A Stable and Efficient Data-Free Model Attack With Label-Noise Data Generation

Zhixuan Zhang, Xingjian Zheng, Linbo Qing, *Member, IEEE*, Qi Liu, *Senior Member, IEEE*, Pingyu Wang, Yu Liu, and Jiyang Liao

***Abstract*—The objective of a data-free closed-box adversarial attack is to attack a victim model without using internal information, training datasets or semantically similar substitute datasets. Concerned about stricter attack scenarios, recent studies have tried employing generative networks to synthesize data for training substitute models. Nevertheless, these approaches concurrently encounter challenges associated with unstable training and diminished attack efficiency. In this paper, we propose a novel query-efficient data-free closed-box adversarial attack method. To mitigate unstable training, for the first time, we directly manipulate the intermediate-layer feature of a generator without relying on any substitute models. Specifically, a label noise-based generation module is created to enhance the intra-class patterns by incorporating partial historical information during the learning process. Additionally, we present a feature-disturbed diversity generation method to augment the inter-class distance. Meanwhile, we propose an adaptive intra-class attack strategy to heighten attack capability within a limited query budget. In this strategy, entropy-based distance is utilized to characterize the relative information from model outputs, while positive classes and negative samples are used to enhance low attack efficiency. The comprehensive experiments conducted on six datasets demonstrate the superior performance of our method compared to six state-of-the-art data-free closed-box competitors in both label-only and probability-only attack scenarios. Intriguingly, our method can realize the highest attack success rate on the online Microsoft Azure model under an extremely low query budget. Additionally, the proposed approach not only achieves more stable training but also significantly reduces the query count for a more balanced data generation. Furthermore, our method can maintain the best performance under the existing defense models and a limited query budget.**

***Index Terms*—Deep neural network, data-free, adversarial examples, closed-box attack.**

I. Introduction

DEEP Neural Networks (DNNs) are advanced artificial intelligence technologies which can learn effective features and are widely used in diverse applications [1], [2], [3], [4], [5], [6], [7]. However, recent studies [8], [9] have demonstrated that DNNs are vulnerable to adversarial examples crafted by adding visually imperceptible perturbations to benign examples. These adversarial examples likely mislead the targeted DNNs, referred to as victim models, to generate incorrect predictions. Generally speaking, the creation of an adversarial example is termed an adversarial attack, posing a potentially serious threat to the applications of DNNs in the field of security [10], [11], [12]. Recently, many researchers have endeavored to expose the vulnerabilities of DNNs through adversarial attacks and subsequently propose several systematic approaches for designing robust models.

The early studies on white-box attack [8], [9], [13], [14], [15], [16], [17] assume that the adversary possesses complete knowledge of the victim model, including its parameters, architecture and gradients. These methods often compute adversarial perturbations using the back-propagating gradients of the victim model, showcasing impressive attack capabilities. However, they are impractical for real-world scenarios due to privacy and security concerns surrounding the internal information of victim models. Unlike white-box attacks, closed-box attacks can only yield probability-based or label-based outputs through a given input. Currently, closed-box attacks can be further divided into estimation-based [18], [19] and substitute-based attacks [20], [21]. The former is not feasible in real-world scenarios due to the substantial number of queries required. The latter utilizes adversarial examples crafted on a substitute model, also known as a clone model, to attack the victim model through adversarial transferability [9], [22]. However, the latter does not impose any restrictions on accessing the training dataset of the target model or a semantically similar substitute dataset, as shown in Fig. 1a.

As illustrated in Fig. 1b, in terms of the limitations of real datasets in realistic scenarios, many research works concentrate on data-free closed-box adversarial attacks. The

Received 7 May 2024; revised 10 November 2024 and 9 February 2025; accepted 1 March 2025. Date of publication 11 March 2025; date of current version 25 March 2025. This work was supported in part by Xizang Key Research and Development Program under Grant XZ202501ZY0064, in part by the National Natural Science Foundation of China under Grant 62301346, in part by Sichuan Science and Technology Program under Grant 2024NSFSC1424, in part by Chengdu Technology Innovation Research and Development Project under Grant 2024-YF05-00652-SN, in part by Chengdu Major Technology Application Demonstration Project under Grant 2023-YF09-00019-SN, and in part by the Talent Introduction and Scientific Research Start-Up Project of Sichuan University under Grant YJ202326. The associate editor coordinating the review of this article and approving it for publication was Prof. Luisa Verdoliva. *(Corresponding authors: Linbo Qing; Pingyu Wang.)*

Zhixuan Zhang and Linbo Qing are with the School of Cyber Science and Engineering, Sichuan University, Chengdu 610207, China (e-mail: zhangzhixuan77@gmail.com; qing_lb@scu.edu.cn).

Xingjian Zheng is with Frost Drill Intellectual Software Pte. Ltd., Singapore 079903 (e-mail: xingjian1972zheng@gmail.com).

Qi Liu is with the School of Future Technology, South China University of Technology, Guangzhou 511442, China (e-mail: drliuqi@scut.edu.cn).

Pingyu Wang, Yu Liu, and Jiyang Liao are with the College of Electronics and Information Engineering, Sichuan University, Chengdu 610065, China (e-mail: wangpingyu@scu.edu.cn; lyuup6@163.com; sc_ljy_0116@163.com).

Digital Object Identifier 10.1109/TIFS.2025.3550066

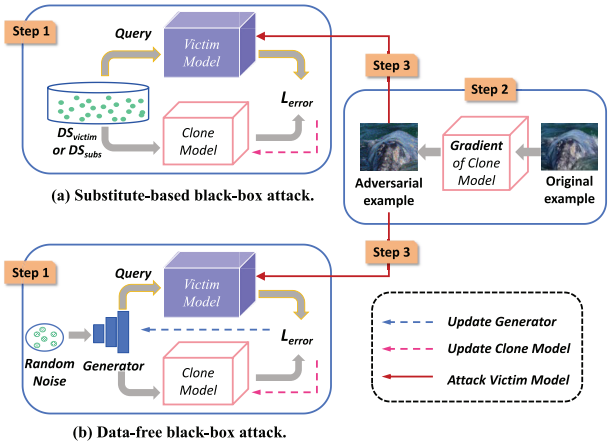


Fig. 1. Visual comparison between substitute-based closed-box attacks and data-free closed-box attacks. Both approaches involve three steps: (1) substitute model learning, (2) adversarial example generation, and (3) attacking the victim model. However, data-free closed-box attacks require no real data at all.

existing literature can be categorized into two main types, i.e., methods [23], [24], [25] based on generative adversarial network [26], [27], [28], and approaches [29], [30] based on knowledge distillation [31], [32], [33]. The former encourages the substitute model to emulate the victim model by a min-max game. The adversarial process, despite its inherent flexibility, often leads to unstable training. The latter involves a two-stage learning process, where data generation is followed by learning the substitute model. This type of methods strongly relies on training generators, while efforts to avoid min-max games should be applied at the same time. In current research, the issue of unstable training may lead to extensive queries (potentially in the hundreds of millions) to the victim model. The aforementioned methods become particularly unfeasible when launching an attack on a security or privacy-focused model with limited query resources. In addition, attack efficiency requires the adversary to achieve successful attacks costing as fewer query counts as possible. However, the problem of low attack efficiency is not only due to unstable training but also attack strategy. Moreover, there exists a highly significant interaction between the unstable training process and the low attack efficiency issues. Specifically, this problem initially impacts surrogate model training and even leads to reduced attack efficiency when dealing with large-scale queries. Consequently, the lack of a well-fitting boundary in the surrogate model inversely hampers generator training, thereby exacerbating the instability of the overall training process. As a result, it is very challenging to simultaneously balance these two issues. The studies in [23], [24], [25], and [34] primarily focus on attack capacity rather than efficiency. Differently, the work in [29] first addresses the issue of unstable training. However, the use of adversarial examples in [29] and [30] significantly impairs the attack efficiency.

In this study, to overcome the limitations of existing methods, we address the issues of unstable training and low attack efficiency simultaneously and propose a novel query-efficient data-free closed-box adversarial attack method, which initiates the knowledge-distillation variant with generator training. In this way, the number of queries to the victim model is greatly reduced. Specifically, for addressing unstable training, focused on directly perturbing the intermediate-layer features, we design the label noise-based generation module to enhance intra-class patterns while preserving layer feature statistics during generator learning. Additionally, we devise the feature-disturbed diversity generation method to increase the distances between different classes. To enhance low attack efficiency by perturbing generated data, we introduce the adaptive intra-class attack strategy. This strategy utilizes entropy-based distance to measure the relationship between positive samples and negative classes, while also considering the impact of positive class probability and the usage of negative samples. Our empirical evaluations on six datasets under both score-only and label-only attack scenarios against six state-of-the-art data-free closed-box competitors demonstrate the superiority of the proposed method. Particularly, our method reaches the highest attack success rate over the online Microsoft Azure model. Furthermore, it not only achieves more stable training but also suites for much lower query budgets and ensures a more balanced data generation. Additionally, it outperforms other baselines with a limited query budget under the existing defense mechanisms. To enhance comprehension of the overall rationale behind our research, we visually present the addressed issues and corresponding proposed methods in Fig. 3, where the former and latter are depicted within the pink and orange regions.

Formally, our main contributions are summarized as follows:

- We introduce a novel query-efficient data-free closed-box adversarial attack method. It is the first attempt to address the issues of unstable training and low attack efficiency simultaneously to our best knowledge.
- For addressing unstable training, by directly perturbing the intermediate-layer features, we design a label noise-based generation module to enhance intra-class patterns and devise a feature-disturbed diversity generation method to increase the distances between different classes.
- To enhance low attack efficiency by perturbing generated data, we introduce the adaptive intra-class attack strategy, by utilization of entropy-based distance and consideration of the impact of positive class probability and the usage of negative samples.
- Our empirical evaluations not only on six datasets but also against the online Microsoft Azure model over six state-of-the-art data-free closed-box competitors demonstrate the superiority of the proposed method. Furthermore, the results prove our methods have the advantages of training models more stably, requiring much lower query budgets, and ensuring more balanced data generation. Even more, it is verified that our method can also evade the existing defense models more easily.

The remainder of the paper is organized as follows. In Section II, background and related works are introduced. The proposed method is stated in Section III. In Section IV, evaluation results show that our method outperforms the competitors. Finally, conclusions and discussions are drawn in Section V.

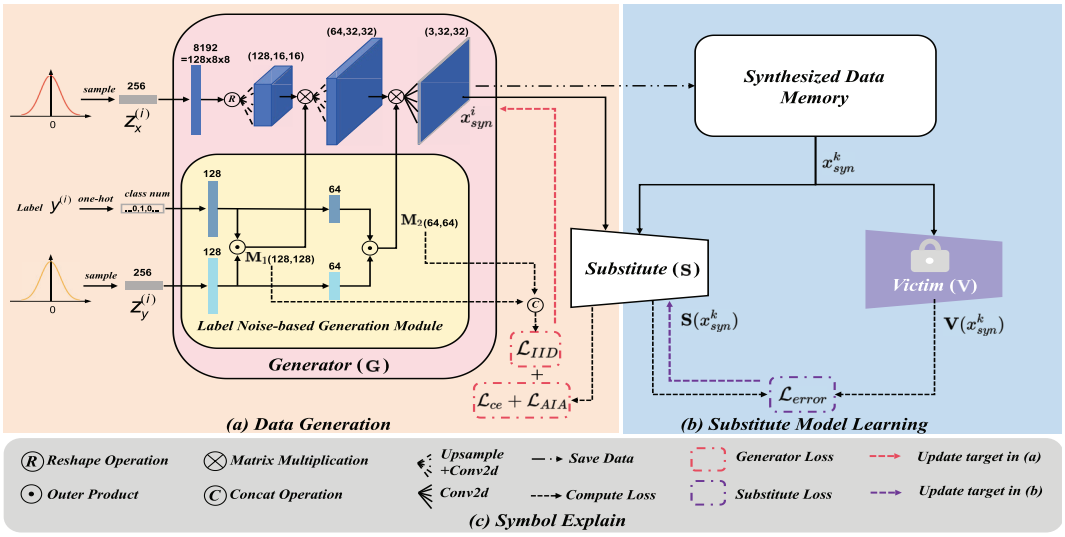


Fig. 2. Overall framework of our proposed method consists of two main components: data generation and substitute model learning. In the data generation phase, we employ our designed label noise-based generation module and feature-disturbed diversity generation method to enhance intra-class patterns using $\mathbf{M}_\ell$ (ℓ=1,2) and increase inter-class distance by maximizing $\mathcal{L}_{IID}$. Notably, our approach focuses on the intermediate-layer feature perspective rather than the output of $\mathbf{G}$ or $\mathbf{S}$. Additionally, we introduce $\mathcal{L}_{AIA}$ from our adaptive intra-class attack strategy. In the substitute model learning phase, we aim to minimize the output difference between $\mathbf{V}$ and $\mathbf{S}$ under both score-only and label-only scenarios.

