Simple Black-box Adversarial Attacks

Chuan Guo [1] **Jacob R. Gardner** [2] **Yurong You** [1] **Andrew Gordon Wilson** [1] **Kilian Q. Weinberger** [1]

Abstract

We propose an intriguingly simple method for the construction of adversarial images in the black-box setting. In constrast to the white-box scenario, constructing black-box adversarial images has the additional constraint on query budget, and efficient attacks remain an open problem to date. With only the mild assumption of continuous-valued confidence scores, our highly query-efficient algorithm utilizes the following simple iterative principle: we randomly sample a vector from a predefined orthonormal basis and either add or subtract it to the target image. Despite its simplicity, the proposed method can be used for both untargeted and targeted attacks – resulting in previously unprecedented query efficiency in both settings. We demonstrate the efficacy and efficiency of our algorithm on several real world settings including the Google Cloud Vision API. We argue that our proposed algorithm should serve as a strong baseline for future black-box attacks, in particular because it is extremely fast and its implementation requires less than 20 lines of PyTorch code.

1. Introduction

As machine learning systems become prevalent in numerous application domains, the security of these systems in the presence of malicious adversaries becomes an important area of research. Many recent studies have shown that decisions output by machine learning models can be altered arbitrarily with imperceptible changes to the input (Carlini & Wagner, 2017b; Szegedy et al., 2014). These attacks on machine learning models can be categorized by the capabilities of the adversary. *White-box* attacks require the adversary to have complete knowledge of the target model, whereas *black-box* attacks require only queries to the target model that may return complete or partial information.

Seemingly all models for classification of natural images are susceptible to white-box attacks (Athalye et al., 2018), which indicates that natural images tend to be close to decision boundaries learned by machine learning classifiers. Although often misunderstood as a property of neural networks (Szegedy et al., 2014), the vulnerability towards adversarial examples is likely an inevitability of classifiers in high-dimensional spaces with most data distributions (Fawzi et al., 2018; Shafahi et al., 2018).

If adversarial examples (almost) always exist, attacking a classifier turns into a search problem within a small volume around a target image. In the white-box scenario, this search can be guided effectively with gradient descent (Szegedy et al., 2014; Carlini & Wagner, 2017b; Madry et al., 2017). However, the black-box threat model is more applicable in many scenarios. Here, queries to the model may incur a significant cost of both time and money, and the number of black-box queries made to the model therefore serves as an important metric of efficiency for the attack algorithm. Attacks that are too costly, or are easily defeated by query limiting, pose less of a security risk than efficient attacks. To date, the average number of queries performed by the best known black-box attacks remains high despite a large amount of recent work in this area (Chen et al., 2017; Brendel et al., 2017; Cheng et al., 2018; Guo et al., 2018; Tu et al., 2018; Ilyas et al., 2018a). The most efficient and complex attacks still typically require upwards of tens or hundreds of thousands of queries. A method for query efficient black-box attacks has remained an open problem.

Machine learning services such as Clarifai or Google Cloud Vision only allow API calls to access the model's predictions and fall therefore in the black-box category. These services do not release any internal details such as training data and model parameters; however, their predictions return continuous-valued confidence scores. In this paper we propose a simple, yet highly efficient black-box attack that exploits these confidence scores using a very simple intuition: if the distance to a decision boundary is small, we don't have to be too careful about the exact direction along which we traverse towards it. Concretely, we repeatedly pick a *random* direction among a pre-specified set of orthog-

[1]Department of Computer Science, Cornell University, Ithaca, New York, USA [2]Uber AI Labs, San Francisco, California, USA. Correspondence to: Chuan Guo <cg563@cornell.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

onal search directions, use the confidence scores to check if it is pointing towards or away from the decision boundary, and perturb the image by adding or subtracting the vector from the image. Each update moves the image further away from the original image and towards the decision boundary.

We provide some theoretical insight on the efficacy of our approach and evaluate various orthogonal search subspaces. Similar to Guo et al. (2018), we observe that restricting the search towards the low frequency end of the discrete cosine transform (DCT) basis is particularly query efficient. Further, we demonstrate empirically that our approach achieves a similar success rate to state-of-the-art black-box attack algorithms, however with an unprecedented low number of black-box queries. Due to its simplicity — it can be implemented in PyTorch in under 20 lines of code[1] — we consider our method a new and perhaps surprisingly strong baseline for adversarial image attacks, and we refer to it as *Simple Black-box Attack (SimBA)*.

2. Background

The study of adversarial examples concerns with the robustness of a machine learning model to small changes in the input. The task of image classification is defined as successfully predicting what a human sees in an image. Naturally, changes to the image that are so tiny that they are imperceptible to humans should not affect the label and prediction. We can formalize such a robustness property as follows: given a model h and some input-label pair $(\mathbf{x}, y)$ on which the model correctly classifies $h(\mathbf{x}) = y$, h is said to be ρ-robust with respect to perceptibility metric $d(\cdot, \cdot)$ if

$$h(\mathbf{x}') = y \;\; \forall \mathbf{x}' \in \{\mathbf{x}' \mid d(\mathbf{x}', \mathbf{x}) \leq \rho \}.$$

The metric d is often approximated by the L_0, L_2 and L_∞ distances to measure the degree of visual dissimilarity between the clean input $\mathbf{x}$ and the perturbed input $\mathbf{x}'$. Following (Moosavi-Dezfooli et al., 2016; Moosavi-Dezfooli et al., 2017), for the remainder of this paper we will use $d(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|_2$ as the perceptibility metric unless specified otherwise. Geometrically, the region of imperceptible changes is therefore defined to be a small hypersphere with radius ρ, centered around the input image $\mathbf{x}$.

Recently, many studies have shown that learned models admit directions of non-robustness even for very small values of ρ (Moosavi-Dezfooli et al., 2016; Carlini & Wagner, 2017b). Fawzi et al. (2018); Shafahi et al. (2018) verified this claim theoretically by showing that adversarial examples are inherent in high-dimensional spaces. These findings motivate the problem of finding adversarial directions δ that alter the model's decision for a perturbed input $\mathbf{x}' = \mathbf{x} + \delta$.

[1] https://github.com/cg563/simple-blackbox-attack

Targeted and untargeted attacks. The simplest success condition for the adversary is to change the original correct prediction of the model to an arbitrary class, i.e., $h(\mathbf{x}') \neq y$. This is known as an *untargeted attack*. In contrast, a *targeted attack* aims to construct $\mathbf{x}'$ such that $h(\mathbf{x}') = y'$ for some chosen target class y'. For the sake of brevity, we will focus on untargeted attacks in our discussion, but all arguments in our paper are also applicable to targeted attacks. We include experimental results for both attack types in section 4.

