger$ | Pre-specific label in targeted attacks |
| $T$ | Query budget | $M$ | Attack iterations |
| $K$ | The number of labels | $N$ | The number of substitute DNNs |

and non-adversarial regions. Based on HSJA, QEBA [14] randomly samples directions from representative subspaces. Motivated by the observation that the curvature of decision boundary is small, qFool [17] and GeoDA [18] are designed to estimate the normal vector of target DNNs with the flat decision boundary. However, such approaches spend many queries on target DNNs to obtain accurate gradient directions.

Substitute-based attacks utilize the prior substitute models to imporve the query efficiency of adversarial attacks, and they are rarely exploited in decision-based attacks. Integrated with BA, BiasedBA [19] exploit gradients from surrogates to biased the distribution of search space, and BAODS [20] design a sampling strategy to maximize the diversity in the output space of substitute models. Considering local models can help direct gradient search, HA [21] exploits the transferability of substitute models to generate new target image for exisitng attacks. However, the performance of the above methods is limited by the approximation of substitute models to the target model. Some attacks [32], [33] execute model extraction attacks to build substitute models with the same functionality as the target model and generate white-box adversarial examples to deceive the target model. However, it required massive queries to train a surrogate model, which is not feasible under limited query budgets. Moreover, The success rate of the attack via transfer is also restricted.

## III. Threat Model and Design Goals

### A. Threat Model

**Adversary Knowledge**. We consider the task of DNN-enpowered image classification, which has a wide range of applications in many fields [1], [34]. In the black-box setting, the adversary is restricted from interacting with the target DNN via queries and the corresponding classification outputs [9]. We also assume a decision-based target model, where the target model returns only the label for each query, rather than predicted probabilities over labels.

Given a target DNN $\mathcal{F}$ with parameters $\theta$ and an original image $\mathbf{x} \in [0,1]^D$ with the ground-truth label $y$ in the label set $[K] = \{0, \cdots, K-1\}$, the classifier $\mathcal{C} : [0,1]^D \to [K]$ recognizes the input $\mathbf{x}$ as the label with the maximal probability (i.e., $y$, if correctly classified), as shown in Eq. (1),

$$\mathcal{C}(\mathbf{x}) = \arg\max_{c \in [K]} \mathcal{F}_c(\mathbf{x}; \theta). \tag{1}$$

Note that the adversary can only get the predicted label with the maximal probability. The main notations used in this paper are summarized in Table II.

We also assume that the adversary knows the dimension $D$ and the label set $[K]$, which is practical in real-world applications. For instance, cloud APIs (e.g., Google Vision API [35]) usually provide API descriptions, including the dimension of input images and the set of labels.

Many efforts have been devoted to developing effective defenses against adversarial attacks, and the defenses adopted by target DNNs are usually kept secret from public. We assume that the adversary has no prior knowledge about the defenses, including whether a specific defense is deployed, and if yes, the details of the deployed defense.

**Adversary Capability**. With the knowledge of the dimension $D$ and label set $[K]$, the adversary has the ability to construct a *shadow* dataset using public datasets (e.g., CIFAR-10 [36]) or Google Images [37]. Using the shadow dataset, the adversary is capable of training multiple local models, which are known as the substitute models of target DNNs.

While general, the substitute DNNs can have a variety of architectures, as recent studies have proposed several different architectures for image classification [29], [30]. Note that, when training substitute DNNs, it is unnecessary for the adversary to query target DNNs.

**Untargeted and Targeted Attacks**. The adversary crafts an adversarial example $\mathbf{x}'$ by adding perturbations on the original image $\mathbf{x}$. In *untargeted* attacks, the adversary attempts to mislead target DNNs by recognizing $\mathbf{x}'$ as any other label $c \in [K] \setminus \{y\}$.. In *targeted* attacks, the adversary aims to make target DNNs classify $\mathbf{x}'$ as a pre-specified label $y^\dagger \in [K] \setminus \{y\}$. Formally, we can define a function $\mathcal{I}_\mathbf{x} : [0,1]^D \to \mathbb{R}$ to indicate whether the adversarial example $\mathbf{x}'$ is effective or not,

$$\mathcal{I}_\mathbf{x}(\mathbf{x}') = \begin{cases} \max\limits_{c \neq y} \mathcal{F}_c(\mathbf{x}'; \theta) - \mathcal{F}_y(\mathbf{x}'; \theta) & \text{Untargeted attacks} \\ \mathcal{F}_{y^\dagger}(\mathbf{x}'; \theta) - \max\limits_{c \neq y^\dagger} \mathcal{F}_c(\mathbf{x}'; \theta) & \text{Targeted attacks} \end{cases}. \tag{2}$$

where $\mathcal{I}_\mathbf{x}(\mathbf{x}') > 0$ indicates an effective adversarial example $\mathbf{x}'$ (i.e., a successful attack). In particular, if $\mathcal{I}_\mathbf{x}(\mathbf{x}') = 0$, the perturbed image $\mathbf{x}'$ is on the decision boundary of the ground-truth label $y$ (in untargeted attacks) or the target label $y^\dagger$ (in targeted attacks).

### B. Design Goals

The proposed decision-based adversarial attack aims to craft effective and imperceptible adversarial examples in a query-efficient manner. Particularly, the attack should achieve three goals described as follows.

**Effectiveness**. The primary goal of an adversarial attack is to mislead target DNNs to generate incorrect outputs. For both untargeted and targeted attacks, the effectiveness in terms of attack success rate is highly desirable, which reflects the ability of this attack to deceive the target models. Especially, when defenses have been deployed, the attack should also be able to undermine these defenses.

**Imperceptibility**. The imperceptibility means that the perturbed images should look similar to the the original images via human perception. Thus, the adversary tries to limit the

This article has been accepted for publication in IEEE Transactions on Dependable and Secure Computing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2023.3289298



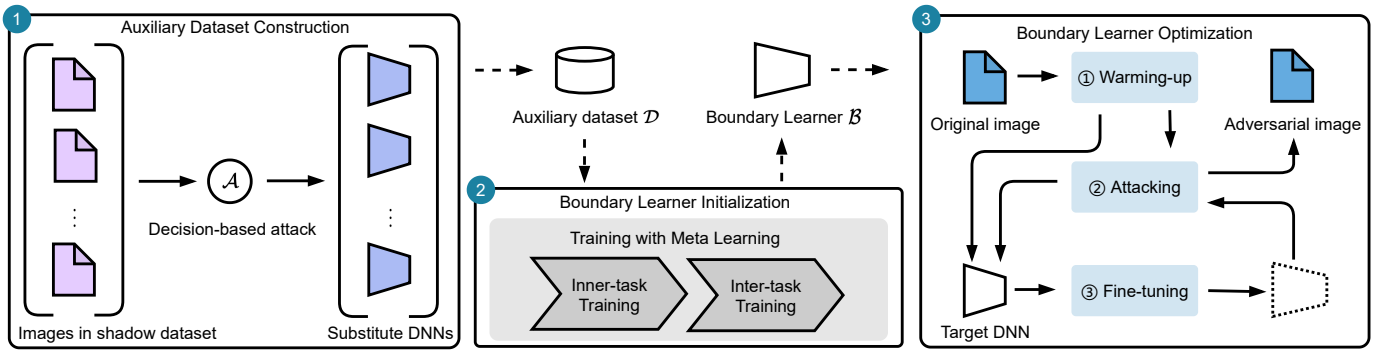



Fig. 1: The overview of DEAL.

magnitude of adversarial perturbations to a certain threshold. A commonly-used measure of magnitude is defined in Eq. (3),

$$\|\mathbf{x}' - \mathbf{x}\|_p \leq \epsilon. \tag{3}$$

where $\epsilon$ is the pre-defined threshold and $\|\cdot\|_p$ is the $L_p$ norms [38]. In this paper, we set $p = 2$ because the $L_2$ norm, also known as the Euclidean distance, is widely used in the literature [12], [15], [18], [29], [30]. In generating an adversarial example, the attackers try to minimize the Euclidean distance between the original input $\mathbf{x}$ and the perturbed input $\mathbf{x}'$.

**Query efficiency**. The number of queries necessary for a successful attack is an important metric in decision-based attacks [29], [30]. The attackers consume resources (e.g., economic costs [39]) to query target DNNs for crafting adversarial examples, thus query overhead can prevent them from launching large-scale attacks. In addition, cloud APIs often limit the number of queries within a certain time period, e.g., Google Vision API allows $1,800$ queries per minute [15].

## IV. Overview of DEAL

In this section, we present the overview of DEAL, a query-efficient decision-based attack that can craft effective adversarial examples to target DNNs.

In this paper, DEAL leverages a query reduction technique named *adaptive boundary learning* for decision-based adversarial attacks. The basic idea is to construct a boundary learner that can gradually learn and approximate the decision boundary of the target model. When generating adversarial examples, the majority of the queries for gradient estimation can be conducted to the boundary learner.

To enable the boundary learner rapidly approximate the decision boundary of target DNNs, we propose a framework that consists of three modules, as illustrated in Figure 1.