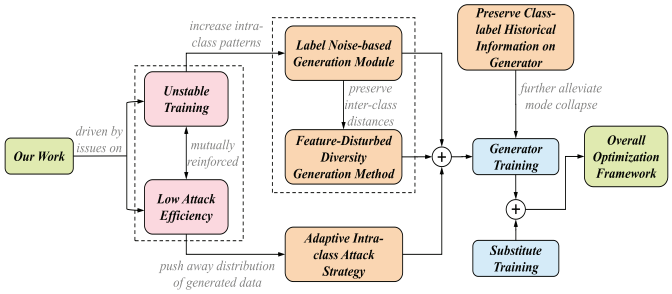


Fig. 3. A diagram to illustrate the whole logical pathway across this work. The pink and orange areas represent the solved problems and proposed methods, respectively.

II. Related Works

A. Adversarial Attack

The remarkable success of DNNs in various tasks and applications has garnered significant attention. The vulnerability of DNNs, particularly in early works like FGSM [8], BIM [13], and PGD [14], has attracted the interest of many researchers. These methods utilize the gradients of the victim model to achieve surprising attack performance and have become standard assessment tools for evaluating network vulnerability. Given an input x, victim model $\mathbf{V}$, true label y and loss function $\mathcal{L}$, the adversarial example x^{adv} in white-box attack under untarget scenario can be formulated as:

$$x_{adv} = x + \min_{\|\sigma\|_p \leq \eta} (-\mathcal{L}(\mathbf{V}(x+\sigma), y)) \tag{1}$$

where σ is the adversarial perturbation, $\|.\|_p$ calculates p norm and η means perturbation constraint. Given a target label $y^t \neq y$, by removing the minus sign and replacing y with y^t, we can straightforwardly reformulate Eq. 1 for the target scenario as:

$$x_{adv} = x + \min_{\|\sigma\|_p \leq \eta} \left(\mathcal{L}(\mathbf{V}(x+\sigma), y^t)\right) \tag{2}$$

However, these white-box attacks are not suitable for the real-world scenario in spite of their excellent attack capacity.

Closed-box adversarial attacks include estimation-based (*e.g.*, ZOO [18], AutoZoom [19]) and substitute-based attacks. To tackle the issue of gradient invisibility in the victim model, the estimation-based attacks employ zeroth-order gradient descent to approximate realistic gradients. However, their direct application is impractical due to limited query budgets. Substitute-based attacks [20], [21] use adversarial examples from a white-box substitute model to mislead the closed-box victim model. This approach is easier to implement but requires access to the training dataset or a semantically similar substitute dataset.

B. Data-Free Closed-Box Attack

Compared to substitute-based attacks, data-free closed-box attacks impose stricter constraints, limiting the adversary to access only the closed-box victim model. This scenario is considered more practical. Zhou et al. [23] pioneer data-free attacks, utilizing a generator for each category in the dataset to train the substitute model. Subsequently, Wang et al. [35] reduce the generator size from [23] using a label-embedding subnetwork and incorporated adversarial examples for substitute model training. Moreover, Sun et al. [25] introduce a label-control generator to achieve attack by uniformly adjusting output probabilities of non-ground-truth classes on correctly classified samples. Additionally, Wang et al. [24] proposes a graph-based learning loss function and a dynamic substitute model to effectively extract information from the victim model. Lastly, Zhang et al. [29] rethink the learning framework as knowledge distillation, involving data generation and substitute model learning. It also utilizes adversarial examples crafted by PGD to approach the boundaries of the substitute and victim models. Similarly,

Yuan et al. [30] focus on model robustness and enable the stealing of both model accuracy and robustness by adversarial training with their proposed high-entropy examples. Whereas, while this method achieves remarkable results, it requires much more queries to the victim model. Most recently, Liu et al. [36] propose to maximize the truncated ratio between positive and negative activations in the middle-layer features of the surrogate model, to strengthen data-free universal attack capability. Despite a similar practical concern (data-free) to ours, our teacher-to-student learning framework is different from it while the universal attack is not our focus. Sequentially, Qian et al. [37] design a substitute structure adaption strategy via an additional policy network, a graph-based loss function through outputs of target and substitute models, and a recycling strategy of hard synthesized examples. Although the dynamic substitute structure adaption strategy is intriguing, it cannot escape from the instability problem in the min-max game while reusing hard synthesized examples may still not address the problem of the tremendous query costs well.

Due to the nature of adversarial training, the aforementioned methods [23], [24], [25], [35] inevitably suffer from the issue of unstable training, which may even damage the attack efficiency. We initiate our learning framework with knowledge distillation like [29] and [30]. However, different from [29] and [30], we do not utilize any adversarial examples and real proxy images at all. Then, to make the generated data distributed more even, we directly manipulate the intermediate-layer features to increase the intra-class diversity and expand the inter-class distances, independent of any specific substitute models. Besides, from the perspective of data generation, we propose a reasonable attack strategy to shift the distribution of the synthesized data and further improve the attack efficiency, instead of simply closing the gap between the victim model and the substitute model at the stage of substitute model learning. Finally, it should be noted that our method feasibly leverages the partial historical information of the generator to make the training convergence faster.

C. Data-Free Closed-Box Knowledge Distillation

In contrast to data-free closed-box adversarial attacks, data-free knowledge distillation focuses on transferring knowledge from a victim model to a substitute model without any access to a training dataset. In the absence of gradients from the victim model, Truong et al. [38] tackle the lack of gradient information by employing zero-order gradient estimation, thus overcoming closed-box limitations. Then, Beetham et al. [34] introduce an additional substitute model to reduce the query count during generator training. This is achieved by giving an upper bound of a gap between the victim model and one substitute model. The upper bound comes from comparing two approximated substitute models. Finally, Zhang et al. [39] utilize the knowledge distillation framework, leveraging the information entropy of the substitute model's output to enhance the diversity of synthesized data.

Different from the reviewed methods, our method does not focus on the gradient information of the victim model. Then, our method can be applied to both label-only and score-only scenarios. Finally, in terms of enhancing the diversity of synthesized data, our method is not related to any specific substitute models.

D. Adversarial Defense Strategies

To eliminate the negative impact of adversarial examples on target models, many works have been devoted to devising various defense strategies for robust models. Tramèr et al. [40] propose adversarial training by incorporating adversarial examples into the training dataset. Subsequently, Xie et al. [41] design a network for Feature Denoising (FD) to suppress features in semantically irrelevant regions. To remove adversarial perturbation, Naseer et al. [42] present Neural Representation Purifier (NRP), which is a self-supervised trained network. Guo et al. [43] apply JPEG compression to images fed into a classifier to remove adversarial perturbations. Xu et al. [44] develop a feature squeezing method that detects adversarial examples by pixel color Bit-depth Reduction (Bit-Red) and applies spatial smoothing.

Typically, evaluating the attack capability in the context of defense strategies can be regarded as a more reliable measure from the adversaries' perspective, signifying a higher level of challenge. Thus, to comprehensively demonstrate the effectiveness of our method, we will conduct a fair comparison between our approach and other baselines under the aforementioned defense strategies.

III. Methodology

In this section, our goal is to solve the problems of unstable training and low attack efficiency as illustrated in Fig. 3. For this, we build a novel data-free closed-box attack framework. Specifically, to address unstable training independently of substitute models, we present a label noise-based module for enhanced intra-class patterns. Then, to avoid the indistinct patterns with the increased intra-class patterns, we design a feature-disturbed diverse generation method to preserve inter-class distances. Additionally, to enhance attack efficiency in the strict data-free scenario, we devise an adaptive intra-class attack strategy. Finally, we propose leveraging class-label historical information on the generator to alleviate mode collapse and further stabilize learning.

A. Framework Overview

To illustrate our methods, we present the framework of our data-free closed-box adversarial attack in Fig. 2. Following the stages similar to [29] and [39], our approach involves two key steps: 1) Data Generation, where the generator $\mathbf{G}$ synthesizes training data, and 2) Substitute Model Learning, where the substitute model $\mathbf{S}$ efficiently imitates the victim model $\mathbf{V}$. Leveraging the transferability of adversarial examples [9], adversarial examples crafted from $\mathbf{S}$ naturally mislead $\mathbf{V}$ into making incorrect decisions.

The distinct features of our approach compared with recent works are: (i) We do not employ proxy images, which are manually selected from external datasets or sources, to ensure stable training. (ii) We hypothesize that incorporating historical information on class-label pattern generation in stage 1 may alleviate the pattern collapse problem. (iii) We consider

both intra-class data diversity and inter-class data distance from the perspective of intermediate layer features. (iv) Our method dispenses with adversarial training to enhance success rates. Instead, we introduce a novel and effective attack strategy in stage 1.

B. Label Noise-Based Generation Module

Recent related works focus on designing generator structures to synthesize class-specific data but often neglect intra-class diversity from the intermediate-layer features of the generator.

To enhance intra-class diversity in synthesized data through the generator structure, we introduce a label noise-based generation module (depicted in the yellow area of Fig. 2a). This module consists of two lightweight networks: E_L creates inter-class diversity, and E_N increases the number of intra-class patterns. Taking the input label $\mathbf{y}$ and label noise $\mathbf{z_y}$ respectively, the outputs of E_L and E_N are $(\mathbf{g}^1_{E_L}, \mathbf{g}^2_{E_L})$ and $(\mathbf{g}^1_{E_N}, \mathbf{g}^2_{E_N})$.

The central element of this module is the matrix M_l, which serves to perturb the features of the subnetwork f at the top of G, thereby enhancing the diversity of data synthesis. Given a batch of input noise $\mathbf{z_x} = (z_x^{(1)}, z_x^{(2)}, \cdots, z_x^{(B)}) \in \mathbb{R}^{B\times nz}$, label $\mathbf{y} = (y^{(1)}, y^{(2)}, \cdots, y^{(B)}) \in \mathbb{R}^{B\times 1}$ and label noise $\mathbf{z_y} = (z_y^{(1)}, z_y^{(2)}, \cdots, z_y^{(B)}) \in \mathbb{R}^{B\times nz}$, where variable B and nz indicate batch size and length of noise vector separately. Then the matrix $\mathbf{M}_\ell$ can be formulated as:

$$\mathbf{M}_\ell = \mathbf{g}^\ell_{E_L} \odot \mathbf{g}^\ell_{E_N} = (\mathbf{g}^\ell_{E_L})^T \mathbf{g}^\ell_{E_N}, \ \ell = 1, 2 \tag{3}$$

where $\mathbf{g}^\ell_{E_L}$ and $\mathbf{g}^\ell_{E_N} \in \mathbb{R}^{B\times 1\times K}$, $K = \{128, 64\}$, and ℓ represents the ℓ-th layer. $\odot$ indicates the outer product operation.

For batch index i, $\mathbf{g}^{\ell(i)}_{E_L}$ naturally signifies the class-related feature (commonly referred to as inter-class diversity), aimed at making the synthesized data as category-wise discriminative as possible. Intuitively, the multiplication of $\mathbf{g}^{\ell(i)}_{E_L}$ by $\mathbf{g}^{\ell(i)}_{E_N}$ serves as perturbation to the class-related feature, effectively augmenting the intra diversity of the class-related feature $\mathbf{g}^{\ell(i)}_{E_L}$. In essence, $\mathbf{M}_\ell$ not only emphasizes the inter-class diversity but also enhances the intra-class diversity.

