Sharpness-Aware Optimization for Real-World Adversarial Attacks for Diverse Compute Platforms with Enhanced Transferability

Muchao Ye [*], Xiang Xu, Qin Zhang, Jonathan Wu
Amazon AI Labs
410 Terry Ave N, Seattle, WA, USA, 98109
muchao@psu.edu, {xiangx, qzaamz, jonwu}@amazon.com

Abstract

In recent years, deep neural networks (DNNs) have become integral to many real-world applications. A pressing concern in these deployments pertains to their vulnerability to adversarial attacks. In this work, we focus on the transferability of adversarial examples in a real-world deployment setting involving both a cloud model and an edge model. The cloud model is a black-box victim model, while the edge model is a surrogate model that is fully accessible to users. We investigated scenarios where attackers leverage information from the known surrogate model to generate adversarial examples to attack the unknown black-box victim model. Existing methods often optimize the adversarial example generation based on the steepest gradients estimated from the surrogate model, which do not generalize effectively to the victim model. To better gauge the for real-world adversarial risks in a cloud-edge deployment setting, we proposed an novel attack mechanism that enhanced transferability by incorporating a sharpness-aware objective into the optimization process. Our evaluation on image classification benchmarks demonstrates that our method significantly improves adversarial example's transferability, even on the foundational computer vision models such as OFA-Large, showcasing its potential as a new standard in assessing attack transferability within a cloud-edge hybrid deployment scenario.

1. Introduction

The widespread adoption of deep neural networks (DNNs) has raised security concerns about their vulnerability to adversarial attacks [1, 13], especially given the observation that attackers can add a small amount of noise to the original input images and fools the model to overturn their originally correct predictions into incorrect ones [8]. In particular, we are concerned about the transferability of the adversarial examples in a cloud-edge deployment scenario. Here, attackers can utilize a white-box edge (surrogate) model to generate adversarial examples and further attack a black-box cloud (victim) model. This problem is critical because if the attack pattern, generated using knowledge of a known model, is transferrable to a target victim model, it enables attackers to compromise the target victim model even with limited access. In high-stakes applications, this transferability can result in significant financial losses [12] and trust erosion among users.

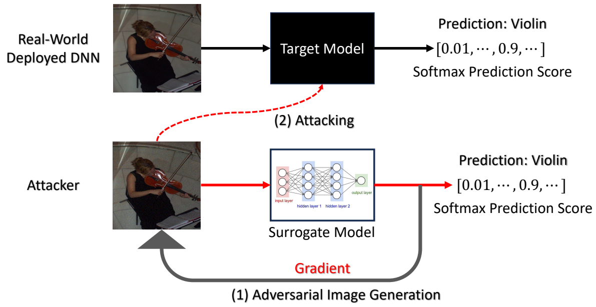


Figure 1. Illustration of the *real-world adversarial attack* setting. Since the target model of the service provider is generally a black box that only outputs prediction score and limits malicious access, the attacker will (1) generate adversarial image through a known surrogate model and then (2) put the generated adversarial example to target model for attacking.

Early research on adversarial example generation mainly operated within the white-box setting, where attackers had full access to the victim model's parameters, enabling a deterministic reverse-engineering process to deceive the model effectively [3, 20]. Subsequently, the black-box scenario was introduced, where attackers only possessed access to prediction logits or top-1 predictions [2]. However, a practical adversarial attack setting tailored for real-world applications, encompassing diverse compute platforms, including the cloud-edge deployment scenario we are investigating, has yet to be established. In this scenario, the edge

[*]This work is done during the internship at AWS.

model is typically a white-box, deployed on the user end and fully accessible, while the cloud model remains a black-box, offering only final prediction logits or top-1 scores to end users, as illustrated in Fig. 1. We refer to this setting as *real-world adversarial attack*. Our objective is to investigate how knowledge acquired from the white-box edge model can be transferred to fool the black-box cloud model, which holds significant security implications. In this context, we consider two practical scenarios:

- *Scenario 1:* Attackers cannot query target victim models before inputting adversarial examples.
- *Scenario 2:* Attackers can query victim models for a limited number of times (finite horizon).

These two scenarios lead us to finding solutions to the following research questions:

- *Q1:* What is the key to generating adversarial examples with high transferability rates and how can we utilize it to design an adversarial example generation method with high transferability rates?
- *Q2:* How can we employ the black-box output from the victim target model to generate adversarial examples with higher transferability when a limited number of queries to the victom models are allowed?