**Auxiliary Dataset Construction**. This module constructs an auxiliary dataset that can be used for building the boundary learner, and the auxiliary dataset is made of query sequences from attacking substitute models. As described in Section III-A, the adversary is capable of training multiple substitute DNNs with different architectures that have the same functionality as target DNNs. This module leverages an existing decision-based attack to explore the decision boundary of these substitute DNNs, i.e., taking every original image in the shadow dataset as input, and generating adversarial examples to deceive each of the substitute DNNs. Then, the adversary can obtain a series of query-result pairs from each substitute DNN for each original image, which is referred to as the *auxiliary* dataset.

**Boundary Learner Initialization**. This module initializes the boundary learner to approximate the decision boundary of target DNNs. The boundary learner is trained using the auxiliary dataset via the *meta learning* mechanism (e.g., MAML [22]). Although the decision boundaries of the substitute DNNs would be different from target DNNs, the initialization process provides the boundary learner the ability to rapidly adapt to the decision boundary of the target DNNs with limited queries. Note that this module can be executed offline.

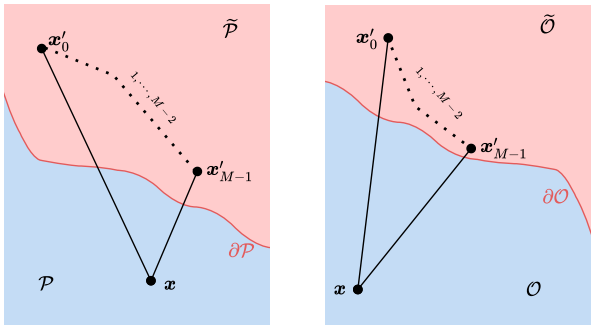
**Boundary Learner Optimization**. This module leverages queries on the target DNNs to further optimize the pre-trained boundary learner via gradually adapting its decision boundary to that of target DNNs. Specifically, it utilizes a three-step optimization strategy, including warming-up, attacking, and fine-tuning. In the warming-up step, this module conducts a small number of queries on the target DNN and utilizes the query-result pairs to improve decision boundary of the boundary learner. Then, in the attacking step, it crafts perturbed images to deceive the target model. If the adversarial examples are ineffective, the query-result pairs are further used to fine-tune the boundary learner. For each specific original image, the attacking and fine-tuning steps can be conducted multiple rounds until an effective adversarial example is obtained or the query budget is exhausted.

## V. Design Details of DEAL

In this section, we present the design details of DEAL, i.e., the design of three main modules in DEAL.

### A. Auxiliary Dataset Construction

In this module, we construct an auxiliary dataset for initializing a boundary learner, which approximate the decision boundary of the target DNN. The quality of this dataset determines the consistency of boundaries between the boundary learner and the target DNN. Motivated by extraordinary performance of meta learning, we use this methodology to pre-train the boundary learner.




(a) Untargeted attacks (b) Targeted attacks

Fig. 2: Boundary between adversarial and non-adversarial regions for untargeted and targeted attacks, where perturbed images located in adversarial regions lead to successful attacks. $\widetilde{\mathcal{P}}$ and $\widetilde{\mathcal{O}}$ are adversarial regions, and $\mathcal{P}$ and $\mathcal{O}$ are non-adversarial regions.

Recall that we assume that the adversary is able to collect a shadow dataset and train $N$ substitute DNNs with different architectures. Specifically, since the attacker has no knowledge of the target model, it cannot obtain a model with the same structure as the target model, so we train 14 substitute models with mainstream image classification networks (e.g., ResNet [40] DenseNet [41]). We setup the target model structure with no overlap with the local for the subsequent experiments, as shown in Section VI-A. We can easily obtain pre-training weights for substitute models from the open-source community, thus avoiding the overhead caused by the training process.

Then, we carry out the decision-based adversarial attack $\mathcal{A}$ to fool the $i$-th substitute model $\mathcal{S}_i$ and get a query sequence which consists of query images and corresponding labels during the attack. Specifically, for an original image $\mathbf{x}$, we obtain a query sequence is query-result pairs

$$\dot{\mathcal{D}}_i = (\mathcal{X}_i, \mathcal{Y}_i) = \left\{(\mathbf{x}'_{i,0}, 0), (\mathbf{x}'_{i,1}, 1), \cdots, (\mathbf{x}'_{i,T-1}, 0)\right\},$$

where $T$ is the number of queries. Given a pre-specified label, for each query with the perturbed image $\mathbf{x}'$, the label of 0 indicates a successful untargeted attack or a failed targeted attack and vice-versa, which means that the image is *outside* the decision boundary of the label.

The reason to select the decision-based adversarial attack lies in that it samples perturbed images that generally distribute on both sides of decision boundary to estimate the direction of adversarial regions. Thus, the sequence of query-result pairs (i.e, $\dot{\mathcal{D}}_i$) contains rich information of decision boundary that corresponds to label $y$ in untargeted attacks or label $y^\dagger$ in targeted attacks on the substitute DNNs $\mathcal{S}_i$.

For each of the $Q$ original images in the shadow dataset, the adversary crafts adversarial images to deceive each of the $N$ substitute DNNs and obtains the auxiliary dataset.

$$\mathcal{D} = \begin{Bmatrix} \left\{\dot{\mathcal{D}}_0, \dot{\mathcal{D}}_1, \cdots, \dot{\mathcal{D}}_{N-1}\right\}_0, \\ \left\{\dot{\mathcal{D}}_0, \dot{\mathcal{D}}_1, \cdots, \dot{\mathcal{D}}_{N-1}\right\}_1, \\ \vdots \\ \left\{\dot{\mathcal{D}}_0, \dot{\mathcal{D}}_1, \cdots, \dot{\mathcal{D}}_{N-1}\right\}_Q \end{Bmatrix}$$

where $Q$ is the number of original images. As illustrated in Figure 2, the adversarial region of untargeted attacks is determined by the decision boundary of label $y$, whereas the adversarial region of targeted attacks is defined by the decision boundary of the pre-specified label $y^\dagger$. Hence, conducting either untargeted attacks or targeted attacks can reflect the decision boundaries of each substitute DNNs.

### *B. Boundary Learner Initialization*

Now we describe the initialization of boundary learner, including its architecture and training process with the auxiliary dataset constructed in the previous subsection.

**Boundary Learner Architecture**. In decision-based attacks, the adversary can only obtain predicted labels instead of full probability distribution. Even the boundary learner and the target DNNs has the same prediction results, the probability distributions of these two models can be different, indicating the differences between decision boundaries of these two models. Thus, it is a challenging task to make the decision boundary of boundary learner match that of target DNNs.

The boundary between adversarial and non-adversarial regions can be formulated as:

$$\mathcal{H}(\mathbf{x}) = \{\mathbf{x}' | \mathcal{I}_{\mathbf{x}}(\mathbf{x}') = 0\}. \tag{4}$$

We also define an indicator function $\mathbb{I}(\mathbf{x}') : [0,1]^D \to \{0,1\}$, which indicates whether a perturbed image $\mathbf{x}'$ is a successful adversarial example,

$$\mathbb{I}(\mathbf{x}') = \text{sign}(\mathcal{I}_{\mathbf{x}}(\mathbf{x}')) = \begin{cases} 1 & \mathcal{I}_{\mathbf{x}}(\mathbf{x}') > 0 \\ 0 & \mathcal{I}_{\mathbf{x}}(\mathbf{x}') \leq 0 \end{cases}. \tag{5}$$

As shown in Figure 2, when generating adversarial examples, we should keep perturbed images stay in adversarial region. This observation motivates us to design an architecture to predict the decision boundary of target DNNs. In this paper, we simplify the task of boundary learner to identify the boundary between adversarial and non-adversarial regions, which is turned into a binary classification problem.

Hence, the boundary learner is designed as a binary classifier, which takes a perturbed image and a specific label as input and determines whether the perturbed image is inside the decision boundary of this label.

The architecture of the boundary learner consists of an image embedding network, a label embedding network, and a classification network, as shown in Figure 3. The backbone of the image embedding network is the deep convolutional neural networks (e.g, ResNet [40]), with the image $\mathbf{x}$ as input and a $d$-dimensional feature vector $\mathbf{h}_1$ as output. The label embedding network and the classification network are made up linear layers. The label embedding network takes one-hot vector $y$ as

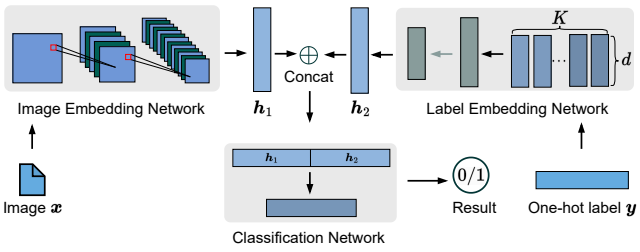



Fig. 3: Architecture of the boundary learner consists of an image embedding network, a label embedding network and a classification network.

input and the output is a vector $\mathbf{h}_2$ with the same dimension as $\mathbf{h}_1$. Finally, the classification network concatenates the two vectors as a new feature $\mathbf{h} = [\mathbf{h}_1, \mathbf{h}_2]$ and predict whether the image $\mathbf{x}$ is inside the decision boundary of the label $y$. For a trained boundary learner, since the dimensions of $\mathbf{h}_1$ and $\mathbf{h}_2$ are fixed, the boundary learner can take inputs of the same size as the training data and it can also attack images with different size by scaling them and avoid retraining the boundary learner.