C. Feature-Disturbed Diversity Generation Method

As more intra-class patterns are generated, unstable training may be alleviated to some extent. However, the presence of indistinguishable patterns still impacts the stability of training. As depicted in Fig. 4, we design a feature-disturbed diversity generation method based on the aforementioned module. This diversity generation method is utilized during both inferring and training. It is noted that the lightweight backbone network in $\mathbf{G}$ (the four blue blocks of Fig. 2a) is the same as that of all above related works except [23].

1) Inferring: We attempt $\mathbf{M}_1 \in \mathbb{R}^{B\times 128\times 128}$ and $\mathbf{M}_2 \in \mathbb{R}^{B\times 64\times 64}$ matrices mentioned above to disturb features of the intermediate layers (the second and third layer) of the top subnetwork f in Fig. 2a. Given features output by the intermediate layers as $f_2(z_x) \in \mathbb{R}^{B\times 128\times W\times H}$ and $f_3(z_x) \in \mathbb{R}^{B\times 64\times W\times H}$, the disturbed feature $\mathbf{g}^{dis}_\ell$ can be computed as follow:

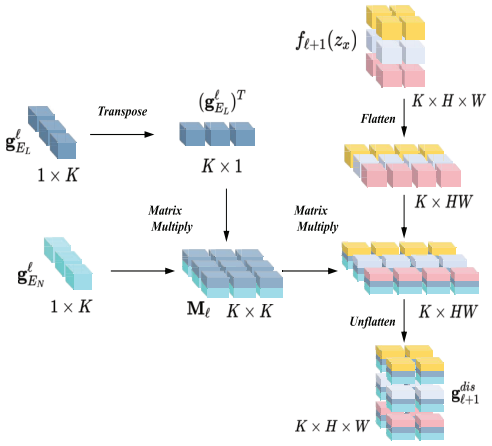


Fig. 4. The detailed processes involved in enhancing the intra-class patterns and perturbing the intermediate-layer features of the generator. The perturbation matrix $\mathbf{M}_\ell$ is generated by enhancing the intra-class patterns. The disturbed feature $\mathbf{g}^{dis}_\ell$ is produced by perturbing the intermediate-layer features $f_{\ell+1}(z_x)$ of generator. K, H and W represent the channel number, height and width of the intermediate-layer feature $f_{\ell+1}(z_x)$ respectively.

$$\begin{aligned}\mathbf{g}^{dis}_\ell &= \mathbf{M}_{\ell-1} \otimes f_\ell(z_x) \\ &= \mathbf{M}_{\ell-1} \times Reshape(f_\ell(z_x), (B, K, W \times H)) \\ \mathbf{g}^{dis}_\ell &= Reshape(\mathbf{g}^{dis}_\ell, (B, K, W, H)), \ \ell = 2, 3\end{aligned} \tag{4}$$

where K, W, and H indicate the channel number, width, and height of the feature, respectively. Additionally, *Reshape*(.) denotes the reshape operation for the dimensional transformation of tensors. $\otimes$ indicates the matrix multiplication along the second and third dimensions.

2) Training: With the enhancement of intra-class patterns, it becomes crucial to expand the inter-class distances. Furthermore, drawing inspiration from the influence of entropy on the uncertainty of data distribution, we introduce a novel disturb-map-based loss, named $\mathcal{L}_{IID}$ (**I**ncreasing **I**nter-class **D**istance). This loss directly operates on the intermediate features of $\mathbf{G}$, such as $\mathbf{M}_\ell$. Importantly, it avoids the back-propagation of gradients of the inter-class diversity loss. Mathematically, this loss can be modeled as follows:

$$\begin{aligned}\mathbf{M}'_\ell &= Reshape(\mathbf{M}_\ell, (B, K \times K)), \ \ell = 1, 2 \\ \mathcal{L}_{IID} &= -\frac{1}{2N_c N_i} \sum_{\ell=1}^{2} \sum_{i=1}^{N_c} \sum_{j=1}^{N_i} \mathbf{M}'^{(i,j)}_\ell \log\left(\left|\mathbf{M}'^{(i,j)}_\ell\right| + \epsilon\right)\end{aligned} \tag{5}$$

where $\mathbf{M}'_\ell$ is the flattened version of $\mathbf{M}_\ell$, N_c and N_i indicate the categorical number and the number of the i-th class within a batch, respectively. $|.|$ represents the absolute value operation. ϵ is a stability factor, set to 10^{-12} in our work.

Due to the existence of distributional variance in $\mathbf{M}_\ell$, it may lead to an unstable training process. Therefore, we choose to smooth $\mathbf{M}'_\ell$ before calculating Eq. 5, as follows:

$$\begin{aligned}\|\mathbf{M}'_\ell\|_2 &= \sqrt{\sum_{i=1}^{K\times K} \mathbf{M}'_\ell(i)^2} \\ \mathbf{M}'_\ell &= \mathbf{M}'_\ell / \|\mathbf{M}'_\ell\|_2, \ \ell = 1, 2\end{aligned} \tag{6}$$

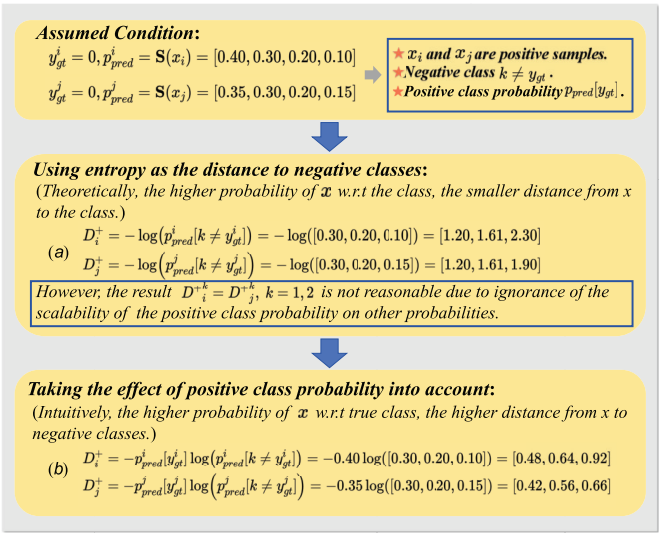


Fig. 5. A mathematical example illustrating the meaning of using entropy-based distance and the impact of positive class probability on negative classes for positive samples.

By maximizing $\mathcal{L}_{IID}$ using Eq. 5, we are able to preserve inter-class differences during the generator training phase. The effectiveness of $\mathcal{L}_{IID}$ will be verified in Section IV. Next, we will explain how to enhance low attack efficiency.

Here, we emphasize that our method distinguishes itself from traditional data augmentation methods in terms of its ultimate objective and operational mechanism. On the one hand, our method is primarily designed to enhance attack efficiency rather than for generative tasks. On the other hand, we address the limitations of traditional data augmentation methods by incorporating considerations for intra-class patterns and mitigating issues related to indistinguishable patterns. Besides, it should be noted that the competitors compared in the following experiments, perturb the intermediate features of the generator through the introduction of random noise or label-related conditional noise, thereby exclusively enhancing inter-class patterns.

D. Adaptive Intra-Class Attack Strategy

In a real-world scenario with limited queries, we opt to reduce queries to $\mathbf{V}$ to eliminate adversarial examples in the training stage of the substitute model. As indicated by [45], it can be observed that solely aligning the distribution of surrogate and victim models may result in overfitting with reduced attack efficiency, while adversarial attacks can be perceived as pushing the learned distribution away from that of the original dataset. Leveraging this perspective, we aim to design an attack strategy where the synthesized data by $\mathbf{G}$ can shift the learned distribution by $\mathbf{S}$ away from that of $\mathbf{V}$. The key challenge is how to guide the generator to achieve this objective.

The proposed closed-box attack.

In Fig. 5, given two input pairs (x_i, y^i_{gt}) and (x_j, y^j_{gt}) with true label y^i_{gt} and y^j_{gt} as the first category, if the output probabilities p^i_{pred} and p^j_{pred} of $\mathbf{S}$ are [0.40, 0.30, 0.20, 0.10] and [0.35, 0.30, 0.20, 0.15], the predicted labels can be correctly classified.

Algorithm 1 The proposed closed-box attack.

Input: Random noise $z_x \in \mathbb{R}^{B \times nz}$; Random label noise $z_y \in \mathbb{R}^{B \times nz}$; Label $y \in \{0, 1, \ldots M-1\}$; Generator $\mathbf{G}$ with parameter $\theta_{\mathbf{G}}$; Target victim model $\mathbf{V}$ with parameter $\theta_{\mathbf{V}}$; Substitute model $\mathbf{S}$ with parameter $\theta_{\mathbf{S}}$; Query budget Q; Batch size B; Generator iterations g_steps; Learning rate $\gamma_{\mathbf{G}}$ of $\mathbf{G}$; Learning rate $\gamma_{\mathbf{S}}$ of $\mathbf{S}$; Minimal loss value $min_{loss_{\mathbf{G}}}$ of $\mathbf{G}$ in every epoch; Data saved into memory, $Data_{save}$; Synthesized data x_{syn} every iteration of $\mathbf{G}$; Output $O_{\mathbf{S}}$ of $\mathbf{S}$; Output $O_{\mathbf{V}}$ of $\mathbf{V}$.

Output: Model parameters $\theta^*_{\mathbf{S}}$.

$N_R \leftarrow Q//B$;
for $i \leftarrow 1$ **to** N_R **do**
  // Data Generation
  $min_{loss_G} \leftarrow 10^6$;
  $Data_{save} \leftarrow$ None;
  $z_x, z_y, y \leftarrow$Random Generation;
  for $j \leftarrow 1$ **to** g_steps **do**
    /* reset $\mathbf{G}$ except BN information in Label Noise-based Generation Module */
    $x_{syn}, \mathbf{M}_1, \mathbf{M}_2 \leftarrow \mathbf{G}(z_x, z_y, y)$;
    $O_{\mathbf{S}} \leftarrow \mathbf{S}(x_{syn})$;
    $\mathcal{L}_{\mathbf{G}} \leftarrow$ Eq. 11;
    $\theta_G \leftarrow \theta_G - \gamma_G \nabla_{\theta_G} \mathcal{L}_{\mathbf{G}}$;
    if $\mathcal{L}_{\mathbf{G}} < min_{loss_{\mathbf{G}}}$ **then**
      $min_{loss_{\mathbf{G}}} \leftarrow \mathcal{L}_{\mathbf{G}}$;
      $Data_{save} \leftarrow x_{syn}$;
    end
  end
  $Data_{save} \rightarrow Memory$;
  // Substitute Model Learning
  $x \leftarrow Memory$;
  $O_{\mathbf{S}} \leftarrow \mathbf{S}(x)$;
  $O_{\mathbf{V}} \leftarrow \mathbf{V}(x)$;
  $\mathcal{L}_{\mathbf{S}} \leftarrow$ Eq. 10;
  $\theta_{\mathbf{S}} \leftarrow \theta_{\mathbf{S}} - \gamma_{\mathbf{S}} \nabla_{\theta_{\mathbf{S}}} \mathcal{L}_{\mathbf{S}}$;
  $\theta^*_{\mathbf{S}} \leftarrow \theta_{\mathbf{S}}$
end

Therefore, these two pairs belong to **positive samples**. We employ entropy as a distance metric to measure the proximity of a sample to a class, exhibiting higher sensitivity towards subtle variations in the data compared to alternative functions such as mean square errors. Moreover, the generation of adversarial examples remains confined within a narrow upper bound around the original samples, rendering it susceptible to even minor fluctuations in the data. Therefore, it is reasonable and natural to incorporate entropy as a distance metric into our work. Theoretically, the higher probability of a sample x w.r.t. the class, the smaller distance from x to the class. Thus to represent the relationship between each positive sample and **negative classes** (all classes except the ground truth), we use entropy to measure the distance from a correct sample to negative classes. The underlying reason for this phenomenon

is an evident yet often overlooked fact that the probabilities of negative classes, rather than the probability of the positive class, can provide more comprehensive distributional information about a sample in relation to the model's classification results. If we only use entropy to calculate the distance, the distance from x_i to the first or second class is equal to that from x_j to the first or second class, as shown in mode (a) of Fig. 5. The result of mode (a) is obviously unreasonable because intuitively, the higher **positive class probability** (equal to the probability of ground truth class), the farther the distance from the sample to negative classes. Therefore, we propose a new description as in mode (b):

$$D^{+k}_{\ i} = -p^i_{pred}[y^i_{gt}] \log(p^i_{pred}[k]),\ k \neq y^i_{gt} \tag{7}$$

where $D^{+k}_{\ i}$ represents the distance from correctly classified sample x_i to the k-th class that is not the true label. Note that Eq. 7 is applicable to positive samples.

