Exploiting Pre-Trained Models and Low-Frequency Preference for Cost-Effective Transfer-based Attack

MINGYUAN FAN, East China Normal University, Shanghai, China
CEN CHEN, East China Normal University, Shanghai, China and The State Key Laboratory of Blockchain and Data Security, Zhejiang University, Shanghai, China
CHENGYU WANG and JUN HUANG, Alibaba Group, Hangzhou, China

The transferability of adversarial examples enables practical transfer-based attacks. However, existing theoretical analysis cannot effectively reveal what factors contribute to cross-model transferability. Furthermore, the assumption that the target model dataset is available together with expensive prices of training proxy models also leads to insufficient practicality. We first propose a novel frequency perspective to study the transferability and then identify two factors that impair the transferability: an unchangeable intrinsic difference term along with a controllable perturbation-related term. To enhance the transferability, an optimization task with the constraint that decreases the impact of the perturbation-related term is formulated and an approximate solution for the task is designed to address the intractability of Fourier expansion. To address the second issue, we suggest employing pre-trained models as proxy models, which are freely available. Leveraging these advancements, we introduce cost-effective transfer-based attack (*CTA*), which addresses the optimization task in pre-trained models. *CTA* can be unleashed *against broad applications, at any time, with minimal effort and nearly zero cost to attackers.* This remarkable feature indeed makes *CTA* an effective, versatile, and fundamental tool for attacking and understanding a wide range of target models, regardless of their architecture or training dataset used. Extensive experiments show impressive attack performance of *CTA* across various models trained in seven black-box domains, highlighting the broad applicability and effectiveness of *CTA*.

CCS Concepts: • **Computing methodologies → Computer vision; Machine learning; • Security and privacy → Systems security;**

Additional Key Words and Phrases: Deep Neural Networks, Adversarial Examples, Black-box Adversarial Attacks, Transferability

ACM Reference format:
Mingyuan Fan, Cen Chen, Chengyu Wang, and Jun Huang. 2025. Exploiting Pre-Trained Models and Low-Frequency Preference for Cost-Effective Transfer-based Attack. *ACM Trans. Knowl. Discov. Data.* 19, 2, Article 52 (February 2025), 18 pages.
https://doi.org/10.1145/3680553

This work was supported by the National Natural Science Foundation of China under grant number 62202170, the Open Research Fund of the State Key Laboratory of Blockchain and Data Security, Zhejiang University, and Alibaba Group through the Alibaba Innovation Research Program.
Authors' Contact Information: Mingyuan Fan, East China Normal University, Shanghai, China; e-mail: fmy2660966@gmail.com; Cen Chen (corresponding author), East China Normal University, Shanghai, China and The State Key Laboratory of Blockchain and Data Security, Zhejiang University, Shanghai, China; e-mail: cenchen@dase.ecnu.edu.cn; Chengyu Wang, Alibaba Group, Hangzhou, China; e-mail: chengyu.wcy@alibaba-inc.com; Jun Huang, Alibaba Group, Hangzhou, China; e-mail: huangjun.hj@alibaba-inc.com.

ACM 1556-472X/2025/2-ART52
https://doi.org/10.1145/3680553

1 Introduction

The significance of adversarial examples' transferability in practical black-box attacks has provoked a surge of interest within the AI community [10, 12, 31, 42]. Specifically, the adversarial examples derived from local proxy models have demonstrated their transferability to attack other unknown models, enabling practical black-box attacks. However, the advancement in this research area is impeded by two primary constraints.

The first limitation stems from the inadequate theoretical analysis of transferability [33, 38]. Existing studies [3, 22, 32] have employed various tools, such as gradient similarity, to explain transferability. However, these analyses still remain rudimentary and have yet to unlock their full potential [3, 38]. To be specific, cross-model transferability is a direct result of gradient similarity. A deeper understanding is needed to determine why certain adversarial examples exhibit high cross-model gradient similarity, what factors contribute to this phenomenon, and which aspects of the model facilitate transferability. Addressing these questions would allow us to more fully comprehend the limitations and effectiveness of transfer-based attacks and develop effective countermeasures.

The second limitation arises from the doubtful practicality of existing transfer-based attacks in real-world scenarios [11, 22, 23], caused by the following reasons. Typically, most attention [6, 7, 19, 30] in this field is paid to explore the cross-model transferability among models trained on the identical dataset, e.g., ImageNet. However, the dataset information associated with the target model is rarely disclosed to untrusted individuals and it is difficult to collect sufficiently similar datasets to train proxy models. For example, in situations where collecting data for a particular label is difficult, training a suitable proxy model becomes infeasible. Moreover, the expensive computational price of training a proxy model [5, 38] also makes a significant bottleneck for the practicality of transfer-based attacks.

Our Contribution. To address the first limitation, we introduce a novel frequency perspective to study transferability. From the frequency perspective, a function can be decomposed as a weighted sum of different frequency functions, offering a clear, concise, and consistent representation of **deep neural networks (DNNs)** with different architectures. This fresh perspective allows for circumventing certain issues that plague existing literature [15, 33, 38]. In particular, structural differences across different DNNs are difficult to handle but have to be taken into consideration when probing the transferability.

Through the lens of the frequency perspective, we identify two terms that impair the transfer effectiveness: namely the inherent difference term along with the perturbation-related term. The former is caused by intrinsic differences between models and is unchangeable. In contrast, the latter is associated with adversarial perturbation and is controllable. To increase the transfer effectiveness, we introduce an optimization constraint that seeks to lower the influence of the perturbation-related term. However, solving this constraint is challenging due to the intractability of Fourier expansion in high-dimensional spaces. To address the challenge, a bound for the original constraint is derived to serve as the new constraint. The optimization task with the new constraint can be approximately solved by exploiting the low-frequency preference principle [26, 39, 41] of DNNs (see Section 5).

The second limitation essentially pertains to the challenge of obtaining a cost-effective proxy model. To address the problem, we propose leveraging pre-trained models as the proxy model. Nowadays, there are a sufficient number of well-trained pre-trained models available to the public [8, 25], which can be accessed free of charge. Moreover, since pre-trained models output embeddings for fed inputs, we can attack the embeddings instead of minimizing probabilities as in typical transfer-based attacks. Doing so eliminates the need to pay attention to the problem of the label space, rendering our attack universally applicable to most deployed models.

By the above analysis, we propose **cost-effective transfer-based attack (CTA)** that produces adversarial examples by solving the formulated task in pre-trained models. The effectiveness

of *CTA* is evaluated through extensive experimentation, where the produced adversarial ones show competitive attack performance across various models trained in seven unknown black-box domains. Besides, our evaluation further consolidates a recently showed counter-intuitive finding: complex-design attacks are commonly less effective than simplistic attacks, likely due to overfitting to commonly used benchmarks [19, 44]. Compared to [22], this article incorporates more state-of-the-art models and transfer attack methods, thereby enhancing the reliability of this counter-intuitive finding. In other words, the complexity of these complex-design attacks may lead to excessive optimization for specific benchmarks at the expense of generalization capability.

2 Related Work

Regarding the underlying source of the transferability, some prior works [14, 35] empirically demonstrated that earlier layers of different models tend to learn similar patterns that induce transferability, but the exact mechanisms at play remain elusive. Prior studies [3, 22, 32] utilized a variety of tools including game theory and gradient similarity to model and explain transferability, as well as measuring differences in model architecture to gain insights into this complex issue. However, these studies only showed the correlation between the transferability of adversarial samples and the tools employed, like gradient similarity. The specific factor that causally leads to the transferability of adversarial samples across different models remains unclear. Therefore, the source of the transferability of adversarial examples continues to remain shrouded in mystery and a deeper analysis is urgently needed.