The basic idea of decision-based attacks is to minimize the distance between the perturbed and the original example and keep the perturbed example in adversarial regions of target DNNs. The adversarial region is outside the decision boundary of label $y$ in untargeted attacks, while inside lable $y^\dagger$ in targeted attacks, as illustrated in Figure 2.

With boundary learner $\mathcal{B}(\mathbf{x}', y) : [0,1]^D \times \{0,1\}^{K\times 1} \rightarrow \{0,1\}$, the indicator function $\mathbb{I}(\mathbf{x}')$ can be re-formulated as

$$\mathbb{I}(\mathbf{x}') = \begin{cases} 1 - \mathcal{B}(\mathbf{x}', y) & \text{Untargeted attacks} \\ \mathcal{B}(\mathbf{x}', y) & \text{Targeted attacks} \end{cases}, \tag{6}$$

We can utilize the boundary learner to obtain $\mathcal{B}(\mathbf{x}', y)$, and mimic boundaries of each label of target DNNs. Note that the adversarial region corresponding to its label will not change when crafting adversarial examples for a specific image.

**Boundary Learner Training**. The training process of boundary learner is to learn an initialization of its parameter set $\phi$ with the auxiliary dataset so as to approximate the decision boundary of target DNNs.

Given an original image $\mathbf{x}$, we define a decision-based attack $\mathcal{A}$ against the $i$-th substitute DNN as a *task* $\mathcal{T}_i$. As described in Section V-A, each task $\mathcal{T}_i$ can generate a sequence of query-result pairs denoted by $\dot{\mathcal{D}}_i$. We leverage MAML [22] to learn the parameter set $\phi$. To facilitate the evaluation of the learned parameters, we split $\dot{\mathcal{D}}_i$ into two parts, namely

$$\dot{\mathcal{D}}_i^{\urcorner} = (\mathcal{X}_i^{\urcorner}, \mathcal{Y}_i^{\urcorner}) = \left\{(\mathbf{x}'_{i,0}, 0), \cdots, (\mathbf{x}'_{i,T/2-1}, 1)\right\},$$

and

$$\dot{\mathcal{D}}_i^{\lrcorner} = (\mathcal{X}_i^{\lrcorner}, \mathcal{Y}_i^{\lrcorner}) = \left\{(\mathbf{x}'_{i,T/2}, 0), \cdots, (\mathbf{x}'_{i,T-1}, 0)\right\}.$$

We employ $\dot{\mathcal{D}}_i^{\urcorner}$ to adapt the initialization in task $\mathcal{T}_i$ using one or multiple rounds of gradient descent, which is referred to as *inner-task training*. A single round of gradient descent is shown in Eq. (7),

$$\phi_i = \phi_i - \lambda_1 \nabla_\phi \mathcal{L}_{\mathcal{T}_i}(\mathcal{B}(\mathcal{X}_i^{\urcorner}, y; \phi), \mathcal{Y}_i^{\urcorner}), \tag{7}$$

where $\lambda_1$ is the step size and $\mathcal{L}(\cdot,\cdot)$ is the cross-entropy loss function for binary classification.

The goal of meta learning is to optimize the model parameters such that a small number of query-result pairs on a new task will produce maximally effective performance. Hence, we apply the *inter-task training* to utilize $\dot{\mathcal{D}}_i^{\lrcorner}$ from multiple tasks to evaluate the performance of adapted boundary learner in a single task $\mathcal{T}_i$ as well as optimizing the parameter set $\phi$ across multiple tasks, which is formally defined in Eq. (8),

$$\phi \leftarrow \phi - \lambda_2 \nabla_\phi \frac{1}{B} \sum_{\mathcal{T}_i \sim P(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(\mathcal{B}(\mathcal{X}_i^{\lrcorner}, y; \phi_i), \mathcal{Y}_i^{\lrcorner}), \tag{8}$$

where $\lambda_2$ is the learning rate, $B$ is the number of tasks (i.e., fooling a substitute DNN with $B$ original images), and $P(\mathcal{T})$ is the distribution of query-result pairs used to attack substitute DNNs. In order to evaluate the performance of the parameter set $\phi$ in adapting to task $\mathcal{T}_i$, the cross-entropy loss function $\mathcal{L}(\cdot,\cdot)$ in Eq. (8) is calculated on the learner with $\phi_i$.

### C. Boundary Learner Optimization

In this subsection, we utilize the queries on target DNNs to gradually optimize the boundary learner so as to improve the consistency of decision boundaries between the boundary learner and the target DNNs. In order to reduce the queries on target DNNs, we propose a three-step attack strategy, including *warming-up*, *attacking* and *fine-tuning*.

**Warming-up**. If the initial boundary learner and target DNNs have the same decision boundary, the adversary can directly use the boundary learner as a substitute model to generate adversarial examples that are also effective on the target DNN. However, there usually exists a gap between the decision boundaries of these two models. Thus, we should optimize the parameter set $\phi$ to approach the decision boundary of target DNNs. Since the boundary learner is initialized in the meta learning manner, we expect a few query-result pairs could help the decision boundary of the boundary learner rapidly adapt to that of target DNNs, which is referred to as the warming-up process. During the adversarial attack for a specific image, this process will be launched only once.

Given an original image $\mathbf{x}$, we employ the decision-based attack $\mathcal{A}$ to deceive the target DNN $\mathcal{F}$, and use the corresponding query-result pairs:

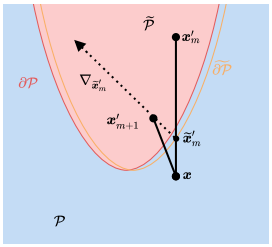
$$\widehat{\mathcal{D}} = (\widehat{\mathcal{X}}, \widehat{\mathcal{Y}}) = \left\{(\mathbf{x}'_0, 0), (\mathbf{x}'_1, 1), (\mathbf{x}'_2, 1), \cdots, (\mathbf{x}'_{R-1}, 0)\right\} \tag{9}$$

to re-train the boundary learner $\mathcal{B}$, as shown in Eq. (10):

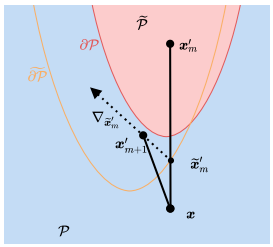
$$\phi' = \begin{cases} \phi - \lambda_3 \nabla_\phi \mathcal{L}(\mathcal{B}(\widehat{\mathcal{X}}, y; \phi), \widehat{\mathcal{Y}}) & \text{Untargeted attacks} \\ \phi - \lambda_3 \nabla_\phi \mathcal{L}(\mathcal{B}(\widehat{\mathcal{X}}, y^\dagger; \phi), \widehat{\mathcal{Y}}) & \text{Targeted attacks} \end{cases}, \tag{10}$$

where $R$ is the number of queries on target DNNs and $\lambda_3$ is the step size. Here, $\widehat{\mathcal{Y}}$ indicates whether examples $\widehat{\mathcal{X}}$ are inside the decision boundary of a specific label of the target DNN.

The difference between warming-up (i.e., Eq. (10)) and the initialization process (i.e., Eq. (7) and Eq. (8)) lies in that the former is optimized using query-result pairs on target DNNs whereas the latter uses query-result pairs on substitute DNNs.

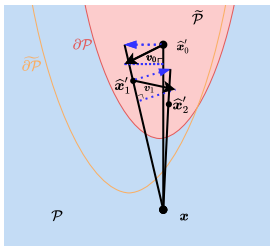


(a) High consistency

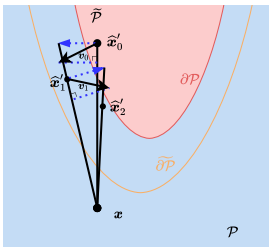


(b) Low consistency

Fig. 4: Attacking results in two different scenarios, where $\partial\mathcal{P}$ is the boundary of the target DNN $\mathcal{F}$ and $\widetilde{\partial\mathcal{P}}$ is the boundary of the boundary learner $\mathcal{B}$.



(a) Successful random walks



(b) Failed random walks

Fig. 5: Successive random walks (e.g., 2 walks) from $\mathbf{x}'_t$ to alleviate the influence of inconsistency of the two decision boundaries $\partial\mathcal{P}$ and $\widetilde{\partial\mathcal{P}}$.

**Attacking**. The consistency of decision boundaries between the boundary learner and the target DNN has a significant influence on attacking results. To reduce the queries necessary for a successful attack, we design an attacking pipeline based on rejection sampling, which consists of *three stages*.