Loss minimization. Since the model outputs discrete decisions, finding adversarial perturbations to change the model's prediction is, at first, a discrete optimization problem. However, it is often useful to define a surrogate loss $\ell_y(\cdot)$ that measures the degree of certainty that the model h classifies the input as class y. The adversarial perturbation problem can therefore be formulated as the following constrained continuous optimization problem of minimizing the model's classification certainty:

$$\min_{\delta} \;\; \ell_y(\mathbf{x} + \delta) \text{ subject to } \;\; \|\delta\|_2 < \rho.$$

When the model h outputs probabilities $p_h(\cdot \mid \mathbf{x})$ associated with each class, one commonly used adversarial loss is the probability of class y: $\ell_y(\mathbf{x}') = p_h(y \mid \mathbf{x}')$, essentially minimizing the probability of a correct classification. For targeted attacks towards label y' a common choice is $\ell_{y'}(\mathbf{x}') = -p_h(y' \mid \mathbf{x}')$, essentially maximizing the probability of a misclassification as class y'.

White-box threat model. Depending on the application domain, the attacker may have various degrees of knowledge about the target model h. Under the *white-box* threat model, the classifier h is provided to the adversary. In this scenario, a powerful attack strategy is to perform gradient descent on the adversarial loss $\ell_y(\cdot)$, or an approximation thereof. To ensure that the changees remain imperceptible, one can control the perturbation norm, $\|\delta\|_2$, by early stopping (Goodfellow et al., 2015; Kurakin et al., 2016) or by including the norm directly as a regularizer or constraint into the loss optimization (Carlini & Wagner, 2017b).

Black-box threat model. Arguably, for many real-world settings the white-box assumptions may be unrealistic. For instance, the model h may be exposed to the public as an API, allowing only queries on inputs. Such scenarios are common when attacking machine learning cloud services such as Google Cloud Vision and Clarifai. This *black-box* threat model is much more challenging for the adversary, since gradient information may not be used to guide the finding of the adversarial direction δ, and each query to the model incurs a time and monetary cost. Thus, the adversary is tasked with an additional goal of minimizing the number of black-box queries to h while succeeding in constructing an imperceptible adversarial perturbation. With a slight abuse of notation this poses a modified constrained

Algorithm 1 SimBA in Pseudocode

procedure SIMBA($\mathbf{x}, y, Q, \epsilon$)
  $\delta = \mathbf{0}$
  $\mathbf{p} = p_h(y \mid \mathbf{x})$
  while $\mathbf{p}_y = \max_{y'} \mathbf{p}_{y'}$ **do**
    Pick randomly without replacement: $\mathbf{q} \in Q$
    for $\alpha \in \{\epsilon, -\epsilon\}$ **do**
      $\mathbf{p}' = p_h(y \mid \mathbf{x} + \delta + \alpha\mathbf{q})$
      if $\mathbf{p}'_y < \mathbf{p}_y$ **then**
        $\delta = \delta + \alpha\mathbf{q}$
        $\mathbf{p} = \mathbf{p}'$
        break
  return δ

optimization problem:

$$\min_{\delta} \ \ell_y(\mathbf{x} + \delta) \text{ subject to: } \ \|\delta\|_2 < \rho, \text{queries} \leq B$$

where B is some fixed budget for the number of queries allowed during the optimization. For iterative methods that query, the budget B constrains the number of iterations the algorithm may take, hence requiring that the attack algorithm converges to a solution very quickly.

3. A Simple Black-box Attack

We assume we have some image $\mathbf{x}$ which a black-box neural network, h, classifies $h(\mathbf{x}) = y$ with predicted confidence or output probability $p_h(y \mid \mathbf{x})$. Our goal is to find a small perturbation δ such that the prediction $h(\mathbf{x}+\delta) \neq y$. Although gradient information is absent in the black-box setting, *we argue that the presence of output probabilities can serve as a strong proxy to guide the search for adversarial images.*

Algorithm. The intuition behind our method is simple (see pseudo-code in Algorithm 1): for *any* direction $\mathbf{q}$ and some step size ϵ, one of $\mathbf{x} + \epsilon\mathbf{q}$ or $\mathbf{x} - \epsilon\mathbf{q}$ is likely to decrease $p_h(y \mid \mathbf{x})$. We therefore repeatedly pick random directions $\mathbf{q}$ and either add or subtract them. To minimize the number of queries to $h(\cdot)$ we always first try adding $\epsilon\mathbf{q}$. If this decreases the probability $p_h(y \mid \mathbf{x})$ we take the step, otherwise we try subtracting $\epsilon\mathbf{q}$. This procedure requires between 1.4 and 1.5 queries per update on average (depending on the data set and target model). Our proposed method – *Simple Black-box Attack* (SimBA) – takes as input the target image label pair $(\mathbf{x}, y)$, a set of orthonormal candidate vectors Q and a step-size $\epsilon > 0$. For simplicity we pick $\mathbf{q} \in Q$ uniformly at random. To guarantee maximum query efficiency, we ensure that no two directions cancel each other out and diminish progress, or amplify each other and increase the norm of δ disproportionately. For this reason we pick $\mathbf{q}$ *without replacement* and restrict all vectors in Q to be *orthonormal*. As we show later, this results in a guaranteed perturbation norm of $\|\delta\|_2 = \sqrt{T}\epsilon$ after T updates. The only hyper-parameters of SimBA are the set of orthogonal search vectors Q and the step size ϵ.

Cartesian basis. A natural first choice for the set of orthogonal search directions Q is the standard basis $Q = I$, which corresponds to performing our algorithm directly in pixel space. Essentially each iteration we are increasing or decreasing one color of a single randomly chosen pixel. Attacking in this basis corresponds to an L_0-attack, where the adversary aims to change as few pixels as possible.

Discrete cosine basis. Recent work has discovered that random noise in low frequency space is more likely to be adversarial (Guo et al., 2018). To exploit this fact, we follow Guo et al. (2018) and propose to exploit the *discrete cosine transform* (DCT). The discrete cosine transform is an orthonormal transformation that maps signals in a 2D image space $\mathbb{R}^{d\times d}$ to frequency coefficients corresponding to magnitudes of cosine wave functions. In what follows, we will refer to the set of orthonormal frequencies extracted by the DCT as Q_{DCT}. While the full set of directions Q_{DCT} contains $d \times d$ frequencies, we keep only a fraction r of the lowest frequency directions in order to make the adversarial perturbation in the low frequency space.

General basis. In general, we believe that our attack can be used with any orthonormal basis, provided that the basis vectors can be sampled efficiently. This is especially challenging for high resolution datasets such as ImageNet since each orthonormal basis vector has dimensionality $d \times d$. Iterative sampling methods such as Gram-Schmidt process cannot be used due to linear memory cost in the number of sampled vectors. Thus, we choose to evaluate our attack using only the standard basis vectors and DCT basis vectors for their efficiency and natural suitability to images.

