Unstoppable Attack: Label-Only Model Inversion via Conditional Diffusion Model

Rongke Liu, Dong Wang, Yizhi Ren, Zhen Wang, Kaitian Guo, Qianqian Qin, Xiaolei Liu

***Abstract*—Model inversion attacks (MIAs) aim to recover private data from inaccessible training sets of deep learning models, posing a privacy threat. MIAs primarily focus on the white-box scenario where attackers have full access to the model's structure and parameters. However, practical applications are usually in black-box scenarios or label-only scenarios, i.e., the attackers can only obtain the output confidence vectors or labels by accessing the model. Therefore, the attack models in existing MIAs are difficult to effectively train with the knowledge of the target model, resulting in sub-optimal attacks. To the best of our knowledge, we pioneer the research of a powerful and practical attack model in the label-only scenario.**

In this paper, we develop a novel MIA method, leveraging a conditional diffusion model (CDM) to recover representative samples under the target label from the training set. Two techniques are introduced: selecting an auxiliary dataset relevant to the target model task and using predicted labels as conditions to guide training CDM; and inputting target label, pre-defined guidance strength, and random noise into the trained attack model to generate and correct multiple results for final selection. This method is evaluated using Learned Perceptual Image Patch Similarity as a new metric and as a judgment basis for deciding the values of hyper-parameters. Experimental results show that this method can generate similar and accurate samples to the target label, outperforming generators of previous approaches.

***Index Terms*—Model inversion attacks, Diffusion model, Deep learning security and privacy, Generative model-based attack model.**

I. Introduction

THe technology of artificial intelligence is developing rapidly and its application brings many conveniences to our daily life nowadays. For example, in the field of image recognition, deep neural networks (DNNs) assist in applications such as face and fingerprint recognition, biomedical diagnosis, and extracting useful information from massive datasets of images. However, building such a model requires massive data for training, which may contain private or sensitive information. Some studies have noted that models tend to memorize the training data [1] [2] [3] [4].

The model inversion attack (MIA) is a type of privacy attack that aims to regenerate data to represent training data, input data, or sensitive attributes by accessing the target model [5] [6] [7]. For instance, the face image of a target individual in the training set can be recovered by a face recognition model, similarly, the input face image can be reconstructed from the prediction vector produced by the face recognition model. Or sensitive attributes of an individual's genome can be inferred by a medical prediction model [5]. Unlike membership inference [8] and model extraction attacks [2], MIA concentrates on recovering data closely resembling the private data itself. Existing model inversion attacks can be classified into two types based on the attacker's background knowledge: the white-box and the black-box scenarios.

In the white-box scenario, the attacker has full access to the structure and parameters of the target model, while in the black-box scenario, the attacker can only access the predictions (confidence vectors or labels) of the target model without providing gradient or other information. For the face recognition model, the existing white-box attacks [6] iteratively optimize the input image by feeding a noise image to the target model and minimizing the loss between the prediction and target label, achieved through a gradient descent algorithm. Whereas, in the attack scenarios where the target model is usually a DNN (e.g., convolutional neural networks (CNNs)), the sensitive features to be recovered often lie in a high-dimensional, continuous data space. Directly optimizing over the high-dimensional space without any constraints may generate unrealistic features lacking semantic information [6] [9]. In order to obtain more semantic and meaningful images on CNNs [10], state-of-the-art methods [9] [11] [12] generate the image by feeding noisy vectors to the generator in a generative adversarial network (GAN) [13] trained with an auxiliary dataset, therefore the optimization turns to the noisy vectors and generator.

In contrast, the existing black-box attack [7] [14] is to optimize a generator by gradient descent to minimize the pixel loss between the generated and auxiliary images. The generated images are produced by the aforementioned generator based on the predictions of the target model for the auxiliary data. In addition, based on the black-box model output confidence or label, the attacks can be categorized into "data reconstruction" and "training class inference" [7]. The "data reconstruction" is mainly based on the prediction confidence vector of the target model to recover the input samples [7], while "training class inference" is based on the one-hot vectors or labels to recover the representative samples of the target [7] [14]. Since this paper studies label-only scenarios, our research belongs to the

Manuscript received 1 August 2023; revised 4 December 2023 and 14 February 2024. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Tondi, Benedetta. *(Corresponding author: Dong Wang.)*