In general, we classify all cases into two typical scenarios, as illustrated in Figure 4. In the high-consistency scenario in Figure 4a, the difference of decision boundaries is relatively small, thus we leverage the *first-stage attack* to fabricate adversarial examples. In another scenario in Figure 4b, there is a larger difference between the two boundaries so that the adversarial examples crafted by the first-stage attack cannot deceive target DNNs. We further employ the *second-stage attack* to generate candidate perturbed images as illustrated in Figure 5a. If fails too, it indicates that there is a much larger gap of decision boundaries between the boundary learner and the target DNN. we utilize the *third-stage* attack which collect the information of decision boundary of the target DNN. Now, We describe our attacking pipeline in detail.

The first-stage attack is based on the assumption that the decision boundaries of the boundary learner and the target DNN largely coincide. We leverage an existing decision-based black-box attack (e.g., HSJA [15]) to generate a new perturbed image $\mathbf{x}'_{m+1}$ [1] to fool target DNNs. The whole procedure is shown in Figure 4a. Since we have obtained $\mathbf{x}'_m$ that is at the opposite side of decision boundary to $\mathbf{x}$, the binary search could be adopted to approximate the decision boundary:

$$\widetilde{\mathbf{x}}'_m = \alpha_m \cdot \mathbf{x} + (1-\alpha_m)\cdot \mathbf{x}'_m, \tag{11}$$

where $\alpha_m \in [0,1]$ is the projection radius and $\widetilde{\mathbf{x}}'_m$ is near the decision boundary while keeping $\mathbb{I}(\widetilde{\mathbf{x}}'_m) = 1$. We could estimate the gradient direction via Monte Carlo method [23]:

$$\nabla_{\widetilde{\mathbf{x}}'_m} = \frac{1}{E}\sum_{i=1}^{E}(2\cdot\mathbb{I}(\widetilde{\mathbf{x}}'_m + \delta\mathbf{u}_i) - 1)\cdot\mathbf{u}_i, \tag{12}$$

where $\{\mathbf{u}_i\}$ are $E$ randomly sampled directions with the unit length, and $\delta$ is a small constant (e.g., $10^{-3}$). Then, we move $\widetilde{\mathbf{x}}'_m$ along direction $\nabla_{\widetilde{\mathbf{x}}'_m}$ to fabricate a candidate image $\mathbf{x}'_{m+1}$:

$$\mathbf{x}'_{m+1} = \max_{\xi_m}(\widetilde{\mathbf{x}}'_m + \xi_m \cdot \frac{\nabla_{\widetilde{\mathbf{x}}'_m}}{\|\nabla_{\widetilde{\mathbf{x}}'_m}\|}), \tag{13}$$

[1]Here, $m$ denotes the current number of attack iterations, while $R$ in Eq. (9) is the number of queries on target DNNs.

subjects to:

$$\begin{cases} \mathbb{I}(\mathbf{x}'_{m+1}) = 1 \\ \left\|\mathbf{x}'_{m+1} - \mathbf{x}\right\|_2 < \left\|\mathbf{x}'_m - \mathbf{x}\right\|_2 \end{cases}, \tag{14}$$

where $\xi_m$ is the step size at the $m$-th step. Eq. (14) guarantees that $\mathbf{x}'_{m+1}$ is in adversarial regions of the boundary learner and the distance from $\mathbf{x}'_{m+1}$ to $\mathbf{x}$ is shorter than that from $\mathbf{x}'_m$ to $\mathbf{x}$. Finally, we feed $\mathbf{x}'_{m+1}$ into the target DNN $\mathcal{F}$ to check whether it is a successful perturbed image. This process is repeated $H_1$ times for exploring different directions $\nabla_{\widetilde{\mathbf{x}}'_m}$ until it finds an image $\mathbf{x}'_{m+1}$ in adversarial regions.

Figure 4 depicts the influence of the consistency of decision boundaries on the location of $\mathbf{x}'_{m+1}$. When the difference between boundaries $\partial\mathcal{P}$ and $\widetilde{\partial\mathcal{P}}$ is relatively small, the candidate perturbed image $\mathbf{x}'_{m+1}$ is accepted with a high probability. Thus, the adversary only needs to query the target DNN $\mathcal{F}$ once. However, if there is a big difference, as shown in Figure 4b, all of the candidate perturbed images along direction $\nabla_{\widetilde{\mathbf{x}}'_m}$ would be rejected, and image $\mathbf{x}'_{m+1}$ would be in non-adversarial regions of the target DNN. To alleviate this problem, we resort to the second-stage attack.

Motivated by BA [12], the second-stage attack is applied to successively walk from $\mathbf{x}'_m$ to craft a candidate image $\mathbf{x}'_{m+1}$, as presented in Figure 5a. Assuming that we randomly walks $S$ steps (i.e., $S=2$ in Figure 5a), the start image $\widehat{\mathbf{x}}'_0$ is $\mathbf{x}'_m$. At the $s$-th step walk, we randomly sample directions $\mathbf{v}_s$ at first, and calculate the vector $\mathbf{v}_s^{\perp}$ orthogonal to vector $\overrightarrow{\widehat{\mathbf{x}}'_s\mathbf{x}}$ with the unit length:

$$\mathbf{v}_s^{\perp} = \mathbf{v}_s - \frac{\mathbf{v}_s \cdot \overrightarrow{\widehat{\mathbf{x}}'_s\mathbf{x}}}{\|\mathbf{v}_s\|_2 \|\overrightarrow{\widehat{\mathbf{x}}'_s\mathbf{x}}\|_2}\overrightarrow{\widehat{\mathbf{x}}'_s\mathbf{x}}. \tag{15}$$

Then, we change length of $\mathbf{v}_s^{\perp}$ to $\gamma_1\|\widehat{\mathbf{x}}'_s - \mathbf{x}\|_2$, as shown in the dashed purple arrow in Figure 5, and generate the perturbed image $\widehat{\mathbf{x}}'_{s+1}$:

$$\widehat{\mathbf{x}}'_{s+1} = \mathbf{x} + (1-\gamma_2)\cdot\|\widehat{\mathbf{x}}'_s - \mathbf{x}\|_2 \cdot \left(\mathbf{v}_s^{\perp} - \overrightarrow{\widehat{\mathbf{x}}'_s\mathbf{x}}\right), \tag{16}$$

subjects to:

$$\begin{cases} \mathbb{I}(\widehat{x}'_{s+1}) = 1 \\ \left\|\mathbf{v}_s^{\perp} - \overrightarrow{\widehat{\mathbf{x}}'_s\mathbf{x}}\right\|_2 = 1 \end{cases}. \tag{17}$$

where $\gamma_1$ and $\gamma_2$ are step sizes for controlling decay rate of the distance between the perturbed and the original images. The

above process is repeated $H_2$ times for exploring different perturbed directions, but there is only one query on the target DNN to obtain the position of $\widehat{\mathbf{x}}'_S$ in the decision space[2]. Finally, if it is in adversarial regions of the target DNN, the perturbed image $\widehat{\mathbf{x}}'_S$ will be accepted.

There are two examples of random walks in Figure 5. Due to the fact that the start point $\widehat{\mathbf{x}}'_0$ of random walks generally stays inside the decision boundaries of the boundary learner $\mathcal{B}$ and the target DNN $\mathcal{F}$, there is a high probability that the perturbed image $\widehat{\mathbf{x}}'_S$ is accepted. Figure 5b shows a failure of random walks, when the boundaries of two models are so different that $\widehat{\mathbf{x}}'_1$ is still inside the decision boundary $\widetilde{\partial\mathcal{P}}$. In this case, we should resort to the fine-tuning process to further optimize the boundary learner.

To collect decision boundary information of target DNN for fine-tuning, we leverage the *third-stage* attack. Specifically, we apply the black-box attack in first-stage attack to fool target DNN directly and collect the query-result pairs. In order to reduce the queries on target DNN, the third-stage attack will be performed only once and we reduce the number of sampled directions (i.e., $E$ in Eq. (12)) to 50.

Note that the first-, second-, and third-stage attacks are launched in sequence, which deal with the situations with a gradually increasing gap between the two decision boundaries. If a perturbed image generated by an attack is accepted by target DNN, all subsequent attacks will not be executed.

**Fine-tuning**. It aims to further optimize the boundary learner to approximate boundaries of target DNN. During the previous attacking process, We keep the most-recent $R$ query-result pairs in $\widehat{\mathcal{D}}$, which is used to optimize the parameter set $\phi$ of the boundary learner:

$$\phi \leftarrow \arg\min_{\phi} \begin{cases} \mathcal{L}(\mathcal{B}(\widehat{\mathcal{X}}, y; \phi), \widehat{\mathcal{Y}}) & \text{Untargeted attacks} \\ \mathcal{L}(\mathcal{B}(\widehat{\mathcal{X}}, y^{\dagger}; \phi), \widehat{\mathcal{Y}}) & \text{Targeted attacks} \end{cases}, \tag{18}$$

subjects to:

$$\begin{cases} \mathbb{I}(\mathbf{x}'_m) = 1 \\ \mathbb{I}(\mathbf{x}) = 0 \end{cases}. \tag{19}$$

Eq. (19) guarantees that $\mathbf{x}'_m$ and $\mathbf{x}$ are different sides of the decision boundary of the fine-tuned boundary learner $\mathcal{B}$ and $\mathbf{x}'_m$ is in the adversarial region of the boundary learner. It serves as the basic condition of searching new perturbed examples as shown in Eq. (11). The entire optimization process is summarized in Algorithm 1.

