

Hard-label based Small Query Black-box Adversarial Attack

Jeonghwan Park, Paul Miller, Niall McLaughlin
Queen's University Belfast, United Kingdom
{jpark04, p.miller, n.mclaughlin}@qub.ac.uk

Abstract

We consider the hard-label based black-box adversarial attack setting which solely observes the target model's predicted class. Most of the attack methods in this setting suffer from impractical number of queries required to achieve a successful attack. One approach to tackle this drawback is utilising the adversarial transferability between white-box surrogate models and black-box target model. However, the majority of the methods adopting this approach are soft-label based to take the full advantage of zeroth-order optimisation. Unlike mainstream methods, we propose a new practical setting of hard-label based attack with an optimisation process guided by a pre-trained surrogate model. Experiments show the proposed method significantly improves the query efficiency of the hard-label based black-box attack across various target model architectures. We find the proposed method achieves approximately 5 times higher attack success rate compared to the benchmarks, especially at the small query budgets as 100 and 250.

1. Introduction

Deep neural networks (DNNs) have become a most successful back-bone technique adopted in many machine learning applications. Especially in the object classification domain, DNN models can classify objects in images with near human accuracy. However, such models are in general vulnerable to adversarial attacks which use maliciously modified input examples to mislead a target model. [28].

Depending on the amount or type of information of target models the attack methods can access, adversarial attack methods are divided into two categories: white-box and black-box attacks. White-box attacks are carried out with full access to necessary information of the target models, such as their weights and structures, to efficiently conduct back-propagation and calculate gradients [14,17]. In the black-box attack settings, however, the attack methods can only monitor

the input examples and corresponding output information from the target model [5,8,20]. The black-box adversarial attacks are, therefore, considered to be more practical attack methods compared to the white-box attacks.

Carlini et al. [7] has emphasised the importance of black-box adversarial attack to evaluate the robustness or security of applications running with DNN model. The applicable black-box attack method was introduced by Papernot et al. [27] in early 2016. Black-box attacks are carried out without knowledge of the target model, and rely on the target model's output data obtained through model querying. Therefore, such attack methods utilise optimisation processes to approximate gradients to generate adversarial examples. Depending on the method to approximate gradient, Black-box attack methods are broadly categorised into transfer based and query based attacks. Transfer based attacks utilise surrogate models to generate gradients which are directly used as the approximate gradient, since such gradients are likely remain adversarial for the target model due to their transferability [26,27]. Although various methods have been introduced to improve the transferability [14,31], the attack success rate (ASR) is yet to be satisfactory. The reason is that there lacks an adjustment procedure in transfer based attacks when the gradient of the surrogate model points to a non-adversarial region of the target model [13]. Query based attacks adopt various gradient-free optimisation processes, which search the optimal point of the functions without using their derivative, to estimate gradients [9,20,30]. Consider a DNN model which classifies an input example into a class. The real-world applications, which deal with the classification problem utilising the DNN, generate a predicted soft-label and/or hard-label as the classification result. According to Galstyan et al. [16], a hard label is one assigned to a member of a class where membership is binary, and a soft label is one which has a score (probability or likelihood) attached to it. Based on this definition, the query based attacks subdivided further into soft-

label based, and hard-label based attacks. As the target model’s classification probability distribution is not observable to the attacks, hard-label based attacks are generally considered to be more challenging task compared with soft-label based attacks. Although query based attack methods generally achieve a higher ASR in comparison with the transfer based attacks, regardless of their categories they typically suffer from a large number of queries required to perform a successful attack.

In this report a novel method is introduced to tackle the query in-efficiency of the hard-label based black-box attack methods by proposing a black-box attack method called Small-Query Black-Box Attack (SQBA). SQBA integrates the transfer based attack to take advantage of the gradients generated from a surrogate model, and applies gradient-free optimisation introduced in [8]. In summary, the contributions of this report are:

- A novel transfer based iterative gradient estimation is proposed to guide gradient direction in the black-box attack settings.
- We design a hard-label based black-box attack method, SQBA, which has an optimisation process guided by the proposed transfer based gradient estimation.
- Through the experiments the improved query-efficiency of SQBA attack method is demonstrated in comparison with several state-of-the-art hard-label based black-box attack methods.

2. Related works

In the black-box adversarial attack settings, the attack methods need to generate adversarial examples without gradient information from the target model, as they have no or limited accessibility to the model. A common choice to tackle this problem is to utilise an approximate gradient instead for generating adversarial examples. While the mainstream approach in generating the approximate gradients is to numerically estimate them by the zeroth-order optimisation algorithms [5, 8, 10, 20, 21], transfer based attacks take an advantage of white-box attack method (surrogate model) to estimate the gradients. Intuition in the transfer based attack methods is that adversarial examples which successfully attack a surrogate model are likely to remain adversarial for the target model due to their transferable characteristics [26].

Tramer et. al. demonstrated in their work [29] that adversarial examples display a high level of transferability between models, which are trained for the

same task, but have different architectures. To exploit this characteristic of the adversarial examples, transfer based attack methods train substitute (surrogate) models to achieve reasonably similar behaviour to the target model, and then apply generic white-box attacks to produce approximate gradients to attack the target model. In this report, however, it is assumed a surrogate model which well clones the target model is given, considering an easy access to the off the shelf pre-trained models, whose network structures are different from the target model.