Our approach to addressing research question Q1 is rooted in the observation that many effective adversarial example generation methods rely on gradient information. In light of this, we explore sharpness-aware optimization [11, 29] and introduce a sharpness regularization term into the attack formulation of PGD [20] to better estimate the loss landscape around the adversarial example, thus enhancing attack transferability. Specifically, we introduce the notion of the ***average-descent direction*** for solving this optimization problem, as opposed to the gradient-descent direction used in prior work[20]. The average-descent direction allows us to assess sharpness around the adversarial example in all directions, rather than just along the gradient-descent direction. We estimate the average-descent direction using Monte Carlo estimation, with randomly sampled Gaussian noise added to the input image. For research question Q2, where attackers are permitted a limited number of queries to the victim target model, we incorporate the output logits from the black-box model into our sharpness-aware optimization solution. This is achieved by modifying the direction of the randomly sampled perturbation based on changes in the output logits. These innovations enable to make the following contributions:

- We incorporate a sharpness-aware regularization into the existing adversarial example generation optimization framework. Moreover, we use the average-descent direction to handle this sharpness regularization term. This approach allows us to consider the flatness of the loss landscape around the input in all directions. Through extensive experiments, comparing our propsed method with various adversarial attack baselines, we demonstrate that our approach significantly enhances the transferability rate on the adversarial samples from 7.29% to 32.41% depending on the surrogate and target models.
- We further propose a mechanism to integrate the black-box output logit information in the optimization process by flipping the direction of the sampled Gaussian noise through the estimated gradient of the target black-box model. This approach can achiever better transferability rate on the adversarial samples up to 21.99% in most of cases.

2. Related Work

2.1. Adversarial Attack and Robustness

It has been observed that DNNs are prone to change their correct prediction results into incorrect ones by being attacked by adversarial examples [14]. Existing adversarial attack can generally be classified as white-box [14, 21] and black-box [4, 15, 17, 25] ones, where the white-box ones assume the full knowledge of the victim models while the black-box ones only have the knowledge of prediction score distribution. Respective white-box methods include FGSM [14] and PGD [21], which generate adversarial examples based on the gradient with respective to the input images. For the black-box ones, since they only have partial knowledge, the attack transferability among different architecture is not always guarantee. In our work, in addition to the respective white-box methods, we choose the black-box method named Square Attack [2] as another type of attack to verify our hypothesis, which is able to generate adversarial examples that be transferable among different DNN structures.

Although existing works have conducted discussion on the adversarial attack transferability [26], they usually do not take the real-world application scenario into consideration. Although there is a recent work [22] exploring attacking foundation model with prompts in the black-box setting, it requires queries to black-box systems to train the prompts, which is unrealistic because real-world systems usually limit the overdue access from users. For our work, we are particularly interested in how the adversarial attack transferability observed by existing research affects DNNs in the real-world application scenario. Such an unexplored discussion can bridge the gap between existing theoretical research and real-world application for DNNs.

2.2. Sharpness-Aware Optimization

Sharpness is the geometry measure of the loss landscape and generalization [11, 29] in parameter optimization. In mathematical terminology, the sharpness of the loss landscape is the largest eigenvalue of the Hessian matrix of the loss function with respect to the model parameters. In model parameter optimization, a model will have good generalization ability if the sharpness value is small. In terms of geometric interpretation, a low degree of sharpness indicates a high degree of flatness in the loss landscape around the optimized parameters. Such a good property in the loss landscape can further help improve the generalization ability of the trained model parameters among different tasks.

In our work, we focus on the adversarial example transferability instead of model generalization ability. However, based on the observation on how sharpness-aware optimization helps increase the generalization ability of the model, we can include the sharpness-aware optimization framework in the formulation of adversarial example generation too to improve the transferability of adversarial examples. This idea can be naturally applied in adversarial example optimization because a good ability of transferability means it can consistently fool different victim models, which can be considered as generalizaiton ability of the adversarial example. Although existing works [30] have applied the sharpness-aware regularization on the formulation of adversarial example generation, their solutions adopt the gradient-descent direction for solving the optimization problem. Our solution differs from them by adopting the average-descent direction instead to solve the optimization problem. Such a treatment is able to take a more comprehensive consideration on the landscape around the adversarial example, and it is a better estimation on the sharpness around the adversarial example.

3. Methodology

3.1. Preliminary

Adversarial Attack: Suppose we are given a C-class image classification dataset $D = (X, Y)$ over a compact image space $X \subset R^{3\times m\times n}$ and a discrete label space $Y = \{y_1, \cdots, y_C\}$, where each image $x \in X$ has 3 channels and $m \times n$ pixels in each channel. Now we are given an image $x \in R^{3\times m\times n}$ whose label is y and a image classification model f that successfully classifies x into the category of y, i.e., $f(x) = y$. In the task of adversarial attack [13] we would like to generate an adversarial image $x' = x + \delta$ by adding a small amount of noise δ into x such that

$$f(x') \neq y, \;\; \text{s.t.}, ||\delta||_p < \epsilon, \tag{1}$$

where $f(x') \neq y$ means that f cannot classify x' correctly after δ is added to x, and the constraint $||\delta||_p < \epsilon$ means that the L_p norm of the added noise δ is confined within the range of $\epsilon > 0$, *i.e.,* the perturbation is imperceptible.