Additionally, for negative samples misclassified by $\mathbf{S}$, we should exclude $p^i_{pred}[y^i_{gt}]$ from Eq. 7 since the incorrect $p^i_{pred}[y^i_{gt}]$ is meaningless. This can be reformulated as:

$$D^{-k}_{\ i} = -\log(p^i_{pred}[k]),\ k \neq y^i_{gt} \tag{8}$$

Combining Eq. 7 with Eq. 8, we can adaptively reduce the difference between all D^k_i for the sample x_i synthesized by $\mathbf{G}$ to further push the learned distribution of $\mathbf{S}$ away from that of $\mathbf{V}$. Therein, the difference between all D^k_i for the sample x_i can be more precisely described as variance computation instead of a simple summation of all D^k_i. Thus, we propose our AIA strategy as a loss used during the training of $\mathbf{G}$. Given a batch of samples x, AIA can be expressed as the loss function L_{AIA}:

$$\mathcal{L}_{AIA} = \frac{1}{N_c}\sum_{i=1}^{N_c}\frac{1}{N_i} STD\left(\sum_{j=1}^{N_i}\log\left(\frac{e^{D_j}}{\sum_{k\neq y^j_{gt}} e^{D^k_j}}\right)\right) \tag{9}$$

where $D_j = \{D^k_j \mid k \neq y^j_{gt}\}$ and Eq. 7 or Eq. 8 is plugged into D^k_j in Eq. 9. Here, standard deviation $STD(.)$ is used to calculate variance of all D^k_i.

Notably, the differences between our proposed AIA and [25] are: 1) We use entropy as a distance to describe the relationship between a sample and negative classes more precisely, which is conducive to correctly calculating the bias of D_j. 2) We adopt the prior knowledge that the probability of the positive class influences the distance from the positive sample to every negative class. This is visually shown using a simple demo in Fig. 5. 3) In our AIA, the negative samples are fully utilized instead of being discarded.

E. Optimization

As shown in Fig. 2, our framework consists of two optimization procedures: generator training and substitute model training. The detailed description of our optimization is shown in Algorithm III-D.

1) Substitute Model Learning: In this stage, our goal is to bring the boundaries of $\mathbf{S}$ and $\mathbf{V}$ closer. Therefore, there is no need to involve the true label of synthesized data and adversarial examples. Given input x extracted from the synthesized data memory, the closed-box $\mathbf{V}$ provides a pseudo target output $y_{pse} = \mathbf{V}(x)$ and pseudo target label $y' = \arg\max(y_{pse})$. We can define the error loss function $\mathcal{L}_{error}$ for $\mathbf{S}$ as follows:

$$\mathcal{L}_{error} = \begin{cases} CE(\mathbf{S}(x), y') + MSE(\mathbf{S}(x), y_{pse}), & \text{if Score-Only} \\ CE\left(\mathbf{S}(x), y'\right), & \text{if Label-Only} \end{cases} \tag{10}$$

where CE represents cross-entropy, and MSE indicates mean square error. It's important to note that *Score-Only* and *Label-Only* scenarios are all considered. However, we refrained from utilizing any proxy dataset in our study due to reported uncertainties about its training effect [35].

2) Generator Training: Our primary focus in this stage is to address the issues of unstable training and low attack efficiency to some extent. Moreover, queries towards $\mathbf{V}$ do not occur in this stage. Given inputs $\mathbf{z_x}$, $\mathbf{z_y}$, and $\mathbf{y}$, the generator $\mathbf{G}$ synthesizes data $x_{\text{syn}} = \mathbf{G}(\mathbf{z_x}, \mathbf{z_y}, \mathbf{y})$. Subsequently, x_{syn} is input into the trained $\mathbf{S}$ obtained from the substitute model learning in the last epoch. The total loss for $\mathbf{G}$ is defined as follows:

$$\mathcal{L}_G = \beta_1\mathcal{L}_{ce} - \beta_2\mathcal{L}_{IID} + \beta_3\mathcal{L}_{AIA} \tag{11}$$

where $\mathcal{L}_{ce} = CE(\mathbf{S}(x_{syn}), y)$. The weights β_1, β_2 and β_3 are utilized to adjust the significance of the according item. The default setting of β_1, β_2 and β_3 are 1, 2 and 1.2, respectively.

During g_steps inner iterations in an epoch, we select a batch of x_{syn} with the minimum loss to save into the memory, where the synthesized data is stored in a queue. The mode collapse problem in GANs refers to the generation of single or excessively similar patterns by the generator, often caused by forgetting previously generated patterns. Building upon the successful application of experience replay in mitigating mode collapse in MetricGAN+ [46], we propose leveraging class-label historical information (e.g., preserving the BN layer in the generator after each iteration) to initially generate samples with statistically similar characteristics as those at the previous iteration, thereby alleviating pattern collapse. Interestingly, we have observed that retaining partial historical information of $\mathbf{G}$ can potentially enhance the efficacy of data-free adversarial attacks in the following experiments, in contrast to the recent works where all such information is either cleared or preserved. This observation will be further validated in Section IV.

IV. Experiments

A. Experimental Settings

1) Datasets and Victim Model:

1) MNIST [1]: The victim model is LeNet [1]. The default substitute model is a small CNN used in [23]. 2) FMNIST [47]: The victim model is LeNet [1]. The default substitute model is a small CNN used in [23]. 3) SVHN [48]: The victim model is pre-trained on ResNet-34 [49]. The default substitute model is ResNet-18. 4) CIFAR10 [50]: The victim model is pre-trained on

AlexNet [51], VGG-16 [52], and ResNet-34. The default substitute model is a ResNet-18. 5) CIFAR100 [50]: The victim model is pre-trained on VGG-19, ResNet-34 and ResNet-50. The default substitute model is ResNet-34. 6) Tiny-Imagenet [53]: The victim model is pre-trained on ResNet-50 [49]. The default substitute model is ResNet-34.

2) Competitors and Defense Models: For fair comparisons, we evaluate our method against two types of state-of-the-art approaches: data-free closed-box attacks (*e.g.*, DaST [23], DST [24], TDFE [29] and DFHL-RS [30]) and data-free model extraction attacks (DFME [38], DS [34], and IDEAL [39]). Notably, DFME is not originally intended for label-only scenarios, so we extend it for such scenarios using the cross-entropy loss function to ensure comparability. All experiments are conducted with a fixed query budget Q. As for the defense models, we take ResNet-34$_{ens4}$, ResNet-101$_{ens4}$, MobileNet-1$_{ens4}$ (width = $\times$1) and MobileNet-075$_{ens4}$ (width = $\times$0.75) [40] as adversarially trained models. Other defense strategies under attack contain FD [41], NRP [42], JPEG [43], and Bit-Red [44].

3) Implementation Details: We implement our method using PyTorch [54]. Training is performed on two NVIDIA GeForce GTX 3090 GPUs. We train the substitute model using SGD with an initial learning rate of 0.01, decayed by a factor of 0.3 at 50% of the training epoch for CIFAR10 and CIFAR100 datasets. We use a momentum of 0.9 and weight decay of 0.005 for MNIST, FMNIST, and SVHN datasets, and 0.0005 for CIFAR10, CIFAR100, and Tiny-ImageNet datasets. The mini-batch size is set to 250.

For training the generator, we use the Adam optimizer with an initial learning rate of 0.001, which is decayed by a factor of 0.3 at 30% and 50% of the training process. At the beginning of each epoch, all generator weights except Batch Normalization (BN) layers in the label noise-based generation module are initialized randomly using a truncated normal distribution with a standard deviation of 0.02. We set the mini-batch size to 250 for MNIST, FMNIST, SVHN, and CIFAR10, and 1000 for CIFAR100 and Tiny-ImageNet datasets. For the g_steps parameter, we use 5 for CIFAR10, CIFAR100, and Tiny-Imagenet, and 10 for MNIST, FMNIST, and SVHN.

4) Evaluation Metrics: We employ three common attack methods, namely FGSM [8], BIM [13], and PGD [14]. For FMNIST and MNIST, we set the perturbation bound $\epsilon = 32/255$ and the step size $\alpha = 0.031$. For SVHN, CIFAR10, CIFAR100 and Tiny-ImageNet, we set $\epsilon = 8/255$ and $\alpha = 2/255$. In untarget attacks, we generate adversarial examples only for images correctly classified by the attacked model. In targeted attacks, we concentrate on images that are categorized as specific incorrect labels. The Attack Success Rate (ASR) is calculated as the ratio of successfully fooled adversarial examples to the total number generated. Furthermore, to assess our method's real-world applicability, we test it against the online model of Microsoft Azure.

B. Empirical Studies of Previous Methods

1) Unstable Training: Unstable training poses a persistent challenge in data-free closed-box adversarial attacks. To highlight this issue among competitors, we examine the training loss of **G** and **S**, as well as the accuracy and ASR curves on CIFAR10 over 150 epochs, with ASR calculated using the infinite norm of BIM. To ensure a fair comparison, we constrain the query budget, setting it to 37.5K for distillation methods like IDEAL, TDFE, and our approach, and 2M for other adversarial competitors and DFHL-RS.

As depicted in Fig. 6, it's evident that competitors requiring significantly more queries exhibit unstable training compared to distillation methods. Moreover, our approach showcases a more seamless training process, as indicated by the initial row of Fig. 6. Additionally, our approach achieves faster and higher improvements in ASR and accuracy compared to others. These results underscore the stability of our method even under a limited query budget. This preliminarily indicates that our method not only solves the problem of instability compared to other competitors but also addresses the issue of low attack efficiency.

2) Historical Information for the Generator: In recent data-free closed-box attacks/distillation, some studies [23], [24], [30], [34], [38] treat the process as an adversarial game between the generator and surrogate model, emphasizing the need to retain all historical generator information despite high query costs. They often lead to model collapse under constrained queries. In contrast, other works [29], [39] prioritize a well-trained surrogate model, disregarding historical information of **G** each iteration to avoid complex adversarial dynamics. However, these approaches still face model collapse due to the limited diversity in generated patterns. Inspired by MetricGAN+ [46], we assume that the class-label historical feature statistic information of **G**, such as the statistic (learned parameters) in BatchNorms of our proposed label noise-based generation module, might alleviate the pattern collapse problem. As is known that mode collapse can lead to unstable training even the worse attack and clone performance, the benefit of our assumption to learning can intuitively prove the alleviation of mode collapse.

To validate the significance of this historical information to learning, we represent our approach as one with partial history, compared with two variants: one with full historical information and another without any historical information, on CIFAR10 and FMNIST under a limited query budget $Q = 20K$.

As depicted in Fig. 7, our approach outperforms the other two variants notably on FMNIST and slightly on CIFAR10. Furthermore, Table I demonstrates that our approach achieves a higher attack success rate. This indicates that preserving partial historical information in our **G** is indeed beneficial for training. Simultaneously, these results in Table I and Fig. 7 can support that incorporating historical information on class-label pattern generation in stage 1 may alleviate the problem of pattern collapse. It is noteworthy that the ASR in Fig. 7 corresponds to the infinite norm of BIM, and BIM in Table I refers to the L_2 norm.