Adversarial examples generated from the surrogate model are not always transferable to the target model, although many efforts have been made to improve the transferability of the adversarial examples by utilising momentum guidances [14, 31], or transforming input examples [15, 24, 32]. The failure of transferability is often presented when the classification probabilities of the models are different, and such discrepancy tends to cause disagreement on gradient directions.

The query based attack methods commonly utilise the iterative-querying approach adopting the zeroth-order optimisation. In details, the approximate gradient is estimated by finite difference [4, 9], random-gradient estimation [30] and natural evolution strategy (NES) [20, 21]. However, such attack methods suffer from impractically large numbers of queries to achieve successful attack. It is because they typically spend many queries to find the intermediate adversarial examples as the nature of zeroth order optimisation becomes effective if and only if examples lie near the decision boundary of the target model [5, 8, 10].

To overcome this drawback of the query based attack methods, an approach to leverage transfer based and query based methods have been adopted in recent studies. Brunner et al. [6] integrated a white-box attack method within Boundary Attack (BA) method [5], and used adversarial example originated from the gradient of the surrogate white-box model to guide optimisation process. Yang et al. [33] improved SimBA [18] query based attack and applied on-line surrogate model updating algorithm in which simulates both forward and backward process of target model. Zhou et al. [34] systematically integrated the surrogate model to generate a Jacobian matrix and improved query efficiency adopting optimisation process of SimBA. Dong et al. [13] obtained the gradient information from a surrogate model, and used it to accelerate searching process. Huang et al. [19] generated gradients from a surrogate model, and applied NES [20] optimisation process to search adversarial examples.

Most of methods reviewed here utilise the optimisation processes rely on the soft-label output of the

target model, whereas [6] focuses on the similar attack settings as SQBA method which uses the hard-label optimisation process.

3. Problem Definition

Suppose $F(x)$ is a model to deal with a k -class object classification problem, and the output of the model is denoted as $f(x) \rightarrow \mathbb{R}^k$ to quantify classification probabilities of the input example $x \in \mathbb{R}^m$ belonging to the classes. The successful classification is represented as $f_{c^\dagger}(x) = \max_k[f_k(x)]$, where c^\dagger is the true class of x .

The goal of the adversarial attack is to find a modified example \tilde{x} such that $f_{\tilde{c}}(\tilde{x}) > f_{c^\dagger}(\tilde{x})$ and $\mathcal{D}(x, \tilde{x})$ is small enough, where \tilde{c} is an adversarial class and \mathcal{D} is a dissimilarity metric. In hard-label based black-box attack settings, the attack methods iteratively inquire the target model with queries using purposely modified examples x'_t and only observe the predicted classes. The parameters θ in the target model and the classification probabilities $\{f_0(x'_t), f_1(x'_t), \dots, f_k(x'_t)\}$ are not accessible to the attack methods.

4. Optimisation Framework

The output vector of model $F(x)$ is a probability distribution over the class set $\mathbf{k} = \{1, \dots, k\}$. The classification function of $F(x)$ is denoted as $C : \mathbb{R}^m \rightarrow \mathbf{k}$ which maps input example x to the class with highest probability, and it is defined as [8]:

$$C(x) := \arg \max_{c \in \mathbf{k}} [f_c(x)] \quad (1)$$

Suppose a function $\mathcal{H}(x)$ is defined for the untargeted attack to divert the true classification $c^\dagger = C(x)$ to any unknown class, which is an adversarial class, $\tilde{c} \in \mathbf{k}$ subject to $\tilde{c} \neq c^\dagger$ as:

$$\mathcal{H}(x + \alpha\mu) := \lim_{\alpha \rightarrow \epsilon} \left(f_{\tilde{c}}(x + \alpha\mu) - \max_{\tilde{c} \neq c^\dagger} [f_c(x + \alpha\mu)] \right) \quad (2)$$

where μ is a perturbation vector generated with an intention to conduct adversarial attack, α is a scaling factor, and $\epsilon \leq 1$ is a small positive value. For the targeted attack, the adversarial class \tilde{c} becomes a designated class. When $\mathcal{H}(x')$ is positive, given modified example x' is clearly adversarial, while negative $\mathcal{H}(x')$ means x' lies in the true class region. Especially when $\mathcal{H}(x')$ is zero, x' is indicated to lie exactly on the decision boundary between two classes, c^\dagger and \tilde{c} . The examples which fall into this case are called boundary-example [23].

In hard-label based attack setting, however, $\mathcal{H}(x')$ is not a linear function and observed as the Heaviside

step function since $f_c(x)$ is not accessible. Therefore, $\mathcal{H}(x')$ needs to be re-defined as:

$$\mathcal{H}(x') := \begin{cases} 1, & \text{if } f_{\tilde{c}}(x') > \max_{\tilde{c} \neq c^\dagger} [f_c(x')] \\ -1, & \text{otherwise.} \end{cases} \quad (3)$$

where $x' = x + \alpha\mu$. When α is acceptably small, the status change of $\mathcal{H}(x')$ is observed if and only if x' is a direct neighbour of the boundary-example. In the hard-label based attack settings, therefore, it is necessary to keep the distance between x'_t and decision boundary close enough in the search phase to guarantee the successful attack with high quality adversarial examples.