Transferable Attack. In this paper, we focus on the real-world adversarial attack setting where attacks targeting a victim modelm, denoted as f, is conducted through a surrogate model, denoted as g. In this context, attackers initially employ the white-box model g to generate an adversarial example x' from x. Subsequently, they input x' into f to assess whether it can deceive f. We define an adversarial example as transferable [28] from g to f if two conditions are met. Firstly, we require that:

$$f(x) = g(x) = y, \tag{2}$$

which means both f and g successfully classify x into the correct class y. Secondly, we demand that:

$$f(x') \neq y \text{ and } g(x') \neq y, \text{subject to } ||\delta||_p < \epsilon, \tag{3}$$

indicating that x' successfully deceives both f and g. In this context, x' is generated from g using an adversarial attack algorithm, while maintaining the constraint $||x' - x||_p < \epsilon$.

PGD Attack: As attackers have knowledge of the surrogate model g, they typically employ white-box adversarial attack methods to generate an adversarial example, denoted as x', given an input x. The generation of an adversarial example can be framed as an optimization problem with the following formulation:

$$x' = \arg\max_{\tilde{x}} L(g(\tilde{x}), y), \text{s.t.}, ||\delta||_p = ||\tilde{x} - x|| < \epsilon, \tag{4}$$

where L is the loss function. This formulation indicates that the optimal adversarial example x' is the one among all possible $\{\tilde{x}\}$ that can fool the surrogate model to the largest degree as measured by the loss function L under the imperceptibility constraint that $||\delta||_p = ||\tilde{x} - x|| < \epsilon$.

Existing solutions to Eq. (4) include PGD [20] and FGSM [13]. We primarily consider PGD here as it is the most representative method for solving Eq. (4), which involves an iterative process. Suppose it has T iteration and the initial solution $x'_0 = x$. In the t-th iteration $(1 \leq t \leq T)$, given the output of previous iteration x'_{t-1}, we have

$$x'_t = \Pi_{B_\epsilon(x'_{t-1})}(x'_{t-1} + \alpha \nabla_{x'_{t-1}} L(g(x'_{t-1}), y), \tag{5}$$

where $\nabla_{x'_{t-1}} L(g(x'_{t-1}), y)$ is the gradient with respect to the previous solution x'_{t-1} and α is the step size for updating x'_{t-1} for the current iteration. $\Pi_{B_\epsilon(x'_{t-1})}$ means projecting the updated solution $x'_{t-1} + \alpha \nabla_{x'_{t-1}} L(g(x'_{t-1}), y)$ in the L_p norm ball centered at x'_{t-1} whose radius is ϵ. After T iterations, the output x'_T will be the final solution to the optimization problem Eq. (4).

3.2. Sharpness-Aware Optimization

Previous research has drawn parallels between enhancing the transferability of adversarial examples and improving a model's generalization capabilities. The latter typically involves favoring solutions within regions characterized by flat loss landscapes, which can be quantified using the concept of loss landscape sharpness [29]. From an optimization perspective, the notion of a model's generalization aligns

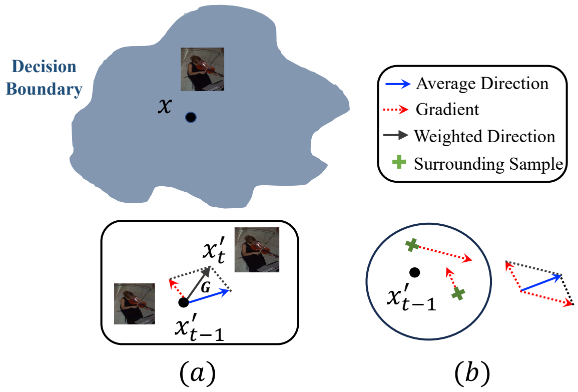


Figure 2. Illustration of sharpness-aware optimization. (a) The gradient direction used for constructing adversarial examples are the weighted sum of the gradient calculated on x'_{t-1} and the average-direction gradient estimated from the samples which are obtained from adding Gaussian noise to perturbed sample x'_{t-1}. (b) For the average-direction gradient, it is an element-wise average of the gradient calculated from the surrounding sample $x'_{t-1} + r\frac{\eta_i}{||\eta_i||}$ which is perturbed sample of x'_{t-1} by adding normalized Gaussian noise $\frac{\eta_i}{||\eta_i||}$.

with the concept of adversarial transferability in adversarial images. Given this observation, we hypothesize that highly transferable adversarial examples should be located in flat regions within the optimization function $L(g(\tilde{x}), y)$ in Eq.(4). To introduce sharpness-awareness into the optimization problem and enhance the transferability of adversarial examples, we incorporate a regularization term and modify the optimization problem as follows:

$$x' = \arg\max_{\tilde{x}}(L(g(\tilde{x}), y) - \lambda||\nabla_{\tilde{x}}L(g(\tilde{x}), y)||_2), \\ \text{s.t.}, ||\delta||_p = ||\tilde{x} - x||_p < \epsilon, \quad (6)$$

where $||\nabla_{\tilde{x}}L(f_s(\tilde{x}), y)||_2$ is the regularization term measured by the L_2 norm, λ is the weight of the regularization term, and we take the negative of the regularization term to fit in the maximization optimization framework.

