Curvature Dynamic Black-box Attack: revisiting adversarial robustness via dynamic curvature estimation

Peiran Sun PEIRANSUN@QQ.COM
Lanzhou University, Lanzhou, China

Editors: List of editors' names

Abstract

Adversarial attack reveals the vulnerability of deep learning models. It is assumed that high curvature may give rise to rough decision boundary and thus result in less robust models. However, the most commonly used *curvature* is the curvature of loss function, scores or other parameters from within the model as opposed to decision boundary curvature, since the former can be relatively easily formed using second order derivative. In this paper, we propose a new query-efficient method, dynamic curvature estimation (DCE), to estimate the decision boundary curvature in a black-box setting. Our approach is based on CGBA, a black-box adversarial attack. By performing DCE on a wide range of classifiers, we discovered, statistically, a connection between decision boundary curvature and adversarial robustness. We also propose a new attack method, curvature dynamic black-box attack (CDBA) with improved performance using the estimated curvature.

Keywords: adversarial attacks, adversarial robustness, decision boundary curvature

1. Introduction

Adversarial attacks are generally classified into two types: white-box Goodfellow et al. (2015); Moosavi-Dezfooli et al. (2016); Carlini and Wagner (2017) and black-box Chen et al. (2017); Papernot et al. (2016); Brendel et al. (2018); Maho et al. (2021); Rahmati et al. (2020); Reza et al. (2023); Wang et al. (2022); Chen et al. (2020) attacks. In white-box setting the attacker has full knowledge about the interior of the model, where using gradient descent via backpropagation is a common practice. However, white-box scenario is too ideal for real-world application. For black- box setting, there are score-based Chen et al. (2017), transfer-based Papernot et al. (2016) and decision-based attacks Brendel et al. (2018); Rahmati et al. (2020); Maho et al. (2021); Reza et al. (2023); Chen et al. (2020). Score-based attack means that the attacker can access the output probability of the last layer. In transfer-based attack, a surrogate model is trained to generate adversarial examples. The decision-based attack is the most practical, yet hardest attack since it only queries the model for the top-1 label.

Early analysis Carlini and Wagner (2017) of adversarial robustness tried to interpret robustness as resilience to attacks. Specifically, there are ℓ_p robustness Croce et al. (2021) and certified robustness Cohen et al. (2019). These approaches can be effective in adversarial training, but cannot help to explain the vulnerability of neural networks. Fawzi et al. (2016) was among the first researching the relationship between decision boundary curvature and robustness. Instead of studying the quantitative curvature of decision boundaries, it resorted to extensive theoretical proof to show the impact of curvature bound on the relative robustness across different noise regimes. In Moosavi-Dezfooli et al. (2019), curvature of the loss function is calculated by the spectral norm of Hessian matrices. In Entesari et al. (2025), Lipschitz constant was shown to be tight upper bound of the second derivative of the output scores. Curvature regularization via Hessian or Lipschitz constant were proposed to improve robustness. However, these curvature are **parameter curvature** as they can be formulated via backpropagation. **Decision boundary curvature** is hard to be formulated since there's no simple expression for decision boundary.

In this paper, we try to estimate decision boundary curvature of image classifiers in a query-efficient way. We focus on the decision-based adversarial attack and compare robustness with decision boundary curvature. Our contributions are:

- We propose a curvature dynamic trajectory alongside the black-box adversarial attack process to perform Dynamic Curvature Estimation (DCE) for every iteration of the attack in the black-box setting.

- We find a connection between curvature and adversarial robustness by conducting DCE over the standard and robust models on CIFAR-10 and ImageNet, the curvature results are compared with common robustness metrics, including robust accuracy from Robustbench Croce et al. (2021) and certified accuracy from Cohen et al. (2019).

- We introduce Curvature Dynamic Black-box Attack (CDBA), by integrating DCE in current state-of-the-art CGBA, to utilize curvature information in adversarial attacks.

2. Related Work

2.1. Adversarial Attacks

Decision-based adversarial attack was first studied in Brendel et al. (2018). HSJA Chen et al. (2020) introduced normal vector estimation to push the adversarial examples into the vast adversarial space indicated by the normal vector. GeoDA Rahmati et al. (2020) was among the first to hypothesize that the decision boundary in non-targeted attacks can approximated by hyperplanes. This hypothesis is the key to the formulation of the semicircular path used in SurFree Maho et al. (2021) and CGBA Reza et al. (2023) since the closest point on the plane should always lie in the perpendicular direction of that plane, and the angle inscribed across a circle's diameter is always a right angle. The difference between SurFree and CGBA is the choice of the restricted 2D plane on which the semicircular search is performed. In CGBA, the plane is spanned by the adversarial examples and the normal vector of the decision boundary while SurFree employs random orthogonal directions.

2.2. Adversarial robustness

Adversarial robustness is commonly evaluated using adversarial examples. There are ℓ_p robustness and certified robustness. In Croce et al. (2021), ℓ_p robustness is evaluated by the robust accuracy of adversarial examples generated by AutoAttack, an ensemble of white- and black-box attacks. In Cohen et al. (2019), certified robustness is evaluated by certified accuracy of images sampled near source images with random Gaussian corruptions.

Common corruption robustness Hendrycks and Dietterich (2019) is another realm of robustness since it is evaluated using algorithmically generated images instead of adversarial examples. The key difference is that adversarial robustness always requires a limited perturbation budget. Common corruption robustness, on the other hand, is generally considered to be more practical for real-world applications.