Rongke Liu, Dong Wang, Yizhi Ren, Zhen Wang, Kaitian Guo and Qianqian Qin are with the School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: liurk@hdu.edu.cn; wangdong@hdu.edu.cn; renyz@hdu.edu.cn; wangzhen@hdu.edu.cn; guokt@hdu.edu.cn; qinqq@hdu.edu.cn)

Xiaolei Liu are with the Institute of Computer Application, China Academy of Engineering Physics, Mianyang 621900, China (e-mail: luxaole@gmail.com)

Digital Object Identifier 10.1109/TIFS.2023.xxxxxxx

“training class inference”.

Current research yielding ideal generation results primarily focuses on white-box scenarios. However, models are typically accessed as black boxes in practical applications and only output predicted labels. This prevents the generator in the state-of-the-art white-box attack from optimizing with the help of gradient and prevents decoupling the potential space of the target class [12]. Moreover, there are several limitations to existing black-box attacks:

1) The generated images [15] [7] [14] [16] [17] [6] are mostly grey-scale, which cannot accurately determine the color characteristics of the target, such as skin tone or pupil color.
2) Since there is currently no attack model that can be optimally trained in the label-only scenario, existing methods [18] [19] [20] had to reach the attack goal by designing additional optimization strategies based on the generator from ordinary GAN, which is trained without the target model’s knowledge.
3) Only a single sample can be generated for the target label [7].
4) The generated results for the target label based on the generator are sub-optimal and evaluation metrics lack comprehensiveness. A comparison of the specific results and evaluation metrics can be seen in Figure 1 and Table I.

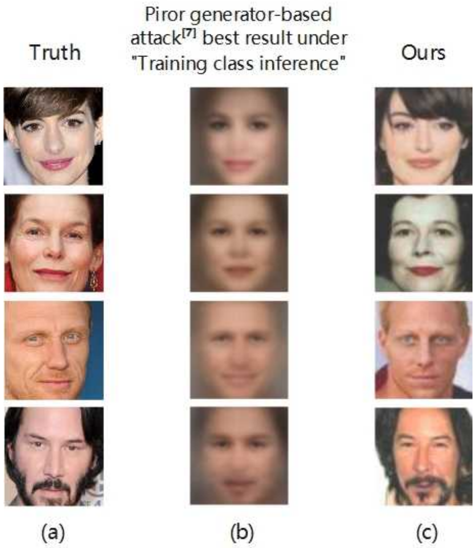
Truth
Piror generator-based attack[7] best result under "Training class inference"
Ours
(a)
(b)
(c)

Fig. 1. Training class inference of our and previous approaches against a facial recognition classifier in the label-only scenario. For better comparison with our method, we turned the labels into correct one-hot vectors to train (b)’s attack model [7] for recovering optimal color images. Note that the correct one-hot vector implies that the confidence value at the target position is 1, while the rest are 0. For instance, if there are a total of 3 classes and the target is class 1, then the one-hot vector would be (1,0,0). Moreover, the auxiliary dataset used by both is the same.

TABLE I
COMPARISON OF EVALUATION METRICS IN WORK RELATED TO BLACK-BOX MODEL INVERSION ATTACKS, WHERE ★ MEANS THAT THE WORK INCLUDES THIS METRIC.

	Attack accuracy	Feature/Pixel-level similarity	Perceptual similarity/ Quantification of qualitative assessment
[17]		★	
[14]		★	
[18]	★		
[15]	★	★	
[19]	★	★	
Ours	★	★	★

In this paper, we develop a novel label-only model inversion attack method to address the above limitations. The core design idea of our method is to train a conditional diffusion model (CDM) [21] guided by the target model predicted label, and in the recovery phase, various samples can be recovered for selection based on the target label guidance. Since our attack method only needs the target model to predict the labels and existing defensive strategies [22] [23] [24] [25] should ensure the availability of the model, this attack is unstoppable. Table II visualizes the strengths and uniqueness of our study.