The difference between the sharpness-aware optimization in Eq. (6) and the vanilla optimization in Eq. (4) lies in the regularization term $||\nabla_{\tilde{x}}L(f_s(\tilde{x}), y)||_2$, which indicates the sharpness of the area around $\tilde{x}$ in the optimization manifold. Upon convergence where $\tilde{x} \rightarrow x'$, if $||\nabla_{x'}L(g(x'), y)||_2$ is small, it means the local area around x' is flat and has a higher degree of transferability.

Denote $L_{\text{reg}} = L(g(\tilde{x}), y) - \lambda||\nabla_{x'}L(g(x'), y)||_2$ and $L_{\tilde{x}} = L(g(\tilde{x}), y)$. Solving Eq. (6) directly would introduce the calculation of the Hessian matrix of L_{reg} with respect to $\tilde{x}$ as follows:

$$\nabla_{\tilde{x}}L_{\text{reg}} = \nabla_{\tilde{x}}L_{\tilde{x}} - \lambda\nabla^2_{\tilde{x}}L_{\tilde{x}} \cdot \frac{\nabla_{\tilde{x}}L_{\tilde{x}}}{||\nabla_{\tilde{x}}L_{\tilde{x}}||} \quad (7)$$

which can be difficult to compute in practice. To relieve it, existing methods [30] usually use Taylor expansion to approximate the Hessian and ignore the high-order terms. Specifically, by Taylor expansion, introducing a small perturbation with a step size of r gives rise to:

$$\nabla_{\tilde{x}}L(g(\tilde{x} + r\frac{\nabla_{\tilde{x}}L_{\tilde{x}}}{||\nabla_{\tilde{x}}L_{\tilde{x}}||}), y) \\ = \nabla_{\tilde{x}}L_{\tilde{x}} + r\frac{\nabla_{\tilde{x}}L_{\tilde{x}}}{||\nabla_{\tilde{x}}L_{\tilde{x}}||}\nabla^2_{\tilde{x}}L_{\tilde{x}} + O(\nabla^2_{\tilde{x}}L_{\tilde{x}}). \quad (8)$$

Thus, we can have

$$\nabla_{\tilde{x}}L_{\text{reg}} \approx (1+\frac{\lambda}{r})\nabla_{\tilde{x}}L_{\tilde{x}} - \frac{\lambda}{r}\nabla_{\tilde{x}}L(g(\tilde{x}+r\frac{\nabla_{\tilde{x}}L_{\tilde{x}}}{||\nabla_{\tilde{x}}L_{\tilde{x}}||}), y) \quad (9)$$

Eq. (9) uses the gradient direction to measure the sharpness around $\tilde{x}$. However, we argue that it only measures the flatness in one direction, i.e., the steepest descent direction which has the highest slope, which may not lead to the global optimum. To alleviate this limitation, we propose to use the average direction for measuring the sharpness. Formally, under L-Lipschitz assumption, the average ascent gradient can be expressed as follows,

$$\nabla_{\tilde{x}}L_{\text{avg}} = \nabla_{\tilde{x}}\big(E_{\eta\sim N(0,I)}L(g(\tilde{x} + r\frac{\eta}{||\eta||}), y)\big) \quad (10)$$

where $N(0, I)$ stands for the isotropic Gaussian distribution. In implementation, we adopt Monte Carlo estimation for calculating Eq. (10). Suppose we sample k times from Gaussian distribution $N(0, I)$, which we denote the i-th one as η_i. Formally, we use the following equation to estimate Eq. (10):

$$\tilde{\nabla}_{\tilde{x}}L_{\text{avg}} = \frac{1}{k}\sum_{i=1}^{k}\nabla_{\tilde{x}}L(g(\tilde{x} + r\frac{\eta_i}{||\eta_i||}), y), \quad (11)$$

Thus, the gradient we use to update adversarial example becomes:

$$\nabla_{\tilde{x}}L_{\text{reg}} = (1 + \frac{\lambda}{r})\nabla_{\tilde{x}}L(g(\tilde{x}), y) - \frac{\lambda}{r}\tilde{\nabla}_{\tilde{x}}L_{\text{avg}}, \quad (12)$$

This method is illustrated in details in Figure 2. For simplicity, we name our attack method as **A**verage **D**irection **R**egulation (ADR) attack.

3.3. Further Justifications For ADR Attack Method

Motivation for sharpness-aware optimization In our sharpness-aware optimization, we put regularization on the decision boundary location where adversarial examples are generated. Since in the general case the victim is less sensitive to adversarial examples in the flat area of its decision

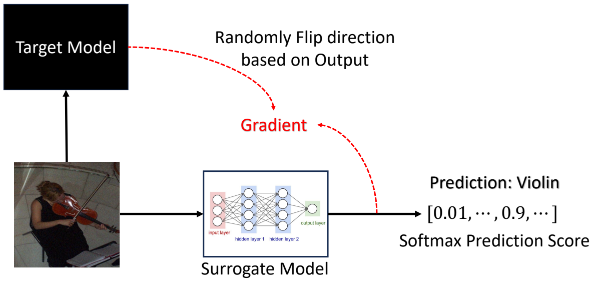


Figure 3. If a limited number of queries are given for querying the black-box target model, we can utilize the output information to randomly flip the estimated average-direction gradient.

boundary [23], adversarial examples generated around this area will be more powerful and be more easily transferred to other models compared to the ones generated in other areas.