**Algorithm 1:** Three-step attack strategy

**Input:** Original image $\mathbf{x}$, target image $\mathbf{x}^{\dagger}$, target model $\mathcal{F}$, indicator function $\mathbb{I}$, boundary learner $\mathcal{B}$, decision-based attack $\mathcal{A}$
**Output:** An adversarial image $\mathbf{x}'$

```
x'_0 ← x† ;
// 1.Warming-up
for i ← 1 to W do
    Computing x'_m using A(F, x, x'_{m-1}) ;
Fine-tune B with query results;
for m ← W to M do
    // 2.Attack
    for h ← 1 to H_1 do
        Compute x'_m from B using Eq. (13);
        if I(x'_m) = 1 then Go to line 5;
    for h ← 1 to H_2 do
        x̂'_0 ← x'_{m-1} ;
        for s ← 1 to S do
            Compute x̂'_s from B using Eq.(16) ;
            if I(x̂'_s) = 0 then Go to line 9;
        x'_m ← x̂'_S ;
        if I(x'_m) = 1 then Go to line 5;
    Compute x'_m from F using Eq. (13);
    // 3.Fine-Tuning
    Optimize B using Eq. (10)
return x'
```

## VI. Experimental Evaluation

In this section, we comprehensively evaluate the performance of DEAL against multiple target DNNs. Particularly, we will answer the following questions:

- Can DEAL benefit from the adaptive boundary learning? (Section VI-B)
- Can DEAL outperform the state-of-the-art adversarial attacks in terms of attack effectiveness and query efficiency? (Section VI-C)
- Can DEAL be robust enough to craft effective adversarial examples on target DNNs with typical defenses? (Section VI-D)

[2]Our second-stage attacks differ from BA in the number of queries required on the target model: BA needs a query for each step of random walk, whereas our method needs one query for $S$ steps. We set $S = 5$ in our implementation.

### A. Experimental Settings

**Implementation**. We implement DEAL in Python. We utilize learn2learn library [42] for building the MAML architecture in meta learning, and PyTorch library [43] for building image classification models.

For implementing the boundary learner, we use ResNet-50 as the backbone of image embedding, and the dimension of feature vector is 256. When training the boundary learner, we set $U = 100$, $B = 32$, $I = 1$ and $T = 300$. When crafting adversarial examples, we set $R = 300$, $M = 80$, $W = 2$, $H_1 = 2$, $H_2 = 4$, $S = 5$, $\gamma_1 = 1e^{-2}$, and $\gamma_2 = 1e^{-2}$, unless otherwise mentioned. As for the pre-specified label $y^{\dagger}$ in targeted attacks, we adopt $y^{\dagger} = (y + 1) \bmod K$ [30].

**Datasets**. We apply three image datasets for evaluation, i.e., CIFAR-10 [36], CIFAR-100 [36], Tiny-ImageNet [44], as summarized in Table III. These datasets are shuffled and normalized before each experiment. Note that decision-based attacks are different from score-based attacks, which requires randomly generating an initial perturbed image $\mathbf{x}'_0$ at the beginning for the original image $\mathbf{x}$. According to existing works [30], [45], $1,000$ pairs of original-initial images are randomly selected from their validation images for evaluation. In untargeted attacks, the label of initial images is randomly

selected from $[K] \setminus \{y\}$, while in targeted attacks, the label of initial images is the pre-specified label $y^{\dagger}$.

**Target Models**. For CIFAR-10, we select four target DNNs [30]: *PyramidNet-272* is a PyramidNet & Shakedrop network with 272 layers, *GDAS* is obtained via the neural architecture search, *WRN-28-10-drop* is a network with 28 layers and 10 times width expansion, and *WRN-40-10-drop* is a network with 40 layers.

In order to train the boundary learner, we select a set of substitute DNNs that have different architectures from target DNNs. Following previous studies [30], a total of 14 models are selected in CIFAR-10 and CIFAR-100 datasets, and 16 models are chosen in Tiny-ImageNet dataset.

**Performance Metrics**. We apply attack success rate (ASR) and average number of queries (AvgQ) on target DNNs as criteria. Generally, the method that has a higher ASR with a lower AvgQ is more favourable by the adversary.

**Baselines**. We employ the state-of-the-art attacks listed in Table I for comparison: BA [12], HSJA [15], QEBA [16], qFool [17], AHA [14], BiasedBA [19], BAODS [20] and HA [21]. As not supporting targeted attacks, TA [13] and GeoDA [18] are not involved in comparison. Note that HA [21] leverage the similar model structure and ensemble identical black-box attack with us. More details of these methods are described in Section II. All methods are limited to the query budget of $5,000$ in both untargeted and targeted attacks. We set the $\epsilon$ is $4.6$ in the $L_2$ norm [30] for all attacks.

**Defenses**. We choose 5 typical defenses to evaluate the robustness of all attacks.

- RND-GF [46] adds random Gaussian noise to inputs for mitigating query-based attacks, which can disrupt the structure of adversarial examples and lead to the correct classification of the target DNN.
- ComDefend [47] aims to purify the adversarial perturbations by compressing and reconstructing the input image with additional networks.
- PCL [48] enforces features of each label to lie inside a convex polytope that is maximally separated from polytopes of other labels. It changes the loss function during the training of the target DNN by adding proximity loss and contrastive proximity loss.
- AdvTrain [49] utilizes adversarial examples to train a network with strong robustness. Adversarial examples are generated by PGD [50], and are used as the training dataset for the target DNN, which makes the target DNN become more difficult to be misled.
- Blacklight [9] detects the adversary by comparing the similarity of the current query with the historical queries. It constructs image fingerprints by calculating hash values of pixels in different windows, then computes the similarity score for the input image by matching its fingerprint.

### *B. Ablation Study*

In this subsection, we separately evaluate the contribution of main modules of the proposed adaptive boundary learning, including the boundary learner initialization module in Section V-B, the boundary learner optimization module in Section V-C, and the fine-tuning strategy.

TABLE III: Details of Datasets.

| Datasets | # Labels | # Train | # Validation | # Dimensions |
|---|---|---|---|---|
| CIFAR-10 | 10 | $50,000$ | $10,000$ | $32 \times 32 \times 3$ |
| CIFAR-100 | 100 | $50,000$ | $10,000$ | $32 \times 32 \times 3$ |
| Tiny-ImageNet | 200 | $100,000$ | $10,000$ | $64 \times 64 \times 3$ |

TABLE IV: Ablation study of DEAL in CIFAR-10.

| Mode | Variations of DEAL | PyramidNet-272 | |
|---|---|---|---|
| | | ASR | AvgQ |
| Untargeted | **Full** | 100.00% | 282.25 |
| | w/o Boundary Learner Optimization | 60.30% | 0.00 |
| | w/o Boundary Learner Initialization | 100.00% | 318.28 |
| | w/o Fine-tuning Step | 99.00% | 403.99 |
| Targeted | **Full** | 97.90% | 963.54 |
| | w/o Boundary Learner Optimization | 0.00% | 0.00 |
| | w/o Boundary Learner Initialization | 99.90% | 1340.40 |
| | w/o Fine-tuning Step | 87.00% | 1486.47 |

To evaluate the contribution of the boundary learner optimization module, we generate a variation *w/o Boundary Learner Optimization*, which utilizes HSJA to directly generate adversarial examples based on the initialization of the boundary learner as described in Section V-B. Note that this variation does not need to query target DNNs.

We generate a variation *w/o Boundary Learner Initialization*, which uses the default settings of PyTorch to randomly initialize the boundary learner to investigate the contribution of the meta learning mechanism in query reduction.

In addition, to investigate the contribution of the optimization by fine-tuning in Section V-C, we also generate a variation *w/o Fine-tuning Step*. This variation removes the fine-tuning step, and thus the boundary learner will not be optimized using the queries on target DNNs.

We conduct untargeted and targeted attacks on PyramidNet-272 in CIFAR-10 dataset. In all experiments, we randomly select $1,000$ images from test images in CIFAR-10, and keep original-initial image pairs the same. The results are listed in Table IV, from which we have several key observations.

(1) Untargeted attacks are much easier than targeted attacks. In the case where we utilize DEAL with full designs to deceive PyramidNet-272, the number of queries increases from $282.25$ for untargeted attacks to $963.54$ for targeted attacks (nearly $71\%$ growth). Adversarial regions of untargeted attacks are full hyperspace except hyperspace of the original label $y$, while those of targeted attacks are hyperspace of the pre-specific label $y^{\dagger}$. Obviously, the former is much larger than the latter.

(2) Without boundary learner optimization, DEAL cannot obtain any prior knowledge about target DNNs, and can only use the transferability of adversarial examples to spoof target DNNs, which significantly reduces ASR (i.e., 0.00% in targeted attacks). This demonstrates that the boundary learner optimization module improves the success rate of DEAL.