As shown in Equation (3), $\mathcal{H}(x')$ is a boolean function, and therefore it is useful to indicate the successful adversarial examples. With the boolean function, the objective of the adversarial attacks can be seen as a process to find an example which satisfies:

$$\min_{x'} \mathcal{D}(x', x), \quad \text{such that } \mathcal{H}(x') = 1 \quad (4)$$

where \mathcal{D} is a distance metric to quantify the amount of applied perturbation vector to modify the given example. Typically l_p -norm, such that $p \in \{0, 2, \infty\}$, is utilised for this purpose. In this work, l_2 -norm distance metric, $\mathcal{D}(x', x) = \|x' - x\|_2$, is chosen.

Consider the objective of the adversarial attacks defined in Equation (4) as an optimisation problem. Two gradients are estimated in this work to solve such problem, and to effectively control the error in estimations. The first gradient $\nabla \mathcal{H}_w(x')$ is estimated via following process:

$$\nabla \mathcal{H}_w(x'_t) := \mu_t^i \quad \text{subject to } \min_{\mu_t^i} \mathcal{D}(x, x'_t + \delta_t \mu_t^i) \quad (5)$$

where $\mu_t^i = \nabla \mathcal{J}_S(x + v_t^i, c^\dagger)$ is a gradient generated from a pre-trained surrogate model $S(x)$, which well clones the classification behaviour of the target model $F(x)$, v_t^i is a vector to differentiate input example x , and δ_t is a scaling factor for the iteration t . The gradient μ_t^i is calculated through a white-box attack by injecting modified example $(x + v_t^i)$ and true class c^\dagger of x into the surrogate model $S(x)$. Specifically, l_∞ gradient from **Dual Gradient Method** (DGM), which is detailed in the Supplemental Material, is used for this purpose. In each iteration n gradients are generated as, $\{\mu_t^0, \mu_t^1, \dots, \mu_t^n\}$, and a gradient which can move current intermediate adversarial example x'_t to the closest point to the input example x preserving adversariality, is selected. This process is detailed further in section Section 4.1 of this report. Boolean

Algorithm 1 SQBA Adversarial Example Computation

input: classifier f , example x , true class c^\dagger , surrogate mode S
output: adversarial example \tilde{x}
 $t = 0, \beta = 1, x'_t = \text{sign}(\nabla_S \mathcal{J}(x, c^\dagger))$
 $x'_{t+1} = \text{Equation (9)} \leftarrow x, x'_t$
while queries \leq query budget **do**
 $t = t + 1, \delta_t = 10^{-2} \mathcal{D}(x, x'_t)$
 $\nabla \mathcal{H}_w = \text{Equation (5)} \leftarrow \delta_t$
 $\nabla \mathcal{H}_b = \text{Equation (6)} \leftarrow \delta_t$
 $g_t = \text{Equation (7)} \leftarrow \nabla \mathcal{H}_w, \nabla \mathcal{H}_b, \beta$
 $\hat{x}_t, \beta = \text{Equation (8)} \leftarrow x'_t, g_t$
 $x'_{t+1} = \text{Equation (9)} \leftarrow x, \hat{x}_t$
end while
 $\tilde{x} = x'_t$

function $\mathcal{H}(x')$ provides an approximation of the gradient direction, which has been utilised in various black-box attack methods [8, 10, 18, 20]. However, Equation (5) estimates $\nabla \mathcal{H}_w(x'_t)$ including its likely direction. Therefore, the proposed attack method doesn't apply such approximation to $\nabla \mathcal{H}_w(x'_t)$ for more effective attack process. The gradient estimation algorithm used in HSJA attack method [8] is adopted in the proposed attack method to estimate the second gradient $\nabla \mathcal{H}_b(x'_t)$. The algorithm generates p_t random examples μ_t^i which are uniformly distributed, where $\mu_t^i \in \mathbb{R}^m$ and $i = \{0, 1, \dots, p_t\}$. At every iteration, the number of random examples is re-calculated. In the proposed method smaller $p_t = 10\sqrt{t+1}$, compared with HSJA, is sufficient as the optimisation has already been progressed with $\nabla \mathcal{H}_w(x'_t)$. The second gradient is then estimated from the generated random examples via the Monte Carlo method as:

$$\nabla \mathcal{H}_b(x'_t) := \frac{1}{p_t} \sum_{i=1}^{p_t} \mathcal{H}(x'_t + \delta_t \mu_t^i) \mu_t^i \quad (6)$$

where $\mu_t^i \sim \mathcal{N}(O, I)$. δ_t is a scaling factor which is also applied in Equation (5), and updated at every iteration as $\delta_t = 10^{-2} \mathcal{D}(x, x'_t)$. With the estimated gradients $\nabla \mathcal{H}_w(x'_t)$ and $\nabla \mathcal{H}_b(x'_t)$, the iterate gradient g_t is approximated as:

$$g_t \approx \beta_t \frac{\nabla \mathcal{H}_w(x'_t)}{\|\nabla \mathcal{H}_w(x'_t)\|_2} + (1 - \beta_t) \frac{\nabla \mathcal{H}_b(x'_t)}{\|\nabla \mathcal{H}_b(x'_t)\|_2} \quad (7)$$