Specifically, we first select an auxiliary dataset that is relevant to the target model task. For example, if the training set utilized by the target model is a facial dataset, then the auxiliary dataset will correspondingly be facial. Secondly, we assign the predicted labels produced by the target model for the corresponding auxiliary data. These predicted labels can reflect the target model’s judgments about various types of target features in the training set. Then, we train the CDM, which consists of forward diffusion and backward prediction [26], to add Gaussian noise to the auxiliary data in the forward diffusion process. This process eventually makes the image close to the standard normal distribution noise image. In the backward prediction process, the added noise is predicted under the guidance of the predicted label, which allows the diffusion model to learn the noise distribution added by the target under the prediction label. After training, we feed random standard normal distribution noise images and the target label into the CDM to recover images with a pre-defined guidance strength. Due to the difference with the traditional CDM training process, the auxiliary data of this model for a specific target comes from public data, and the real labels do not correspond to the actual data, which makes the difference in the learned noise distribution and leads to poor noise reduction in the generated images. Accordingly, we perform gamma correction [27] on the generated images to make them more consistent with human visual judgment. Finally, we randomly change multiple generated images, submit them to the target model for prediction, and then select the top-k robust generated images. Experiments show that our attack model can generate more accurate, realistic, and similar images than existing attack models in the label-only scenario.

As shown in Table I, in the field of MIA, evaluation metrics are not standardized, and commonly used measures such as attack accuracy and feature distance may not accurately reflect the quality of generated results. For instance, an over-fitted and semantically meaningless optimized image may still exhibit high attack accuracy and low feature distance. We argue that

TABLE II
STRENGTHS AND UNIQUENESS OF OUR STUDY COMPARED TO RELATED WORK, WHERE ★ MEANS THAT THE WORK INCLUDES THIS CAPABILITY.

Capability	White-box		Black-box			
	GMI [9]	PLG [12]	BERP [18]	RL-MIA [19]	LB-MIA [7]	**Ours**
Access to the target model only output label			★			★
Attack methods focus on generative models	★	★			★	★
No need to obtain gradient information from the target model to attack			★	★	★	★
The generator does not require additional optimization strategies to achieve targeted attacks		★			★	★

a qualitative evaluation may be more important than a quantitative one for this work [6] [28] [15]. As such, we propose the use of Learned Perceptual Image Patch Similarity (LPIPS) [29] as a novel evaluation metric for MIA. LPIPS approximates human judgment of similarity between two sets of data and can serve as a proxy for qualitative evaluation. Our results demonstrate that our method is capable of generating accurate and similar data to target labels in the label-only scenario and outperforms the previous attack models regardless of the individual overlap between the auxiliary set and the training set.

Contributions. In summary, we make the following contributions to this paper:

- We pioneer a new MIA attack model that effectively leverages target model knowledge even in label-only scenarios. Our experiments validate the practicality and effectiveness of our approach.
- We propose to utilize gamma correction to address the degradation of generation quality due to differences in auxiliary data under the same target.
- We can use the target model to filter out multiple generated samples under the same target. Compared with the existing methods in the label-only scenario, which can only generate a unique sample for the same target, this method is more fault-tolerant and has a large optimization space.
- We conduct a systematic evaluation of our attack in terms of attack accuracy, similarity, and realism, both quantitatively and qualitatively. Our experimental results demonstrate that our attack model can generate more accurate, realistic, and similar target samples in the label-only scenario, regardless of the individual overlap between the auxiliary set and the training set.

II. Related Work and Preliminary Knowledge

Privacy attacks on machine learning and deep learning models can be categorized into three types: model extraction attacks [2], inference attacks [8], and model inversion attacks [6]. Model extraction attacks aim to infer the parameters or features of the target model to replicate a similar machine-learning model, while inference attacks aim to reveal information related to the training set of the target model. Model inversion attack aims to regenerate data to represent training data, input data, or sensitive attributes by accessing the trained target model. Among them, the MIA reveals privacy information at a finer level.

A. Traditional Model Inversion Attacks

Fredrikson et al. [5] were the first to propose a model inversion attack, using warfarin dose personalization [30] as a case study to show how MIA can infer patient-specific genetic markers by accessing linear regression models as well as maximum posterior probabilities. Subsequently, Fredrikson et al. [6] extended the attack to decision trees [31] and face recognition neural networks using confidence and gradient descent algorithms. However, both works explored black-box attacks, but [5] assumed too strongly on the knowledge held by the adversary, and [6] required 50-80 days for experiments on multi-layer perception network or denoising autoencoder network to complete, i.e., estimating gradients for optimization. In addition, this type of attack does not successfully recover the training set of DNNs.