(3) The meta learning mechanism in boundary learner initialization module dramatically reduce the number of queries on target DNNs in both untargeted and targeted attacks. In general, a larger number of queries are required for targeted

TABLE V: Results of Untargeted and Targeted Attacks in CIFAR-10.

| Mode | Methods | PyramidNet-272 | | GDAS | | WRN-28-10-drop | | WRN-40-10-drop | |
|---|---|---|---|---|---|---|---|---|---|
| | | ASR | AvgQ | ASR | AvgQ | ASR | AvgQ | ASR | AvgQ |
| Untargeted | BA [12] | 55.30% | 2807.14 | 66.10% | 2284.50 | 52.60% | 2864.53 | 59.20% | 2622.96 |
| | HSJA [15] | 100.00% | 529.05 | 100.00% | 272.05 | 100.00% | 347.70 | 100.00% | 350.59 |
| | QEBA [16] | 99.70% | 967.03 | 100.00% | 481.70 | 100.00% | 517.79 | 100.00% | 545.61 |
| | qFool [17] | 73.20% | 1765.45 | 95.70% | 558.13 | 93.60% | 664.98 | 92.70% | 648.44 |
| | AHA [14] | 99.50% | 882.20 | 99.90% | 619.06 | 100.00% | 665.83 | 99.90% | 644.35 |
| | BiasedBA [19] | 100.00% | 905.72 | 99.90% | 939.26 | 100.00% | 861.54 | 100.00% | 860.17 |
| | BAODS [20] | 97.30% | 976.46 | 100.00% | 399.65 | 100.00% | 299.04 | 100.00% | 234.36 |
| | HA [21] | 100.00% | 492.79 | 100.00% | 205.38 | 100.00% | 296.58 | 100.00% | 317.04 |
| | DEAL | **100.00%** | **282.25** | **100.00%** | **103.67** | **100.00%** | **156.64** | **100.00%** | **154.36** |
| Targeted | BA [12] | 36.50% | 3645.00 | 42.60% | 3394.02 | 36.80% | 3643.69 | 37.80% | 3604.18 |
| | HSJA [15] | 99.90% | 1391.73 | 100.00% | 1077.81 | 100.00% | 835.41 | 100.00% | 881.42 |
| | QEBA [16] | 98.80% | 2429.60 | 99.90% | 1429.52 | 99.60% | 1240.22 | 99.90% | 1315.60 |
| | qFool [17] | 69.10% | 2212.64 | 58.50% | 2495.81 | 91.90% | 1107.95 | 92.80% | 1135.06 |
| | AHA [14] | 99.20% | 1126.26 | 99.70% | 902.67 | 99.80% | 899.21 | 99.30% | 905.43 |
| | BiasedBA [19] | 99.90% | 1325.60 | 90.00% | 2551.60 | 100.00% | 1062.74 | 100.00% | 1002.66 |
| | BAODS [20] | 91.70% | 1583.56 | 95.90% | 839.75 | 99.50% | **530.03** | 99.40% | **500.20** |
| | HA [21] | 100.00% | 1316.04 | 100.00% | 1046.73 | 100.00% | 810.27 | 100.00% | 876.15 |
| | DEAL | 97.90% | **963.54** | 99.10% | **758.84** | 99.70% | **557.03** | 99.70% | **585.74** |
| | DEAL$^{\diamondsuit}$ | 100.00% | **988.48** | 100.00% | **808.26** | 100.00% | **590.90** | 100.00% | **594.47** |

[1] $\diamondsuit$: We set the budget of queries against target DNNs to 10,000.

attacks, thus the query reduction targeted attacks is more prominent than that for untargeted attacks (i.e., $28\%$ vs. $11\%$). This demonstrates the contribution of the meta learning mechanism on query reduction.

(4) Removing the fine-tuning step leads to a lower ASR and a higher AvgQ than that of the full DEAL. This demonstrates that fine-tuning can make the decision boundary of boundary learner more consistent with that of the target model.

(5) The ASR of the full DEAL is slightly lower than that of the variation without boundary learner initialization module for targeted attacks (i.e., $97.90\%$ vs. $99.90\%$). There is a situation where initialized boundary learner has the biased knowledge in some labels, which means the decision boundaries of target DNNs are significantly different from those of the $N$ substitute DNNs. Hence, DEAL needs to query target DNNs for fine-tuning, which would result in the failure of attack due to exhausted query budget.

### *C. Evaluation on Effectiveness and Query Efficiency of Attacks*

In this subsection, we will compare the performance of DEAL with 6 state-of-the-art attacks. The target models (see Section VI-A) are not protected by any defenses.

**Untargeted Attacks**. In untargeted attacks, we randomly select $1,000$ pairs of original-initial image pairs from the validation set of CIFAR-10 (see Table III). The results are listed in Table V, where the query reduction marked along AvgQ of DEAL is calculated using HSJA as the baseline. We can make three *key observations*:

(1) DEAL can reduce the AvgQ on target DNNs while maintaining an ASR of 100%. Specifically, for the 4 target models, DEAL decreases the query number by over $45\%$.

(2) PyramidNet-272 is relatively hard to be deceived, compared with the other three models in CIFAR-10, as each attack achieves its lowest attack success rate and requires the largest number of queries on PyramidNet-272. This is the reason why we choose PyramidNet-272 as the target DNN in ablation studies in Section VI-B.

(3) Substitute-based attack (e.g., BiasedBA, BAODS and HA) obtains a higher ASR with smaller AvgQ than substitute-free attacks (e.g., HSJA and qFool). Specifically, HA can achieve second-best results when attacking all target models.

**Targeted Attacks**. We randomly choose $1,000$ original-initial image pairs from the validation set of CIFAR-10 in Table III. For ease of comparison, we also employ a variation of DEAL named DEAL$^{\diamondsuit}$, which has a query budget of 10,000 to ensure an ASR of 100%. The results are summarized in Table V, where the query reduction of DEAL is also compared with HSJA. We can make the following key observations.

(1) DEAL can achieve a relatively high ASR with the smallest AvgQ in all attacks. Specifically, compared with the second-best value, DEAL achieves an ASR of more than $97.90\%$ with a maximal query reduction of $15\%$. In addition, the variation is comparable with HA in terms of ASR , and can reduce the AvgQ significantly.

(2) Compared with the existing attacks, the number of queries reduced by DEAL in targeted attacks is less than that in untargeted attacks. This may be due to untargeted attacks have a much larger solution space than target attacks, which hinders DEAL to diminish queries

(3) Adversarial attacks achieve similar success rates on WRN-28-10-drop and WRN-40-10-drop, in both untargeted and targeted attacks. This result is consistent with the findings of Ma et al. [30] and indicates that black-box models with

TABLE VI: Results of Untargeted Attacks on Defensive Models.

| | | RND-GF [46] | | ComDefend [47] | | PCL [48] | | AdvTrain [49] | | Blacklight [9] | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ASR | AvgQ | ASR | AvgQ | ASR | AvgQ | ASR | AvgQ | ASR | AvgQ |
| CIFAR-10 | HSJA [15] | 100.00% | 423.41 | 63.30% | 2313.34 | 100.00% | 333.29 | 62.50% | 2488.08 | 65.00% | 1891.82 |
| | QEBA [16] | 100.00% | 530.71 | 34.64% | 3346.56 | 100.00% | 546.16 | 96.70% | 1032.57 | 38.00% | 3182.73 |
| | qFool [17] | 79.70% | 1599.43 | 23.56% | 2895.28 | 92.40% | 854.08 | 35.20% | 3944.14 | 62.00% | 1968.53 |
| | BiasedBA [19] | 100.00% | 868.66 | **96.40%** | 1497.70 | 100.00% | 875.45 | 89.50% | 2116.84 | 0.00% | 5000.00 |
| | BAODS [20] | 99.30% | 1040.35 | 63.50% | 3477.32 | 99.90% | 322.65 | 31.30% | 4195.22 | 0.00% | 5000.00 |
| | HA [21] | 100.00% | 406.89 | 67.30% | 2184.77 | 100.00% | 254.24 | **98.40%** | 1035.97 | 59.70% | 2015.00 |
| | DEAL | 99.90% | **253.88** | 88.00% | **960.29** | **100.00%** | **140.03** | 95.80% | **884.82** | **100.00%** | **149.20** |
| CIFAR-100 | HSJA [15] | 98.00% | 861.93 | 77.90% | 1595.75 | 100.00% | 266.14 | 55.80% | 2686.56 | 83.30% | 1014.82 |
| | QEBA [16] | 98.00% | 589.42 | 4.00% | 4684.66 | 100.00% | 290.94 | 95.00% | 1035.28 | 95.00% | 1035.28 |
| | qFool [17] | 96.40% | 412.32 | 86.50% | 1165.92 | 99.60% | 184.34 | 51.30% | 2462.85 | 87.50% | 1014.82 |
| | BiasedBA [19] | 99.50% | 906.70 | 96.40% | 1186.40 | 99.90% | 888.57 | 94.90% | 1324.77 | 0.00% | 5000.00 |
| | BAODS [20] | 99.20% | 393.87 | 78.60% | 1236.71 | 99.30% | 214.33 | 80.10% | 1947.80 | 0.00% | 5000.00 |
| | HA [21] | **100.00%** | 222.85 | 77.60% | 1566.40 | 100.00% | 103.72 | **98.40%** | 831.22 | 69.00% | 1550.00 |
| | DEAL | 99.90% | **152.08** | **88.80%** | **902.00** | **100.00%** | **101.28** | 93.30% | **782.41** | **100.00%** | **96.70** |
| Tiny-ImageNet | HSJA [15] | 26.70% | 3876.13 | 44.50% | 3177.71 | 69.20% | 1914.50 | 85.35% | 2260.05 | 0.80% | 4960.10 |
| | QEBA [16] | 70.10% | 1977.87 | 46.50% | 3064.58 | 87.50% | 1114.24 | **92.50%** | 1839.73 | 0.80% | 4960.11 |
| | qFool [17] | 43.00% | 3193.22 | 51.80% | 2913.01 | 70.50% | 1709.44 | 37.50% | 4263.68 | 6.70% | 4673.63 |
| | BiasedBA [19] | 92.00% | 1974.45 | **84.20%** | 2099.04 | 79.70% | 1599.43 | 91.20% | 1814.77 | 0.00% | 5000.00 |
| | BAODS [20] | 97.30% | 1231.14 | 76.90% | **1406.57** | 99.30% | **500.97** | 82.80% | **1417.31** | 0.00% | 5000.00 |
| | HA [21] | **97.70%** | **1095.41** | 46.90% | 2968.73 | **99.70%** | 584.96 | 76.50% | 2195.76 | 38.69% | 3065.69 |
| | DEAL | 72.50% | 1837.29 | 55.80% | 2561.16 | 89.20% | 931.94 | 85.80% | 1492.48 | **76.67%** | **1655.19** |