C. Attack Results With Competitors

1) Using Various White-Box Adversarial Sample Generation Methods: In both score-only and label-only scenarios, we

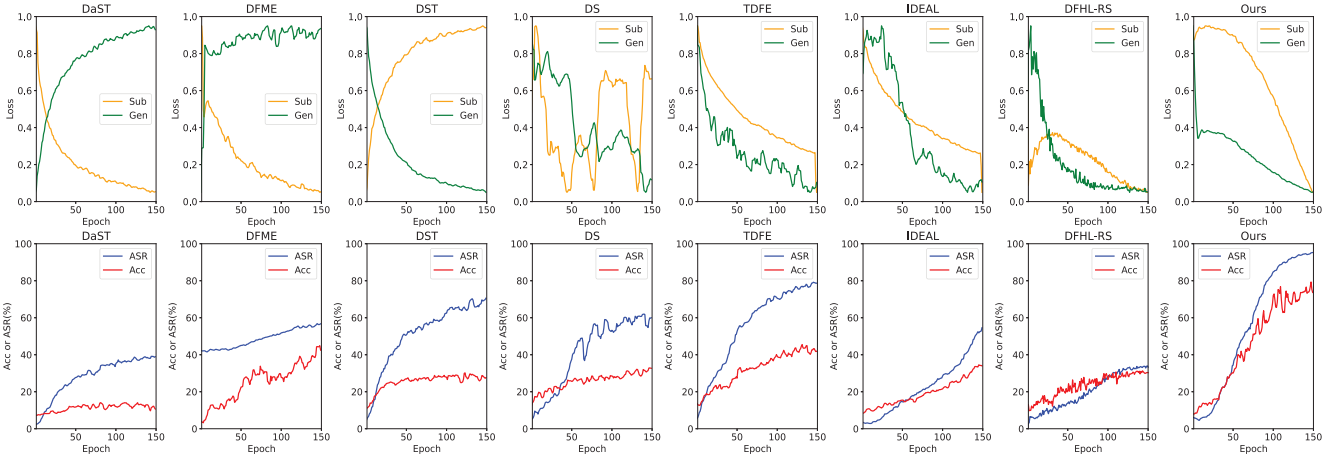
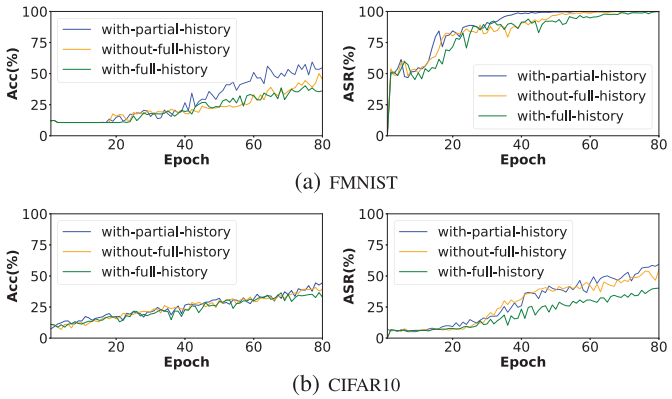


Fig. 6. Analysis of unstable training between our proposed method and other related methods.



(a) FMNIST

(b) CIFAR10

Fig. 7. Validation of the effectiveness of preserving partial historical information of **G** on FMNIST and CIFAR10, comparing variants with and without full historical information of **G**.

TABLE I

PERFORMANCE COMPARISON BETWEEN OUR METHOD AND TWO VARIANTS UNDER THE SCORE-ONLY SCENARIO WITH A QUERY BUDGET OF $Q = 20K$ ON FMNIST AND CIFAR10. **BOLD** INDICATES THE BEST

Type	Target, Score-Only			Untarget, Score-Only		
Method	FGSM	BIM	PGD	FGSM	BIM	PGD
FMNIST						
With full history	38.38	69.63	49.80	96.39	98.93	95.61
Without history	39.94	68.46	51.56	97.75	99.80	97.75
With partial history	**40.62**	**74.61**	**54.76**	**98.44**	**100.0**	**99.12**
CIFAR10						
With full history	4.00	10.35	8.79	31.35	48.63	46.68
Without history	5.66	15.04	13.96	36.04	54.10	52.73
With partial history	**6.45**	**19.34**	**17.97**	**46.09**	**62.89**	**62.01**

compare the ASR of our method with competitors across six datasets under varying query budgets. Specifically, we set the query budget Q to 20K for MNIST and FMNIST, and 250K for the other datasets. We employ three classic attacks to generate adversarial samples over substitute models for attacking the target models on these datasets.

As presented in Table II and Table III, our method almost outperforms all competitors across all datasets in terms of ASR performance. Notably, even in the most challenging label-only and target scenarios, our method demonstrates superior performance over competitors by a significant margin in almost all cases.

TABLE II

PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHODS UNDER A QUERY BUDGET OF $Q = 20K$ ON MNIST AND FMNIST. **BOLD** INDICATES THE BEST

Dataset	Type	Label-Only, Target			Label-Only, Untarget			Score-Only, Target			Score-Only, Untarget		
	Method	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD
MNIST	DaST	1.07	1.17	0.39	7.52	7.91	4.88	0.78	1.07	0.39	7.03	8.11	5.27
	DFME	0.98	0.49	0.10	11.04	3.91	3.32	1.66	0.48	0.39	9.96	3.52	2.44
	DST	1.46	0.49	0.29	14.55	17.38	4.79	3.32	2.73	1.07	11.33	12.50	4.39
	DS	1.27	0.39	0.29	15.23	6.64	3.71	0.39	0.39	0.29	9.57	10.84	5.47
	TDFE	19.43	39.45	9.38	65.23	87.30	42.19	17.38	36.82	8.20	62.40	88.57	42.48
	IDEAL	16.89	40.92	10.06	63.57	91.89	45.12	15.72	37.30	11.62	63.18	87.79	45.41
	DFHL-RS	4.69	6.45	1.86	23.93	31.64	9.38	4.98	8.50	1.76	26.07	28.42	11.04
	Ours	**41.31**	**77.05**	**35.25**	**91.11**	**99.61**	**82.03**	**45.21**	**78.42**	**37.30**	**94.63**	**99.90**	**83.50**
FMNIST	DaST	7.81	2.64	2.44	55.27	54.88	51.56	8.20	2.73	2.05	55.47	54.00	50.78
	DFME	7.91	6.05	4.00	61.62	56.74	50.20	7.13	2.83	4.69	59.77	39.55	40.43
	DST	6.35	4.69	6.05	52.54	57.03	50.88	4.79	3.81	5.57	53.52	53.22	46.88
	DS	3.61	1.66	1.95	59.96	58.40	63.67	5.96	2.15	2.83	60.84	60.25	63.77
	TDFE	28.12	45.41	30.96	93.55	95.90	86.62	25.20	38.57	23.54	91.50	94.43	86.72
	IDEAL	31.15	54.00	30.86	94.14	97.95	89.94	29.39	55.86	36.62	95.41	98.24	91.99
	DFHL-RS	12.99	11.33	9.57	79.39	75.88	67.97	9.77	6.64	6.93	73.63	63.57	59.57
	Ours	**39.45**	**76.86**	**52.73**	**97.56**	**100.0**	**97.95**	**40.62**	**74.61**	**54.76**	**98.44**	**100.0**	**99.12**

TABLE III

PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHODS UNDER A QUERY BUDGET OF $Q = 250K$ ON SVHN, CIFAR10, CIFAR100 AND TINY-IMAGENET. **BOLD** INDICATES THE BEST

Dataset	Type	Label-Only, Target			Label-Only, Untarget			Score-Only, Target			Score-Only, Untarget		
	Method	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD
SVHN	DaST	19.34	42.29	39.45	55.57	66.70	65.33	19.63	41.41	38.18	54.39	65.23	64.26
	DFME	1.20	1.27	1.20	12.01	13.48	13.48	26.95	64.36	61.43	77.25	92.38	92.58
	DST	**29.49**	59.86	61.62	73.63	87.60	87.11	**34.28**	66.21	67.87	79.98	91.70	91.11
	DS	27.25	56.74	53.91	67.77	82.62	82.13	11.43	25.78	24.02	55.66	73.05	72.17
	TDFE	24.02	37.79	34.67	64.39	75.59	75.98	24.90	44.92	41.99	67.97	84.28	83.50
	IDEAL	19.63	35.68	35.72	53.71	72.46	70.21	18.65	39.10	39.14	65.86	85.39	82.95
	DFHL-RS	0.29	0.20	0.49	7.52	7.71	7.81	0.29	0.20	0.49	6.93	6.64	6.84
	Ours	18.65	**68.26**	**67.58**	**78.42**	**92.09**	**91.80**	22.07	**70.31**	**68.58**	**81.35**	**93.95**	**93.75**
CIFAR10	DaST	2.15	1.46	1.95	17.58	19.24	18.36	2.05	1.66	2.05	17.48	18.46	18.07
	DFME	0.88	0.68	1.17	14.94	15.33	15.23	0.88	1.07	0.88	15.92	18.07	18.16
	DST	2.93	5.96	5.47	23.34	28.12	27.44	3.91	7.81	7.91	26.07	33.59	34.47
	DS	0.78	1.37	1.46	15.43	14.75	14.65	1.76	2.15	2.34	18.36	20.51	19.43
	TDFE	24.02	62.40	57.62	76.86	96.39	95.61	24.51	62.99	57.40	77.05	96.39	95.31
	IDEAL	26.17	69.73	64.36	81.74	98.14	97.17	25.39	66.50	62.21	80.27	97.17	96.68
	DFHL-RS	0.49	0.59	0.88	12.30	12.40	12.21	0.59	0.88	0.78	12.21	12.40	12.40
	Ours	**30.57**	**87.40**	**85.16**	**86.52**	**100.0**	**99.90**	**29.88**	**87.79**	**85.64**	**87.21**	**100.0**	**100.0**
CIFAR100	DaST	0.20	0.39	0.59	49.22	48.05	48.54	0.20	0.29	0.10	49.51	47.85	47.46
	DFME	0.49	0.49	0.20	45.70	45.90	45.90	0.68	0.98	0.20	41.80	42.38	42.48
	DST	0.49	1.27	0.98	60.25	74.32	71.58	0.68	1.86	2.05	61.23	79.30	77.05
	DS	0.29	0.29	0.10	44.14	43.85	44.34	0.39	0.39	0.39	50.88	45.70	44.24
	TDFE	3.91	41.60	34.28	81.25	94.14	93.75	4.49	50.68	44.04	83.69	95.70	95.51
	IDEAL	3.61	42.58	35.84	81.93	95.02	95.02	3.22	35.45	29.39	80.86	92.38	92.09
	DFHL-RS	0.20	0.29	0.39	39.75	38.96	38.87	0.39	0.29	0.10	37.70	38.28	38.09
	Ours	**5.96**	**63.48**	**56.64**	**89.45**	**99.41**	**99.22**	**6.15**	**62.99**	**55.96**	**89.45**	**99.61**	**99.51**
Tiny-ImageNet	DaST	0.10	0.20	0.29	45.02	46.97	45.90	0.10	0.20	0.39	45.51	45.70	45.41
	DFME	0.20	0.10	0.10	43.75	43.55	43.65	0.20	0.10	0.20	46.00	45.61	45.31
	DST	0.20	0.10	0.10	46.68	45.21	45.31	0.10	0.10	0.10	44.14	43.65	43.85
	DS	0.10	0.20	0.29	44.73	44.24	44.24	0.39	0.20	0.68	46.39	45.90	45.70
	TDFE	0.29	0.20	0.29	46.97	46.68	46.39	0.39	0.10	0.10	48.83	46.48	46.58
	IDEAL	0.20	0.20	0.29	49.32	46.48	45.61	0.10	0.10	0.29	49.51	46.97	46.78
	DFHL-RS	0.10	0.20	0.20	46.48	47.95	47.56	0.10	0.20	0.10	47.85	47.07	47.27
	Ours	**0.39**	**0.59**	**0.78**	**57.62**	**53.03**	**52.44**	**0.49**	**0.59**	**0.80**	**59.08**	**52.54**	**53.52**

TABLE IV

PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHODS UNDER VARIOUS VICTIM MODELS AND SCORE-ONLY SCENARIO WITH QUERY BUDGET $Q = 250K$ ON CIFAR10 AND CIFAR100 USING PGD. **BOLD** INDICATES THE BEST