B. Generative Model Inversion Attacks

To address the limitations that the recovered results are often non-semantic or meaningless when facing DNNs, Zhang et al. [9] first proposed to train GAN [13] using fuzzy or incomplete training set data to generate target label samples. Chen et al. [11] proposed to improve the training of GAN using public auxiliary data, and with the help of soft labels predicted by the target model. Yuan et al. [12] further proposed to use of the target model to provide pseudo-labels and guide the training of conditional GAN [32] to optimize the latent space and decouple the generated data. However, the above methods are all white-box attacks and therefore may not be practical for real-world scenarios.

To address this issue, Yang et al. [7] resorted to an autoencoder (AE) [33] architecture to train an inversion model as a decoder to reconstruct the input data and perform data encoding with the target model as the encoder. In addition to these approaches, Zhu et al. [14] and Kahla et al. [18] investigate the direction of label-only model inversion attacks. Zhu et al. used the target model error rate to estimate the confidence of predictions and trained an attack model based on the framework of Yang et al. While Kahla et al. do not adopt the architecture of AE, train a GAN using a public dataset and propose a gradient estimation algorithm to estimate the true gradient, and optimize the potential input vector. Similar research only conducted for optimization algorithms continues to evolve. For example, Dionysiou et al. [15] proposed the use of evolutionary algorithms [34], and Han et al. [19] used reinforcement learning [35] to optimize for input noise. However, the attack model in these related works is nothing more than the inversion model [7] or the same GAN [9] [18] [19].

Currently, there is no powerful attack model other than the inversion model that can be trained using the target model in the label-only scenario. To the best of our knowledge, we are the first to investigate the practicable and powerful attack model in the label-only scenario.

C. Diffusion Models

Diffusion Models (DMs) were first introduced in 2015 by Sohl-Dickstein et al. [36]. These models are inspired by non-equilibrium statistical physics [37] and work by systematically destroying the structure in a data distribution through an iterative forward diffusion process. The model then learns the reverse diffusion process to restore the data structure, resulting in highly flexible and easy-to-handle data generation models. In 2020, Ho et al. [26] simplified this approach by proposing Denoising Diffusion Probabilistic Models (DDPMs). These models use parameterized Markov chains [38] trained by variational inference to generate samples matching the data after a finite amount of time. Dhariwal et al. [39] later built on this work by proposing that prediction of noise through specific classifiers can guide sample generation. They demonstrated through systematic experiments that DM outperforms GAN. Ho et al. [21] then proposed "classifier-free diffusion guidance", which aims to jointly train a conditional and unconditional diffusion model. The conditional diffusion model (CDM) is guided by the true labels, and the resulting conditional and unconditional diffusion models are combined to achieve a trade-off between sample quality and diversity. This is similar to the results obtained using classifier guidance.

Zheng et al. [40] first used pre-trained diffusion models for MIA, and they adopted the idea of previous white-box attacks, i.e., minimizing the cross-entropy loss between the target model prediction of the generated image and the true label, to optimize the x_t in the noise reduction path. However, the final result of the attack makes the generation results unsatisfactory even though the classification accuracy is guaranteed. Therefore, a complete evaluation system is essential for MIA.

Combining label-only MIA and CDM, it was found that diffusion models guided by labels can be well applied to attack models in label-only MIA. However, unlike traditional diffusion models, the attacker will not have access to real training data and labels. To better reflect the knowledge behind the target model, an attack method was designed in Section IV.

III. Threat Model

In real attack scenarios, the target models are usually DNNs (e.g., face recognition models), and the sensitive data (face images) to be recovered often lie in a high-dimensional, continuous data space. However, recent works show that MIAs could even successfully reconstruct high-dimensional data, such as images [6] [9]. In this study, we also adopt the convolutional neural network image classification model as our target model. Our focus is on the label-only scenario, where the adversary has access only to label predictions $F_W(x)$, obtained by inputting an image x into the target model F_W.