Learning rate ϵ. Given any set of search directions Q, some directions may decrease $p_h(y \mid \mathbf{x})$ more than others. Further, it is possible for the output probability $p_h(y \mid \mathbf{x} + \epsilon\mathbf{q})$ to be non-monotonic in ϵ. In Figure 1, we plot the relative decrease in probability as a function of ϵ for randomly sampled search directions in both pixel space and the DCT space. The probabilities correspond to prediction on an ImageNet validation sample by a ResNet-50 model. This figure highlights an illuminating result: the probability $p_h(y \mid \mathbf{x} \pm \epsilon\mathbf{q})$ *decreases monotonically* in ϵ with surprising consistency (across random images and vectors $\mathbf{q}$)! Although some directions eventually increase the true class probability, the expected change in this probability is negative with a relatively steep slope. This means that our algorithm is not overly sensitive to the choice of ϵ and the iterates will decrease the true class probability quickly. The figure also shows that search in the DCT space tends to lead to steeper descent directions than pixel space. As we show in the next section, we can tightly bound the final L_2-norm of the perturbation given a choice of ϵ and maximum number of

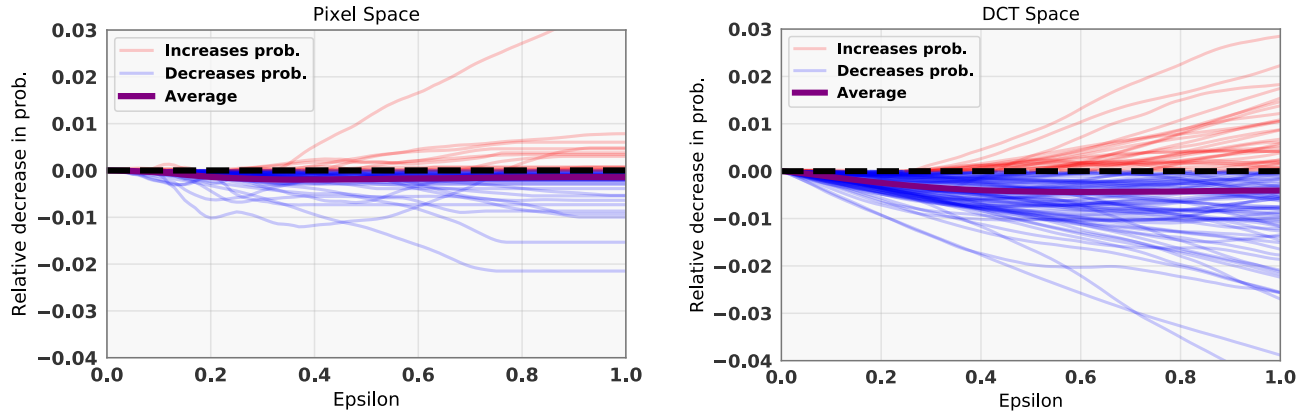


Figure 1. Plot of the change in predicted class probability when a randomly picked basis direction $\mathbf{q}$ is added or subtracted (whichever decreases the loss more) with step size ϵ. The left plot shows pixel space and the right plot shows low frequency DCT space. The average change (purple line) is almost linear in ϵ with the slope being steeper when the direction is sampled in DCT space. Further, 98% of the directions sampled in DCT space have either $-\mathbf{q}$ or $\mathbf{q}$ descending, while only 73% are descending in pixel space.

steps T, so the choice of ϵ depends *primarily on budget considerations* with respect to $\|\delta\|_2$.

Budget considerations. By exploiting the orthonormality of the basis Q we can bound the norm of δ tightly. Each iteration a basis vector is either added, subtracted, or discarded (if neither direction yields a reduction of the output probability.) Let $\alpha_i \in \{-\epsilon, 0, \epsilon\}$ denote the sign of the search direction chosen at step t, so

$$\delta_{t+1} = \delta_t + \alpha_t \mathbf{q}_t.$$

We can recursively expand $\delta_{t+1} = \delta_t + \alpha_t \mathbf{q}_t$. In general, the final perturbation δ_T after T steps can be written as a sum of these individual search directions:

$$\delta_T = \sum_{t=1}^{T} \alpha_t \mathbf{q}_t.$$

Since the directions $\mathbf{q}_t$ are orthogonal, $\mathbf{q}_t^\top \mathbf{q}_{t'} = 0$ for any $t \neq t'$. We can therefore compute the L_2-norm of the adversarial perturbation:

$$\|\delta_T\|_2^2 = \left\| \sum_{t=1}^{T} \alpha_t \mathbf{q}_t \right\|_2^2 = \sum_{t=1}^{T} \|\alpha_i \mathbf{q}_t\|_2^2 = \sum_{t=1}^{T} \alpha_t^2 \|\mathbf{q}_t\|_2^2 \leq T\epsilon^2.$$

Here, the second equality follows from the orthogonality of $\mathbf{q}_t$ and $\mathbf{q}_{t'}$, and the last inequality is tight if all queries result in a step of either ϵ or $-\epsilon$. Thus the adversarial perturbation has L_2-norm at most $\sqrt{T}\epsilon$ after T iterations. This result holds for any orthonormal basis (e.g. Q_{DCT}).

Our analysis highlights an important trade-off: for query-limited scenarios, we may reduce the number of iterations by setting ϵ higher, incurring higher perturbation L_2-norm. If a low norm solution is more desirable, reducing ϵ will allow quadratically more queries at the same L_2-norm. A more thorough theoretical analysis of this trade-off could improve query efficiency.

4. Experimental Evaluation

In this section, we evaluate our attack against a comprehensive list of competitive black-box attack algorithms: the Boundary Attack (Brendel et al., 2017), Opt attack (Cheng et al., 2018), Low Frequency Boundary Attack (LFBA) (Guo et al., 2018), AutoZOOM (Tu et al., 2018), the QL attack (Ilyas et al., 2018a), and the Bandits-TD attack (Ilyas et al., 2018b). There are three dimensions to evaluate black-box adversarial attacks on: how often the optimization problem finds a feasible point (*success rate*), how many queries were required (B), and the resulting perturbation norms (ρ).

4.1. Setup

We first evaluate our method on ImageNet. We sample a set of 1000 images from the ImageNet validation set that are initially classified correctly to avoid artificially inflating the success rate. Since the predicted probability is available for every class, we minimize the probability of the correct class as adversarial loss in untargeted attacks, and maximize the probability of the target class in targeted attacks. We sample a target class uniformly at random for all targeted attacks.

Next, we evaluate SimBA in the real-world setting of attacking the Google Cloud Vision API. Due to the extreme budget required by baselines that might cost up to $150 per image[2], here we only compare to LFBA, which we found to be the most query efficient baseline.

[2]The Google API charges $1.50 for 1000 image queries.