Moreover, another important line of this field is to make adversarial examples even more transferable across different models. To achieve this, these works modify the backpropagation process [36, 42], diversifying inputs and models [18, 19, 31, 37], tuning gradients [30], and so on. However, these studies only showed the correlation between the transferability of adversarial samples and the tools employed, like gradient similarity. The specific factor that causally leads to the transferability of adversarial samples across different models remains unclear. Besides, Wang et al. [34] introduce a regularization term to penalize the distance between adversarial perturbations and their transformed versions generated by randomly removing certain frequency components of the original perturbation. In contrast, we conduct theoretical analysis to demonstrate the importance of low-frequency components to transferability and thus focus on corrupting low-frequency components by adding random noises to inputs. Notice that, in these works, the employed proxy models typically share the same training dataset of the target model, i.e., ImageNet, which mismatches with a complete black-box setting. Generator-based attacks [23, 43] use a non-identical dataset to train a generative model, which can produce adversarial examples for arbitrary samples. The generated ones sometimes can still remain threatening to target models trained in different black-box domains. Compared to the abovementioned attacks, our attack leverages an off-the-shelf pre-trained model and eliminates the need to collect data and train models. Therefore, our approach offers better practicality in real-world scenarios.

3 Approach

We focus on transfer-based black-box adversarial attacks, where adversarial examples generated on the proxy model sometimes can also fool other unknown target models. Let $P_{\theta_p}(\cdot)$ denote the proxy model parameterized by θ_p. Given a sample $x \in [0,1]^d$ with ground-truth label $y \in [0,1]^K$ where d, K is the dimension of x and K is the number of categories. Formally, the vanilla transfer-based attacks typically solve the following task to craft the adversarial perturbation δ^*:

$$\delta^* = \arg\max_{\delta} \mathcal{L}(P_{\theta_p}(x+\delta), y),\ s.t.,\ ||\delta||_\infty \leq \epsilon, \tag{1}$$

where $\mathcal{L}(\cdot,\cdot)$ is the loss function and ϵ is the perturbation budget to account for similarity between x and $x+\delta$. The standard practice to solve Equation (1) is to leverage gradient-based optimization algorithms. However, during the optimization process, the adversarial perturbation is updated greedily at each iteration to maximize the loss function. This often leads to overfitting of δ^* to the proxy model, i.e., rendering δ^* less effective against the target model.

To mitigate the overfitting problem, we propose a variant of Equation (1). Intuitively, in Equation (1), the overfitting problem is generally caused by excessive reliance on corrupting model-specific features of x. A straightforward solution is to introduce random noises that serve to disrupt these model-specific features to a certain degree. In this way, during the optimization process, attention is redirected toward perturbing a broader range of features rather than exclusively focusing on the model-specific ones. With this consideration in mind, we rewrite Equation (2) as follows:

$$\delta^* = \arg\max_{\delta} \mathcal{L}(P_{\theta_p}(x+\delta+\upsilon), y),\ \upsilon \sim N(0,\rho I),\ s.t., ||\delta||_\infty \leq \epsilon, \tag{2}$$

where υ is isotropic Gaussian noises and ρ is the noise magnitude. Section 5 presents the formal justification for the effectiveness of Equation (2). In detail, given the tendency of DNNs toward capturing low-frequency components with greater fidelity [39, 40], the divergence in the high-frequency components across different DNNs is commonly more pronounced compared to the low-frequency components. We demonstrate that the incorporation of Gaussian noises in Equation (2) is capable of reducing the impact of the high-frequency components during the optimization process, thereby enhancing the cross-model transferability of the resulting adversarial examples.

The remaining problems are the choice of proxy models and the concretization of the loss function.

Proxy Model. We suggest employing pre-trained models as proxy models, motivated by two key considerations. First, there are many off-the-shelf pre-trained models. Second, *CTA* corrupts high-level semantic information in images given by pre-trained models, which results in a more general-purpose and convenient attack compared to existing transfer-based attacks that reduce model confidence for specific labels. For example, existing attacks reduce model confidence associated with either cats or dogs to attack a DNN distinguishing cats and dogs. However, when switching from attacking the DNN to another DNN, the adversarial ones for the original DNN cannot be reused and it is required to train a brand-new proxy model from scratch, i.e., traditional transfer-based attacks are highly tailored to individual DNN and lack universality. In contrast, *CTA* does not require a new proxy model when the target model changes since it corrupts overall semantic information rather than label-specified information. This feature reduces attack costs and makes *CTA* more efficient when attacking multiple DNNs.

Loss Function Design. Let z denote the embedding vector outputted by the pre-trained model $P_{\theta_p}(\cdot)$ for $x+\delta$. It is believed that the high-level semantic information of samples is well encoded in z and the information is sufficient to linearly separate different categories of samples. Therefore, we suppose that the target model is similar to a concatenation of the proxy model and a linear layer. Formally, the target model passes z through the linear layer, parameterized by $w_i, i = 1, \cdots, K$, to yield the confidence score for ith category. The predicted probability of a sample for the ith category, denoted by p_i, can be expressed as follows:

$$p_i = \frac{e^{z^T w_i}}{\sum_{j=1}^{K} e^{z^T w_j}}, \tag{3}$$

Algorithm 1: *CTA*

Input: $P_{\theta_p}(\cdot)$: the proxy model; x, y: the natural sample and its ground-truth label; $\mathcal{L}$: the loss function (Equation (5)); ϵ: perturbation budget; ρ: noise magnitude; T: the number of iterations; α: step size; N: the number of sampled Gaussian noise υ.

Initialize adversarial perturbation $\delta = 0$.
for each iteration $i = 1$ to T **do**
  for $j = 1$ to N **do**
    Sample Gaussian noise $\upsilon \sim N(0, \rho I)$.
    Compute the loss $\mathcal{L}(P_{\theta_p}(x+\delta+\upsilon), y)$ and get the gradients $g_j = \nabla_\delta \mathcal{L}(P_{\theta_p}(x+\delta+\upsilon), y)$.
  end for
  Update adversarial perturbation $\delta = \delta + \frac{\alpha}{N} \sum_{j=0}^{j=N-1} \text{sign}(g_j)$.
  Clip adversarial perturbation $\delta = \max(\min(\delta, \epsilon), -\epsilon)$.
end for
Return: the crafted adversarial example $x + \delta$.

where w_i can be regarded as the representation vector of the ith category. Although employing the cosine similarity between z^T and w_i as the loss function is a straightforward option, the loss function in fact can be further improved. To show this, we rewrite p_i as follows:

$$p_i = \frac{e^{\left(\frac{||z||_2}{||z||_2} z\right)^T \frac{||w_i||_2}{||w_i||_2} w_i}}{\sum_{j=1}^{K} e^{\left(\frac{||z||_2}{||z||_2} z\right)^T \frac{||w_j||}{||w_j||} w_j}} = \frac{e^{||w_i||_2 ||z||_2 \ cos\_sim(z, w_i)}}{\sum_{j=1}^{K} e^{||w_j||_2 ||z||_2 \ cos\_sim(z, w_j)}} = \frac{(a_i)^{||z||_2}}{\sum_{j=1}^{K} (a_i)^{||z||_2}}, \tag{4}$$

$$a_i = e^{||w_i||_2 \ cos\_sim(z, w_i)}.$$

In addition to the cosine similarity between z and w_i, the norm of z also impacts the predicted probability. To be specific, the magnitude of a_i, which is associated with the norm magnitude of w_i and the cosine similarity between z and w_i, determines the prediction category of x by the model. Moreover, the magnitude of the norm of z reflects the confidence of the model in its decision. In detail, assuming that the model assigns x to the ith category, we have $p_i = 1/(\sum_{j=1}^{K} (\frac{a_j}{a_i})^{||z||_2})$ and $\frac{a_j}{a_i} < 1, i \neq j$. As the norm of z increases, so does the corresponding probability p_i, i.e., the model strengthens its belief of a sample belongs to ith category and is less convinced of other categories. Similarly, a small magnitude leads to a uniform distribution of prediction probabilities, indicating that the model is uncertain about which category the sample belongs to. If $||z||_2 = 0$, there is $p_1 = p_2 = \cdots = p_K = \frac{1}{K}$. Inspired by the above observation, the ultimate loss function comprises two terms: the cosine similarity and the norm of z. Notably, the form of the loss differs slightly between white-box and black-box scenarios.