Gradient combining factor β is obtained from a boolean function whose output is switched between $\{0, 1\}$. The proposed method begins with $\beta = 1$ meaning $\nabla \mathcal{H}_w(x'_t)$ is solely used to approximate iterate gradients. When $\nabla \mathcal{H}_w(x'_t)$ is indicated to be in a local minima, β is then switched to 0 to start using $\nabla \mathcal{H}_b(x'_t)$ instead. Consider an iterative optimisation process which is given

access to the approximate gradient g_t with an intermediate adversarial example x'_t . The process performs an update as $\hat{x}_t = x'_t + \alpha g_t$. The optimisation problem in Equation (4) becomes meaningful when \hat{x}_t is a close neighbour to the boundary-example. Therefore, a tuning process is necessary to move \hat{x}_t towards the boundary such as:

$$\hat{x}_t = \min_{\alpha} (x'_t + \alpha g_t), \quad \text{such that } \mathcal{H}(\hat{x}_t) = 1 \quad (8)$$

where α is a line search parameter. The tuning process shall guarantee obtained \hat{x}_t to satisfy $\mathcal{H}(\hat{x}_t) = 1$ and also $\min \mathcal{D}(x'_t, \hat{x}_t)$, so that the next iterate example lies on the boundary. It can be noticed when an approximate gradient g_t gets into a local or global minima, the line search parameter α likely becomes small to find \hat{x}_t . Therefore, α is used to switch the boolean function output β in Equation (7). In practice $\alpha \leq 1.0$ is used for this purpose.

The final step of the iterative optimisation is increasing the correlation between input example x and calculated intermediate adversarial example \hat{x}_t . A binary search algorithm is applied for the improved correlation as

$$x'_{t+1} = \lim_{\zeta \rightarrow 0} (\zeta x + (1 - \zeta) \hat{x}_t), \quad \text{such that } \mathcal{H}(x'_{t+1}) = 1 \quad (9)$$

The process shown in Equation (9) is a simple and greedy search algorithm, however it is useful enough to be adopted in many hard-label based methods [8, 10].

4.1. Gradient from surrogate model

The transferability of gradients generated from the surrogate model is mitigated when the decision boundary of the surrogate model does not closely match the target model. However, even when this is the case, the decision boundaries of both models may still be reasonably nearby [6]. To address this drawback, the proposed SQBA attack utilises the multi-gradient method.

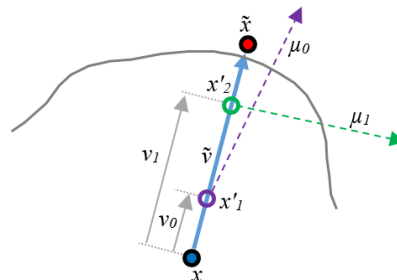


Figure 1. Illustration of the process to find iterate example search direction.

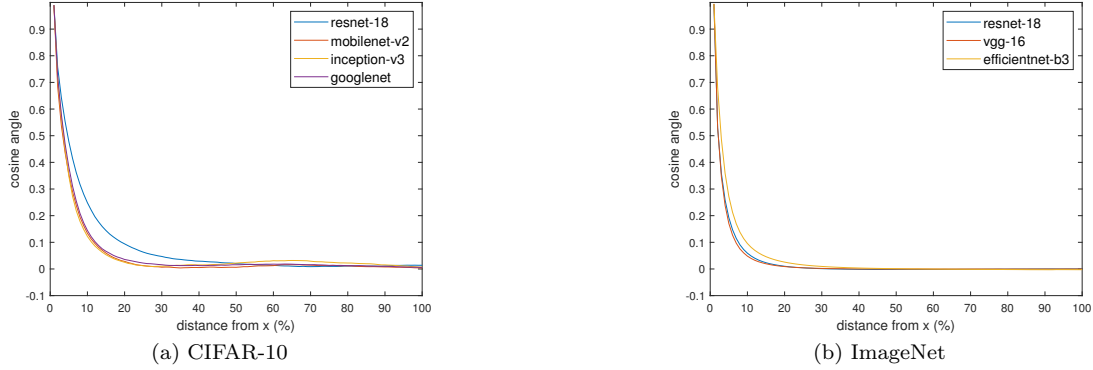


Figure 2. Change of Angle of the gradient vectors from surrogate model.

Let \tilde{x} be an adversarial example generated from surrogate model $S(x)$, and $\tilde{v} = \nabla_S \mathcal{J}(x, c^\dagger)$ be a perturbation vector to transpose the input example x to \tilde{x} . The examples that lie on \tilde{v} are calculated by $x'_i = x + \eta_i \tilde{v}$, where $\eta_i \leq 1$ is a positive scaling factor which configures the distance from input example, such as $x'_i = x$ subject to $\eta = 0$, and $x'_i = \tilde{x}$ subject to $\eta_i = 1$. As illustrated in Figure 1, a set of gradients then can be generated from $S(x'_i)$ as:

$$\mu_i = \nabla_S \mathcal{J}(x'_i, c^\dagger), \quad \text{where } i = 0, 1, \dots, n \quad (10)$$

Many works have proven that the optimal search direction in hard-label based attack is perpendicular to \tilde{v} [5, 8]. Suppose a function $\mathcal{A}(\bar{a}_1, \bar{a}_2)$ is defined to derive the angle between two vectors as $\mathcal{A}(\bar{a}_1, \bar{a}_2) := \cos \angle(\bar{a}_1, \bar{a}_2)$, where $\cos \angle(\bar{a}_1, \bar{a}_2) = (\bar{a}_1 \cdot \bar{a}_2) / \|\bar{a}_1\|_2 \|\bar{a}_2\|_2$. The angle

between \tilde{v} and a gradient vector μ_i is then calculated with $\mathcal{A}(\tilde{v}, \mu_i)$.