Motivation for using randomized direction for optimization. In adversarial example generation, diverse input patterns are proved to be effective in improving the transferability of adversarial examples across different models. [31]. Since adding random noise to the input introduces diversity to the input compared to only using the gradient direction, the adversarial examples generated through random estimation will have a higher probability of fooling the victim model based on the principle we have discussed above. Therefore, we adopt a random estimation in solving sharpness-aware optimization for adversarial example generation, which can lead to higher transferability among different models, as illustrated in Sec. 4.3.

3.4. Utilizing Target Model with Limited Query

In the real-world scenario, attackers may have a limited number of queries (finite horizon) they can use on the black-box target model to aid in the generation of adversarial examples (e.g., by employing the black-box model's outputs). In this context, our question is how to leverage the output from the black-box target model to enhance the transferability of adversarial examples. To address this question, we propose utilizing the estimated gradient from the black-box model to adjust the average ascent direction, which is initially derived entirely from the white-box model.

Firstly, we would like to introduce how gradient is estimated given the black-box prediction [5]. Given the white-box surrogate model g and black-box target model f. Suppose at the time step $T-1$, we have adversarial example x'_{T-1}. We can calculate the loss at x'_{T-1} and denote it as $L(f(x'_{T-1}), y)$. Suppose we spend the last k queries on black-box target model, we can estimate the gradient of loss function with respect to x'_t by Gaussian noise as follows:

$$\tilde{\nabla}_{x'_{T-1}} L_k \approx \frac{\eta_i}{r||\eta_i||} \cdot L\big(f(x'_{T-1} + r\frac{\eta_i}{||\eta_i||}), y\big) - \frac{\eta_i}{r||\eta_i||} \cdot L\big(f(x'_{T-1}), y\big) \quad (13)$$

where η_i is the Gaussian noise sampled at query k.

This can be consider as an estimate of gradient by one-sample Monte Carlo estimation. We can see from this equation that $\frac{L(f((x'_{T-1}+r\frac{\eta_i}{||\eta_i||}),y)-L(f(x'_{T-1}),y)}{r}$ decide the direction of gradient. We can use this information to improve our calculation in Eq (11) as follows, which is demonstrated in Figure 3:

(1) If $\frac{L(f((x'_{T-1}+r\frac{\eta_i}{||\eta_i||}),y)-L(f(x'_{T-1}),y)}{r} > 0$, we keep the sampled Gaussian noise to calculate the average direction in Eq (11). (2) If $\frac{L(f((x'_{T-1}+r\frac{\eta_i}{||\eta_i||}),y)-L(f(x'_{T-1}),y)}{r} < 0$, we adjust the sampled direction to make it align with the gradient estimate direction of black-box target model. In particular, we sample ρ of all elements in η_i and change their values by multiplying them by -1. As a result, we could make the estimated average direction more aligned with the gradient direction of black-box model, which helps improve the transferability rate.

4. Experiments

4.1. Experimental Setup

Dataset The image classification dataset we use in experiment is ImageNet-1k [7]. We randomly sample 5,000 images as samples for attack.

Baselines Methods The used baselines in our comparison experiments are as follows.

- PGD Attack [20]. PGD attack is representative white-box adversarial attack method that iteratively updates the adversarial examples through gradient ascent.
- FGSM Attack [13]. It is another representative white-box adversarial attack that uses gradient information to generate adversarial examples. It takes the sign of the gradient as the perturbation added to the images.
- Square Attack [2] is a black-box adversarial attack that uses randomized search to generated adversarial examples given the black-box prediction information from the target model.
- Universal Adversarial Attack [24]. This is a method that train a general adversarial pattern that can be added universally among different models.
- GNP Attack [30]. Compared to PGD attack, this method introduces an l_2 norm regularizer for gradient and solve it by ascent-direction solution as shown in Eq. (9).

The step size of optimization is $2/255$, and we the maximum perturbation between the perturbed image x' and the original one x is $||x' - x||_\infty = 8/255$, which means the l_∞ norm between x' and x is at most $8/255$.

Evaluation Metric. Naturally, transferability rate (TR) is used in our experimental evaluation. Given a set of testing images $D_{test} = \{(x, y)\}$, we calculate it as follows:

$$TR = \frac{\sum_{(x,y)\ D_{test}} \mathbb{1}_{f(x)=g(x)=y, f(x')\neq y \text{ and } g(x')\neq y}}{\sum_{(x,y)\ D_{test}} \mathbb{1}(f(x) = g(x) = y)}, \tag{14}$$

where $\mathbb{1}$ is the indicator function which equals to 1 when the condition satisfies and equals to 0 when it does not.