In the white-box scenario, the proxy model is the target model. Thus, by reducing the cosine distance between z and w_i on the proxy model, it is guaranteed that the target model no longer correctly identifies $x + \delta$, and increasing the norm of z makes the proxy model more confident. In the black-box scenario, the target model is not the proxy model, indicating $x + \delta$ may not be misclassified by the target model due to the divergence between the proxy and target models. Increasing the norm of z probably reduces attack effectiveness. Thus, a more suitable option is to reduce the norm of z, as doing so can make the model less confident about its decision. In a nutshell, we define

$$\begin{aligned} &\mathcal{L}(P_{\theta_p}(x+\delta+\upsilon), y) = cos\_sim(z, z_0) - \beta \cdot ||z||_2, \\ &z = P_{\theta_p}(x+\delta+\upsilon), \ z_0 = P_{\theta_p}(x), \ \upsilon \sim N(0, \rho I), \beta \geq 0, \end{aligned} \tag{5}$$

Table 1. Comparison of Eight Domains

Domain	Number of Categories	Granularity	Image Size	Label Space
SVHN	10	Coarse-grained	32×32	Digit
CIFAR-10	10	Coarse-grained	32×32	Animals, Transportations
CIFAR-100	100	Fine-grained	32×32	Animals, People, Transportations
STL-10	10	Coarse-grained	96×96	Animals, Transportations
FGVC AirCraft	102	Fine-grained	224×224	Aircraft
CUB-200-2011	200	Fine-grained	224×224	Birds
ImageNet	1,000	Mid-grained	224×224	Universal Objects
CLIP	N/A	N/A	224×224	Universal Objects

where z_0 is used to replace w_i and β serves a balance factor. In practice, Equation (5) is solved by gradient descent algorithm with multiple sampled v. Algorithm 1 summarizes the overall process of *CTA*.

4 Experimental Evaluation

4.1 Experimental Settings

Proxy Model. We employ the vision encoder of CLIP [25], one of the most common pre-trained models, as the proxy model to construct *CTA*.

Competitor. Six state-of-the-art transfer-based attacks are considered to compare: BIM [17], DI [37], MI [6], TI [7], VR [30], and SSA [19]. These attacks are originally implemented using the cross-entropy loss function based on ImageNet-trained models. For fair comparisons, we substitute the proxy models and loss function with the CLIP model and our loss function to evaluate.

Black-Box Domains. To deliver a reliable evaluation of transfer-based attacks, we carefully select seven different black-box domains together with various model architectures, including CIFAR-10 [16], CIFAR-100 [16], SVHN [24], STL-10 [1], FGVC AirCraft [21], CUB-200-2011 [29], and ImageNet [4]. As shown in Table 1, these black-box domains enjoy significant distribution differences from CLIP. For SVHN, CIFAR-10, and CIFAR-100, we train five model architectures, DenseNet, EfficientNet, MobileNetV2, ResNet18, and ShuffleNetV2, using publicly available codes, while for STL-10, FGVC AirCraft, CUB-200-2011, and ImageNet, we use publicly available models to attack, including CNNs, Vision Transformers, and L_2, L_∞ adversarially trained models.

Metric. **Attack success rate (ASR)** serves as the evaluation metric, which is the misclassification rate of the target models over adversarial ones. Higher ASR indicates better attack performance.

Implementation Details. If not otherwise specified, we use the same hyperparameters as [6, 7, 37] for all attacks, including $\epsilon = \frac{16}{255}$, step size of $\alpha = \frac{1.6}{255}$, and iteration of 10. Furthermore, we set $N = 10$, $\beta = 1$, and $\rho = 1.0$ for our method and the hyperparameter setup for other attacks aligns with their original papers. N is the number of sampled v. Moreover, the test sets of CIFAR-10, CIFAR-100, and SVHN are hired and a random subset of 10,000 samples is extracted from ST-L10, FGVC AirCraft, and CUB-200-2011 datasets for the attack evaluations. For ImageNet, we employ the benchmark sub-dataset [44] to evaluate.

4.2 Attacks over Various Black-Box Domains

Attacks on Coarse-Grained Domains. Table 2 reports the empirical performance of attacks across three coarse-grained domains: SVHN, CIFAR-10, and STL-10. Overall, it is observed from Table 2 that *CTA* is significantly more effective than the baselines, with improvements of up to 3.46%,

Table 2. The Attack Success Rates (ASR) (%) of Different Attacks on Coarse-Grained Black-Box Domains

Domain	Target Model	FGSM	BIM	DI	MI	TI	VR	SSA	*CTA* (Ours)
SVHN	DenseNet	14.58	13.73	15.07	14.51	12.17	15.07	14.84	**18.53**
	EfficientNet	17.59	16.74	18.30	15.85	13.95	17.63	17.52	**19.53**
	MobileNetV2	15.82	14.51	17.19	17.19	14.73	16.18	16.96	**20.20**
	ResNet	14.92	13.62	**17.52**	15.40	13.73	15.40	15.29	17.08
	ShuffleNetV2	17.06	16.18	16.41	16.63	16.07	19.08	17.08	**19.98**
CIFAR-10	DenseNet	55.85	54.02	53.79	57.37	47.66	54.35	56.47	**60.94**
	EfficientNet	51.08	49.67	52.68	53.35	48.88	52.46	52.23	**58.15**
	MobileNetV2	46.74	45.20	46.54	46.88	41.07	45.31	47.32	**51.34**
	ResNet	43.02	41.85	42.41	46.76	38.17	43.19	44.98	**50.56**
	ShuffleNetV2	39.53	38.84	41.07	42.52	38.06	38.39	40.40	**42.86**
STL-10	ResNet50	43.08	41.55	42.20	45.30	42.53	41.47	41.57	**46.30**

The best results are in bold.

4.80%, and 1.00% in SVHN, CIFAR-10, and STL-10, respectively. In addition to this main finding, two intriguing observations emerge from Table 2. Firstly, all attacks only achieve relatively low ASRs, less than 20.20%, in SVHN as compared to CIFAR-10 and STL-10. This could be due to the limited discriminative features learning by CLIP for digit images, caused by the huge discrepancy in SVHN and training data distribution of CLIP. Specifically, as reported in [25], CLIP encounters a major difficulty in classifying digits as its training dataset only contains a few images of digits. Secondly, except for *CTA*, the results in Table 2 reveal no consistent outperformance of any single attack method over others. More specifically, some sophisticated attack methods, such as VR and SSA, instead yield inferior performance to simpler alternatives in STL-10. In contrast, MI, a relatively simple attack, achieves competitive performance across most cases. This indicates that the attacks may overfit to specified cases like when the training dataset between the proxy model and the target model is identical. In other words, a simple design may offer a more universally effective solution.

Attacks on Fine-Grained Domains. Table 3 reports the performance of different attacks in three fine-grained domains. The principal conclusions in the fine-grained domains align with those observed in coarse-grained domains. Moreover, we observe that the attack effectiveness in fine-grained domains is commonly better than in coarse-grained domains. Such phenomena can be attributed to the high semantic similarity among images within fine-grained domains, which leads to a substantial degree of feature overlap and the high compactness of images over decision spaces. As a result, even slight perturbations are sufficient for moving them across decision boundaries.

Attacks on ImageNet. We examine the performance of *CTA* in ImageNet, a large-scale domain widely used as a benchmark in transfer-based attacks. Notice that the evaluation differs from existing transfer-based attacks in a realistic black-box scenario used here by not training the proxy model in ImageNet, providing more reliable results. The results are reported in Table 4. Besides convolutional network architectures, *CTA* also are evaluated over the transformer-liked model architectures, Vit and Swin. As can be seen, regardless of the model architecture used, our attack achieves the best ASRs.