To exam the direction of gradients of the set of examples x'_i , $i = 0, 1, \dots, n$, dedicated 500 examples to each CIFAR-10 and ImageNet-100 datasets were taken, and average direction of the gradients from various DNN models were plotted in Figure 2. As one can expect, when η_i is small, in other words, x'_i is close to input example x , the direction of gradient vector μ_i is near parallel to \tilde{v} as $\mathcal{A}(\tilde{v}, \mu_i) \approx 1$. The direction of μ_i , however, tends to be exponentially changed even at a small increment in η , and rapidly saturated to the perpendicular direction to \tilde{v} as $\mathcal{A}(\tilde{v}, \mu_i) \approx 0$. Regardless of DNN models or dataset, the similar pattern of changes is observed that is an example which lies at $\eta_i \approx 0.2$, and afterwards their gradients are almost saturated to $\mathcal{A}(\tilde{v}, \mu_i) \approx 0$. The iterate approximate gradient, therefore, can be chosen from the calculated gradient vectors such that $\eta_i \geq 0.2$ and $\mathcal{H}(\tilde{x} + 2\delta\mu_i) = 1$ as:

$$\nabla \mathcal{H}_w(\tilde{x}) = \mu_i \quad \text{where } i = \arg \min_{\mathcal{D}} [\mathcal{D}(x, \tilde{x} + 2\delta_t \mu_i)] \quad (11)$$

Dataset	Model	Accuracy
CIFAR-10	ResNet-18	93.07%
	MobileNet-V2	93.90%
	Inception-V3	93.74%
	GoogLeNet	92.96%
ImageNet	EfficientNet-b3	83.58%
	VGG-16	78.98%
	ResNet-18	76.74%

Table 1. List of DNN Models and Classification Accuracies

Dataset	Target Model	Surrogate Model
CIFAR-10	ResNet-18	MobileNet-V2 GoogLeNet
	MobileNet-V2	Inception-V3 ResNet-18
ImageNet	EfficientNet-b3	ResNet-18 VGG-16
	VGG-16	EfficientNet-b3 ResNet-18

Table 2. Target and Surrogate models mapping

5. Experiments

Experimental analysis carried out to evaluate SQBA attack method is detailed here. The efficiency of SQBA attack was compared with several state-of-the-art attack methods which utilise the hard-label based adversarial attack setting.

5.1. Benchmarks

SQBA attack method was compared with four state-of-the-art black-box attack methods: HSJA Attack [8], Sign-Opt Attack [10], BA Attack [5] and Biased-BA Attack [6]. These attack methods commonly fall into the hard-label based black-box adversarial attack setting as they solely observe the predicted class against

CIFAR-10: ResNet-18 at $\rho(\tilde{x}) \leq 0.10$						
Query Budget		100	250	500	750	1000
Attack Methods	HSJA [8]	1.2%	9.7%	33.2%	46.9%	60.5%
	BA [5]	2.9%	3.7%	4.0%	4.2%	4.5%
	Sign-Opt [10]	8.5%	10.5%	11.4%	24.5%	32.1%
	Biased-BA(GoogLeNet) [6]	3.0%	5.5%	13.7%	19.2%	30.3%
	SQBA(MobileNet-V2)	38.4%	52.1%	66.1%	74.7%	82.0%
	SQBA(GoogLeNet)	28.1%	40.4%	60.5%	73.0%	79.2%

Table 3. Query Budgets and ASR of hard-label based black-box attacks on ResNet-18 trained on CIFAR-10

CIFAR-10: MobileNet-V2 at $\rho(\tilde{x}) \leq 0.10$						
Query Budget		100	250	500	750	1000
Attack Methods	HSJA [8]	3.9%	23.4%	67.6%	78.4%	85.6%
	BA [5]	9.3%	10.0%	11.9%	13.1%	16.1%
	Sign-Opt [10]	26.8%	32.0%	37.3%	61.3%	68.9%
	Biased-BA(Inception-V3) [6]	11.9%	28.0%	56.3%	68.0%	73.8%
	SQBA(Inception-V3)	71.9%	80.2%	89.4%	91.9%	95.6%
	SQBA(ResNet-18)	72.0%	83.6%	91.1%	95.6%	97.6%

Table 4. Query Budgets and ASR of hard-label based black-box attacks on MobileNet-V2 trained on CIFAR-10

the given example. Especially Biased-BA is taking the similar approach as SQBA which embeds a transfer based algorithm within the base line hard-label based attack method. Model parameters or model output logits are not accessible to the attack methods used in the experiments. All the benchmark methods were configured to perform the untargeted attack scenario. In the untargeted attack scenario, an attack becomes successful if the target model predicts a different class from the true class of the given example. Note that Biased-BA attack was initialised with a random noise instead of a target example to conduct such an attack scenario. Attack strategies of the benchmark methods commonly have three stages: First, they find an initial adversarial example (starting point); Second, they iteratively search a path towards a further optimised example, which is closer to the original example but still maintains the adversariality; and finally, they stop searching when the optimisation process satisfies stopping criteria which is given the query budget.