Attack goal. Given access to a target model $F_W : [0,1]^d \rightarrow L$, the attacker aimed to regenerate the representative samples $\tilde{x}$ of the training dataset of the target label L; d represents the dimension of the input; L represents the predicted label, and "representative samples" means the generated images are similar to the target individual.

Task Knowledge. We assume that the attacker knows the task of the target model, e.g., a face classification model. This assumption is reasonable as this information is typically available from the network or can be inferred through direct access to the target model.

Data Knowledge. Based on the above assumptions, it is then reasonable to assume that the attacker can construct similarly distributed auxiliary datasets D_{aux}. Previous work has assumed that there is no target class overlap between the two datasets, but we argue that the possibility of target class overlap in the image recognition domain exists if an attacker can combine a large amount of auxiliary data for an attack. We discuss the impact of this scenario on the attack results in Section VI, but we weaken the assumption that the attacker does not know how many classes or images overlap, i.e., the attacker does not know which classes are overlapped.

IV. Attack Method Design

A. Overview of Our Method

This section details the design of our proposed method. As illustrated in Figure 2, our approach comprises two primary phases: the training phase and the recovery phase.

During the training phase, we train a generator for model inversion attacks. This involves the following steps:

- **Step 1:** Selecting an auxiliary dataset D_{aux} that is relevant to the target model task.
- **Step 2:** Inputting $\mathbf{x}_0 \in D_{aux}$ into the target model F_W yields the predicted labels $F_W(\mathbf{x}_0)$.
- **Step 3:** Training a conditional diffusion model G_θ for the attack using the auxiliary dataset D_{aux} from step 1 and employing the prediction labels $F_W(\mathbf{x}_0)$ from step 2 as conditions to guide training.

Our method improves existing MIA by using CDM as the attack model which is tailored for label-only scenarios. Instead of traditional CDM training, we use the target model's predicted label for auxiliary data as guidance. This aligns the auxiliary data under the predicted label with the target model's decision for the target, helping the attack model learn consistent features of the target from the data.

During the recovery phase, we use the trained generator to recover target label data. This involves the following steps:

- **Step 1:** Inputting multiple standards normally distributed noise images $\mathbf{z}$ and target attack label l into the trained conditional diffusion model to recover images $G_\theta(\mathbf{z})$ of this label with a pre-defined guidance strength. Unlike traditional CDM, there is a quality impact on the generated data since the data under identical guidance is not the