Table 3. The ASR (%) of Different Attacks on Fine-Grained Black-Box Domains

Domain	Target Model	FGSM	BIM	DI	MI	TI	VR	SSA	*CTA* (Ours)
CIFAR-100	DenseNet	81.08	78.79	82.48	83.26	77.57	82.16	82.76	**84.58**
	EfficientNet	75.52	73.77	75.00	75.78	73.66	74.50	72.08	**77.92**
	MobileNetV2	76.87	74.89	76.45	76.00	71.76	74.70	76.01	**79.23**
	ResNet	70.60	67.86	70.09	74.11	66.07	68.45	71.07	**73.29**
	ShuffleNetV2	75.25	74.67	74.33	74.22	74.22	71.37	72.98	**75.60**
FGVC AirCraft	ResNet50	66.28	64.83	64.93	68.70	63.03	65.73	67.20	**68.80**
	SeNet	64.51	62.13	62.70	65.73	62.33	62.57	65.40	**67.27**
	SeRes101	80.25	78.43	78.33	80.90	76.97	78.97	79.87	**84.50**
CUB-200-2011	ResNet50	64.78	62.37	62.33	66.90	62.37	63.23	63.13	**76.40**
	SeNet	49.23	46.90	49.00	50.50	52.50	47.40	47.23	**59.87**
	SeRes101	49.47	47.50	47.23	51.40	45.53	45.97	47.47	**55.43**

The best results are in bold.

Table 4. The ASR (%) of Different Attacks on ImageNet against Regular Models

Target Model	FGSM	BIM	DI	MI	TI	VR	SSA	*CTA* (Ours)
ResNet50	55.80	52.70	59.50	53.60	55.90	50.70	53.20	**68.60**
WideResNet50	48.20	46.40	51.70	45.90	51.40	44.80	49.70	**68.50**
DenseNet121	56.70	53.20	59.50	51.60	56.50	51.80	54.40	**73.20**
EfficientNet	55.60	51.60	56.80	51.40	52.60	49.30	54.30	**73.00**
Inc-V3	57.30	54.70	63.90	55.90	61.60	57.00	58.20	**74.30**
VGG19	64.60	62.20	66.70	62.90	65.00	61.80	64.50	**78.20**
MNASNet	70.20	68.50	72.70	69.50	68.80	69.00	70.70	**86.80**
RegNet	53.70	50.70	55.30	48.90	54.40	49.10	51.80	**70.90**
MobilenetV2	68.20	65.70	73.00	65.60	71.40	65.40	66.30	**78.50**
ShuffleNet	70.10	69.40	72.10	71.20	70.20	68.80	69.00	**80.40**
SqueezeNet	86.30	84.40	85.00	87.80	84.50	84.30	84.20	**90.00**
ConvNext	22.10	21.40	23.10	23.20	21.30	22.20	26.20	**39.00**
ViT	30.80	29.70	34.00	30.60	30.20	31.10	35.50	**57.50**
SwinT	24.80	23.60	25.70	26.70	21.20	25.30	27.10	**40.90**

The best results are in bold.

4.3 Attacks over Secured (Adversarially Trained) Models

We here report the attack results of *CTA* against secured models trained in dataset corrupted [13] and dataset augmented with adversarial examples [27] (known as adversarial training). The adversarially trained models include both single models and ensemble models, which are trained with multiple combinations of constraint norm type and perturbation budget. Therefore, the evaluation nearly covers all possible secured models. Table 5 reports the results and *CTA* considerably outperforms the baselines for most of the secured models, except against EnsAdvIncV2 and Linf-8.0. In cases against EnsAdvIncV2 and Linf-8.0, our method only slightly underperforms MI and TI by less than 0.3%.

Table 5. The ASRs (%) of Different Attacks on ImageNet against Secured Models

Target Model	FGSM	BIM	DI	MI	TI	VR	SSA	*CTA* (Ours)
SIN	65.10	63.30	66.00	63.00	61.50	63.10	62.70	**75.20**
SIN-IN	49.30	47.40	54.00	46.60	53.50	48.40	49.30	**66.40**
AdvIncV3	57.20	56.10	57.60	58.40	59.90	55.00	55.40	**60.50**
EnsAdvIncV2	31.30	29.30	37.30	30.20	**37.50**	31.30	31.40	37.40
L2-0.03	56.70	54.40	62.00	55.30	56.90	54.30	55.60	**73.80**
L2-0.05	50.10	47.70	58.10	50.30	52.50	47.40	49.20	**68.40**
L2-0.1	46.80	44.60	54.00	45.40	51.10	44.70	46.00	**66.50**
L2-0.5	42.90	39.80	44.30	45.00	44.90	40.10	39.50	**55.00**
L2-1.0	44.50	41.80	46.50	46.30	46.70	41.50	42.10	**54.50**
L2-3.0	44.60	42.10	43.30	45.70	46.10	41.70	41.50	**51.00**
L2-5.0	47.20	44.90	47.30	48.20	47.50	45.80	46.90	**50.70**
Linf-0.5	38.40	36.90	41.60	40.30	42.30	38.40	37.40	**53.20**
Linf-1.0	35.60	33.40	36.60	38.70	37.10	34.20	33.20	**44.20**
Linf-2.0	33.20	31.00	33.60	35.20	32.70	31.80	32.10	**36.30**
Linf-4.0	38.40	36.90	38.00	40.60	38.40	37.00	37.30	**41.10**
Linf-8.0	46.70	45.50	45.80	**47.30**	45.40	45.60	45.40	47.00

The best results are in bold.

Table 6. The ASR (%) of Attacks against ResNet50 with Different Defenses in ImageNet

Defense	RP	NIPS-R3	FD	ComDefend	RS
MI	25.70	29.40	35.60	44.90	22.60
VT	26.80	29.80	39.50	46.90	29.90
SSA	34.80	37.30	43.80	49.50	31.80
CTA (Ours)	**39.50**	**47.90**	**60.50**	**58.70**	**34.60**

The best results are in bold.

4.4 Attacks over State-of-the-Art Defenses

In this subsection, we examine the attack performance of different attacks against state-of-the-art defenses. Table 6 reports the attack results of four attack methods against five state-of-the-art defenses, including RP, NIPS-R3, FD, ComDefend, and RS, with the target model ResNet50 on ImageNet. As can be seen, *CTA* consistently outperforms all baselines across all defenses tested, further supporting the superior effectiveness of *CTA*.

4.5 Attacks with Different Proxy Models

In this subsection, we investigate the attack performance of different pre-trained models for constructing our method. Particularly, we substitute the visual encoders of CLIP with the backbones of MNASNet, VGG19, InceptionV3 (Inc-v3), InceptionV4 (Inc-v4), Inception-Resnet-v2 (IncRes-v2), and Resnet-v2-152 (Res-152) trained on ImageNet. Tables 7, 8, and 9 report the attack performance using these models as proxy models against target models trained on four black-box domains including CIFAR-10, CIFAR-100, FGVC AirCraft, and ImageNet. Firstly, we observe that the attack performance of our method consistently outperforms state-of-the-art attacks, VR and SSA, across

Table 7. The ASR (%) of Attacks with Different Proxy Models on CIFAR-10 and CIFAR-100

Proxy Model	Attack	CIFAR-10			CIFAR-100		
		ResNet	DenseNet	EfficientNet	ResNet	DenseNet	EfficientNet
MNASNet	VR	22.98	33.91	31.19	46.88	61.26	54.91
	SSA	25.73	36.69	34.25	50.66	61.98	51.24
	CTA (Ours)	**34.03**	**42.02**	**38.29**	**52.27**	**64.61**	**61.39**
VGG19	VR	29.66	36.46	38.77	56.37	72.92	61.79
	SSA	32.90	39.92	39.14	57.78	71.88	59.99
	CTA (Ours)	**37.59**	**47.55**	**44.14**	**59.26**	**73.81**	**63.84**
Inc-v3	VR	40.29	51.22	48.22	63.74	78.53	70.03
	SSA	40.59	52.10	48.71	66.62	80.03	69.18
	CTA (Ours)	**45.87**	**57.73**	**53.70**	**69.07**	**83.21**	**77.98**

The best results are in bold.