5.2. Datasets and Models

The experiments were conducted over two standard image datasets: CIFAR-10 [22] and ImageNet-100 [11]. CIFAR-10 dataset is composed of RGB image examples in the $3 \times 32 \times 32$ dimensional spaces. It has 10 classes with 6,000 examples per class. ImageNet-100 is a subset of ImageNet-1k dataset. It consists of random 100 classes out of 1K classes of the full dataset, and each class has 1,350 examples. All the models used in the experiments are listed in Table 1. The models were off the shelf, which were not re-trained nor mod-

ified, obtained from publicly available libraries, such as Zenodo [1] for CIFAR-10 and Pytorch torchvision library [3] for ImageNet-1K models. All the examples were transformed as instructed by the libraries. For example, examples in ImageNet-100 dataset were rescaled to $3 \times 256 \times 256$ dimensional space, and centre cropped to $3 \times 224 \times 224$ dimensional RGB examples. From the validation split of each dataset, dedicated subsets were created with the examples correctly classified by the target models to avoid artificial inflation of the success rate. 1000 examples were randomly down sampled further from the subsets to be used in the evaluation. As the objective dataset, ImageNet-100, is a direct subset of ImageNet-1K dataset, DNN models trained on ImageNet-1K dataset are naturally ready for the objective dataset. In the rest of the report, ImageNet-100 dataset is presented as ImageNet unless otherwise specified.

For fair and easy comparison, the DNN models, which are publicly well known and pre-trained with the datasets described above, were chosen for the experiments. Specifically, for the experiments with CIFAR-10 dataset, two DNN models, ResNet-18 and MobileNet-V2 were used as the target models. SQBA attack method conducted two attacks on each target model utilising different surrogate models, whose architectures are different from others, to demonstrate the generalisation of the method. To attack ResNet-18 target model, MobileNet-V2 and GoogLeNet were chosen for the surrogate model, while Inception-V3 and ResNet-18 were used to attack MobileNet-V2. For the experiments with ImageNet dataset, VGG-16 and

ImageNet: EfficientNet-b3 at $\rho(\tilde{x}) \leq 0.10$						
Query Budget		100	250	500	750	1000
Attack Methods	HSJA [8]	1.7%	5.2%	12.0%	19.3%	24.6%
	BA [5]	2.7%	3.0%	3.0%	3.2%	3.4%
	Sign-Opt [10]	5.5%	7.5%	7.2%	7.5%	7.3%
	Biased-BA(VGG-16) [6]	2.0%	3.4%	7.5%	15.6%	21.6%
	SQBA(ResNet-18)	33.8%	38.2%	46.8%	49.4%	55.2%
	SQBA(VGG-16)	30.4%	37.4%	45.0%	48.7%	53.2%

Table 5. Query Budgets and ASR of hard-label based black-box attacks on EfficientNet-b3 trained on ImageNet

ImageNet: VGG-16 at $\rho(\tilde{x}) \leq 0.10$						
Query Budget		100	250	500	750	1000
Attack Methods	HSJA [8]	2.2%	10.2%	24.0%	31.2%	40.8%
	BA [5]	5.0%	6.0%	6.8%	6.2%	9.0%
	Sign-Opt [10]	11.4%	13.6%	14.4%	15.8%	15.7%
	Biased-BA(ResNet-18) [6]	6.3%	9.0%	22.7%	34.4%	40.6%
	SQBA(Inception-V3)	57.6%	62.8%	68.4%	72.4%	75.8%
	SQBA(ResNet-18)	65.5%	69.2%	73.3%	78.0%	80.9%

Table 6. Query Budgets and ASR of hard-label based black-box attacks on VGG-16 trained on ImageNet

EfficientNet-b3 were attacked by the methods. SQBA attack method utilised ResNet-18 and EfficientNet-b3 models to perform attacks on VGG-16 target model. ResNet-18 and VGG-16 models were used as the surrogate models to attack EfficientNet-b3 target model. Each group of models used in the experiments are detailed in Table 2.

5.3. Adversarial Attack Budgets

In order to generate meaningful adversarial examples, it is necessary to impose reasonable constraints to the attack. An unconstrained attack perhaps achieve a successful attack by, for example, disturbing target model with tremendous number of queries, or introducing large and brutal perturbations to an input example that would alter its semantics [7]. Since such attacks are not considered to satisfy the goal of adversary, which is an adversarial example shall have human-imperceptible perturbation, two restrictions were introduced in the experiments. In the rest of this report, the restrictions are called attack budgets.

The first attack budget used in order to quantify the performance of the attack methods is a metric used in [12] as:

$$\rho(\tilde{x}) := \frac{\|\mu\|_2}{\|x\|_2} \quad (12)$$

where μ is the perturbation vector added to the original example x to achieve a successful adversarial attack. This metric was originally used to measure the average robustness. In this report, however, $\rho(\tilde{x})$ is used as the perturbation budget to determine the successfulness of an individual attack. $\rho(\tilde{x})$ in fact indicates the distance

between x and \tilde{x} , therefore the lower value means the higher robustness.

As the second attack budget, the number of queries that an attack method can use is also restricted by the five different budgets such as {100, 250, 500, 750, 1000}. The highest allowed number of queries 1000 may seem to be in-efficient. However, some object classification applications which utilise modern techniques such as GPU or FPGA accelerated systems are already capable of processing more than 1000 images per second [2].