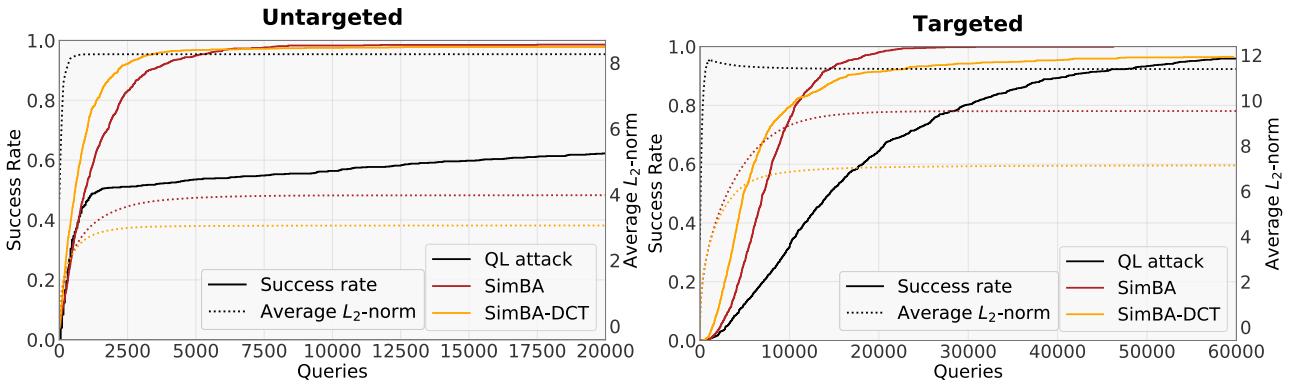


Figure 2. Comparison of success rate and average L_2-norm versus number of model queries for untargeted (left) and targeted (right) attacks. Horizontal axis shows number of model queries. The increase in success rate for SimBA and SimBA-DCT are dramatically faster than that of QL-attack in both untargeted and targeted scenarios. Both methods also achieve lower average L_2-norm than QL-attack. Note that although SimBA-DCT has faster initial convergence, its final success rate is lower than SimBA.

In our experiments, we limit SimBA and SimBA-DCT to at most $T = 10,000$ iterations for untargeted attacks and to $T = 30,000$ for targeted attacks. For SimBA-DCT, we keep the first $1/8$th of all frequencies, and add an additional $1/32$nd of the frequencies whenever we exhaust available frequencies without succeeding. For both methods, we use a fixed step size of $\epsilon = 0.2$.

4.2. ImageNet results

Success rate comparison (Figure 2). We demonstrate the query efficiency of our method in comparison to the QL attack – arguably the state-of-the-art black-box attack method to date – by plotting the average success rate against the number of queries. Figure 2 shows the comparison for both untargeted and targeted attacks. The dotted lines show progression of average L_2-norm throughout optimization. Both SimBA and SimBA-DCT achieve dramatically faster increase in success rate in both untargeted and targeted scenarios. The average L_2-norm for both methods are also significantly lower.

Query distributions (Figure 3). In Figure 3 we plot the histogram of model queries made by both SimBA and SimBA-DCT over 1000 random images. Notice that the distributions are highly right skewed so the median query count is a much more representative aggregate statistic than average query count. These median counts for SimBA and SimBA-DCT are only 944 and 582, respectively. In the targeted case, SimBA-DCT can construct an adversarial perturbation within only $4,854$ median queries but failed to do so after $60,000$ queries for approximately 2.5% of the images. In contrast, SimBA achieves a success rate of 100% with a median query count of $7,038$.

This result shows a fundamental trade-off when selecting the orthonormal basis Q. Restricting to only the low frequency DCT basis vectors for SimBA-DCT results in faster average rate of descent for most images, but may fail to admit an adversarial perturbation for some images. This phenomenon has been observed by Guo et al. (2018) for optimization-based white-box attacks as well. Finding the right spectrum to operate in on a per-image basis may be key to further improving the query efficiency and success rate of black-box attack algorithms. We leave this promising direction for future work.

Aggregate statistics (Table 1). Table 1 computes aggregate statistics of model queries, success rate, and perturbation L_2-norm across different attack algorithms. We reproduce the result for LFBA, QL-attack and Bandits-TD using default hyperparameters, and present numbers reported by the original authors' papers for Boundary Attack[3], Opt-attack, and AutoZOOM. The target model is a pretrained ResNet-50 (He et al., 2016) network, with the exception of AutoZOOM, which used an Inception v3 (Szegedy et al., 2016) network. Some of the attacks operate under the harder label-only setting (i.e., only the predicted label is observed), which may impact their query efficiency due to observation of partial information. Nevertheless, we include these methods in the table for completeness.

The three columns in the table show all the relevant metrics for evaluating a black-box attack. Ideally, an attack should succeed often, construct perturbation with low L_2 norm, and do so within very few queries. It is possible to artificially reduce the number of model queries by lowering success rate and/or increasing perturbation norm. To ensure fair comparison, we enforce our methods achieve close to 100% success rate and compare the other two metrics. Note that the success rate for boundary attack and LFBA are always 100% since both methods begin with very large perturba-

[3]Result reproduced by Cheng et al. (2018)

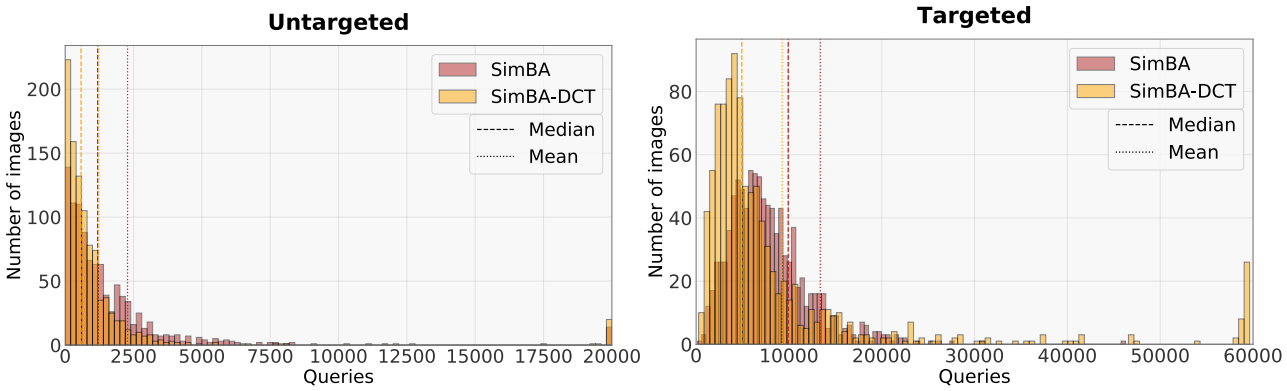


Figure 3. Histogram of number of queries required until a successful attack (over 1000 target images). SimBA-DCT is highly right skewed, suggesting that only a handful of images require more than a small number of queries. The *median* number of queries required by SimBA-DCT for untargeted attack is only 582. However, limiting to the low frequency basis results in SimBA-DCT failing to find a successful adversarial image after $60,000$ queries, whereas SimBA can achieve 100% success rate consistently.