Table 8. The ASR (%) of Attacks with Different Proxy Models on FGVC AirCraft and ImageNet

Proxy Model	Attack	FGVC AirCraft			ImageNet		
		ResNet50	SeNet	SeRes101	ResNet50	DenseNet121	EfficientNet
MNASNet	VR	48.05	44.00	58.44	30.30	31.70	31.30
	SSA	48.66	44.56	61.63	33.40	34.10	32.20
	CTA (Ours)	**50.81**	**47.25**	**64.51**	**46.40**	**51.50**	**53.90**
VGG19	VR	52.77	48.17	65.38	39.30	36.30	36.70
	SSA	54.86	51.04	66.57	39.10	39.20	43.20
	CTA (Ours)	**55.20**	**54.24**	**70.81**	**57.00**	**59.90**	**61.40**
Inc-v3	VR	63.32	60.49	73.93	45.80	48.40	47.40
	SSA	64.14	62.02	74.45	49.50	50.20	51.00
	CTA (Ours)	**68.04**	**62.95**	**82.09**	**63.60**	**67.60**	**68.10**

The best results are in bold.

all cases. Furthermore, we find that employing CLIP as the proxy model yields the best attack performance, followed by Inception, VGG19, and MNASNet. It is intuitive that the effectiveness of transfer-based attacks is correlated with the degree of overlap between the features learned by the proxy models and the target models. Thus, CLIP, which learns more general object features [25], is more likely to possess overlapping features with the target models, leading to better attack performance compared with other models.

4.6 Time Complexity Comparison

We here are interested in the time complexity of different attack methods. The runtime of these attacks is primarily determined by the number of forward and backward passes needed. Specifically, FGSM requires only one forward and backward pass through the proxy model, resulting in a time complexity of $O(1)$. BIM is an iterative version of FGSM, and its running time scales linearly with the number of iterations T, i.e., $O(T)$. Similarly, DI, MI and TI share the same time complexity $O(T)$, since they only conduct simple arithmetic operations which can be neglected. On the other

Table 9. The ASR (%) of Attacks with Four Different Proxy Models on ImageNet

Proxy Model	Attack	Inc-v3	Inc-v4	IncRes-v2	Res-152	Inc-v3$_{ens3}$	Inc-v3$_{ens4}$	IncRes-v2$_{ens}$
Inc-v3	DI	99.7	64.6	59.6	47.8	14.1	14.5	7.0
	VR	**99.9**	75.5	69.0	62.7	32.9	30.9	17.4
	SSA	99.2	87.6	85.9	81.4	52.3	51.7	37.8
	CTA (Ours)	99.3	**90.4**	**88.3**	**83.4**	**53.5**	**54.1**	**39.3**
Inc-v4	DI	72.7	**99.9**	61.8	51.2	17.4	15.8	8.5
	VR	78.4	**99.9**	71.9	64.0	37.6	39.0	22.8
	SSA	89.9	**99.9**	86.3	82.5	60.8	60.7	45.5
	CTA (Ours)	**90.9**	99.7	**87.7**	**84.7**	**63.2**	**63.5**	**46.8**
IncRes-v2	DI	70.7	66.9	**100.0**	58.1	25.2	20.8	14.6
	VR	79.1	76.4	**100.0**	67.3	47.1	40.5	34.3
	SSA	89.8	88.5	99.8	85.3	64.5	59.8	54.0
	CTA (Ours)	**91.9**	**91.5**	99.9	**88.0**	**67.0**	**62.1**	**55.8**
Res-152	DI	79.4	76.2	74.2	**100.0**	34.9	28.8	19.1
	VR	73.9	70.0	66.3	**100.0**	45.0	41.1	30.4
	SSA	89.0	86.6	86.9	99.7	66.5	62.6	50.6
	CTA (Ours)	**90.6**	**90.7**	**89.2**	99.8	**69.8**	**64.2**	**52.9**

Inc-v3$_{ens3}$, Inc-v3$_{ens4}$, and IncRes-v2$_{ens}$ come from [28]. The best results are in bold.

Table 10. The Empirical Running Time of Eight Attack Methods for Crafting a Single Adversarial Example Using an NVIDIA A10

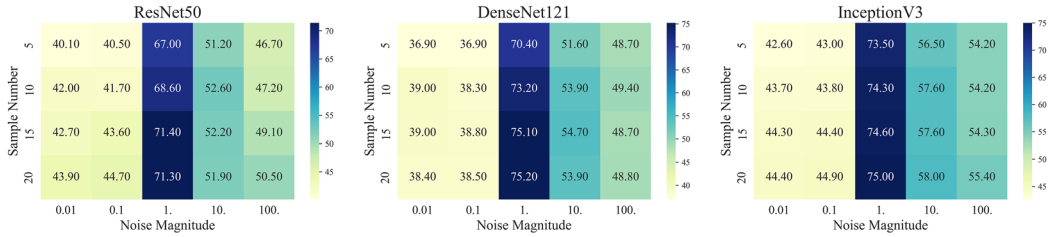
Attack	FGSM	BIM	DI	MI	TI	VR	SSA	*CTA* (Ours)
Running Time (s)	0.051	0.498	0.501	0.499	0.498	5.012	5.076	5.008

Table 11. The ASR (%) of Attacks with Different Perturbation Budget ϵ against ResNet50 in ImageNet

Perturbation Budget ϵ	8	10	12	14	16
MI	39.50	43.20	48.40	50.80	53.60
VT	35.30	39.50	46.50	48.30	50.70
SSA	41.30	44.30	47.20	50.20	53.20
CTA (Ours)	**50.60**	**55.30**	**62.00**	**65.10**	**68.60**

The best results are in bold.

hand, VR, SSA, and our proposed method require computing gradients of N transformed inputs per iteration. Accordingly, their running time can be approximated as N times that of BIM, i.e., $O(TN)$. Table 10 reports the empirical time required by each attack to generate adversarial samples for a single input, which aligns with our analysis. In general, *CTA* has an empirical running time similar to that of VR and SSA.



(a) (b) (c)

Fig. 1. The attack effectiveness of *CTA* with varying N (sample number) and ρ (noise magnitude) against three target models (ResNet50, DenseNet121, and Inc-v3) over ImageNet.

Table 12. The Attack Effectiveness of *CTA* with Varying β.

β	0	0.01	0.1	1	10	100
ResNet50	66.90	67.20	68.20	68.60	68.80	68.50
DenseNet121	71.80	72.00	72.60	73.20	73.30	73.50
VGG19	75.30	75.80	77.30	78.20	78.40	78.20

The best results are in bold.

4.7 Sensitive Analysis and Ablation Study

Attacks with Varying Perturbation Budget ϵ. We here evaluate the attack performance of *CTA* under varying perturbation budgets. Specifically, Table 11 presents the attack effectiveness of different attack methods with varying ϵ on the target model, ResNet50, trained on ImageNet. Overall, reducing the perturbation budget weakens the attack performance. Moreover, it is observed that *CTA* surpasses baselines by a clear margin regardless of the chosen perturbation budget.

The Impact of N and ρ. Figure 1 shows the impact of sample number N and noise magnitude ρ on *CTA*. Overall, increasing N leads to a better attack performance, as it induces more accurate estimation for Equation (5). Besides, the ASRs increase until the peak at a noise magnitude of 1.0, followed by a gradual decrease. Intuitively, as shown in Figure 4, too small a noise magnitude results in relatively lower ASRs due to ineffective elimination of high-frequency components; vice versa.