3. Problem statement

We introduce the following notations.

The image classifier to be attacked can be written as $f(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}^C$, $\mathbf{x} \in \mathbb{R}^D$ being the input image, D being the dimension of the input image and C being the number of classes. In the decision-based setting, the only information available for attack is the top-1 label $\mathsf{cl}(\mathbf{x}) := \arg\max_k f_k(\mathbf{x})$, $f_k(\mathbf{x})$ standing for the probability predicted for each class k, $1 \leq k \leq C$.

In non-targeted attacks, the attacker only has one correctly classified image $\mathbf{x}_s$. The goal of the attack is to find a perturbed image $\mathbf{x}_b$ satisfying $\mathsf{cl}(\mathbf{x}_b) \neq \mathsf{cl}(\mathbf{x}_s)$. The adversarial space is defined by $\mathcal{O}_{\text{non-targeted}} := \{\mathbf{x}_b \in \mathbb{R}^D : \mathsf{cl}(\mathbf{x}_b) \neq \mathsf{cl}(\mathbf{x}_s)\}$. The decision boundary is $\partial\mathcal{O}_{\text{non-targeted}} := \{\mathbf{x}_b \in \mathbb{R}^D : f_{\mathsf{cl}(\mathbf{x}_s)}(\mathbf{x}_b) = f_i(\mathbf{x}_b), i \neq \mathsf{cl}(\mathbf{x}_s)\}$.

In targeted attacks, the attacker has a correctly classified source image $\mathbf{x}_s$ and a target image $\mathbf{x}_t$. The goal of the attack is to find a perturbed image $\mathbf{x}_b$ satisfying $\mathsf{cl}(\mathbf{x}_b) = \mathsf{cl}(\mathbf{x}_t)$. The adversarial space is defined by $\mathcal{O}_{\text{targeted}} := \{\mathbf{x}_b \in \mathbb{R}^D : \mathsf{cl}(\mathbf{x}_b) = \mathsf{cl}(\mathbf{x}_t)\}$. The decision boundary is $\partial\mathcal{O}_{\text{targeted}} := \{\mathbf{x}_b \in \mathbb{R}^D : f_{\mathsf{cl}(\mathbf{x}_s)}(\mathbf{x}_b) = f_{\mathsf{cl}(\mathbf{x}_t)}(\mathbf{x}_b)\}$.

We simplify the notation for two adversarial spaces $\mathcal{O}_{\text{non-targeted}}$, $\mathcal{O}_{\text{targeted}}$ as $\mathcal{O}$ where it's applicable for both. The optimization of adversarial attacks can be written as:

$$\mathbf{x}_b^\star = \arg\min_{\mathbf{x} \in \mathcal{O}} \|\mathbf{x}_b - \mathbf{x}_s\|. \tag{1}$$

4. Proposed Methods

We carry out the attack following CGBA Reza et al. (2023), by conducting boundary search on normal vector guided semicircular path. Specifically, for a clean image $\mathbf{x}_s$ and adversarial point $\mathbf{x}_{b_t} \in \partial\mathcal{O}$ at tth iteration, we first perform normal vector estimation of the decision boundary as Rahmati et al. (2020):

$$\hat{\boldsymbol{\eta}}_t = \frac{\sum_{i=1}^{N_t} \phi(\mathbf{x}_{b_t} + \boldsymbol{\delta}_i)\boldsymbol{\delta}_i}{\| \sum_{i=1}^{N_t} \phi(\mathbf{x}_{b_t} + \boldsymbol{\delta}_i)\boldsymbol{\delta}_i\|_2}. \tag{2}$$

where $\boldsymbol{\delta}_i \sim \mathcal{N}(0, \sigma^{\mathbf{2}})$, N_t is the number of queries used for estimation and $\phi(\cdot)$ is the hard label indicator function:

$$\phi(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \mathcal{O} \\ -1, & \text{otherwise} \end{cases} \tag{3}$$

Since the semicircular search and DCE are all done on the restricted plane spanned by $\mathbf{x}_{b_t} - \mathbf{x}_s$ and the estimated normal vector $\hat{\boldsymbol{\eta}}_t$, we wish to introduce coordinates on the plane for the sake of discussion. We set the origin at $\mathbf{x}_s$, set the $\frac{\mathbf{x}_{b_t} - \mathbf{x}_s}{\|\boldsymbol{x}_{b_t} - \boldsymbol{x}_s\|_2}$ as the x-axis base $\mathbf{x}_i$, use Schmidt orthogonalization on $\hat{\boldsymbol{\eta}}_t$ to get the normalized perpendicular direction for y-axis base $\mathbf{y}_i$.

It should be noted that there is no guarantee that the estimated $\hat{\boldsymbol{\eta}}_t$ is perpendicular to the decision boundary on this plane since N_t is far less than D (this will be further discussed in ablation study). But in general, the orthogonal $\mathbf{y}_i$ still points towards the adversarial region.

Then, any point on the spanned plane can be written as:

$$\mathbf{x} = \mathbf{x}_s + \langle \mathbf{x} - \mathbf{x}_s, \mathbf{x}_i \rangle \mathbf{x}_i + \langle \mathbf{x} - \mathbf{x}_s, \mathbf{y}_i \rangle \mathbf{y}_i \tag{4}$$

where $\langle , \rangle$ denotes the inner product. For simplicity, we denote the coordinates $(\langle \mathbf{x} - \mathbf{x}_s, \mathbf{x}_i \rangle, \langle \mathbf{x} - \mathbf{x}_s, \mathbf{y}_i \rangle)$ as (x, y). Similarly we can get the polar coordinates (ρ, θ) by $(\frac{\|\mathbf{x}-\mathbf{x}_s\|_2}{\|\boldsymbol{x}_{b_t}-\boldsymbol{x}_s\|_2}, \arctan \frac{\langle \mathbf{x}-\mathbf{x}_s,\mathbf{y}_i \rangle}{\langle \mathbf{x}-\mathbf{x}_s,\mathbf{x}_i \rangle})$.