Untargeted

Attack	Average queries	Average L_2	Success rate
Label-only			
Boundary attack	123,407	5.98	100%
Opt-attack	71,100	6.98	100%
LFBA	30,000	6.34	100%
Score-based			
QL-attack	28,174	8.27	85.4%
Bandits-TD	5,251	5.00	80.5%
SimBA	1,665	3.98	98.6%
SimBA-DCT	**1,283**	3.06	97.8%

Targeted

Attack	Average queries	Average L_2	Success rate
Score-based			
QL-attack	20,614	11.39	98.7%
AutoZOOM	13,525	26.74	100%
SimBA	**7,899**	9.53	100%
SimBA-DCT	8,824	7.04	96.5%

Table 1. Average query count for untargeted (left) and targeted (right) attacks on ImageNet. Methods are evaluated on three different metrics: average number of queries until success (lower is better), average perturbation L_2-norm (lower is better), and success rate (higher is better). Both SimBA and SimBA-DCT achieve close to 100% success rate, similar to other methods in comparison, but require significantly fewer model queries while achieving *lower* average L_2 distortion.

tions to guarantee misclassification and gradually reduce the perturbation norm.

Both SimBA and SimBA-DCT have *significantly lower* average L_2-norm than all baseline methods. For untargeted attack, our methods require 3-4x fewer queries (at $1,665$ and $1,232$, respectively) compared to the strongest baseline method – Bandits-TD – which only achieves a 80% success rate. For targeted attack (right table), the evaluated methods are much more comparable, but both SimBA and SimBA-DCT still require significantly fewer queries than QL-attack and AutoZOOM.

Evaluating different networks (Figure 4). To verify that our attack is robust against different model architectures, we evaluate SimBA and SimBA-DCT additionally against DenseNet-121 (Huang et al., 2017a) and Inception v3 (Szegedy et al., 2016) networks. Figure 4 shows success rate across the number of model queries for an untargeted attack against the three different network architectures. ResNet-50 and DenseNet-121 exhibit a similar degree of vulnerability against our attacks. However, Inception v3 is noticeably more difficult to attack, requiring more than $10,000$ queries to successfully attack with some images. Nevertheless, both methods can successfully construct adversarial perturbations against all models with high probability.

Qualitative results (Figure 5). For qualitative evaluation of our method, we present several randomly selected images before and after adversarial perturbation by untargeted attack. For comparison, we attack the same set of images using QL attack. Figure 5 shows the clean and perturbed images along with the perturbation L_2-norm and number of queries. While all attacks are highly successful at changing the label, the norms of adversarial perturbations constructed by SimBA and SimBA-DCT are much smaller than that of QL attack. Both methods requires consistently fewer queries than QL attack for almost all images. In fact, SimBA-DCT was able to find an adversarial image in as few as 36 model queries! Notice that the perturbation produced by SimBA contains sparse but sharp differences, constituting a low L_0-norm attack. SimBA-DCT produces perturbations that are sparse in frequency space, and the resulting change in

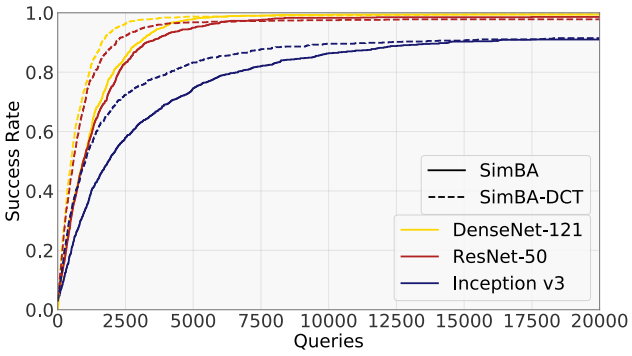


Figure 4. Comparison of success rate versus number of model queries across different network architectures for untargeted SimBA (solid line) and SimBA-DCT (dashed line) attacks. Both methods can successfully construct adversarial perturbations within $20,000$ queries with high probability. DenseNet is the most vulnerable against both attacks, admitting a success rate of almost 100% after only 6,000 queries for SimBA and 4000 queries for SimBA-DCT. Inception v3 is much more difficult to attack for both methods.

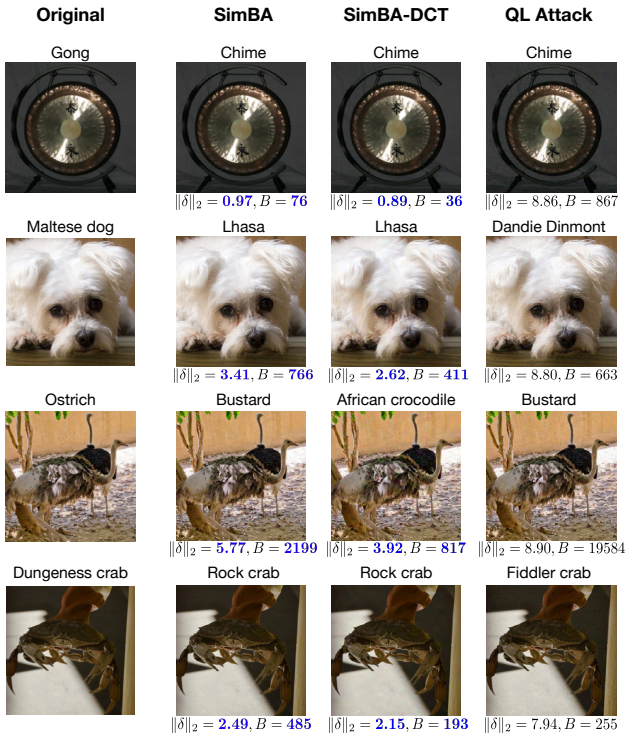


Figure 5. Randomly selected images before and after adversarial perturbation by SimBA, SimBA-DCT and QL attack. The constructed perturbation is imperceptible for all three methods, but the perturbation L_2-norms for SimBA and SimBA-DCT are significantly lower than that of QL attack across all images. Our methods are capable of constructing an adversarial example in comparable or fewer queries than QL attack – as few as 36 queries in some cases! Zoom in for detail.

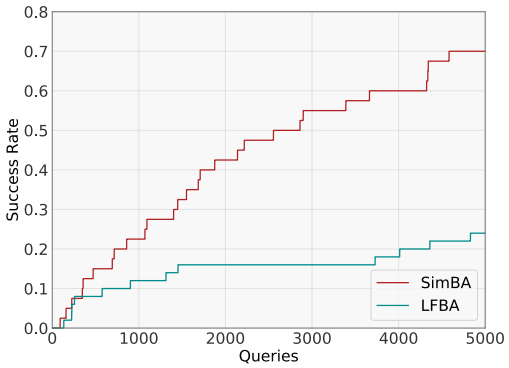


Figure 6. Plot of success rate across number of model queries for Google Cloud Vision attack. SimBA is able to achieve close to 70% success rate after only 5000 queries, while the success rate for LFBA has only reached 25%.

pixel space is spread out across all pixels.

4.3. Google Cloud Vision attack

To demonstrate the efficacy of our attack against real world systems, we attack the Google Cloud Vision API, an online machine learning service that provides labels for arbitrary input images. For a given image, the API returns a list of top concepts contained in the image and their associated probabilities. Since the full list of probabilities associated with every label is unavailable, we define an untargeted attack that aims to remove the top 3 concepts in the original. We use the maximum of the original top 3 concepts' returned probabilities as the adversarial loss and use SimBA to minimize this loss. Figure 7 shows a sample image before and after the attack. The original image (left) contains concepts related to camera instruments. SimBA successfully replaced the top concepts with weapon-related objects with imperceptible change to the original image. Additional samples are included in the supplementary material.