The Impact of β. We here evaluate the effectiveness of *CTA* with different β. Table 12 presents the attack results over different β. As shown in Table 12, the performance of *CTA* is improved by setting $\beta \geq 0$. Moreover, we see that the marginal gains from increasing β diminish as the value of β surpasses 1. Worse, increasing β beyond a certain threshold (1 or 10) may even result in a decline in attack performance. We hypothesize that an excessively large value of β makes $||z||_2$ dominate the optimization process, neglecting the cosine similarity.

5 Theoretical Justification

In this section, we justify the effectiveness of Equation (2) from the frequency perspective. Fourier expansion of Equation (1) yields

$$\delta^* = \arg\max_{\delta} \sum_j a_j exp\{iw_j^T[x+\delta, y]\},\ s.t.,\ ||\delta||_\infty \leq \epsilon, \tag{6}$$

where $w_j \in \mathbb{R}^{d+K}$, a_j, i, $[\cdot,\cdot]$ stand for the frequency vector, the corresponding frequency coefficient, the imaginary number, and the concatenation operation, respectively. In the frequency domain,

the difference between two functions can be characterized by the difference in their frequency coefficients. By denoting the frequency coefficient difference between the proxy and target models for w_j as Δa_j and assuming ϵ to be small, the loss of $x+\delta^*$ in the target model is expressed as

$$\begin{aligned}
&\mathcal{L}(T_{\theta_t}(x+\delta^*), y) \\
&= \sum_j (a_j + \Delta a_j) exp\{iw_j^T[x+\delta^*, y]\} = \sum_j \left\{a_j exp\{iw_j^T[x+\delta^*, y]\} + \Delta a_j exp\{iw_j^T[x+\delta^*, y]\}\right\} \\
&\approx \sum_j a_j exp\{iw_j^T[x+\delta^*, y]\} + \Delta a_j exp\{iw_j^T[x, y]\} + \Delta a_j exp\{iw_j^T[x, y]\} iw_j^T[\delta^*, 0]. \qquad (7)
\end{aligned}$$

The third line of Equation (7) hires linear approximation. The loss is decomposed into three terms, namely the attack effectiveness term, the inherent difference term, and the perturbation-related term, from left to right. The attack effectiveness term is identical to the loss of $x+\delta^*$ in the proxy model, quantifying the attack performance of $x+\delta^*$ against the proxy model. Commonly, δ^* is threatening to the proxy model, i.e., $x+\delta^*$ has a huge loss in the proxy model. However, $x+\delta^*$ is often of low attack performance against the target model, indicating that the inherent difference term and the perturbation-related term lower the loss of $x+\delta^*$ in the target model and thus undermining the transferability of $x+\delta^*$.

In light of the above insights, producing transferable adversarial ones can be formulated as follows:

$$\begin{aligned}
&\delta^* = \arg\max_{\delta} \mathcal{L}(P_{\theta_p}(x+\delta), y), \\
&s.t., ||\delta||_\infty \le \epsilon, \sigma \ge 0, \left| \sum_j \Delta a_j exp\{iw_j^T[x, y]\} + \sum_j \Delta a_j exp\{iw_j^T[x, y]\} iw_j^T[\delta, 0] \right| \le \sigma. \qquad (8)
\end{aligned}$$

In Equation (8), the optimization target promotes $x+\delta$ to be effective against the proxy model and the constraint ensures the similarity of loss of $x+\delta$ in proxy and target models (small σ makes better loss similarity). By leveraging Equation (8), the produced adversarial examples $x+\delta^*$ are more transferable, which is formally demonstrated in Theorem 5.1.

Theorem 5.1. *Given ϵ and σ, let δ^* be the optimal solution for Equation (8), and loss stands for the corresponding value of $L(P(x+\delta^*), y)$. Assume cross-entropy loss function is used. In particular, if $e^{\max\{-loss+\sigma, -loss-\sigma\}} \le \frac{1}{K}$, $x+\delta^*$ can fool the target model.*

Proof. Notice *loss* stands for the loss value of $x+\delta^*$ in the proxy model. According to Equation (8), the loss value of $x+\delta^*$ in the target model is bounded between $loss \pm \sigma$. Moreover, as cross-entropy loss function is used, the predicted probability of the target model to $x+\delta^*$ is bounded between $e^{-loss\pm\sigma}$. For a K-classification task, if there exists a predicted probability value less than $\frac{1}{K}$, then another value greater than $\frac{1}{K}$ must exist. Otherwise, the sum of the probabilities over all categories is less than 1. Therefore, if $e^{\max\{-loss+\sigma, -loss-\sigma\}} \le \frac{1}{K}$, $x+\delta^*$ can fool the target model. □

However, the constraint in Equation (8) poses a significant challenge, primarily due to the requirement of full access to the target model to evaluate Δj, compounded by the intractability of Fourier expansion in high-dimensional spaces.

We first simplify Equation (8) by using an upper bound of the constraint as a substitution. Furthermore, we show the new constraint becomes more manageable by exploiting the fundamental

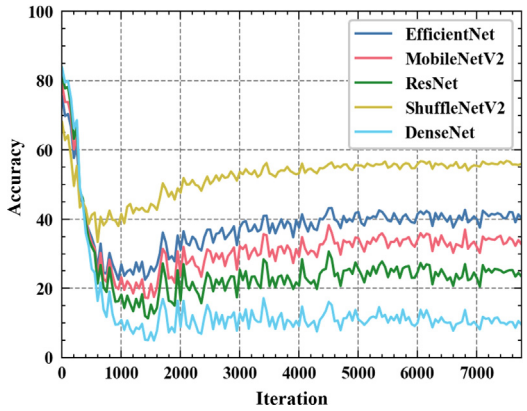


Fig. 2. The accuracy of target models in adversarial examples produced by DenseNet over different training iterations.

characteristics of DNNs. In detail, there is

$$\begin{aligned} &\left| \sum_j \Delta a_j exp\{iw_j^T[x,y]\} + \sum_j \Delta a_j exp\{iw_j^T[x,y]\} iw_j^T[\delta,0] \right| \\ &\leq \left| \sum_j \Delta a_j exp\{iw_j^T[x,y]\}| + \sum_j |\Delta a_j| \, |exp\{iw_j^T[x,y]\}i| \, |w_j^T[\delta,0]\right|. \end{aligned} \tag{9}$$

In Equation (9), $|\sum_j \Delta a_j exp\{iw_j^T[x,y]\}|$ is not associated with δ and can be discarded. As such, $\sum_j |\Delta a_j| \, |exp\{iw_j^T[x,y]\}i||w_j^T[\delta^*,0]| \leq \sigma$ is used to replace the original constraint.

Although the new constraint is less complex than the original, it still needs to resort to Fourier expansion. We here present an approximate solution. The core idea behind the solution is that: if δ is unrelated to high-frequency components, the new constraint can be approximately satisfied. Consequently, the constrained optimization task can be converted into an easy-to-handle unconstrained task, if δ keeps unrelated to high-frequency components throughout the optimization process.

Specifically, on the one hand, compared to low-frequency vectors, high-frequency vectors possess larger norms, i.e., high magnification strength. Therefore, if δ is related to high-frequency vectors, the constraint is more likely to be breached. On the other hand, a lot of studies [39, 41], both theoretically and empirically, show that the coefficients corresponding to low-frequency components across different models are usually quite close to each other. In detail, the training of DNNs places a high priority on fitting low-frequency vectors and thus it is more possible that the low-frequency coefficients between DNNs have a nearly negligible small gap. In contrast, the fitting of high-frequency coefficients is affected by various factors.

Empirical validation of this is observed in Figures 2 and 3, where we measure the transferability of adversarial ones generated by DenseNet and EfficientNet over different training iterations to five black-box target models. Wherein, the transferability increases during the early training stage since the networks first fit the low-frequency vectors and the resultant adversarial noises are prone to attack the low-frequency vectors. After that, the transferability decreases as the networks start to learn high-frequency vectors that attract a part of the attention of adversarial noises. Thus, it is believed that keeping noises correlated with low-frequency vectors can better align the constraint compared to high-frequency vectors.