The semicircular path used in CGBA can be written in polar coordinates as:

$$\rho = cos(\theta) \quad \text{s.t. } \theta \in [0, \pi/2] \tag{5}$$

By binary searching on θ until the error between adversarial and clean points is less than a set tolerance, the algorithm determines the boundary point $\mathbf{x}_{b_{t+1}} \in \partial \mathcal{O}$ for next iteration.

4.1. Dynamic Curvature Estimation (DCE)

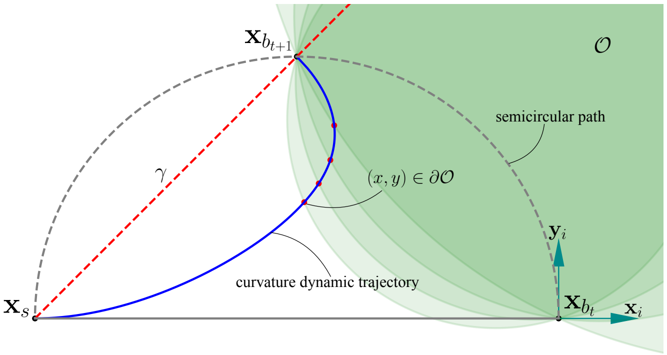

Figure 1: Curvature dynamic searching process. The decision boundary $\partial\mathcal{O}$ between two iterations is approximated by circles with different curvature. The dashed line indicates the original semicircular path, the red dots are the points (x, y) with minimum ℓ_2-norm perturbation on the estimated boundary. The solid line is the trajectory of these closest points on the circles, which is the curvature dynamic trajectory.

We propose a curvature dynamic trajectory to conduct DCE in a query-efficient way. As shown in Figure 1, after the semicircular path has searched another adversarial example $\mathbf{x}_{b_{t+1}}$ on the plane spanned by $\mathbf{x}_{b_t} - \mathbf{x}_s$ and the estimated normal vector $\hat{\boldsymbol{\eta}}_t$, we assume that the local 2-D decision boundary between $\mathbf{x}_{b_{t+1}}$ and $\mathbf{x}_{b_t}$ can be approximated by circles with different curvatures. The trajectory of the points with minimum ℓ_2 perturbation on these circles is the curvature dynamic trajectory.

We write the curvature dynamic trajectory in the form of polar coordinates for the sake of binary searching (proof in Appendix A):

$$\begin{gathered}(\tan\gamma\cos\theta-\sin\theta)r^2-r\tan\gamma+\sin\theta=0,\\ \text{s.t. } \theta\in[0,\gamma].\end{gathered}\tag{6}$$

γ is the angle between $\mathbf{x}_{b_t}-\mathbf{x}_s$ and $\mathbf{x}_{b_t}-\mathbf{x}_s$. By searching on this trajectory, the algorithm will be able to find a minimum $(\hat{\rho},\hat{\theta})\in\partial\mathcal{O}$ dynamically, with respect to different curvature of the circles. And inversely, we can also get the estimated curvature by solving:

$$\hat{\kappa}=((\frac{0.5\tan\gamma}{\tan\gamma-\tan\hat{\theta}}-1)^2+(\frac{0.5\tan\gamma\tan\hat{\theta}}{\tan\gamma-\tan\hat{\theta}})^2)^{-1/2}\tag{7}$$

Algorithm 1 One iteration of DCE

```
Input: Original image $\mathbf{x}_s$, boundary point $\mathbf{x}_{b_t}\in\partial\mathcal{O}$, estimated normal vector $\hat{\boldsymbol{\eta}}_t$
Output: $\mathbf{x}_{b_t+1}\in\partial\mathcal{O}$, $\hat{\kappa}$
$\mathbf{x}_i,\mathbf{y}_i$ = Schmidt_Orthogonalization($\frac{\mathbf{x}_{b_t}-\mathbf{x}_s}{\|\boldsymbol{x}_{b_t}-\boldsymbol{x}_s\|_2},\hat{\boldsymbol{\eta}}_t$)
clean = $\mathbf{x}_s$, adv = $\mathbf{x}_s+\mathbf{x}_i$
Semicircular search by Eq 5 until $\|$clean $-$ adv$\|_2\leq$ tol
Curvature Dynamic Search by Eq 6 between $\mathbf{x}_s$ and adv
adv = $\mathbf{x}_s+\hat{r}\cos\hat{\theta}\mathbf{x}_i+\hat{r}\sin\hat{\theta}\mathbf{y}_i$
$\hat{\kappa}=((\frac{0.5\tan\gamma}{\tan\gamma-\tan\hat{\theta}}-1)^2+(\frac{0.5\tan\gamma\tan\hat{\theta}}{\tan\gamma-\tan\hat{\theta}})^2)^{-1/2}$
Return adv, $\hat{\kappa}$
```