Since our attack can be executed efficiently, we evaluate its effectiveness over an aggregate of 50 random images. For the LFBA baseline, we define an attack as successful if the produced perturbation has an L_2-norm of at most the highest L_2-norm in a successful run of our attack. Figure 6 shows the average success rate of both attacks across number of queries. SimBA achieves a final success rate of 70% after only 5000 API calls, while LFBA is able to succeed only 25% of the time under the same query budge. To the best of our knowledge, this is the first adversarial attack result on Google Cloud Vision that has a high reported success rate within very limited number of queries.

5. Related Work

Many recent studies have shown that both white-box and black-box attacks can be applied to a diverse set of tasks.

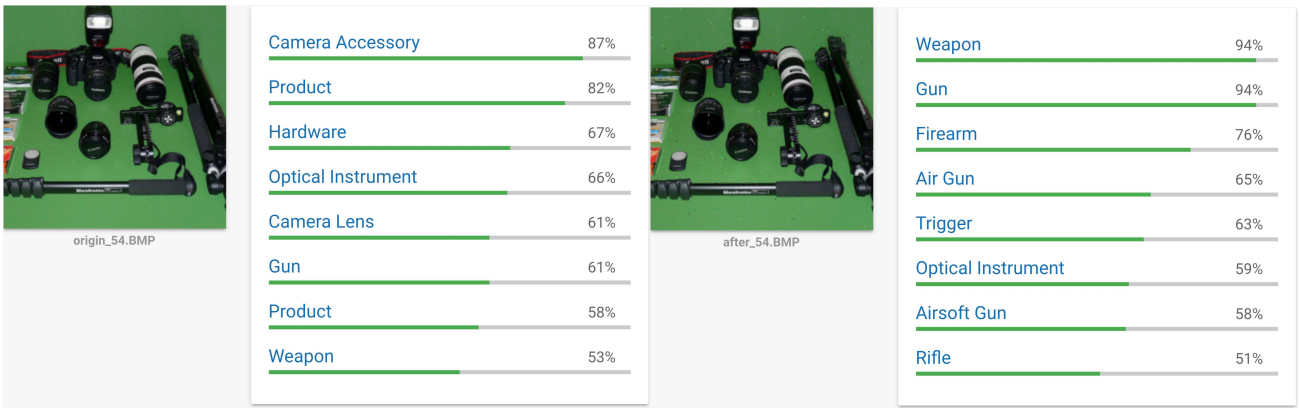


Figure 7. Screenshot of Google Cloud Vision labeling results on a sample image before and after adversarial perturbation. The original image contains a set of camera instruments. The adversarial image successfully replaced the top concepts with guns and weapons. See supplementary material for additional samples.

Computer vision models for image segmentation and object detection have also been shown to be vulnerable against adversarial perturbations (Cisse et al., 2017a; Xie et al., 2017). Carlini & Wagner (2018) performed a systematic study of speech recognition attacks and showed that robust adversarial examples that alter the transcription model to output arbitrary target phrases can be constructed. Attacks on neural network policies (Huang et al., 2017b; Behzadan & Munir, 2017) have also been shown to be permissible.

As these attacks become prevalent, many recent works have focused on designing defenses against adversarial examples. One common class of defenses applies an image transformation prior to classification, which aims to remove the adversarial perturbation without changing the image content (Xu et al., 2017; Dziugaite et al., 2016; Guo et al., 2017). Instead of requiring the model to correctly classify all adversarial images, another strategy is to detect the attack and output an adversarial class when certain statistics of the input appear abnormal (Li & Li, 2017; Metzen et al., 2017; Meng & Chen, 2017; Lu et al., 2017). The training procedure can also be strengthened by including the adversarial loss as an implicit or explicit regularizer to promote robustness against adversarial perturbations (Tramèr et al., 2017; Madry et al., 2017; Cisse et al., 2017b). While these defenses have shown great success against a passive adversary, almost all of them can be easily defeated by modifying the attack strategy (Carlini & Wagner, 2017a; Athalye & Carlini, 2018; Athalye et al., 2018).

Relative to defenses against white-box attacks, few studies have focused on defending against adversaries that may only access the model via black-box queries. While transfer attacks can be effectively mitigated by methods such as ensemble adversarial training (Tramèr et al., 2017) and image transformation (Guo et al., 2017), it is unknown whether existing defense strategies can be applied to adaptive adversaries that may access the model via queries. Guo et al. (2018) have shown that the boundary attack is susceptible to image transformations that quantize the decision boundary, but employing the attack in low frequency space can successfully circumvent these transformation defenses.

6. Discussion and Conclusion

We proposed SimBA, a simple black-box adversarial attack that takes small steps iteratively guided by continuous-valued model output. The unprecedented query efficiency of our method establishes a strong baseline for future research on black-box adversarial examples. Given its real world applicability, we hope that more effort can be dedicated towards defending against malicious adversaries under this more realistic threat model.

While we intentionally avoid more sophisticated techniques to improve SimBA in favor of simplicity, we believe that additional modifications can still dramatically decrease the number of model queries. One possible extension could be to further investigate the selection of different sets of orthonormal bases, which could be crucial to the efficiency of our method by increasing the probability of finding a direction of large change. Another area for improvement is the adaptive selection of the step size ϵ to optimally consume the distance and query budgets.

Given that our method has very few requirements, it is conceptually suitable for applications to any task for which the target model returns a continuous score for the prediction. For instance, speech recognition systems are trained to maximize the probability of the correct transcription (Amodei et al., 2016), and policy networks (Mnih et al., 2015) are trained to maximize some reward function over the set of actions conditioned on the current environment. A simple iterative algorithm that modifies the input at random may prove to be effective in these scenarios. We leave these directions for future work.

References

Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Chen, J., Chrzanowski, M., Coates, A., Diamos, G., Elsen, E., Engel, J., Fan, L., Fougner, C., Hannun, A. Y., Jun, B., Han, T., LeGresley, P., Li, X., Lin, L., Narang, S., Ng, A. Y., Ozair, S., Prenger, R., Qian, S., Raiman, J., Satheesh, S., Seetapun, D., Sengupta, S., Wang, C., Wang, Y., Wang, Z., Xiao, B., Xie, Y., Yogatama, D., Zhan, J., and Zhu, Z. Deep speech 2 : End-to-end speech recognition in english and mandarin. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pp. 173–182, 2016.

Athalye, A. and Carlini, N. On the robustness of the CVPR 2018 white-box adversarial example defenses. *CoRR*, abs/1804.03286, 2018.

Athalye, A., Carlini, N., and Wagner, D. A. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. *CoRR*, abs/1802.00420, 2018.

Behzadan, V. and Munir, A. Vulnerability of deep reinforcement learning to policy induction attacks. *CoRR*, abs/1701.04143, 2017.