Victim Model. We adopt the following representative models of different types as victim models, which including convolutional neural networks, Transformer-based networks, and foundation models.

- ResNet-18 and ResNet-50 [16]. ResNet is a type of representative convolutional neural networks that use residual connection to relieve the vanishing gradient problem in convolutional structures. It has been widely used as the base model for different computer vision tasks. In our experiments, we use two relatively light-weight ResNet structures including ResNet-18 and ResNet-50 as the black-box victim models.
- ViT-b/16 and ViT-b/32 [10]. Vision Transformer is a recently favored DNN architecture for computer vision inspired by the success of Transformer [9] in natural language processing tasks. One representative structure is named ViT, which treats each image patch as Transformer token for computer vision tasks such as image classification. In our experiments, we adopt the relatively lightweight ViT base model (ViT-b) as victim target model, and two of its variants include ViT-b/16 and ViT-b/32, which divide a image into 16 and 32 patches, respectively.
- Swin-T [19]. Swin Transformer is another representative Transformer structure. Compared to ViT structures, it includes shifted windows that can improve the computation efficiency and helps obtain better performance. We used the configuration tiny in our experiment.
- OFA-Large [27]. OFA is a recently proposed foundation model which can be fine-tuned into different types of structures. It uses convolutional structure (e.g., ResNet50) to extract visual features, and the visual features are put to Transformer-based encoder-decoder for predictions. Compared to previously mentioned encoder-only structure (ViT and Swin Transformer), OFA has a brand new encoder-decoder structure that output the prediction through generative response. It helps obtain better performance.

Surrogate Models. In our experiments, the surrogate models include ViT-b/16 and ResNet-50. We use them because they are representative for lightweight convolutional neural networks and Transformer-based model, which can reduce the computation resources the attackers require.

4.2. Comparison Results

4.2.1 Comparison with Baseline Methods

We show the comparison results with baselines in Table 1. Firstly, we can find that attacks such as square attack and universal attack generally cannot generate adversarial example with high transferability. This is basically because (1) black-box attacks only utilize the output logit information for adversarial example generation and they cannot use the gradient information. As a result, since the generation adversarial example does not take the loss landscape of adversarial example generation into consideration, the generated ones tailors only to one specific models and has a lower degree of transferability. (2) The data distribution of images used to learn the universal adversarial pattern in the Universal Attack is usually different from the test data, and thus the attack pattern cannot be easily applied to other target models.

Secondly, among white-box adversarial attacks, PGD attack usually generate adversarial examples with higher transferability rates. Based on that, the gradient regularization is able to further improve the adversarial transferability rate against different black-box target models by the regularizing the loss landscape. As for our optimization framework, it achieves better performance over the ascent-direction optimization framework. For example, in the case of using ResNet-50 as a surrogate model, our method achieves an increase by 14.23% compared to using the ascent-direction for optimization. Those improvement results further demonstrate the effectiveness and reasonableness of utilizing average direction for sharpness-aware optimization in adversarial example generation, which leads to higher transferability rate by utilizing the loss landscape information in different directions around the current adversarial example.

4.2.2 Influence of Using Black-Box Target Model Output Information

After validating the effectiveness of using average direction for solving the adversarial example optimization problem with gradient norm regularization, we further conduct a comparison of using black-box information for adversarial example optimization. We denote the proposed method using output logits to generate adversarial example ADR-BB. In Table 2, in most cases ADR-BB outperforms ADR. Especially, in the case of attacking ViT-b/32 by using ResNet-50 as the surrogate model, ADR-BB achieves up to 3.33% in terms of attack transferability rate. These results show that the adjustment of sample direction through the designed mechanism is able to generated adversarial examples that

Surrogate Model	Target Model	#Param.	PGD	FGSM	Square	Universal Attack	GNP	ADR	Improvement (%)
ViT-b/16	OFA-Large	450.0	11.87	4.24	8.74	/	12.72	**14.87**	14.46
	ViT-b/32	88.2	21.62	10.71	5.41	/	23.85	**25.96**	8.13
	Swin-T	28.3	21.65	8.93	4.22	/	22.51	**24.91**	9.36
	ResNet50	25.6	17.68	7.93	4.68	7.69	18.95	**20.44**	7.28
	ResNet18	11.7	31.82	32.13	7.16	/	33.36	**37.37**	10.73
ResNet-50	OFA-Large	450.0	5.92	2.50	9.47	3.39	8.35	**14.01**	32.41
	ViT-b/32	88.2	12.63	8.24	8.47	6.25	15.14	**18.48**	18.07
	Swin-T	28.3	13.02	6.90	7.17	6.45	16.77	**23.85**	29.69
	ViT-b/16	25.6	9.66	5.70	6.05	7.69	12.42	**16.15**	23.10
	ResNet18	11.7	28.45	14.38	14.63	9.69	37.22	**51.45**	27.66

Table 1. Results of attack transferability rate (%) obtained by different attack methods in ImageNet.