4.2. Curvature Dynamic Black-box Attack (CDBA)

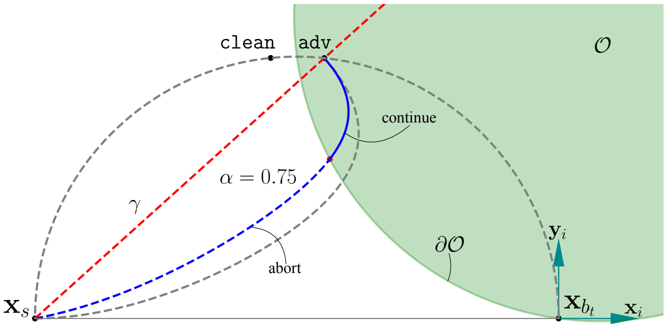

Figure 2: CDBA with abort protocol and step parameter. The gray dashed line indicates the vanilla curvature dynamic trajectory. The blue line is added step parameter $\alpha=0.75$, the dashed part is the interval to abort curvature dynamic search

Since the curvature dynamic search poses an extra query cost, we propose an abort protocol to control the trade-off between this extra query cost and the ℓ_2-norm perturbation of the attack. Abort protocol can stop the curvature dynamic search when the curvature is low. To compensate for the error caused by this protocol and avoid reaching the local minimum, we also proposed a step parameter to control the step length in curvature dynamic searching process.

Algorithm 2 One iteration of CDBA

```
Input: Original image $\mathbf{x}_s$, boundary point $\mathbf{x}_{b_t} \in \partial\mathcal{O}$, estimated normal vector $\hat{\boldsymbol{\eta}}_t$
Output: $\mathbf{x}_{b_t+1} \in \partial\mathcal{O}$
$\mathbf{x}_i, \mathbf{y}_i = \text{Schmidt\_Orthogonalization}(\frac{\mathbf{x}_{b_t} - \mathbf{x}_s}{\|\boldsymbol{x}_{b_t} - \boldsymbol{x}_s\|_2}, \hat{\boldsymbol{\eta}}_t)$
$\texttt{clean} = \mathbf{x}_s$, $\texttt{adv} = \mathbf{x}_s + \mathbf{x}_i$
Semicircular search by Eq 5 until $\|\texttt{clean} - \texttt{adv}\|_2 \leq 1000 \cdot \texttt{tol}$
$\gamma = \arctan(\frac{\langle \texttt{adv} - \mathbf{x}_s, \mathbf{y}_i \rangle}{\langle \texttt{adv} - \mathbf{x}_s, \mathbf{x}_i \rangle})$
Solve $\theta$ for $r = \|\texttt{clean} - \mathbf{x}_s\|_2$ in Eq 6
$\theta = \gamma - \alpha * (\gamma - \theta)$
$\texttt{tmp} = \mathbf{x}_s + r\cos\theta\mathbf{x}_i + r\sin\theta\mathbf{y}_i$
if $\phi(\texttt{tmp}) = 1$ then
    Curvature Dynamic Search by Eq 6 with step parameter between $\mathbf{x}_s$ and tmp
    Return adv
else
    Semicircular search by Eq 5 between clean and adv
    Return adv
end if
```

4.2.1. Abort protocol

At the initial semicircular search, the search will not reach maximum tolerance in the first place. Instead, it will stop when the norm between clean example `clean` and adversarial example `adv` is smaller than 1000 times the final tolerance as shown in Figure 2. Then, we use `adv` to determine the tangent γ, and we check the corresponding point `tmp` with the same norm as `clean` on the trajectory, to see whether the curvature dynamic search will lead to a better result.

If `tmp` is adversarial, then we start the curvature dynamic search between $\mathbf{x}_s$ and `tmp`. If not, the search will continue on the original semicircular path until it reaches the final tolerance.

4.2.2. Step parameter

We also found that searching along the curvature dynamic trajectory sometimes leads to a local minimum, making the descent on later iterations less ideal. Consequently, we proposed a step parameter α to control the step length. In the curvature dynamic searching process, instead of query at the point$(\rho^\star, \theta^\star)$, we query at the point $(\rho^\star, \gamma - \alpha * (\gamma - \theta^\star))$. By setting $\alpha \in [0, 1]$, the curvature dynamic trajectory is flattened as is shown in Figure 2. It should be noted that when $\alpha = 0$, the algorithm will be reduced to CGBA-H which uses a binary line search.

5. Experiments

5.1. Experimental setting

5.1.1. DCE

For ℓ_p robust classifiers, we selected seven WideResNet-28-10 classifiers on CIFAR-10 including three ℓ_2 robust ones Rony et al. (2019); Wang et al. (2023); Rebuffi et al. (2021) and four ℓ_∞ robust ones Hendrycks et al. (2019); Sehwag et al. (2020); Zhang et al. (2021); Cui et al. (2021). We also selected three ℓ_∞ robust ResNet-50 models Wong et al. (2020); Engstrom et al. (2019); Salman et al. (2020) on ImageNet. These robust models and the standard models are available from Robustbench library Croce et al. (2021) and demonstrate different adversarial robustness. The clean accuracy and robust accuracy of these classifiers are available in Appendix C. 100 pairs of images randomly drawn from CIFAR-10 and ILSVRC2012's validation set are used for DCE in ℓ_p robustness analysis.

For certified robust classifiers, we chose seven classifiers including four ResNet-101 and three ResNet-50 classifiers from Cohen et al. (2019). It used randomized smoothing, a voting strategy of samples near source image with Gaussian noise, to improve certified robustness. 50 pairs of images are used for DCE in certified robustness analysis.