	Dataset	CIFAR10			CIFAR100	
	Victim Model	AlexNet	VGG-16	ResNet-34	VGG-19	ResNet-34
Untarget	DaST	16.11	19.82	18.36	51.76	38.76
	DFME	19.92	33.20	17.87	56.84	60.06
	DST	18.85	31.25	33.57	63.67	54.59
	DS	15.62	14.65	18.38	47.27	41.89
	TDFE	67.77	83.30	95.02	90.62	95.98
	IDEAL	66.21	84.96	96.40	89.06	96.19
	DFHL-RS	11.43	13.38	12.42	40.43	42.09
	Ours	**90.43**	**99.12**	**100.0**	**95.61**	**99.71**
Target	DaST	0.88	2.05	2.15	0.59	0.39
	DFME	1.56	3.52	0.78	0.98	0.29
	DST	0.98	1.86	7.25	0.78	0.49
	DS	0.59	0.88	2.39	0.59	0.59
	TDFE	23.24	36.82	57.44	21.78	56.64
	IDEAL	20.41	39.26	62.21	22.46	54.39
	DFHL-RS	0.68	1.07	0.80	0.29	0.29
	Ours	**44.53**	**73.13**	**86.13**	**32.91**	**72.75**

TABLE V

PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHODS UNDER VARIOUS SUBSTITUTE MODELS AND SCORE-ONLY SCENARIO WITH QUERY BUDGET $Q = 250K$ ON CIFAR10 AND CIFAR100 USING PGD. **BOLD** INDICATES THE BEST

	Dataset	CIFAR10			CIFAR100		
	Substitute Model	AlexNet	VGG-16	ResNet-18	AlexNet	VGG-16	ResNet-18
Untarget	DaST	14.26	15.53	17.97	72.66	46.39	64.26
	DFME	12.70	13.57	18.10	39.84	40.23	40.92
	DST	13.09	12.60	34.27	42.87	46.19	71.48
	DS	11.13	9.96	19.56	48.63	45.41	46.78
	TDFE	22.27	23.34	95.40	54.20	44.63	93.55
	IDEAL	33.11	51.66	96.65	57.71	48.73	92.29
	DFHL-RS	13.67	12.40	12.40	46.00	41.99	43.65
	Ours	**94.82**	**97.56**	**100.0**	**93.95**	**87.11**	**99.22**
Target	DaST	1.17	0.68	2.05	0.10	0.29	0.20
	DFME	1.07	0.59	1.02	0.10	0.29	0.10
	DST	0.78	0.49	8.05	0.20	0.30	0.78
	DS	0.10	0.20	2.34	0.39	0.20	0.29
	TDFE	3.91	4.20	57.40	1.07	0.49	36.91
	IDEAL	10.25	13.77	62.12	1.76	0.88	34.38
	DFHL-RS	1.17	0.59	0.78	0.20	0.20	0.10
	Ours	**61.72**	**67.38**	**85.70**	**18.26**	**16.31**	**53.12**

2) Under Various Victim Models: We further validate the effectiveness of our proposed method under a query budget $Q = 250K$ on the CIFAR10 and CIFAR100 datasets across different victim models with various architectures. To ensure a fair comparison, we employ ResNet-18 as the substitute model to craft adversarial examples for attacking all victim models. Additionally, we utilize PGD as the white-box attack in this experiment.

As depicted in Table IV, the ASR of each method fluctuates across different victim models. However, the effectiveness of all competitors is consistently inferior to ours. Particularly on CIFAR100, our method maintains state-of-the-art attack performance.

3) Under Various Substitute Models: We also validate the effectiveness of our proposed method under a query budget $Q = 250K$ on the CIFAR10 and CIFAR100 datasets across different substitute models with various architectures. Besides, we employ ResNet-34 and ResNet-50 as the victim model for CIFAR10 and CIFAR100 respectively. Finally, PGD is used as the white-box attack in this experiment.

As shown in Table V, the ASR of each method changes across different substitute models. Especially, the ASR may decay quickly when the architecture of the substitute model differs from that of the victim model too large. However, our method can still outperform all the competitors by a large margin.

TABLE VI

PERFORMANCE COMPARISON WITH STATE-OF-THE-ART METHODS UNDER A QUERY BUDGET OF $Q = 10K$ ON THE MICROSOFT AZURE ONLINE MODEL. **BOLD** INDICATES THE BEST

Type	Label-Only, Target			Label-Only, Untarget			Score-Only, Target			Score-Only, Untarget		
Method	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD
DaST	5.08	6.64	4.88	54.79	51.27	44.73	4.88	6.74	4.49	55.08	52.54	44.73
DFME	5.08	5.57	3.32	58.01	50.59	46.00	6.84	6.54	5.96	56.93	54.79	46.78
DST	5.47	3.12	2.34	63.28	62.50	56.15	6.64	4.10	3.32	64.16	63.18	56.05
DS	6.15	7.03	4.20	56.54	54.88	47.17	5.86	5.76	4.59	59.47	58.30	52.83
TDFE	45.31	58.98	48.83	97.75	99.41	97.85	44.04	58.59	47.75	99.61	100.0	98.44
IDEAL	39.36	56.74	47.66	98.34	99.90	97.85	40.82	57.91	47.66	99.51	100.0	99.41
DFHL-RS	4.59	5.96	4.88	53.52	45.31	41.80	8.11	6.84	5.86	58.69	51.07	42.87
Ours	**46.09**	**59.18**	**48.83**	**99.90**	**100.0**	**99.80**	**46.19**	**63.38**	**50.49**	**100.0**	**100.0**	**99.90**

4) Against Microsoft Azure Online Model: Evaluating the effectiveness of adversarial example generation methods against real-world closed-box models is crucial. Therefore, we conducted a comparison between our method and other approaches using the online Microsoft Azure model under an extremely low query budget of $Q = 10K$, which conforms to reality. We set a small CNN used in MNIST as the substitute model. Additionally, the hyperparameter g_steps is set as 30.

As illustrated in Table VI, our method outperforms all competitors, demonstrating its practical capacity for closed-box attacks in real-world applications.

All in all, the extensive experiments in this section sufficiently reflect that our method can almost achieve the best attack performance on different datasets with diverse query budget settings, especially against the real-world closed-box model under only 10 K queries limited. Therefore, these results unequivocally establish our method as possessing superior attack efficiency.

D. Query Efficiency Validation

To verify the query efficiency of our method compared to all competitors, we conduct two groups of experiments under the score-only scenario on CIFAR10 within 20 epochs. These experiments correspond to the two columns in the results, where the smaller query is needed every epoch in the first group, and the larger query is set for the other group. Here, we use the infinite norm of BIM to compute the ASR.

It's important to note that since adversarial competitors such as DaST, DFME, DST, and DS themselves require high query budgets such as 2M and 20M, especially the distillation competitor DFHL-RS with 38M, the much lower queries ($1K$ or $10K$) in an epoch, represented as $\times 1K$ or $\times 10K$, might result in their accuracy and ASR curves being close-to-zero straight lines with almost no fluctuation. Specifically, DFHL-RS focuses on enhancing both model accuracy and robustness through adversarial training with high-entropy examples. However, DFHL-RS requires a significantly higher number of queries to achieve satisfactory performance. Thus, for better illustration, we set $\times 50K$ and $\times 500K$ for their two groups, respectively.

Through Fig. 8, it is observed that in the small query group, our method achieves higher accuracy and ASR than other

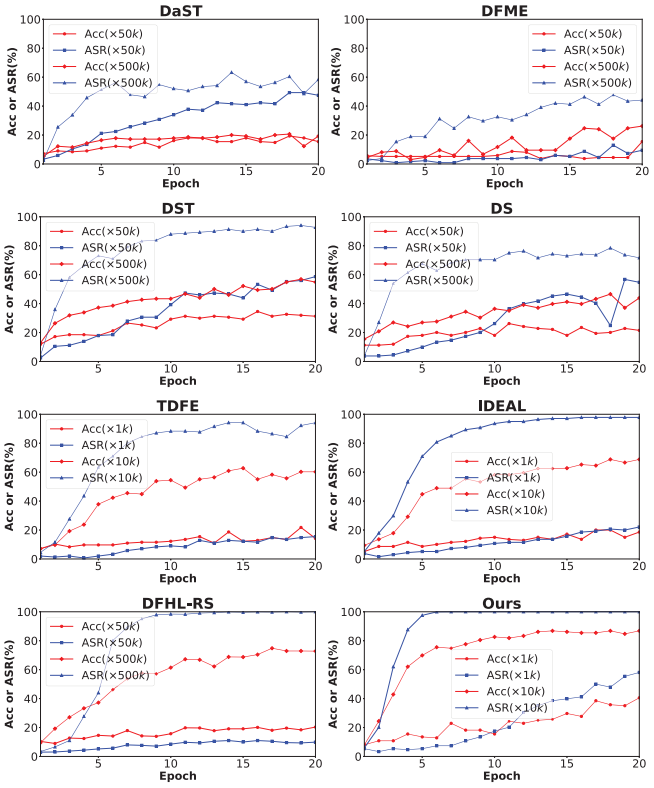


Fig. 8. Comparison of queries to the victim model on CIFAR10 between our method and all state-of-the-art competitors.

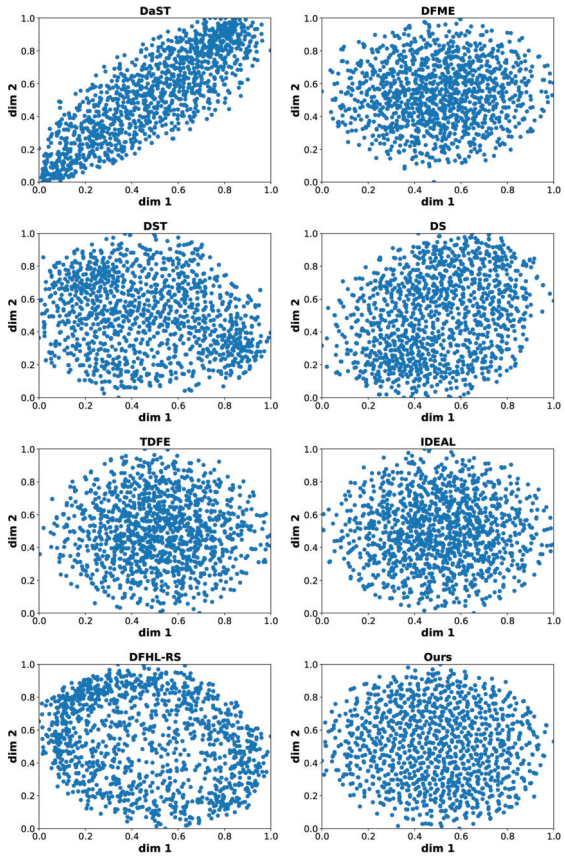


Fig. 9. Visualization of 1250 randomly generated data distributions using t-SNE on CIFAR10 comparing our method with all state-of-the-art competitors.

methods under a query budget of $Q = 20K$. Additionally, in the second column of Fig. 8, these adversarial competitors can barely reach 95% ASR even if they consume 50 times more queries. However, in terms of achieving 95% ASR, our method, TDFE, IDEAL and DFHL-RS expend $50K$, $200K$, $110K$ and 4M queries, respectively. Besides, only our method and DFHL-RS can reach 100% ASR. To achieve 100% ASR, in comparison with DFHL-RS using 6M queries, our method merely consumes $70K$ queries. Clearly, this reflects the prominent efficiency of our method under a lower query budget.

Thereby, the conducted additional experiments on query efficiency as depicted in Fig. 8 reinforce this claim that our method can obtain the higher query efficiency. Besides, the smoother and faster Acc curve in our method under the much less query budget also suggests more stable training compared with other competitors to a large extent.

E. Visual Analysis

1) Uniformity of Synthesized Data: As reported in [24] and [25], we visualize the uniformity of the randomly synthesized data using a data dimension reduction tool t-SNE [55]. Specifically, the primary objective is to ensure that the synthesized data is uniformly distributed across the feature space, thereby theoretically enhancing the training of substitute models [24].

To compare the generative data distribution of our method with that of all competitors on CIFAR10 under a query budget $Q = 250K$, we randomly generate 1250 data points in 5 epochs using the trained generator for each method. We then utilize t-SNE to reduce the dimensionality of the generated data into a two-dimensional space.

The results, shown in Fig. 9, reveal that data produced by all competitors exhibits a significant level of internal coupling, with the data almost centralized in a singular location. In contrast, there is much less overlap between the data generated by our method, indicating that our approach can produce much more evenly distributed data than all competitors. This further validates the effectiveness of our method in data generation.