Brendel, W., Rauber, J., and Bethge, M. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *CoRR*, abs/1712.04248, 2017.

Carlini, N. and Wagner, D. A. Adversarial examples are not easily detected: Bypassing ten detection methods. *CoRR*, abs/1705.07263, 2017a.

Carlini, N. and Wagner, D. A. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, pp. 39–57, 2017b.

Carlini, N. and Wagner, D. A. Audio adversarial examples: Targeted attacks on speech-to-text. *CoRR*, abs/1801.01944, 2018.

Chen, P., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C. ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec@CCS 2017, Dallas, TX, USA, November 3, 2017*, pp. 15–26, 2017. doi: 10.1145/3128572.3140448.

Cheng, M., Le, T., Chen, P., Yi, J., Zhang, H., and Hsieh, C. Query-efficient hard-label black-box attack: An optimization-based approach. *CoRR*, abs/1807.04457, 2018.

Cisse, M., Adi, Y., Neverova, N., and Keshet, J. Houdini: Fooling deep structured prediction models. *CoRR*, abs/1707.05373, 2017a.

Cisse, M., Bojanowski, P., Grave, E., Dauphin, Y., and Usunier, N. Parseval networks: Improving robustness to adversarial examples. *CoRR*, abs/1704.08847, 2017b.

Dziugaite, G. K., Ghahramani, Z., and Roy, D. A study of the effect of JPG compression on adversarial images. *CoRR*, abs/1608.00853, 2016.

Fawzi, A., Fawzi, H., and Fawzi, O. Adversarial vulnerability for any classifier. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pp. 1186–1195, 2018.

Goodfellow, I., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *Proc. ICLR*, 2015.

Guo, C., Rana, M., Cissé, M., and van der Maaten, L. Countering adversarial images using input transformations. *CoRR*, abs/1711.00117, 2017.

Guo, C., Frank, J. S., and Weinberger, K. Q. Low frequency adversarial perturbations. *CoRR*, abs/1809.08758, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proc. CVPR*, pp. 770–778, 2016.

Huang, G., Liu, Z., Weinberger, K., and van der Maaten, L. Densely connected convolutional networks. In *Proc. CVPR*, pp. 2261–2269, 2017a.

Huang, S., Papernot, N., Goodfellow, I., Duan, Y., and Abbeel, P. Adversarial attacks on neural network policies. *CoRR*, abs/1702.02284, 2017b.

Ilyas, A., Engstrom, L., Athalye, A., and Lin, J. Black-box adversarial attacks with limited queries and information. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, pp. 2142–2151, 2018a.

Ilyas, A., Engstrom, L., and Madry, A. Prior convictions: Black-box adversarial attacks with bandits and priors. *CoRR*, abs/1807.07978, 2018b.

Kurakin, A., Goodfellow, I., and Bengio, S. Adversarial machine learning at scale. *CoRR*, abs/1611.01236, 2016.

Li, X. and Li, F. Adversarial examples detection in deep networks with convolutional filter statistics. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 5775–5783, 2017. doi: 10.1109/ICCV.2017.615.

Lu, J., Issaranon, T., and Forsyth, D. A. Safetynet: Detecting and rejecting adversarial examples robustly. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 446–454, 2017. doi: 10.1109/ICCV.2017.56.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. *CoRR*, abs/1706.06083, 2017.

Meng, D. and Chen, H. Magnet: A two-pronged defense against adversarial examples. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, pp. 135–147, 2017. doi: 10.1145/3133956.3134057.

Metzen, J. H., Genewein, T., Fischer, V., and Bischoff, B. On detecting adversarial perturbations. *CoRR*, abs/1702.04267, 2017.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M. A., Fidjeland, A., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533, 2015. doi: 10.1038/nature14236.

Moosavi-Dezfooli, S., Fawzi, A., and Frossard, P. Deepfool: A simple and accurate method to fool deep neural networks. In *Proc. CVPR*, pp. 2574–2582, 2016.

Moosavi-Dezfooli, S.-M., Fawzi, A., Fawzi, O., and Frossard, P. Universal adversarial perturbations. In *Proc. CVPR*, pp. 86–94, 2017.

Shafahi, A., Huang, W. R., Studer, C., Feizi, S., and Goldstein, T. Are adversarial examples inevitable? *CoRR*, abs/1809.02104, 2018.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. In *In Proc. ICLR*, 2014.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proc. CVPR*, pp. 2818–2826, 2016.

Tramèr, F., Kurakin, A., Papernot, N., Boneh, D., and McDaniel, P. D. Ensemble adversarial training: Attacks and defenses. *CoRR*, abs/1705.07204, 2017.

Tu, C., Ting, P., Chen, P., Liu, S., Zhang, H., Yi, J., Hsieh, C., and Cheng, S. Autozoom: Autoencoder-based zeroth order optimization method for attacking black-box neural networks. *CoRR*, abs/1805.11770, 2018.

Xie, C., Wang, J., Zhang, Z., Zhou, Y., Xie, L., and Yuille, A. L. Adversarial examples for semantic segmentation and object detection. In *ICCV*, pp. 1378–1387. IEEE Computer Society, 2017.

Xu, W., Evans, D., and Qi, Y. Feature squeezing: Detecting adversarial examples in deep neural networks. *CoRR*, abs/1704.01155, 2017.

Supplementary Material for Simple Black-box Adversarial Attacks

S1. Experiment on CIFAR-10

In this section, we evaluate SimBA and SimBA-DCT on a ResNet-50 model trained on CIFAR-10. Both attacks remain very efficient on this new dataset without any hyperparameter tuning.

Figure S1 shows the distribution of queries required for a successful targeted attack to a random target label. In contrast to the experiment on ImageNet, the use of low frequency DCT basis is less effective due to the reduced image dimensionality. Both SimBA and SimBA-DCT perform similarly, with SimBA-DCT having a slightly heavier tail.

Table S1 shows aggregate statistics for the attack on CIFAR-10. Both methods achieve a success rate of 100% when limited to a maximum of $10,000$ queries. The actual required queries is much fewer, with both methods averaging to approximately 300 queries, matching the median. SimBA-DCT has a slightly worse performance compared to SimBA due its query distribution having a slightly heavier tail. Nevertheless, the average query count is in line with state-of-the-art attacks on CIFAR-10. For instance, AutoZOOM achieves a mean query count of 259 with an average L_2-norm of 3.53.

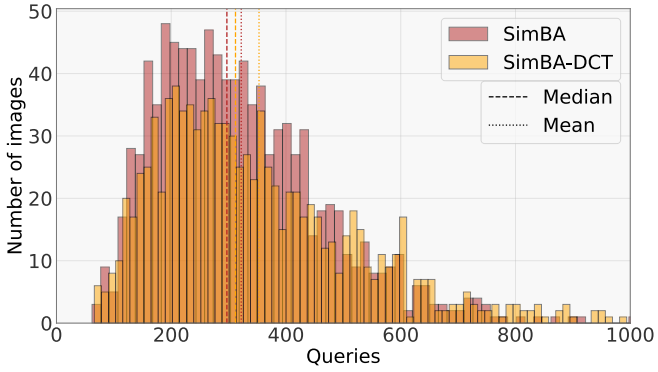


Figure S1. Histogram of number of queries required until a successful targeted attack on CIFAR-10 (over 1000 target images).