For common corruption robust classifiers, three ResNet-50 robust classifiers Geirhos et al. (2018); Erichson et al. (2022); Hendrycks et al. (2021) on ImageNet are selected. 50 pairs of images are used for DCE in common corruption robustness analysis. The classifiers and their respective robust accuracy are also available in Robustbench library Croce et al. (2021).

Considering the difference in converging speed, DCE is performed for 20 iterations on standard CIFAR-10 classifier and 30 for robust ones. DCE is performed for 40 iterations on standard ImageNet classifier and 60 for robust ones. Moreover, instead of taking the average $\hat{\kappa}$ over the entire iteration, the results are first divided into different bins according to their ℓ_2-norm, curvature over 1000 is dropped because of the ill estimation around local minima. We take the average of $\hat{\kappa}$ for every bin in the interval we are interested in. The ℓ_2-norm bins used for CIFAR-10 and ImageNet classifiers are `linspace(1,6,6)` and `linspace(30,80,6)` in adversarial and common corruption robustness analysis. We also take the logarithm of $\hat{\kappa}$ for better visualization.

The ablation study in Appendix B includes limitation of normal vector estimation, effect of normal vector estimation and sensitivity of curvature dynamic trajectory. Moreover, we evaluated the curvature difference between targeted and non-targeted attack decision boundary in standard CIFAR-10 and ImageNet classifiers. 50 pairs of images are used for curvature analysis in different attacks.

5.1.2. CDBA

We evaluate CDBA using standard ResNet-50 ImageNet classifier in RobustBench Croce et al. (2021), since targeted ImageNet decision boundary has the highest curvature. We run CDBA targeted with abort protocol and step parameter $\alpha = 0.75$ for 15 iterations on 100 pairs of images from ILSVRC2012's validation set.

For baselines, we choose CGBA and CGBA-H Reza et al. (2023), the state-of-the-art non-targeted and targeted attack, which is also what DCE is based on. All hyperparameters including query number N_0, final tolerance, DCT factors in all experiments except ablation study are set the same as CGBA and CGBA-H for fair comparison. Note that normalization is dropped to keep aligned with the Robustbench setting.

5.2. Experimental results

5.2.1. Decision boundary curvature and adversarial robustness

Datasets	Structure	Metric	Classifier	$\overline{log(\hat{\kappa})}$				
CIFAR-10	WideResNet	-	**Standard**	**-0.63**	**0.18**	**1.17**	**1.58**	**1.43**
		ℓ_2	Rony et al. (2019)	-1.75	-1.64	-0.93	-0.97	-1.41
			Rebuffi et al. (2021)	**-3.32**	**-3.41**	**-3.18**	**-2.78**	**-2.39**
			Wang et al. (2023)	-3.12	-3.33	-2.65	-2.34	-1.37
		ℓ_∞	Hendrycks et al. (2019)	-1.90	-1.82	-1.10	-1.01	-0.74
			Sehwag et al. (2020)	-1.64	-1.48	-1.59	-1.46	-1.34
			Zhang et al. (2021)	-1.70	-1.55	-1.64	-1.37	-1.20
			Cui et al. (2021)	**-3.03**	**-2.57**	**-2.10**	**-1.74**	**-1.53**
	ResNet-101	-	**Standard**	**-0.26**	**0.44**	**0.75**	**1.08**	**1.54**
		Certified	$\sigma = 0.12$	-0.72	-0.26	-0.32	0.03	0.65
			$\sigma = 0.25$	-0.65	-0.73	-0.49	-0.15	-0.96
			$\sigma = 0.50$	-1.47	**-1.17**	-0.92	-1.36	**-1.35**
			$\sigma = 1.00$	**-2.32**	-1.10	**-1.18**	**-1.70**	-0.59
ImageNet	ResNet-50	-	**Standard**	**3.24**	**3.34**	**3.39**	**3.42**	**3.44**
		ℓ_∞	Wong et al. (2020)	0.59	1.24	1.93	1.93	2.14
			Engstrom et al. (2019)	0.80	1.60	1.79	2.36	2.36
			Salman et al. (2020)	**0.17**	**1.07**	**1.45**	**1.91**	**2.10**
		Certified	$\sigma = 0.25$	0.91	0.77	1.12	1.52	1.70
			$\sigma = 0.50$	0.34	**0.44**	0.73	1.33	1.04
			$\sigma = 1.00$	**0.20**	0.48	**0.19**	**0.01**	**-0.07**

Table 1: Decision boundary curvature in adversarial robust image classifiers. The greatest curvature of every basic classifier structure are marked in bold, the least curvature of every robustness metric are marked in red. Classifiers are listed in ascending order in terms of their robustness.

Table 1 shows the average targeted decision boundary curvature $\overline{log(\hat{\kappa})}$ versus different ℓ_2-norm bins in CIFAR-10 and ImageNet classifiers. Note that the classifiers are listed in ascending order in terms of their robustness. We observe that for all classifier structures on all datasets, the decision boundary curvature in standard classifiers is significantly higher than that in robust classifiers. Moreover, it is also apparent that the curvature in robust classifiers is associated with the exact robustness as measured by the robustness accuracy. The higher curvature of decision boundary, the more vulnerable the classifier will be to adversarial attacks.

This is especially true when the perturbation is small enough when it is closer to the perturbation of adversarial examples used to train and test these models. This might be intuitive in ℓ_2 robust models. But such finding also holds in ℓ_∞ robust models, which means that adversarial robustness is generally able to generate flatter local boundaries regardless of the perturbation distribution.