2) The Classification Distribution of the Trained Substitute Model: To better demonstrate our superior performance, we visualize the classification distribution of the trained substitute model generated by our method and all state-of-the-art competitors.

We adopt ResNet-18 as the substitute model on CIFAR10 and train it within a query budget $Q = 250K$. Utilizing the features extracted from its pre-softmax layer (the final layer before softmax), we apply t-SNE to illustrate the classification distributions of the trained ResNet-18 across the entire testing dataset in a two-dimensional space.

As depicted in Fig. 10, our method exhibits more distinct boundaries between different categories classified by the substitute model compared to all other competitors. This indicates a larger inter-class distance and a smaller intra-class variance. Such observations further validate the effectiveness of our approach in optimizing the substitute model.

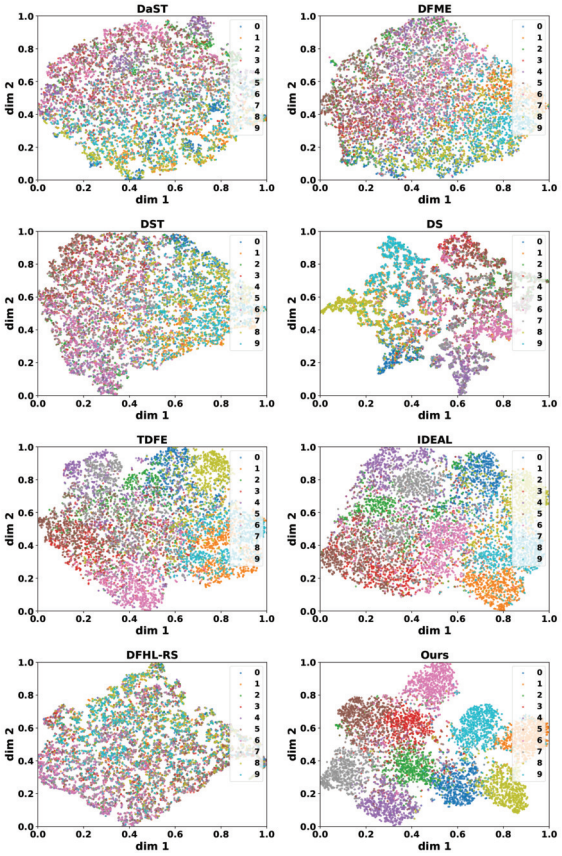


Fig. 10. Comparison of classification boundary over trained substitute models by our method with all state-of-the-art competitors using t-SNE on CIFAR10.

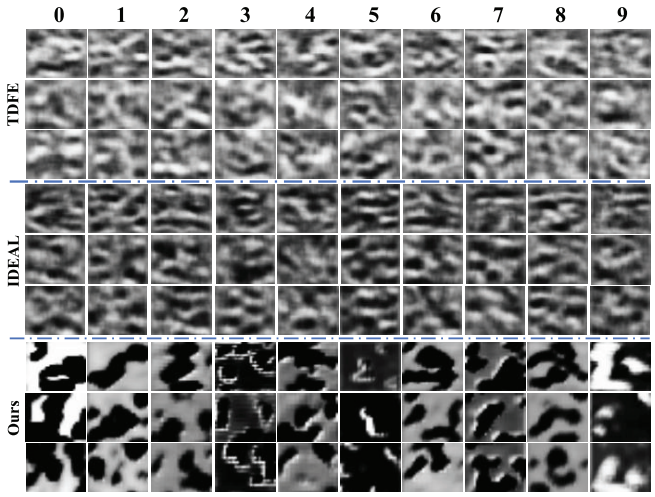


Fig. 11. Samples of generated images under a query budget of $Q = 20K$ on MNIST. Every three rows up to down represent the randomly sampled generation results from the three best methods (i.e., Ours, IDEAL and TDFE). Left to right represents 0–9 hand-written number classes, and the samples belonging to the same column have the same label.

TABLE VII

COMPARISON OF QUALITY CORRESPONDING TO FIG. 11 AMONG THE THREE BEST METHODS (I.E., OURS, IDEAL AND TDFE) UNDER SCORE-ONLY SCENARIO WITH QUERY BUDGET $Q = 20K$ ON MNIST. **BOLD** INDICATES THE BEST

Method	IS_{mean}	IS_{std}	Acc	ASR
TDFE	6.8e-6	6.4e-6	76.60	92.40
IDEAL	3.6e-6	5.7e-6	70.40	94.20
Ours	**1.3e-2**	**1.1e-2**	**93.75**	**100.0**

3) The Quality Analysis of Generated Images: To further verify our generative performance, we not only visualize the diversity of generated data but also show the corresponding quantity results among the three best methods, including ours and other two state-of-the-art competitors (i.e., IDEAL and TDFE). The exclusion of other competitors (i.e., DasT, DFME, DST, DS and DFHL-RS) is due to their significantly low attack efficiency under a limited query budget caused by the huge queries required for victim models as illustrated in Fig. 8

First, we adopt a small CNN and LeNet as the substitute model and victim model on MNIST, and train it within a query budget $Q = 20K$. We randomly sample two generated images for each method to display the synthesized diversity. As well known that the mean (IS_{mean}) and the standard deviation (IS_{std}) of Inception Scores [56] represent the clarity and the diversity of the generated images, we adapted the original Inception Scores on Inception-v3 [57] to align with ResNet-18, considering the single channel representation of MNIST images. And ASR is calculated using the infinite norm of BIM under untarget scenario.

As illustrated in Fig. 11, it can be easily observed that our method obtains more distinguished diverse patterns compared to the other two competitors under the limited query budget. The visual representation effectively illustrates the increased diversity of the images generated by our proposed method. Furthermore, as shown in Table VII, our approach demonstrates superior capabilities in generating high-quality data.

Recall that the diverse patterns generation can alleviate the mode collapse problem and stabilize the learning process, thus it is obvious that our generated data with not only more uniformity and diversity but also superior generative quality for stage 2, actually verify the stronger generated capability of our method. In other words, it indicates that we can mitigate the issue of mode collapse and solve the unstable training problem more successfully.

F. Evaluations on Defense Models

1) Under the Adversarial Training Strategy: Concerned with the query limitation in real-world applications, we evaluate our method on the existing defensive models to further validate our superiority while limiting the query budget to $Q = 250K$. Specifically, we first compare the ASR of our method with competitors in the score-only scenario, over four adversarially trained models on CIFAR10. The competitors selected are TDFE and IDEAL, due to their satisfactory performance with the low query requirement shown in Fig. 8. And the adversarially trained models are ResNet-34_{ens4}, ResNet-101_{ens4}, MobileNet-1_{ens4} (width = $\times 1$) and MobileNet-075_{ens4} (width = $\times 0.75$). They are taken as the other unknown target models to test the transferable attack capacity of adversarial examples, crafted on ResNet-18 as the substitute model and ResNet-34 as the victim model.

TABLE VIII

PERFORMANCE COMPARISON OF ATTACKING ADVERSARIALLY TRAINED MODELS AMONG THE THREE BEST METHODS (I.E., OURS, IDEAL AND TDFE) UNDER SCORE-ONLY SCENARIO WITH QUERY BUDGET $Q = 250K$ ON CIFAR10. **BOLD** INDICATES THE BEST

	Adv-Model	ResNet-34_{ens4}			ResNet-101_{ens4}			$\text{MobileNet-l}_{ens4}$			$\text{MobileNet-075}_{ens4}$		
	Method	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD
Untarget	TDFE	30.76	31.35	29.30	45.21	51.37	51.95	46.48	52.34	51.27	35.25	38.77	37.40
	IDEAL	31.54	33.40	32.13	44.14	51.76	52.15	46.39	55.08	53.81	35.55	36.43	37.11
	Ours	**38.48**	**37.40**	**37.01**	**58.30**	**75.59**	**72.07**	**58.69**	**70.51**	**70.02**	**46.00**	**54.49**	**52.93**
Target	TDFE	4.10	3.42	4.49	7.71	10.64	9.57	7.23	9.86	10.35	5.76	6.64	4.59
	IDEAL	4.49	4.39	5.08	7.32	11.82	10.94	8.30	11.62	12.99	5.47	7.32	5.57
	Ours	**6.74**	**5.96**	**7.13**	**11.52**	**27.05**	**24.22**	**12.11**	**23.44**	**21.29**	**8.79**	**12.30**	**12.11**

TABLE IX

PERFORMANCE COMPARISON OF ATTACKING MODELS USING OTHER DEFENSE METHODS AMONG THE THREE BEST METHODS (I.E., OURS, IDEAL AND TDFE) UNDER SCORE-ONLY SCENARIO WITH QUERY BUDGET $Q = 250K$ ON CIFAR10. **BOLD** INDICATES THE BEST

	Defense Method	FD			NRP			JPEG			Bit-Red		
	Method	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD	FGSM	BIM	PGD
Untarget	TDFE	57.91	77.64	75.88	31.84	52.64	56.64	61.52	71.09	71.68	72.95	90.92	90.14
	IDEAL	55.96	76.86	75.49	30.37	52.64	55.57	63.18	74.71	74.02	72.17	91.02	91.02
	Ours	**70.31**	**94.53**	**94.34**	**43.16**	**74.80**	**78.32**	**75.10**	**84.08**	**86.13**	**83.40**	**99.61**	**99.32**
Target	TDFE	10.16	30.18	28.71	3.91	17.68	18.07	12.70	19.53	19.63	19.82	50.29	49.41
	IDEAL	10.74	33.89	33.50	4.39	15.92	18.36	14.06	22.36	22.75	20.51	52.44	51.37
	Ours	**16.21**	**58.89**	**58.20**	**6.64**	**32.91**	**34.18**	**19.82**	**35.45**	**36.04**	**29.98**	**79.69**	**77.25**

As suggested in Table VIII, our method keeps the same degradation trend of ASR as TDFE and IDEAL compared with Table III. Yet, it is evident that our approach maintains superior attack capability compared to TDFE and IDEAL, even when adversarial examples in adversarial training enhance model robustness to the greatest extent possible.

2) Under the Other Defense Strategies: Except for the famous adversarial training, we also verify our method over the other existing defense strategies, including FD, NRP, JPEG and Bit-Red. Different from the above setting used to attack adversarially trained methods, we set ResNet-34 individually equipped with various defense strategies as the corresponding unknown target models.

As compared with Table III, these existing defense strategies successfully reduce the ASR performance of ours and the two others simultaneously in Table IX. Importantly, despite the more challenging target scenarios, our method still maintains its superior performance over other competitors by a significant margin.

G. Ablation Study

To comprehensively explore the impact of components on our overall framework, we conduct extensive ablation studies over the following variants on CIFAR10 under the score-only scenario with a query budget $Q = 250K$: (1) "$Module^*$": Using the generator without our designed label noise-based generation module. (2) "$\mathcal{L}_{IID}$": Removing the second item in Eq. 11 in training $\mathbf{G}$. (3)"$\mathcal{L}_{AIA}$": Deleting the last item in Eq. 11 in training $\mathbf{G}$. (4) "$\mathcal{L}_{AIA}(A)$": Without considering the efficacy of positive class probability in Eq. 9. (5) "$\mathcal{L}_{AIA}(B)$": Discarding negative samples used in Eq. 9. (6) "$\mathcal{L}_{AIA}(C)$": Only using the entropy-based distance description in Eq. 9. (7) "D": Omitting the entropy-based distance description based on (6). In this experiment, we still leverage PGD as the white-box attack to calculate ASR under both target and untarget scenarios. Notably, several ablation studies in (3)(4)(5)(6)(7) are used to systematically validate the efficacy of our proposed Adaptive Intra-class Attack Strategy.