Attack	Average queries	Median queries	Average L_2	Success rate
SimBA	322	297	2.04	100%
SimBA-DCT	353	312	2.21	100%

Table S1. Average query count for SimBA and SimBA-DCT on CIFAR-10.

S2. Additional image samples for attack on Google Cloud Vision

To demonstrate the generality of our evaluation of the Google Cloud Vision attack, we show 10 additional random images before and after perturbation by SimBA. In all cases, we successfully remove the top 3 original labels.

origin_60.BMP

Dog Breed	93%
Dog Like Mammal	91%
Dog	90%
Lakeland Terrier	85%
Wire Hair Fox Terrier	84%
Terrier	83%
Snout	62%
Carnivoran	62%
Companion Dog	60%

after_60.BMP

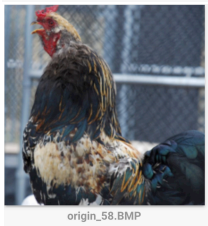
Grass	65%
Snout	63%
Terrier	60%

origin_14.BMP

Bird	96%
Fauna	89%
Branch	84%
Tree	82%
Beak	80%
Perching Bird	56%
Wildlife	55%
Twig	52%

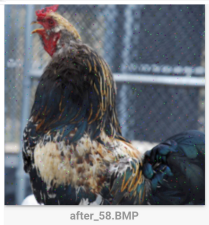
after_14.BMP

Tree	73%
Water	52%



origin_58.BMP

Chicken	98%
Beak	92%
Galliformes	91%
Rooster	89%
Feather	81%
Fowl	79%
Poultry	74%
Bird	70%
Phasianidae	58%



after_58.BMP

Snow	77%
Winter	69%

origin_59.BMP

Lighthouse	98%
Tower	96%
Landmark	89%
Beacon	81%
Sky	78%
Promontory	77%
Coast	59%
Sea	58%
Inlet	56%

after_59.BMP

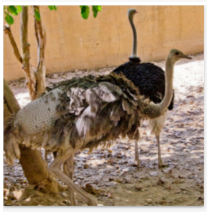
Sky	78%
Sea	73%
Computer Wallpaper	56%

origin_25.BMP

Marine Mammal	95%
Fauna	94%
Mammal	93%
Wildlife	82%
Terrestrial Animal	80%
Organism	77%
Snout	74%
Mouth	71%
Whales Dolphins And Porpoises	69%

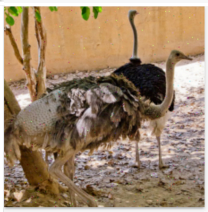
after_25.BMP

Fish	94%
Ecosystem	91%
Fish	86%
Shark	75%
Mouth	69%
Cartilaginous Fish	67%
Organism	67%
Marine Biology	63%
Jaw	53%



origin_2.BMP

Ostrich	99%
Ratite	96%
Fauna	93%
Flightless Bird	87%
Beak	81%
Wildlife	76%
Bird	75%
Emu	72%
Terrestrial Animal	62%



after_2.BMP

Wildlife	51%

origin_6.BMP

Cycle Sport	96%
Vehicle	91%
Mode Of Transport	88%
Unicycle	85%
Bicycle	77%
Flatland Bmx	76%
Sports Equipment	75%
Freestyle Bmx	75%
Bmx Bike	73%

after_6.BMP

Vehicle	81%
Cycle Sport	80%
Sports Equipment	72%
Unicycle	71%
Flatland Bmx	68%
Bicycle	68%
Bmx Bike	64%
Freestyle Bmx	64%
Recreation	58%

origin_19.BMP

Sky	87%
Energy	84%
Solar Power	78%
Solar Energy	72%
Technology	66%
Daylighting	63%
Roof	62%
Solar Panel	60%
Facade	56%

after_19.BMP

Roof	54%
Daylighting	53%

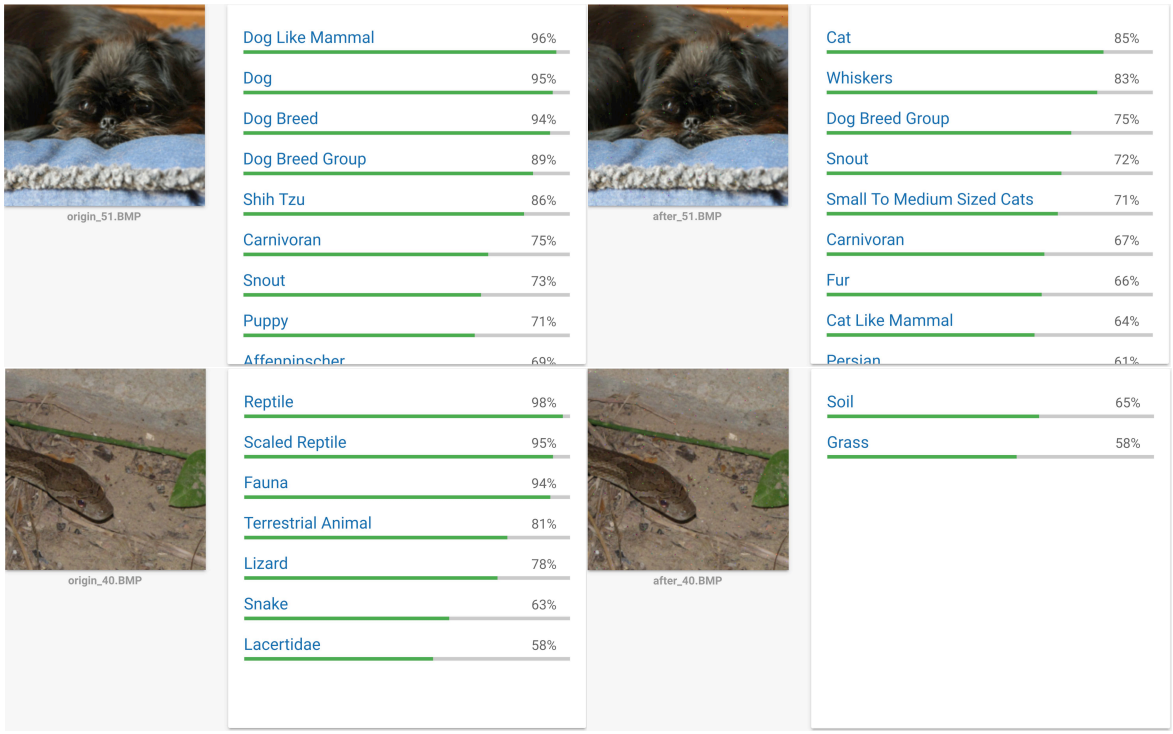


Figure S2. Additional adversarial images on Google Cloud Vision.