Surrogate Model	Target Model	ADR	ADR-BB	Improvement (%)
ViT-b/16	ViT-b/32	25.96	25.51	-1.73
	Swin-T	24.91	25.17	1.04
	ResNet18	37.37	38.55	3.16
	ResNet50	20.44	21.41	4.75
ResNet-50	ViT-b/32	15.14	18.47	21.99
	Swin-T	23.85	23.94	0.38
	ResNet18	51.45	50.82	-1.22
	ViT-b/16	16.15	16.61	2.85

Table 2. Comparison results of attack transferability rate (%) on using black-box victim model output information.

can achieve better transferability on the black-box target model.

4.3. Ablation Study

Without loss of generality, we adopt the case of using ViT-b/16 as surrogate model to conduct the ablation study for understanding the effectiveness of the proposed method.

4.3.1 Influence of Injecting Noise in Different Layer

As shown in Eq. 11, one of the key calculation of our method is to adopt the average direction for measuring the sharpness to conduct regularization. This operation can be conducted in the latent layer of ViT too. Thus, firstly, we conduct an ablation study on how the number of layer where noise is injected influence the performance. The results are shown in Table 3.

Since ViT-b/16 has 12 self-attention layers, we test on other variants where noise is injected before 1st, 3rd, 6th, 9th and 12th self-attention layers. From the results, we can find that the quality of transferability generally decreases as the noise injected layer go deeper, and the decrease usually converges in the mid-layer (6th) self-attention blocks. These results show that using average direction for sharpness-aware optimization is the most effective for input images and becomes less useful when the noise is added to high-level features. This is because adding Gaussian noise to images is basically constructing a randomized smoothed classifier [6] , which could generate a more robust prediction than the base model. Thus, the gradient calculated from this randomized smoothed classifier is more instructed because it is a feedback signal from attacking a more robust model. And as the added noise go deeper, it is found that the modular sensitivity will become less bounded and the added noise will bring more harm to the model performance [18]. Thus, we shall inject noise in the input image.

4.3.2 Influence of Number of Queries

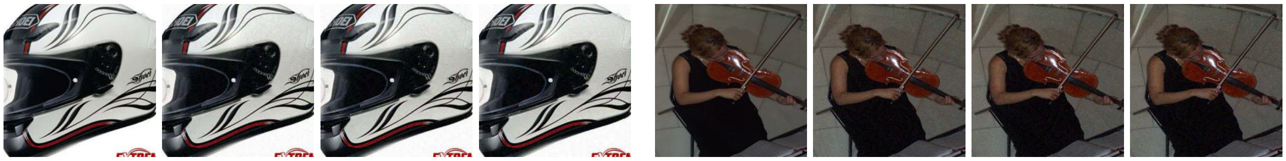
Another key idea for our sharpness-aware optimization is using Monte Carlo estimation for getting an estimate for Eq. (10). Table 4. Whether the results are sensitive to the choice of k is another important aspect of our solution. Thus, we further conduct an experiment on how k affects the results our solution, whose results are shown in Table 4. From those results, it can found that the proposed method can achieve relatively good results on transferability against ViT-b/32, Swin Transformer and ResNet18 when $k = 2$. Thus, it does not take a large k for the proposed method to get a good estimate for Eq. (10) in generating adversarial examples with high transferability rate, which shows the insensitivity of our method with respect to k.

4.3.3 Influence of Number of Optimization Steps

As previous solution, the proposed method is also an iterative optimization solution. We also conduct an ablation study on the influence of optimization steps, as shown in Table 5. We can find in some cases, setting the number of optimization step to be 4 has already helped us obtain adversarial examples with high transferability. Thus, the

Surrogate Model	Target Model	Position of injecting noise					
		Image	1st	3rd	6th	9th	12th
ViT-b/16	OFA-Large	14.87	12.49	12.35	12.69	11.95	12.42
	ViT-b/32	25.96	22.51	20.26	20.40	20.26	21.18
	Swin-T	24.91	21.88	20.44	19.93	19.40	19.85
	ResNet18	37.37	31.94	30.24	29.90	29.19	29.90
	ResNet50	20.44	18.08	16.98	16.30	16.76	16.84

Table 3. Results of attack transferability rate (%) influenced by injecting noise in different layers.



(a) Original (b) PGD (c) GNP (d) ADR (e) Original (f) PGD (g) GNP (h) ADR

Figure 4. Visualization of generated adversarial examples.

Target Model	#Query				
	1	2	3	4	5
OFA-Large	13.24	13.82	13.82	14.61	14.87
ViT-b/32	24.07	25.82	25.29	25.98	25.96
Swin-T	24.42	26.55	26.52	25.78	24.91
ResNet18	35.30	36.63	36.69	38.17	37.37
ResNet50	19.25	18.32	18.72	19.51	20.44

Table 4. Results of attack transferability rate (%) influenced by k.