5.2.2. Decision boundary curvature and common corruption robustness

Classifier	$\overline{log(\hat{\kappa})}$				
Standard	**3.24**	**3.34**	**3.39**	**3.42**	**3.44**
Geirhos et al. (2018)	2.75	2.99	3.08	3.05	3.20
Erichson et al. (2022)	2.71	2.85	3.03	3.17	3.29
Hendrycks et al. (2021)	2.87	2.73	2.68	2.77	2.85

Table 2: Decision boundary curvature in common corruption robust models

In common corruption robust classifiers, however, things are a little bit different. As is shown in Table 2, although the decision boundary curvature is still slightly lower in robust classifiers, they don't seem to be strongly related to their exact robustness. This might be because the perturbation of common corruptions is generally higher than adversarial perturbations, which doesn't necessarily result in flattened local boundary as shown in adversarial training.

5.2.3. CDBA

We run CGBA, CGBA-H, CDBA ($\alpha = 1, 0.75$) separately for 15 iterations of targeted attack on standard ResNet-50 classifier to show the average ℓ_2-norm versus query number results in different attack methods. The least ℓ_2 perturbation is marked in bold.

Query	250	500	750	1000
CGBA	96.93	94.11	91.63	89.05
CGBA-H	82.78	74.13	68.60	63.91
CDBA($\alpha = 1$)	**81.72**	73.86	68.37	64.23
CDBA($\alpha = 0.75$)	82.25	**73.85**	**67.69**	**62.78**

Table 3: mean ℓ_2-norm distortion in targeted attack on ImageNet

From Table 3 we observe that CDBA($\alpha = 0.75$) achieves the least ℓ_2-norm perturbation under a limited query budget 1000. Moreover, standard CDBA outperforms other attacks during the initial descent, with relatively high curvature of decision boundary. In later iterations, as perturbation decreases, the step parameter added successfully compensates for the error caused in the previous abort protocol and prevents the algorithm from reaching the local minima. The effectiveness of step parameter within each iteration will be further discussed in ablation study.

6. Conclusion

In this paper, we proposed Dynamic Curvature Estimation, which performs query-efficient estimation of curvature during the process of black-box attacks. By conducting DCE on standard and robust image classifiers, we revealed a connection between decision boundary curvature and adversarial robustness. We also designed an ablation study on the proposed curvature dynamic trajectory to prove its effectiveness. Furthermore, using curvature dynamic search, we also crafted a new decision-based adversarial attack CDBA, which uses the curvature information to improve the attack under a limited query budget.

References

Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. URL https://openreview.net/forum?id=SyZI0GWCZ.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57, 2017. doi: 10.1109/SP.2017.49.

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

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

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1310–1320. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/cohen19c.html.

Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. URL https://openreview.net/forum?id=SSKZPJCt7B.

Jiequan Cui, Shu Liu, Liwei Wang, and Jiaya Jia. Learnable boundary guided adversarial training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15721–15730, 2021.

Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019. URL https://github.com/MadryLab/robustness.

Taha Entesari, Sina Sharifi, and Mahyar Fazlyab. Compositional curvature bounds for deep neural networks. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2025.

N. Benjamin Erichson, Soon Hoe Lim, Francisco Utrera, Winnie Xu, Ziang Cao, and Michael W. Mahoney. Noisymix: Boosting robustness by combining data augmentations, stability training, and noise injections. *CoRR*, abs/2202.01263, 2022. URL https://arxiv.org/abs/2202.01263.

Alhussein Fawzi, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Robustness of classifiers: from adversarial to random noise. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 1632–1640, Red Hook, NY, USA, 2016. Curran Associates Inc. ISBN 9781510838819.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *CoRR*, abs/1811.12231, 2018. URL http://arxiv.org/abs/1811.12231.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

Dan Hendrycks and Thomas G. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *CoRR*, abs/1903.12261, 2019. URL http://arxiv.org/abs/1903.12261.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2712–2721. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/hendrycks19a.html.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021.

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

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: A simple and accurate method to fool deep neural networks. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2574–2582. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.282. URL https://doi.org/10.1109/CVPR.2016.282.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9070–9078, 2019. doi: 10.1109/CVPR.2019.00929.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in machine learning: from phenomena to black-box attacks using adversarial samples. *ArXiv preprint*, abs/1605.07277, 2016. URL https://arxiv.org/abs/1605.07277.

Ali Rahmati, Seyed-Mohsen Moosavi-Dezfooli, Pascal Frossard, and Huaiyu Dai. Geoda: A geometric framework for black-box adversarial attacks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 8443–8452. IEEE, 2020. doi: 10.1109/CVPR42600.2020.00847. URL https://doi.org/10.1109/CVPR42600.2020.00847.

Sylvestre-Alvise Rebuffi, Sven Gowal, Dan A. Calian, Florian Stimberg, Olivia Wiles, and Timothy Mann. Fixing data augmentation to improve adversarial robustness, 2021. URL https://arxiv.org/abs/2103.01946.

Md Farhamdur Reza, Ali Rahmati, Tianfu Wu, and Huaiyu Dai. Cgba: curvature-aware geometric black-box attack. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 124–133, 2023.

Jérôme Rony, Luiz G. Hafemann, Luiz S. Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4317–4325, 2019. doi: 10.1109/CVPR.2019.00445.

Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3533–3545. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/24357dd085d2c4b1a88a7e0692e60294-Paper.pdf.

Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Jana. Hydra: Pruning adversarially robust neural networks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 19655–19666. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/e3a72c791a69f87b05ea7742e04430ed-Paper.pdf.

Xiaosen Wang, Zeliang Zhang, Kangheng Tong, Dihong Gong, Kun He, Zhifeng Li, and Wei Liu. Triangle attack: A query-efficient decision-based adversarial attack. In *European conference on computer vision*, pages 156–174. Springer, 2022.

Zekai Wang, Tianyu Pang, Chao Du, Min Lin, Weiwei Liu, and Shuicheng Yan. Better diffusion models further improve adversarial training. In *International Conference on Machine Learning (ICML)*, 2023.

Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=BJx040EFvH.

Jingfeng Zhang, Jianing Zhu, Gang Niu, Bo Han, Masashi Sugiyama, and Mohan Kankanhalli. Geometry-aware instance-reweighted adversarial training. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=iAX0l6Cz8ub.

Appendix A. Proof for curvature dynamic trajectory

We assume that the local 2-D decision boundary between $\mathbf{x}_{b_{t+1}}$ and $\mathbf{x}_{b_t}$ can be approximated by circles with different curvatures.

For these circles to pass $\mathbf{x}_{b_t}$, the equation of the set of circles can be written as:

$$(x - x_0)^2 + (y - y_0)^2 = (x_0 - 1)^2 + y_0^2 \tag{8}$$

where (x_0, y_0) is the center of circle.

Denote the angle between $\mathbf{x}_{b_t} - \mathbf{x}_s$ and $\mathbf{x}_{b_t} - \mathbf{x}_s$ as γ. For the circles to pass both $\mathbf{x}_{b_t}$ and $\mathbf{x}_{b_{t+1}}$, the center should be on the perpendicular bisector of them, which is:

$$y_0 = \tan\gamma(x_0 - 0.5) \tag{9}$$

Finally, the closest points (x, y) to $\mathbf{x}_s$ on the circles should also satisfy that:

$$y = \frac{y_0}{x_0}x \tag{10}$$

By solving Eq9 and Eq10, we can get:

$$\begin{aligned} x_0 &= \frac{0.5\tan\gamma \cdot x}{\tan\gamma \cdot x - y}, \\ y_0 &= \frac{0.5\tan\gamma \cdot y}{\tan\gamma \cdot x - y}. \end{aligned} \tag{11}$$

By inserting Eq11 into Eq 8, we can get the trajectory of the points with minimum ℓ_2 perturbation on these circles, which is the curvature dynamic trajectory:

$$\begin{gathered} (\tan\gamma\cos\theta - \sin\theta)r^2 - r\tan\gamma + \sin\theta = 0, \\ \text{s.t. } \theta \in [0, \gamma]. \end{gathered} \tag{12}$$

Appendix B. Ablation study

B.1. Limitation of normal vector estimation

As mentioned before, the estimated normal vector on the entire space(even after the dimension reduction via DCT) differs greatly from the normal vector in the restricted plane because the query number is far less than the dimension. Here we propose an experiment with normal vector estimation on the restricted plane immediately after the initial full space normal vector estimation in each iteration. We first generate a set of random vectors $\{\mathbf{x}_k\}_{k=1}^K$ and project them to $\{\mathbf{x}_k^{\text{proj}}\}_{k=1}^K$ on the plane using:

$$\mathbf{x}_k^{\text{proj}} = \langle \mathbf{x}_k, \mathbf{x}_i \rangle \mathbf{x}_i + \langle \mathbf{x}_k, \mathbf{y}_i \rangle \mathbf{y}_i \tag{13}$$

where $\mathbf{x}_i$ and $\mathbf{y}_i$ are the xy basis of the plane. Then $\{\mathbf{x}_k^{\text{proj}}\}_{k=1}^K$ is normalized to $\boldsymbol{\delta}_k^{\text{proj}} \sim \mathcal{N}(0, \sigma^{\mathbf{2}})$. We can obtain the normal vector on the plane by:

$$\hat{\boldsymbol{\eta}}_t^{\text{proj}} = \frac{\sum_{k=1}^K \phi(\mathbf{x}_{b_t} + \boldsymbol{\delta}_i^{\text{proj}})\boldsymbol{\delta}_i^{\text{proj}}}{\| \sum_{k=1}^K \phi(\mathbf{x}_{b_t} + \boldsymbol{\delta}_i^{\text{proj}})\boldsymbol{\delta}_i^{\text{proj}} \|_2}. \tag{14}$$

where $\phi(\cdot)$ is the hard label indicator function. The error angle between $\hat{\boldsymbol{\eta}}_t^{\mathrm{proj}}$ and $\hat{\boldsymbol{\eta}}_t$ can be obtained by $\psi_t = \arccos \frac{\langle \hat{\boldsymbol{\eta}}_t, \hat{\boldsymbol{\eta}}_t^{\mathrm{proj}} \rangle}{\|\hat{\boldsymbol{\eta}}_t\|_2 \cdot \|\hat{\boldsymbol{\eta}}_t^{\mathrm{proj}}\|_2}$

Specifically, we choose $\sigma = 0.0002$, which is the same as full space normal vector estimation in CGBA. Four initial query number N_0 are selected for normal vector estimation with dimension reduction factor $f = 4$ and $K = 100$ for normal vector estimation within the restricted plane. The experiment is conducted for 20 iterations on 100 pairs of random images in standard CIFAR-10 classifier.