similar architectures have near decision boundaries.

(4) The substitute-based attacks also show their superiority in targeted attacks. As illustrated in Table. V, BAODS can achieve the minimal AvgQ when attacking WRN-28-10-drop and WRN-40-10-drop. In the following, we focus on evaluating the performance of the attacks where the target models are protected with defenses.

### *D. Evaluation on Robustness of Attacks under Defenses*

In this subsection, we investigate the performance of DEAL when attacking black-box models protected by 5 typical defensive methods as described in Section VI-A. We utilize ResNet-50 as the target model, and these defenses have demonstrated their effectiveness in protecting ResNet- 50.

Based on the previous observations, untargeted attacks lead to higher success rates and fewer queries than targeted attacks, which poses a severe threat to the security of black-box models. Thus, we focus on evaluating the performance of decision-based attacks with the untargeted attack setting.

We randomly select $1,000$ original-initial image pairs from validation sets of CIFAR-10, CIFAR-100 and Tiny-ImageNet in all experiments. The results are listed in Table VI, from which we make several key observations.

(1) DEAL is robust to these defenses in different datasets, it achieves the SOTA ASR with a small AvgQ in all attacks. The exceptions include the cases to deal with Tiny-ImageNet. Images in Tiny-ImageNet have a lager dimensions than CIFAR-10 and CIFAR-100, which makes DEAL difficult to mimic the decision boundary. Thus, DEAL should conduct more queries on target DNNs, which makes the number of queries exceed the budget (i.e., 5000) and results in the drop of ASR.

The substitute-based attacks are effective in terms of ASR and AvgQ. Specially, BAODS can achieve the minimal AvgQ in the evaluation on three defenses in Tiny-ImageNet.

(2) DEAL remains effective under the defense of Blacklight, which is the state-of-the-art defense. In particular, the ASR of DEAL against Blacklight in CIFAR-10 and CIFAR-100 can achieve 100% and the ASR of Tiny-ImageNet has the best result of 76.67% compared with other attack method. This is because Blaclight detects suspicious images based on image fingerprint similarity, while DEAL generates candidate perturbed images in the boundary learner, which avoids a large number of sensitive queries on the target model and decreases the probability of being recognized as anomalous. Beside, the adversarial examples generated by the boundary learner have a large difference from the image in previous step, and even if a query image is flagged as a perturbation image, DEAL samples another perturbation direction from the boundary learner and cannot be stuck like other three decision-based attacks.

Since the label set of CIFAR-10 is smaller than that of CIFAR-100, CIFAR-100 tends to have larger adversarial regions, so the untargeted attacks in CIFAR-100 are much easier than CIFAR-10. In Tiny-ImageNet, the dimension of images becomes larger (see Table III), requiring more gradient estimations to make a successful attack. A large image dimension also results in generating more uniform adversarial perturbations, which increases the similarity of the queries on target model and influences the attack success rate.

## VII. Discussion

**Consistency of Decision Boundaries**. As described in Section IV, the consistency of boundaries between the boundary

learner and the target DNN determines how many queries on target DNNs could be reduced. In fact, it is a challenging task to quantitatively measure the consistency. But we can infer the consistency from the attack success rate and the query number, as illustrated in Section VI-B. The results prove that DEAL can gradually minimize the gap between the decision boundaries of the boundary learner and the target model.

**Distance Metrics**. We select $L_2$ norm as the metric of imperceptibility in this paper, as it is widely adopted in adversarial attacks [12], [13], [17], [31]. DEAL is sampled at the decision boundary to update the adversarial example, so we only need to make DEAL sample under other distance metrics to enable it to be extended to other metrics. In contrast, some substitute-based attacks (e.g., BiasedBA and BAODS) are difficult to extend to other distance metrics (e.g., $L_\infty$), as they combine the gradients of substitute models, which makes the adversarial perturbation more concentrated, leading to a large value on $L_\infty$.

**Shadow Dataset**. The shadow dataset is collected by the adversary which is applied to train substitute DNNs, and it may have different distribution from the training dataset of the target model. We believe that the distribution of the shadow dataset has limited impact on the attack performance of DEAL, as DEAL aims to learn how to mimic the unknown model from the auxiliary dataset and the distribution of the shadow dataset is not relevant to the learning performance.

**Potential Countermeasures**. The experimental results in Section VI-D show that Blacklight performs relatively better in defending against decision-based attacks. Thus, constructing fingerprinting and detecting adversarial perturbations based on input similarity can be a promising direction [51]. However, those defenses also have a high false positive rate on benign images. We leave the design of practical countermeasures of DEAL as future work.

**Time Consumption of Attack**. Though DEAL achieves excellent results in both undefened and defended scenarios, its time consumption is still non-negligible especially in fine-tune. Fortunately, processes before Boundary Learner Optimization (see Figure 3) are conducted only once. We leave how to improve attack time efficiency to future work.

## VIII. Conclusion

In this paper, we proposed DEAL, a decision-based query-efficient attack via adaptive boundary learning. We introduced the boundary learner, a novel architecture of local DNNs model, to approximate the decision boundaries of target DNNs. We utilized the meta learning mechanism to initialize the boundary learner, and developed a three-step optimization pipeline to make its decision boundary adapt to the target model. Extensive experimental results demonstrated that DEAL reduced the number of queries necessary for a successful attack while achieving a high attack success rate. DEAL can also maintain its effectiveness against defended target models. In future work, we will investigate techniques to further improve the effectiveness of DEAL and explore powerful defenses.

## References


[1] H. Zhang, Y. Li, Y. Huang, Y. Wen, J. Yin, and K. Guan, "Mlmodelci: An automatic cloud platform for efficient mlaas," in *MM '20*, 2020, pp. 4453–4456.

[2] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "Magface: A universal representation for face recognition and quality assessment," in *CVPR*, 2021, pp. 14 225–14 234.

[3] M. Shen, K. Ye, X. Liu, L. Zhu, J. Kang, S. Yu, Q. Li, and K. Xu, "Machine learning-powered encrypted network traffic analysis: A comprehensive survey," *IEEE Commun. Surv. Tutorials*, vol. 25, no. 1, pp. 791–824, 2023. [Online]. Available: https://doi.org/10.1109/COMST.2022.3208196

[4] M. Shen, Y. Liu, L. Zhu, X. Du, and J. Hu, "Fine-grained webpage fingerprinting using only packet length information of encrypted traffic," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 2046–2059, 2021. [Online]. Available: https://doi.org/10.1109/TIFS.2020.3046876

[5] M. Shen, J. Zhang, L. Zhu, K. Xu, and X. Du, "Accurate decentralized application identification via encrypted traffic analysis using graph neural networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 2367–2380, 2021. [Online]. Available: https://doi.org/10.1109/TIFS.2021.3050608

[6] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," in *ICLR*, 2014.

[7] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *ICLR*, 2015.

[8] M. Shen, H. Yu, L. Zhu, K. Xu, Q. Li, and J. Hu, "Effective and robust physical-world attacks on deep learning face recognition systems," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 4063–4077, 2021. [Online]. Available: https://doi.org/10.1109/TIFS.2021.3102492

[9] H. Li, S. Shan, E. Wenger, J. Zhang, H. Zheng, and B. Y. Zhao, "Blacklight: Scalable defense for neural networks against {Query-Based}{Black-Box} attacks," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 2117–2134.

[10] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin, "Black-box adversarial attacks with limited queries and information," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 80, 2018, pp. 2142–2151.