TABLE X

ABLATION STUDY CONDUCTED UNDER VARIOUS CONDITIONS AND THE SCORE-ONLY SCENARIO WITH A QUERY BUDGET OF $Q = 250K$ ON CIFAR10 USING PGD

Without Condition							Target	Untarget
$Module^*$	$\mathcal{L}_{IID}$	$\mathcal{L}_{AIA}$	$\mathcal{L}_{AIA}(A)$	$\mathcal{L}_{AIA}(B)$	$\mathcal{L}_{AIA}(C)$	D		
							85.64	**100.0**
✓							79.00	98.34
	✓						84.69	99.80
		✓					82.81	99.90
			✓				85.35	99.90
				✓			81.15	99.61
					✓		82.23	99.90
						✓	81.25	99.61

As shown in Table X, by comparing the results among the variants, several observations become apparent: (1) Without our label noise-based generation module, the attack effectiveness severely decays, especially in the target attack scenario, highlighting the significance of intra-class diversity. (2) Increasing the intra-class patterns influences the attack performance beyond just expanding the inter-class distance. (3) The attack strategy we designed is more effective for improving the attack performance in more challenging target scenarios. (4) Consideration for the positive class probability does contribute slightly to the attack effectiveness. (5) Negative samples provide more patterns to enhance the attack capability. (6) Under conditions where no negative samples are available, and the generator may fall into overfitting, the usage of positive class probability may inadvertently exacerbate this problem. (7) The idea of the entropy-based distance description notably improves the attack effectiveness.

The ablation study conducted above individually assesses the effectiveness of each component within our method. It further explains that our work achieves addressing the two problems including the unstable training and the low attack efficiency.

V. CONCLUSION

In this paper, we address the challenges posed by a limited query budget setting, which manifests as unstable training and low attack efficiency. To tackle unstable training, we introduce a label noise-based generation module that enhances intra-class patterns by leveraging partial historical information of $\mathbf{G}$. However, an abundance of intra-class patterns can blur the boundaries between classes. Hence, we propose a feature-disturbed diversity generation method to amplify the inter-class distance by maximizing our novel loss $\mathcal{L}_{IID}$. Furthermore, to enhance low attack efficiency within the query budget constraints, we devise a novel attack strategy that meticulously considers the entropy-based distance description, the influence of positive class probability, and the utilization of negative samples in data-free adversarial attacks. Extensive evaluations validate the effectiveness of our approach.

REFERENCES

[1] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[2] L. Deng, G. Hinton, and B. Kingsbury, "New types of deep neural network learning for speech recognition and related applications: An overview," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Feb. 2013, pp. 8599–8603.

[3] W. Yu et al., "Learning from inside: Self-driven Siamese sampling and reasoning for video question answering," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, Dec. 2021, pp. 26462–26474.

[4] H. Qiu, B. Yu, D. Gong, Z. Li, W. Liu, and D. Tao, "SynFace: Face recognition with synthetic data," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10860–10870.

[5] Q. Liu, X. Li, H. Cao, and Y. Wu, "From simulated to visual data: A robust low-rank tensor completion approach using lp-regression for outlier resistance," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 6, pp. 3462–3474, Jun. 2022.

[6] Q. Liu and X. Li, "Efficient low-rank matrix factorization based on $l_{1,\varepsilon}$-norm for online background subtraction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 7, pp. 4900–4904, Jul. 2022.

[7] Z. Zhang and Q. Liu, "Spike-Event-Driven deep spiking neural network with temporal encoding," *IEEE Signal Process. Lett.*, vol. 28, pp. 484–488, 2021.

[8] I. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. Int. Conf. Learn. Represent.*, Dec. 2014, pp. 1–7.

[9] C. Szegedy et al., "Intriguing properties of neural networks," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2014, pp. 1–11.

[10] S. Wang, Z. Zhang, G. Zhu, X. Zhang, Y. Zhou, and J. Huang, "Query-efficient adversarial attack with low perturbation against end-to-end speech recognition systems," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 351–364, 2023.

[11] H. Ma, K. Xu, X. Jiang, Z. Zhao, and T. Sun, "Transferable black-box attack against face recognition with spatial mutable adversarial patch," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 5636–5650, 2023.

[12] Z. Sun, Y. Ren, Y. Huang, W. Liu, and H. Zhu, "AFPM: A low-cost and universal adversarial defense for speaker recognition systems," *IEEE Trans. Inf. Forensics Security*, vol. 19, pp. 2273–2287, 2024.

[13] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Int. Conf. Learn. Represent. (Workshop)*, 2017, pp. 1–16.

[14] A. Mądry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2017, pp. 1–8.

[15] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "DeepFool: A simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2574–2582.

[16] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1765–1773.

[17] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z. B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," in *Proc. IEEE Eur. Symp. Secur. Privacy*, Mar. 2016, pp. 372–387.

[18] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh, "ZOO: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models," in *Proc. 10th ACM Workshop Artif. Intell. Secur.*, Nov. 2017, pp. 15–26.

[19] C.-C. Tu et al., "AutoZOOM: Autoencoder-based zeroth order optimization method for attacking black-box neural networks," in *Proc. 33rd AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 1, pp. 742–749.

[20] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, "Practical black-box attacks against machine learning," in *Proc. ACM Asia Conf. Comput. Commun. Secur.*, Apr. 2017, pp. 506–519.

[21] T. Orekondy, B. Schiele, and M. Fritz, "Knockoff Nets: Stealing functionality of black-box models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 4954–4963.

[22] J. Dong, Y. Cong, G. Sun, Z. Fang, and Z. Ding, "Where and how to transfer: Knowledge aggregation-induced transferability perception for unsupervised domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 3, pp. 1664–1681, Mar. 2021.

[23] M. Zhou, J. Wu, Y. Liu, S. Liu, and C. Zhu, "DaST: Data-free substitute training for adversarial attacks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 234–243.

[24] W. Wang, X. Qian, Y. Fu, and X. Xue, "DST: Dynamic substitute training for data-free black-box attack," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 14341–14350.

[25] X. Sun, G. Cheng, H. Li, L. Pei, and J. Han, "Exploring effective data for surrogate training towards black-box attack," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15355–15364.

[26] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2017, pp. 1–17.

[27] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of Wasserstein GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, Dec. 2017, pp. 5769–5779.

[28] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. Int. Conf. Learn. Represent.*, 2016, pp. 1–9.

[29] J. Zhang et al., "Towards efficient data free blackbox adversarial attack," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 15094–15104.

[30] X. Yuan, K. Chen, W. Huang, J. Zhang, W. Zhang, and N. Yu, "Data-free hard-label robustness stealing attack," in *Proc. AAAI Conf. Artif. Intell.*, Mar. 2024, vol. 38, no. 7, pp. 6853–6861.

[31] R. Gontijo Lopes, S. Fenu, and T. Starner, "Data-free knowledge distillation for deep neural networks," 2017, *arXiv:1710.07535*.

[32] H. Yin et al., "Dreaming to distill: Data-free knowledge transfer via DeepInversion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8715–8724.

[33] H. Chen et al., "Data-free learning of student networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3514–3522.

[34] J. Beetham, N. Kardan, A. Mian, and M. Shah, "Dual Student networks for data-free model stealing," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2023, pp. 1–16.

[35] W. Wang et al., "Delving into data: Effectively substitute training for black-box attack," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4761–4770.

[36] Y. Liu, X. Feng, Y. Wang, W. Yang, and D. Ming, "TRM-UAP: Enhancing the transferability of data-free universal adversarial perturbation via truncated ratio maximization," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 4739–4748.

[37] X. Qian, W. Wang, Y.-G. Jiang, X. Xue, and Y. Fu, "Dynamic routing and knowledge re-learning for data-free black-box attack," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 47, no. 1, pp. 486–501, Jan. 2025.

[38] J.-B. Truong, P. Maini, R. J. Walls, and N. Papernot, "Data-free model extraction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 4771–4780.

[39] J. Zhang, C. Chen, J. Dong, R. Jia, and L. Lyu, "IDEAL: Query-efficient data-free learning from black-box models," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2022, pp. 1–15.

[40] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2017, pp. 1–22.

[41] C. Xie, Y. Wu, L. Van Der Maaten, A. L. Yuille, and K. He, "Feature denoising for improving adversarial robustness," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 501–509.

[42] M. Naseer, S. Khan, M. Hayat, F. S. Khan, and F. Porikli, "A self-supervised approach for adversarial robustness," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 259–268.

[43] C. Guo, M. Rana, M. Cissé, and L. Van Der Maaten, "Countering adversarial images using input transformations," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2017, pp. 1–12.

[44] W. Xu, D. Evans, and Y. Qi, "Feature squeezing: Detecting adversarial examples in deep neural networks," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2018, pp. 1–15.

[45] Y. Zhu et al., "Toward understanding and boosting adversarial transferability from a distribution perspective," *IEEE Trans. Image Process.*, vol. 31, pp. 6487–6501, 2022.

[46] S. Fu et al., "MetricGAN+: An improved version of MetricGAN for speech enhancement," in *Proc. Annu. Conf. Int. Speech Commun. Assoc.*, Aug. 2021, pp. 201–205.

[47] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.

[48] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop*, Jan. 2011, pp. 1–9.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[50] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, Tech. Rep. 1.4, 2009.
[51] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 60, May 2017, pp. 84–90.
[52] K. Simonyan, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, Jan. 2015, pp. 1–14.
[53] Y. Le and X. Yang, "Tiny ImageNet visual recognition challenge," *CS 231N*, vol. 7, no. 7, p. 3, 2015.
[54] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, Jan. 2019, pp. 1–17.
[55] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–27, 2008.
[56] S. Barratt and R. Sharma, "A note on the inception score," 2018, *arXiv:1801.01973*.
[57] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

Zhixuan Zhang received the M.S. degree in computer technology from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2021. He is currently pursuing the Ph.D. degree in cyberspace security with Sichuan University, Chengdu. His research interests include adversarial attacks, spiking neural networks, and image processing.

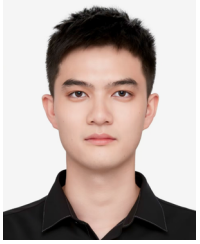
Xingjian Zheng received the M.S. degree in science from the National University of Singapore in 2022. He was with A*STAR, for two years, and he is currently a free Researcher. His research interests include signal processing, image processing, and generative models.

Linbo Qing (Member, IEEE) received the B.S. degree in electronic information science and technology and the Ph.D. degree in communication and information system from Sichuan University, China, in 2003 and 2008, respectively. He is currently a Professor with the College of Electronics and Information Engineering, Sichuan University. His research interests include artificial intelligence and computer vision, image processing, visual computing, data mining, and digital health. He is a member of Chinese Institute of Electronics.

Qi Liu (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the City University of Hong Kong, Hong Kong, China, in 2019. From 2018 to 2019, he was a Visiting Scholar with the University of California Davis, Davis, CA, USA. From 2019 to 2022, he was a Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He is currently a Professor with the School of Future Technology, South China University of Technology. His research interests include human-object interaction, AIGC, 3D scene reconstruction, and affective computing. He was a recipient of the Best Paper Award of IEEE ICSIDP in 2019. He has been an Associate Editor of the IEEE SYSTEMS JOURNAL (since 2022) and *Digital Signal Processing* (since 2022). He was also the Guest Editor of the IEEE INTERNET OF THINGS JOURNAL and *IET Signal Processing*.

Pingyu Wang received the Ph.D. degree from Beijing Key Laboratory of Network System and Network Culture, Beijing University of Posts and Telecommunications, Beijing, China, in 2021. From 2021 to 2023, he was a Senior Algorithm Researcher with Tencent Technology (Beijing) Company Ltd. The techniques he has developed/involved have been shipped to several products in Tencent, such as WeChat, QQ, Tencent Video, and ZenVideo. He is currently an Associate Research Fellow with the College of Electronics and Information Engineering, Sichuan University, Chengdu, China. His research interests include image processing, computer vision, multi-modality learning, and deep learning.

Yu Liu received the bachelor's degree from the College of Electrical and Mechanical Engineering, Chengdu University of Technology, Chengdu, China, in 2022. He is currently pursuing the master's degree with the School of Electronic and Information Engineering, Sichuan University, Chengdu. His research interests include the fields of computer vision and urban perception.



Jiyang Liao received the B.S. degree from the College of Electronics and Information Engineering, Sichuan University, Chengdu, China, in 2022, where he is currently pursuing the M.S. degree. His research interests include scene graph generation, human-object interaction recognition, and social relationship recognition in the field of computer vision.