N_0	10	20	30	40
MSE($^\circ$)	48.34	43.28	40.41	37.69

Table 4: MSE of normal vector estimation with different initial query number N_0

Table 4 shows the MSE of the normal vector estimation measured by ψ_t between $\hat{\boldsymbol{\eta}}_t^{\mathrm{proj}}$ and $\hat{\boldsymbol{\eta}}_t$ in degree. Since the error is quite large, it is barely possible to utilize the geometric property of normal vectors in the restricted plane. This is why we propose the circular interpolation of decision boundary in DCE regardless of the normal vector. However, in DCE we still follow the normal vector estimation setting in CGBA since it successfully minimizes the ℓ_2 perturbation in a query-efficient way. This might be because although there is estimation error, the vector is still potentially pointing towards the vast convex adversarial region.

B.2. Effect of normal vector estimation in curvature estimation

Normal vector estimation can still affect curvature estimation in a way that it determines the direction from the whole space to construct the restricted plane. To reveal its effect, we compare 300 random estimated curvature with ℓ_2-norm $\in [1, 2)$ in the previous experiment. Similarly, estimated curvature over 1000 is dropped empirically because of bad estimation around local minima.

N_0	10	20	30	40
$log(\hat{\kappa})$	$1.79_{\pm 1.11}$	$1.64_{\pm 1.00}$	$1.51_{\pm 1.07}$	$1.49_{\pm 1.11}$

Table 5: $log(\hat{\kappa})$ with different initial query number N_0

Table 5 shows the mean and standard deviation of the logarithm of the estimated curvature in four N_0. We observe a modest drop in curvature as query number increases. An ANOVA test of four sets is conducted with a result of $F = 4.88$ and $p = 0.002$, indicating there is a significant shift in the mean curvature using different initial query numbers. Normal vector estimation with higher query number can not only result in reduced error, but also points towards flatter adversarial region.

B.3. Sensitivity of curvature dynamic trajectory

We compare the result of CGBA-H and three DCE steps ($\alpha = 0.5, 1, 1.5$) without abort protocol in each iteration to reveal the effectiveness of curvature dynamic trajectory. The

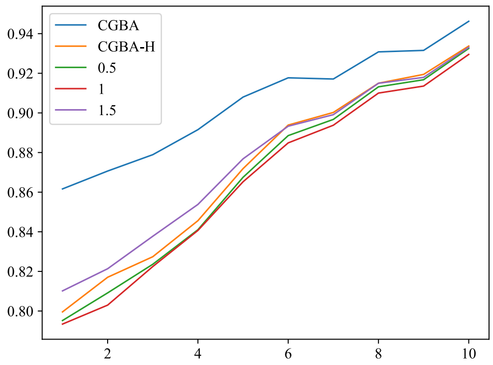

Figure 3: $\|\mathbf{x}_{b_{t+1}}\|_2/|\mathbf{x}_{b_t}\|_2$ in first 10 iterations

experiment is conducted for 10 iterations on 100 pairs of random images in standard CIFAR-10 classifier. We record their mean ℓ_2-norm decrease percentage in each iteration.

As shown in Figure 3, DCE with $\alpha = 1$ achieves the best performance over other step parameters and CGBA-H per iteration, indicating that the vanilla DCE can best locate the minima on the restricted plane. It also means that the local circular assumption of decision boundary is effective in constructing the curvature dynamic trajectory.

B.4. Targeted and Non-targeted decision boundary curvature

Dataset	ℓ_2-norm bins			
CIFAR-10	[0,1)	[1,2)	[2,3)	[3,4)
Targeted	**-1.66**	**-0.63**	**0.18**	**1.17**
Non-targeted	-2.38	-1.77	-1.19	-0.82
ImageNet	[0,10)	[10,20)	[20,30)	[30,40)
Targeted	**2.23**	**2.43**	**2.93**	**3.24**
Non-targeted	-0.67	0.12	-0.37	-0.89

Table 6: $\overline{log(\hat{\kappa})}$ versus ℓ_2-norm for different attacks

Table 6 shows the curvature difference in targeted and non-targeted decision boundaries. The ℓ_2-norm bins used here are slightly different since the attack converges faster in non-targeted scenario. It is clear that the decision boundary in targeted attacks is higher than that in non-targeted attacks, which is in accordance with previous research.

Although the non-targeted decision boundary curvature is actually lower, most attacks including CGBA converge faster in non-targeted attacks. This means that the estimated curvature cannot be simply interpreted as an indicator of converging speed or attack success rate and is thus associated with robustness. Higher curvature is not a direct result of better attack results.

Appendix C. Adversarial robustness of models

	Classifier	Clean acc.	Robust acc.
CIFAR(ℓ_2)	Standard	94.78%	0
	Rony et al. (2019)	89.05%	66.44%
	Rebuffi et al. (2021)	91.79%	78.80%
	Wang et al. (2023)	95.16%	83.68%
CIFAR(ℓ_∞)	Standard	94.78%	0
	Hendrycks et al. (2019)	87.11%	54.92%
	Sehwag et al. (2020)	88.98%	57.14%
	Zhang et al. (2021)	89.36%	59.64%
	Cui et al. (2021)	92.16%	67.73%
ImageNet	Standard	76.52%	0
	Wong et al. (2020)	55.62%	26.24%
	Engstrom et al. (2019)	62.56%	29.22%
	Salman et al. (2020)	64.02%	34.96%

Table 7: Clean and robust accuracy of ℓ_p robust classifiers on two datasets