Target Model	#Optimization steps				
	2	4	6	8	10
OFA-Large	13.56	15.85	15.52	15.32	14.87
ViT-b/32	26.73	28.34	27.34	26.79	25.96
Swin-T	23.79	24.98	24.05	25.25	24.91
ResNet18	36.14	38.79	38.45	38.51	37.37
ResNet50	18.32	19.44	19.61	19.58	20.44

Table 5. Results of attack transferability rate (%) influenced by optimization step.

introduced average-direct gradient does not add further burden to the iterative optimization.

5. Conclusion

Real-world application systems that use deep neural networks usually are designed as black box services for users. In this paper, we mainly investigate the research question of generating adversarial examples with high transferability rate in the setting of using surrogate models to attack black-box target models. Enlightened by the sharpness-aware optimization framework, we introduce the average direction for regularizing the sharpness of loss landscape at the adversarial example to achieve higher transferability. In addition, when attackers are allowed to access the black-box with a limited number of queries, we introduce a direction adjustment mechanism that change the perturbation to obtain a more accurate estimated gradient for adversarial example generation. Experiment results conducted in ImageNet dataset with an extensive selection of target models show that the proposed ADR attack generally produce adversarial examples with higher transferability rate. In addition, the devised mechanism with black-box output information can further help improve the transferability of generated adversarial examples.

References

[1] Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430, 2018. 1

[2] Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In *European conference on computer vision*, pages 484–501. Springer, 2020. 1, 2, 5

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

[4] Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security, AISec@CCS 2017, Dal-*

las, TX, USA, November 3, 2017, pages 15–26. ACM, 2017. 2

[5] Minhao Cheng, Thong Le, Pin-Yu Chen, Jinfeng Yi, Huan Zhang, and Cho-Jui Hsieh. Query-efficient hard-label black-box attack: An optimization-based approach. *arXiv preprint arXiv:1807.04457*, 2018. 5

[6] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *international conference on machine learning*, pages 1310–1320. PMLR, 2019. 7

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[8] Yingpeng Deng and Lina J Karam. Universal adversarial attack via enhanced projected gradient descent. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1241–1245. IEEE, 2020. 1

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 6

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 6

[11] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *arXiv preprint arXiv:2010.01412*, 2020. 2

[12] Ivan Fursov, Matvey Morozov, Nina Kaploukhaya, Elizaveta Kovtun, Rodrigo Rivera-Castro, Gleb Gusev, Dmitry Babaev, Ivan Kireev, Alexey Zaytsev, and Evgeny Burnaev. Adversarial attacks on deep models for financial transaction records. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2868–2878, 2021. 1

[13] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1, 3, 5

[14] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 2

[15] Chuan Guo, Jacob R. Gardner, Yurong You, Andrew Gordon Wilson, and Kilian Q. Weinberger. Simple black-box adversarial attacks. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 2484–2493. PMLR, 2019. 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. 6

[17] Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. 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*, pages 2142–2151. PMLR, 2018. 2

[18] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE symposium on security and privacy (SP)*, pages 656–672. IEEE, 2019. 7

[19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 6

[20] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017. 1, 2, 3, 5

[21] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 2

[22] Natalie Maus, Patrick Chao, Eric Wong, and Jacob Gardner. Adversarial prompting for black box foundation models. *arXiv preprint arXiv:2302.04237*, 2023. 2

[23] Seyed-Mohsen Moosavi-Dezfooli, Omar Fawzi Alhussein Fawzi, Pascal Frossard, and Stefano Soatto. Analysis of universal adversarial perturbations. corr abs/1705.09554 (2017). *arXiv preprint arxiv:1705.09554*, 2017. 5

[24] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017. 5

[25] Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2017, Abu Dhabi, United Arab Emirates, April 2-6, 2017*, pages 506–519. ACM, 2017. 2

[26] Jacob M. Springer, Melanie Mitchell, and Garrett T. Kenyon. A little robustness goes a long way: Leveraging robust features for targeted transfer attacks. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 9759–9773, 2021. 2

[27] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International Conference on Machine Learning*, pages 23318–23340. PMLR, 2022. 6

[28] Futa Waseda, Sosuke Nishikawa, Trung-Nghia Le, Huy H Nguyen, and Isao Echizen. Closer look at the transferability of adversarial examples: How they fool different models differently. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1360–1368, 2023. 3

[29] Kaiyue Wen, Tengyu Ma, and Zhiyuan Li. How sharpness-aware minimization minimizes sharpness? In *The Eleventh International Conference on Learning Representations*, 2022. 2, 3

[30] Tao Wu, Tie Luo, and Donald C Wunsch. Gnp attack: Transferable adversarial examples via gradient norm penalty. *arXiv preprint arXiv:2307.04099*, 2023. 3, 4, 5

[31] Cihang Xie, Zhishuai Zhang, Yuyin Zhou, Song Bai, Jianyu Wang, Zhou Ren, and Alan L Yuille. Improving transferability of adversarial examples with input diversity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2730–2739, 2019. 5