5.4. Attacks on CIFAR-10 Dataset

In this section the results of black-box adversarial attacks with CIFAR-10 dataset are reported. Pre-trained ResNet-18 and MobileNet-V2, which achieves 93.07% and 93.90% classification accuracies respectively, were chosen for the target models. All the attacks were limited with the perturbation budget $\rho(\tilde{x}) < 0.1$, and to at most the five query budgets as described in Section 5.3. Any successful attack achieved with queries or perturbation which exceeded the budgets was reported as a failed attack.

As shown in Table 2, Two SQBA with different surrogate models performed black-box attacks to each target model, and their performance was compared with four state-of-the-art hard-label based attack methods listed in Section 5.1. The attack results over ResNet-18 target model is presented in Table 3. SQBA with both MobileNet-V2 and GoogLeNet surrogate models outperformed benchmark methods across all the query budgets, specifically SQBA with MobileNet-V2

CIFAR-10: ResNet-18-AT at $\rho(\bar{x}) \leq 0.10$				CIFAR-10: MobileNet-V2-AT at $\rho(\bar{x}) \leq 0.10$			
Query Budget		750	1000	Query Budget		750	1000
Attack Methods	HSJA [8]	16.5%	22.1%	HSJA [8]	24.5%	27.6%	
	Sign-Opt [10]	8.6%	9.5%	Sign-Opt [10]	12.1%	13.5%	
	Biased-BA(GoogLeNet) [6]	3.0%	5.9%	Biased-BA(Inception-V3) [6]	10.5%	25.1%	
	SQBA(GoogLeNet)	18.0%	22.5%	SQBA(Inception-V3)	25.2%	31.9%	

Table 7. Comparison of black-box attacks on defended models for CIFAR-10 dataset

showed the best attack performance achieving 82.0% ASR at 1000 query budget. Especially in the very small query scenarios, where queries are limited with 100 and 250 at most, SQBA attacks performed approximately 4 and 5 times higher ASR respectively compared with the best performing benchmark. Table 4 details attack performances on MobileNet-V2 target model. SQBA attacks also outperformed benchmarks in this experiment. At 100 query budget, both SQBA attacks with Inception-V3 and ResNet-18 surrogate models achieved over 70.0% ASR while other benchmark methods didn't show meaningful attack performances yet.

5.5. Attacks on ImageNet Dataset

This section details the performances of black-box adversarial attacks conducted against target-models trained on ImageNet dataset. For the experiments, pre-trained EfficientNet-b3 and VGG-16 DNN models were chosen for the target models, and they respectively achieve 83.58% and 78.98% classification accuracies over ImageNet-100 dataset. Surrogate models used in SQBA attacks are detailed in Table 2. The benchmark methods and attack budgets used in CIFAR-10 experiments were also applied in this series of experiments to evaluate the performance of the SQBA methods.

Table 5 and Table 6 show the performances of the attack methods over ImageNet models, EfficientNet-b3 and VGG-16 respectively. While it is a general observation that SQBA attacks outperformed all other methods, in detail SQBA attacks achieved approximately 5 times higher ASR compared with the benchmarks at small query budgets 100 and 250, interestingly all methods struggled to achieve successful attacks to EfficientNet-b3 model. At 1000 query budget, HSJA [8] performed 40% degraded ASR compared to the same attack to VGG-16 target model, while SQBA with ResNet-18 achieved 32% degraded ASR compared with the attack on VGG-16.

5.6. Attacks on Defended models

Attack methods were further evaluated with the defended models. ResNet-18-AT and MobileNet-V2-

AT models have identical architectures to the original models used in Section 5.4. They were trained with CIFAR-10 dataset using the PGD adversarial training method [25] to achieve 67.24% and 68.49% standard classification accuracies respectively. Table 7 show the comparison of black-box attacks on both defended models. In these experiments, two query budgets, 750 and 1000, were considered, and BA attack method which didn't show meaningful ASR in the previous experiments was not included. GoogLeNet and Inception-V3 were chosen for the surrogate models to attack ResNet-18-AT and MobileNet-V2-AT respectively. It can be observed that the defended models successfully reduced the ASR of all the attack methods to approximately 70.0% less than their attacks on the original models. Although SQBA attacks showed the improved performances compared with the benchmarks, their performance was also heavily degraded achieving 22.5% on ResNet-18-AT and 31.9% on MobileNet-V2-AT target models at 1000 query budget, which were 71.5% and 66.6% respectively less ASR compared with the same attacks on the original models.

6. Conclusion

SQBA method which combines transfer based and hard-label based adversarial attacks was introduced. SQBA is designed to generate likely adversarial examples for an unknown classifier whose predicted class is only accessible. The method takes an advantage of the transfer based algorithm for the rapid search of an initial convergence point which is followed by the guided optimisation procedure. Through the experiments it was demonstrated that SQBA is capable of achieving higher ASR with smaller query budgets compared with the benchmarks. One drawback of black-box attack methods which utilise transfer based algorithms degrades the performance when gradients from the transfer based algorithms disagree with the target model, and SQBA is not immune from this problem. Future work may focus on improving the adversarial transferability in the SQBA method to address this issue.