[11] S. Bhambri, S. Muku, A. Tulasi, and A. B. Buduru, "A survey of black-box adversarial attacks on computer vision models," *CoRR*, vol. abs/1912.01667, 2019.

[12] W. Brendel, J. Rauber, and M. Bethge, "Decision-based adversarial attacks: Reliable attacks against black-box machine learning models," in *ICLR*, 2018.

[13] X. Wang, Z. Zhang, K. Tong, D. Gong, K. He, Z. Li, and W. Liu, "Triangle attack: A query-efficient decision-based adversarial attack," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part V*. Springer, 2022, pp. 156–174.

[14] J. Li, R. Ji, P. Chen, B. Zhang, X. Hong, R. Zhang, S. Li, J. Li, F. Huang, and Y. Wu, "Aha! adaptive history-driven attack for decision-based black-box models," in *ICCV*, October 2021, pp. 16 168–16 177.

[15] J. Chen, M. I. Jordan, and M. J. Wainwright, "Hopskipjumpattack: A query-efficient decision-based attack," in *SP*, 2020, pp. 1277–1294.

[16] H. Li, X. Xu, X. Zhang, S. Yang, and B. Li, "QEBA: query-efficient boundary-based blackbox attack," in *CVPR*, 2020, pp. 1218–1227.

[17] Y. Liu, S. Moosavi-Dezfooli, and P. Frossard, "A geometry-inspired decision-based attack," in *ICCV*, 2019, pp. 4889–4897.

[18] A. Rahmati, S. Moosavi-Dezfooli, P. Frossard, and H. Dai, "Geoda: A geometric framework for black-box adversarial attacks," in *CVPR*, 2020, pp. 8443–8452.

[19] T. Brunner, F. Diehl, M. T. Le, and A. Knoll, "Guessing smart: Biased sampling for efficient black-box adversarial attacks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4958–4966.

[20] Y. Tashiro, Y. Song, and S. Ermon, "Diversity can be transferred: Output diversification for white-and black-box attacks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4536–4548, 2020.

[21] F. Suya, J. Chi, D. Evans, and Y. Tian, "Hybrid batch attacks: Finding black-box adversarial examples with limited queries," in *29th USENIX Security Symposium*, 2020.

[22] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 70, 2017, pp. 1126–1135.

[23] P. Chen, H. Zhang, Y. Sharma, J. Yi, and C. Hsieh, "ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *AISec@CCS*, 2017, pp. 15–26.

[24] A. Ilyas, L. Engstrom, and A. Madry, "Prior convictions: Black-box adversarial attacks with bandits and priors," in *ICLR*, 2019.
[25] M. Alzantot, Y. Sharma, S. Chakraborty, H. Zhang, C. Hsieh, and M. B. Srivastava, "Genattack: practical black-box attacks with gradient-free optimization," in *GECCO*, 2019, pp. 1111–1119.
[26] B. Ru, A. D. Cobb, A. Blaas, and Y. Gal, "Bayesopt adversarial attack," in *ICLR*, 2020.
[27] Y. Liu, X. Chen, C. Liu, and D. Song, "Delving into transferable adversarial examples and black-box attacks," in *ICLR*, 2017.
[28] W. Wang, B. Yin, T. Yao, L. Zhang, Y. Fu, S. Ding, J. Li, F. Huang, and X. Xue, "Delving into data: Effectively substitute training for black-box attack," in *CVPR*, 2021, pp. 4761–4770.
[29] J. Du, H. Zhang, J. T. Zhou, Y. Yang, and J. Feng, "Query-efficient meta attack to deep neural networks," in *ICLR*, 2020.
[30] C. Ma, L. Chen, and J. Yong, "Simulating unknown target models for query-efficient black-box attacks," in *CVPR*, 2021, pp. 11 835–11 844.
[31] T. Maho, T. Furon, and E. L. Merrer, "Surfree: A fast surrogate-free black-box attack," in *CVPR*, 2021, pp. 10 430–10 439.
[32] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 2017, pp. 506–519.
[33] X. Sun, G. Cheng, H. Li, L. Pei, and J. Han, "Exploring effective data for surrogate training towards black-box attack," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 355–15 364.
[34] M. Shen, H. Lu, F. Wang, H. Liu, and L. Zhu, "Secure and efficient blockchain-assisted authentication for edge-integrated internet-of-vehicles," *IEEE Trans. Veh. Technol.*, vol. 71, no. 11, pp. 12 250–12 263, 2022. [Online]. Available: https://doi.org/10.1109/TVT.2022.3194008
[35] Google. (2022) https://cloud.google.com/vision. Accessed April 12, 2022.
[36] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.
[37] Google. (2022) Google images. https://images.google.com/. Accessed February 18, 2022.
[38] W. Lin, C. P. Lau, A. Levine, R. Chellappa, and S. Feizi, "Dual manifold adversarial robustness: Defense against lp and non-lp adversarial attacks," in *NeurIPS*, 2020.
[39] C. Guo, J. R. Gardner, Y. You, A. G. Wilson, and K. Q. Weinberger, "Simple black-box adversarial attacks," in *ICML*, ser. Proceedings of Machine Learning Research, vol. 97, 2019, pp. 2484–2493.
[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[41] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017, pp. 2261–2269.
[42] S. M. R. Arnold, P. Mahajan, D. Datta, I. Bunner, and K. S. Zarkias, "learn2learn: A library for meta-learning research," *CoRR*, vol. abs/2008.12284, 2020.
[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *NeurIPS*, 2019, pp. 8024–8035.
[44] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
[45] Y. Guo, Z. Yan, and C. Zhang, "Subspace attack: Exploiting promising subspaces for query-efficient black-box attacks," in *NeurIPS*, 2019, pp. 3820–3829.
[46] Z. Qin, Y. Fan, H. Zha, and B. Wu, "Random noise defense against query-based black-box attacks," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
[47] X. Jia, X. Wei, X. Cao, and H. Foroosh, "Comdefend: An efficient image compression model to defend adversarial examples," in *CVPR*, 2019, pp. 6084–6092.
[48] A. Mustafa, S. H. Khan, M. Hayat, R. Goecke, J. Shen, and L. Shao, "Adversarial defense by restricting the hidden space of deep neural networks," in *ICCV*, 2019, pp. 3384–3393.
[49] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6572
[50] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *ICLR*, 2018.
[51] S. Song, Y. Chen, N. Cheung, and C. J. Kuo, "Defense against adversarial attacks with saak transform," *CoRR*, vol. abs/1808.01785, 2018.

**Meng Shen** (Member, IEEE) is a Professor at Beijing Institute of Technology, Beijing, China. He received the B.Eng degree from Shandong University, Jinan, China in 2009, and the Ph.D. degree from Tsinghua University, Beijing, China in 2014, both in computer science. His research interests include data privacy and security, blockchain applications, and encrypted traffic classification. He has authored over 50 papers in top-level journals and conferences, such as ACM SIGCOMM, IEEE JSAC, and IEEE TIFS. He has guest edited special issues on emerging technologies for data security and privacy in IEEE Network and IEEE Internet-of-Things Journal. He received the Best Paper Runner-Up Award at IEEE IPCCC 2014 and IEEE/ACM IWQoS 2020. Dr. Shen was selected by the Beijing Nova Program 2020 and was the winner of the ACM SIGCOMM China Rising Star Award 2019. He is a member of the IEEE.

**Changyue Li** received the B.Eng degree in electronic and information from Northwestern Polytechnical University, Xi'an, China in 2021. Currently he is a master student in the Department of Cyberspace Science and Technology, Beijing Institute of Technology. His research interests include Neural Network and Adversarial Attack.

**Hao Yu** is an Machine Learning Engineer of Ant Group. He received the B.Eng degree in computer science from Inner Mongolia University, Hohhot, China in 2019, and the MA.Sc in cyberspace science and technology from Beijing Institute of Technology, Beijing, China in 2022. His research interest is the security of AI models.

**Qi Li** (Senior Member, IEEE) received the Ph.D. degree from Tsinghua University. He is currently an Associate Professor with the Institute for Network Sciences and Cyberspace, Tsinghua University. He has worked with ETH Zurich and the University of Texas at San Antonio. His research interests include network and system security, particularly in Internet and cloud security, mobile security and big data security. He is currently an Editorial Board Member of the IEEE TDSC and ACM DTRAP.

**Liehuang Zhu** (Senior Member, IEEE) is a professor at the Department of Cyberspace Science and Technology at Beijing Institute of Technology. He is selected into the Program for New Century Excellent Talents in University from the Ministry of Education, P.R. China. His research interests include Internet of Things, Cloud Computing Security, Internet and Mobile Security.

**Ku Xu** (Senior Member, IEEE) received his Ph.D. from the Department of Computer Science & Technology of Tsinghua University, Beijing, China, where he serves as a full professor. He has published more than 200 technical papers and holds 11 US patents in the research areas of next-generation Internet, blockchain systems, Internet of Things (IoT), and network security. He is a member of ACM and senior member of IEEE. He has guest-edited several special issues in IEEE and Springer Journals. He is an editor of IEEE IoT Journal. He is also the Steering Committee Chair of IEEE/ACM IWQoS.