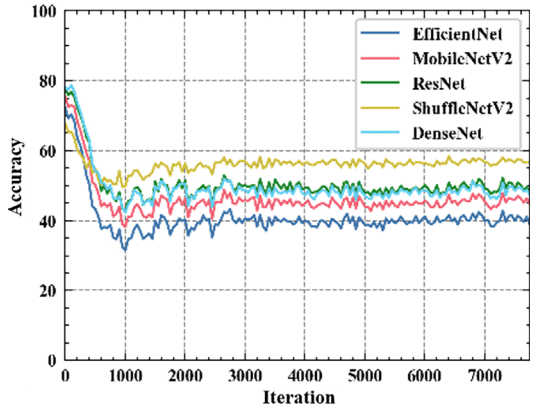


Fig. 3. The accuracy of target models in adversarial examples produced by EfficientNet over different training iterations.

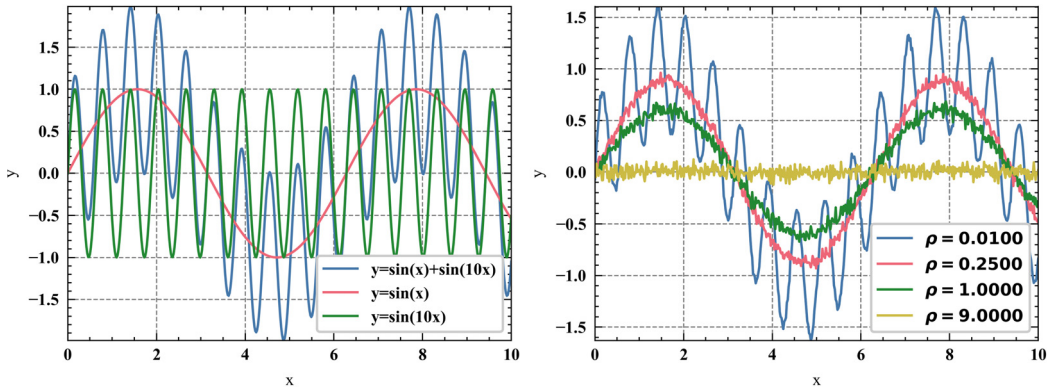


Fig. 4. The left image shows a function consisting of a low-frequency and a high-frequency function. Based on Theorem 5.2, the right image illustrates the smoothed versions of $y = sin(x) + sin(10x)$ over different ρ. With a suitable value of ρ, the high-frequency component $y = sin(10x)$ can be effectively removed while the low-frequency component $y = sin(x)$ remains.

Theorem 5.2 [9]. *Let v be isotropic Gaussian noises, i.e., $v \sim N(0, \rho I)$ where ρ is the noise magnitude. Assume $\mathcal{L}(P_{\theta_p}(\cdot), \cdot)$ is α-Lipschitz continuous and gradient function ∇L is β-Lipschitz continuous. The expected gradient of random function $\mathcal{L}(P_{\theta_p}(\cdot + v), \cdot)$ is $min\{\frac{\alpha}{\sqrt{\rho}}, \beta\}$-Lipschitz continuous.*

The optimal solution for Equation (8) can be approximated by selecting the adversarial noises that maximize the loss function among all noises being unrelated to high-frequency vectors. However, directly optimizing $\mathcal{L}(P_{\theta_p}(\cdot), \cdot)$ fails to guarantee the resulting δ^* to be unrelated to high-frequency vectors. To address this issue, we optimize a proxy loss function $\mathcal{L}(P_{\theta_p}(\cdot + v), \cdot)$ instead. Specifically, Theorem 5.2 shows that the proxy loss function changes more slowly over its domain than the original one and is considered to be more smooth, because the gradient function of the proxy enjoys a smaller Lipschitz constant. Moreover, a smooth function tends to contain fewer high-frequency components, which in turn ensures that the resultant adversarial noises to less unrelated to high-frequency vectors. Figure 4 intuitively demonstrates Theorem 5.2, with a plot of a function composed of a low and a high-frequency component, as well as its proxy versions with varying ρ. As shown in Figure 4, a bigger noise magnitude delivers a stronger smoothing

effect, and, given a fixed magnitude, the suppression effect on high-frequency components is more pronounced, compared with low-frequency ones. In this way, with an approximate noise magnitude, high-frequency components can be effectively removed while preserving low-frequency ones.

In fact, adding Gaussian noises leads to the value of the proxy at a given point being the aggregated value of the original around a small neighborhood centered at the point. As the period of the high-frequency components is short, the neighborhood can be viewed as consisting of several complete periods together with some incomplete ones. According to the fact that the sum of values within a period of a periodic function equals 0, we can consider the average value of the remaining points to represent the function value at that point. With an increase in frequency, the average value of these points decreases, thus allowing for the effective elimination of high-frequency components using this approach. In contrast, the slow change in function value corresponding to the low-frequency component indicates that it is highly preserved within a given neighborhood, making this method effective for preserving the low-frequency component as well.

Based on our analysis, it is clear that Gaussian noises introduced in Equation (2) in fact effectively suppress the high-frequency components of the proxy model. By doing so, the adversarial examples generated by Equation (2) are less related to the high-frequency components, thereby facilitating their transferability across different models. Moreover, there exist some defenses [2] that leverage random smoothing, i.e., adding random noises to inputs, to bolster the robustness of models. Theorem 5.2 shows that models that add random noises to inputs enjoy a smaller Lipschitz constant, which indicates less sensitivity to input changes and thus explains the effectiveness of random smoothing. From this interesting perspective, *CTA* in fact crafts adversarial examples that are threatening against robust models. Intuitively, the adversarial examples crafted via more robust models should own better attack effectiveness [20]. In summary, we grasp why the adversarial examples generated by Equation (2) are more transferable.

6 Conclusion

In this article, we probed the source of the transferability of adversarial ones from the frequency perspective and identified the intrinsic difference term and the perturbation-related term that impairs transfer effectiveness. To mitigate the impact of the perturbation-related term, we formulated an optimization task with a constraint and designed an approximation solution. Moreover, we customized a loss function when using pre-trained models for launching transfer-based attacks. Extensive experiments demonstrated the superior attack performance of *CTA*.

References

[1] Adam Coates, A. Ng, and Honglak Lee. 2011. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In *International Conference on Artificial Intelligence and Statistics*, 215–223.

[2] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. 2019. Certified Adversarial Robustness via Randomized Smoothing. In *International Conference on Machine Learning*. PMLR, 1310–1320.

[3] Ambra Demontis, Marco Melis, Maura Pintor, Matthew Jagielski, Battista Biggio, Alina Oprea, Cristina Nita-Rotaru, and Fabio Roli. 2019. Why Do Adversarial Attacks Transfer? Explaining Transferability of Evasion and Poisoning Attacks. In *28th USENIX Security Symposium (USENIX Security 19)*. USENIX Association, Santa Clara, CA, 321–338. Retrieved from https://www.usenix.org/conference/usenixsecurity19/presentation/demontis

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

[5] Shi Dong, Ping Wang, and Khushnood Abbas. 2021. A Survey on Deep Learning and Its Applications. *Computer Science Review* 40 (2021), 100379 (2021). DOI: https://doi.org/10.1016/j.cosrev.2021.100379

[6] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. 2018. Boosting Adversarial Attacks with Momentum. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9185–9193.

[7] Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. 2019. Evading Defenses to Transferable Adversarial Examples by Translation-Invariant Attacks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4307–4316.