References

- [1] <https://github.com/huyvnphan>. 6
- [2] <https://github.com/u39kun/deep-learning-benchmark>. Technical report. 7
- [3] <https://pytorch.org/vision/stable/models.html>. 6
- [4] A. N. Bhagoji, W. He, B. Li, and D. Song. Practical black-box attacks on deep neural networks using efficient query mechanisms. *The European Conference on Computer Vision*, 2018. 2
- [5] W. Brendel, J. Rauber, and M. Bethge. Decision based adversarial attacks: Reliable attacks against black-box machine learning models. *International Conference on Learning Representation*, 2018. 1, 2, 5, 6, 7
- [6] T. Brunner, F. Diehl, M. T. Le, and A. Knoll. Guessing smart: Biased sampling for efficient black-box adversarial attacks. *arXiv preprint arXiv:1812.09803v3*, 2019. 2, 3, 4, 5, 6, 7, 8
- [7] N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras, I. Goodfellow, A. Madry, and A. Kurakin. On evaluating adversarial robustness. *arXiv preprint arXiv:1902.06705*, 2019. 1, 7
- [8] J. Chen, M. I. Jordan, and M. J. Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. *IEEE Symposium on Security and Privacy*, 2020. 1, 2, 3, 4, 5, 6, 7, 8
- [9] P. Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C. J. Hsieh. Zoo: Zeroth order optimisation based black-box attacks to deep neural networks without training substitute models. *ACM Workshop on Artificial Intelligence and Security*, 2017. 1, 2
- [10] M. Cheng, S. Singh, P. Chen, P.Y. Chen, S. Liu, and C.J. Hsieh. Sign-opt: A query-efficient hard-label adversarial attack. *International Conference on Learning Representations*, 2020. 2, 4, 5, 6, 7, 8
- [11] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. 6
- [12] S.M.M. Dezfooli, A. Fawzi, P. Frossard, and E.P.F. Lausanne. "deepfool": A simple and accurate method to fool deep neural networks. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 7
- [13] Y. Dong, S. Cheng, T. Pang, H. Su, and J. Zhu. Query-efficient black-box adversarial attacks guided by a transfer-based prior. *arXiv:2203.06560v1*, 2022. 1, 2
- [14] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li. Boosting adversarial attacks with momentum. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1, 2
- [15] Y. Dong, T. Pang, H. Su, and J. Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2
- [16] A. Galstyan and P. R. Cohen. Empirical comparison of "hard" and "soft" label propagation for relational classification. *The 17th International Conference on Inductive Logic Programming*, 2007. 1
- [17] I. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *International Conference on Learning Representations*, 2015. 1
- [18] C. Guo, J. R. Gardner, Y. You, A. G. Wilson, and K. Q. Weinberger. Simple black-box adversarial attacks. *International Conference on Learning Representation*, 2019. 2, 4
- [19] Z. Huang and T. Zhang. Black-box adversarial attack with transferable model-based embedding. *International Conference on Learning Representations*, 2020. 2
- [20] A. Ilyas, L. Engstrom, A. Athalye, and J. Lin. Black-box adversarial attacks with limited queries and information. *International Conference on Machine Learning*, 2018. 1, 2, 4
- [21] A. Ilyas, L. Engstrom, and A. Madry. Prior convictions: Black-box adversarial attacks with bandits and priors. *International Conference on Learning Representations*, 2019. 2
- [22] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Technical Report, Citeseer*, 2009. 6
- [23] H. Li, X. Xu, X. Zhang, S. Yang, and B. Li. Qeba: Query-efficient boundary-based blackbox attack. *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 3
- [24] J. Lin, C. Song, K. He, L. Wang, and J. E. Hopcroft. Nesterov accelerated gradient and scale invariance for adversarial attacks. *International Conference on Learning Representations*, 2020. 2
- [25] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *International Conference on Learning Representations*, 2018. 8
- [26] N. Papernot, P. McDaniel, and I. Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *arXiv preprint arXiv:1605.07277*, 2016. 1, 2
- [27] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z.B. Celik, and A. Swami. Practical black-box attacks against machine learning. *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, 2017. 1
- [28] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *International Conference on Learning Representations*, 2013. 1
- [29] F. Tramer, N. Papernot, I. Goodfellow, D. Boneh, and P. D. McDaniel. The space of transferable adversarial examples. *arXiv preprint arXiv:1704.03453*, 2017. 2
- [30] C. Tu, P. Ting, P. Chen, S. Liu, H. Zhang, J. Yi, C. Hsieh, and S. Cheng. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. *The Thirty-Third AAAI Conference on Artificial Intelligence*, 2019. 1, 2
- [31] Xiaosen Wang and Kun He. Enhancing the transferability of adversarial attacks through variance tuning.

IEEE Conference on Computer Vision and Pattern Recognition, 2021. [1](#), [2](#)

- [32] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. Yuille. Improving transferability of adversarial examples with input diversity. *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. [2](#)
- [33] J. Yang, Y. Jiang, X. Huang, B. Ni, and C. Zhao. Learning black-box attackers with transferable priors and query feedback. *Conference on Neural Information Processing System*, 2020. [2](#)
- [34] L. Zhou, P. Cui, X. Zhang, Y. Jiang, and S. Yang. Adversarial eigen attack on black-box models. *arXiv:2009.00097*, 2020. [2](#)