[8] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. 2022. A Survey of Vision-Language Pre-Trained Models. In *International Joint Conference on Artificial Intelligence*, 5436–5443.
[9] John C. Duchi, Peter L. Bartlett, and Martin J. Wainwright. 2011. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization* 22 (2011), 674–701. Retrieved from https://api.semanticscholar.org/CorpusID:1182594
[10] Mingyuan Fan, Cen Chen, Chengyu Wang, and Jun Huang. 2023a. On the Trustworthiness Landscape of State-of-the-Art Generative Models: A Comprehensive Survey. arXiv:2307.16680. Retrieved from https://arxiv.org/abs/2307.16680
[11] Mingyuan Fan, Cen Chen, Chengyu Wang, Wenmeng Zhou, and Jun Huang. 2023b. On the Robustness of Split Learning against Adversarial Attacks. In *European Conference on Artificial Intelligence (ECAI '23)*. IOS Press, 668–675.
[12] Mingyuan Fan, Wenzhong Guo, Zuobin Ying, and Ximeng Liu. 2023c. Enhance Transferability of Adversarial Examples with Model Architecture. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '23)*. IEEE, 1–5.
[13] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. 2019. ImageNet-trained CNNs Are Biased towards Texture; Increasing Shape Bias Improves Accuracy and Robustness. In *International Conference on Learning Representations*. Retrieved from https://openreview.net/forum?id=Bygh9j09KX
[14] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial Examples Are Not Bugs, They Are Features. In *Advances in Neural Information Processing Systems*. H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.), Vol. 32. Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper_files/paper/2019/file/e2c420d928d4bf8ce0ff2ec19b371514-Paper.pdf
[15] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey E. Hinton. 2019. Similarity of Neural Network Representations Revisited. In *International Conference on Machine Learning*, 3519–3529.
[16] Alex Krizhevsky. 2009. Learning Multiple Layers of Features from Tiny Images, University of Toronto.
[17] A. Kurakin, I. Goodfellow, and S. Bengio. 2017. Adversarial Examples in the Physical World. (2017). In *5th International Conference on Learning Representations*, Workshop Track Proceedings.
[18] Jiadong Lin, Chuanbiao Song, Kun He, Liwei Wang, and John E. Hopcroft. 2020. Nesterov Accelerated Gradient and Scale Invariance for Adversarial Attacks. arXiv:1908.06281.
[19] Yuyang Long, Qi li Zhang, Boheng Zeng, Lianli Gao, Xianglong Liu, Jian Zhang, and Jingkuan Song. 2022. Frequency Domain Model Augmentation for Adversarial Attack. *In Computer Vision - ECCV 2022 - 17th European Conference*. Shai Avidan, Gabriel J. Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner (Eds.), Vol. 13664. Springer, 549–566.
[20] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*.
[21] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. 2013. Fine-Grained Visual Classification of Aircraft. arxiv:1306.5151. Retrieved from https://arxiv.org/abs/1306.5151
[22] Yuhao Mao, Chong Fu, Sai gang Wang, Shouling Ji, Xuhong Zhang, Zhenguang Liu, Junfeng Zhou, Alex X. Liu, Raheem A. Beyah, and Ting Wang. 2022. Transfer Attacks Revisited: A Large-Scale Empirical Study in Real Computer Vision Settings. In *2022 IEEE Symposium on Security and Privacy (SP)*, 1423–1439.
[23] Muzammal Naseer, Salman Hameed Khan, M. H. Khan, Fahad Shahbaz Khan, and Fatih Murat Porikli. 2019. Cross-Domain Transferability of Adversarial Perturbations. In *Advances in Neural Information Processing Systems*, 12885–12895.
[24] Yuval Netzer, Tao Wang, Adam Coates, A. Bissacco, Bo Wu, and A. Ng. 2011. Reading Digits in Natural Images with Unsupervised Feature Learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, Vol. 2011. Granada, 4.
[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *International Conference on Machine Learning*, 8748–8763.
[26] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. 2019. On the Spectral Bias of Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning* (Proceedings of Machine Learning Research, Vol. 97). Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 5301–5310. Retrieved from https://proceedings.mlr.press/v97/rahaman19a.html
[27] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. 2020. Do Adversarially Robust ImageNet Models Transfer Better? In *Advances in Neural Information Processing Systems (NIPS '20)*. Curran Associates Inc., Red Hook, NY, Article 298, 13 pages.
[28] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. 2018. Ensemble Adversarial Training: Attacks and Defenses. In *International Conference on Learning Representations*.
[29] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. 2011. *FGVC-Aircraft Benchmark*. Technical Report CNS-TR-2011-001. California Institute of Technology.

[30] Xiaosen Wang and Kun He. 2021. Enhancing the Transferability of Adversarial Attacks through Variance Tuning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 1924–1933.
[31] Xiaosen Wang, Xu He, Jingdong Wang, and Kun He. 2021a. Admix: Enhancing the Transferability of Adversarial Attacks. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 16138–16147.
[32] Xin Wang, Jie Ren, Shuyu Lin, Xiangming Zhu, Yisen Wang, and Quanshi Zhang. 2021b. A Unified Approach to Interpreting and Boosting Adversarial Transferability. In *International Conference on Learning Representations (ICLR)*.
[33] Yilin Wang and Farzan Farnia. 2022. On the Role of Generalization in Transferability of Adversarial Examples. arXiv:2206.09238.
[34] Yajie Wang, Yu-an Tan, Haoran Lyu, Shangbo Wu, Yuhang Zhao, and Yuanzhang Li. 2022. Toward Feature Space Adversarial Attack in the Frequency Domain. *International Journal of Intelligent Systems* 37, 12 (2022), 11019–11036.
[35] Futa Waseda, Sosuke Nishikawa, Trung-Nghia Le, Huy Hoang Nguyen, and Isao Echizen. 2021. Closer Look at the Transferability of Adversarial Examples: How They Fool Different Models Differently. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 1360–1368.
[36] Dongxian Wu, Yisen Wang, Shutao Xia, James Bailey, and Xingjun Ma. 2020. Skip Connections Matter: On the Transferability of Adversarial Examples Generated with ResNets. arXiv:2002.05990.
[37] Cihang Xie, Zhishuai Zhang, Jianyu Wang, Yuyin Zhou, Zhou Ren, and Alan Loddon Yuille. 2019. Improving Transferability of Adversarial Examples with Input Diversity. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2725–2734.
[38] Han Xu, Yao Ma, Hao-Chen Liu, Debayan Deb, Hui Liu, Ji-Liang Tang, and Anil K. Jain. 2020. Adversarial Attacks and Defenses in Images, Graphs and Text: A Review. *International Journal of Automation and Computing* 17, 2 (2020), 151–178.
[39] Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yan Xiao, and Zheng Ma. 2019. Frequency Principle: Fourier Analysis Sheds Light on Deep Neural Networks. arXiv:1901.06523. Retrieved from https://arxiv.org/abs/1901.06523
[40] Zhi-Qin John Xu, Yaoyu Zhang, and Yan Xiao. 2018. Training Behavior of Deep Neural Network in Frequency Domain. In *International Conference on Neural Information Processing*, 264–274.
[41] Zhi-Qin John Xu and Hanxu Zhou. 2020. Deep Frequency Principle towards Understanding Why Deeper Learning Is Faster. In *AAAI Conference on Artificial Intelligence*, 10541–10550.
[42] Jianping Zhang, Weibin Wu, Jen tse Huang, Yizhan Huang, Wenxuan Wang, Yuxin Su, and Michael R. Lyu. 2022b. Improving Adversarial Transferability via Neuron Attribution-based Attacks. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 14973–14982.
[43] Qilong Zhang, Xiaodan Li, Yuefeng Chen, Jingkuan Song, Lianli Gao, Yuan He, and Hui Xue. 2022a. Beyond ImageNet Attack: Towards Crafting Adversarial Examples for Black-box Domains. In *International Conference on Learning Representations (ICLR)*.
[44] Zhengyu Zhao, Zhuoran Liu, and Martha Larson. 2021. On Success and Simplicity: A Second Look at Transferable Targeted Attacks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 6115–6128.

Received 15 September 2023; revised 9 April 2024; accepted 14 July 2024