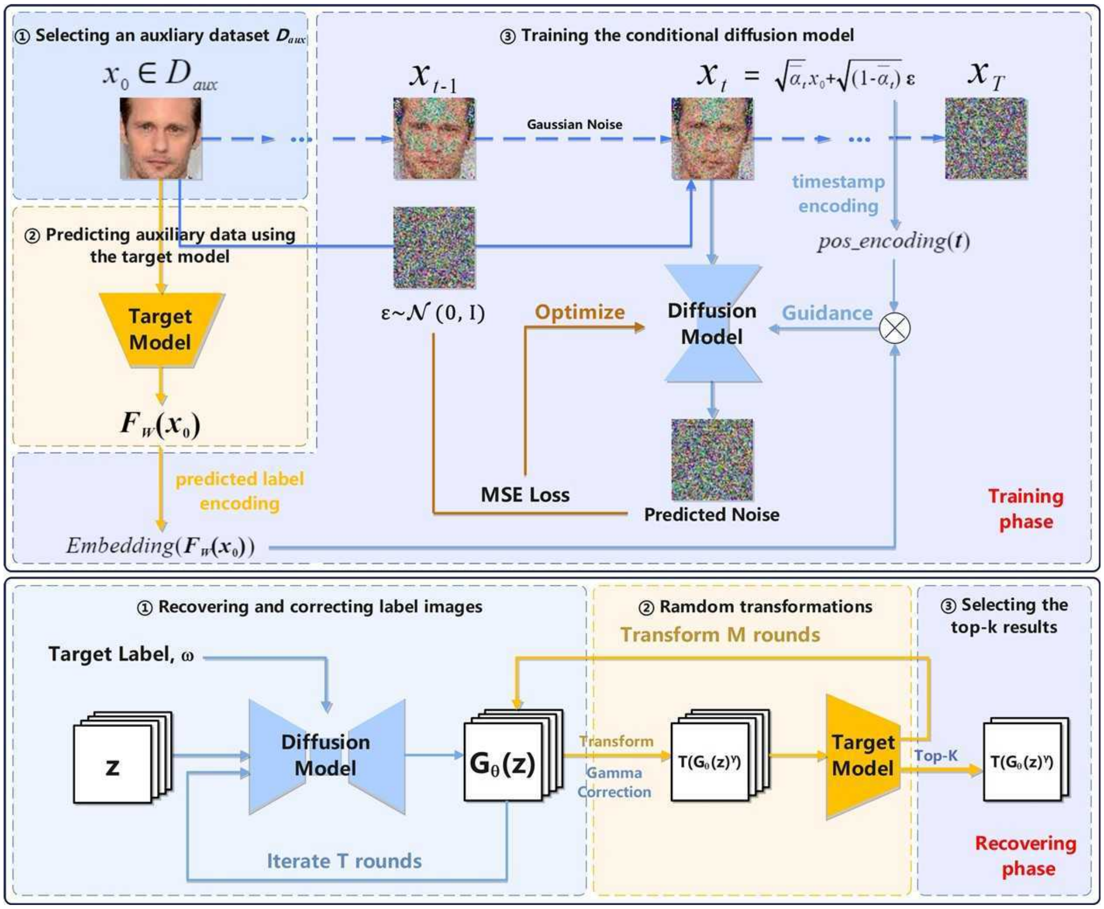

Fig. 2. The attack overview of the proposed label-only model inversion attack method.

same entity. To solve this issue, we use gamma correction to correct the generated images to $G_\theta(\mathbf{z})^\gamma$.

- **Step 2:** Transforming the corrected generated image $G_\theta(\mathbf{z})^\gamma$ randomly into $T(G_\theta(\mathbf{z})^\gamma)$ and inputting it into the target model for prediction, then repeating this step M times.
- **Step 3:** Selecting the top-k robust generated images from $T(G_\theta(\mathbf{z})^\gamma)$, i.e., the top-$k$ images that still predict the target label with the highest ratio among M random changes. In contrast to the low fault-tolerance limitation of existing attack models, which can only generate a single image for the target, our approach is not only powerful but also has unique advantages.

B. The Training Phase

This section presents a detailed analysis of each step in the training phase.

1) Selecting an auxiliary dataset: To achieve better generator results, it is essential to choose an auxiliary dataset that closely resembles the training set in terms of data distribution and task relevance. Previous research [28] has demonstrated that the auxiliary dataset significantly affects the results of the generator. For instance, if the target model is for face recognition but the generator is trained on an oil painting face dataset, the inversion attack results will produce unsatisfactory oil painting images of the target face. Additionally, we opted for a larger-scale public dataset and preprocess to extract a considerable amount of feature information. The generated results can approximate private ones by learning from these public features. For instance, for face image recovery, the auxiliary data should include only facial regions to avoid background influence and enable direct learning by the generator.

2) Predicting auxiliary data using the target model: In previous black-box attacks, the assistance of the target model in training a powerful generator was often overlooked due to technical and scenario limitations. As a result, these researches focused on optimization strategies. The conditional diffusion model is a type of generator that does not require backpropagation of gradient optimization after loss calculation through label and prediction. However, real training data and labels cannot be obtained. Therefore, auxiliary data $x_0 \in D_{aux}$ selected in the first step is fed into the target model F_W for prediction, to obtain the target model's classification task labels $F_W(x_0)$. This reflects the target model's judgment on the feature of the training data. For a simple instance, if the target model is a "0-4" handwritten digit classifier, then input digits "7, 9" will have a high probability of outputting "1". Thus, we can learn "1" by "7, 9".

3) Training the conditional diffusion model: We train the conditional diffusion model G_θ using the data and labels provided in the preceding two steps. In reference to previous work [26] [21], the training procedure for the conditional diffusion model comprises two primary stages: forward diffusion and backward prediction. During forward diffusion, Gaussian noise is added to the auxiliary data x_0 T times, ultimately resulting in standard normally distributed noise. At each step, Gaussian noise is added to the data x_{t-1} obtained in the previous step as follows:

$$q\left(x_t \mid x_{t-1}\right)=\mathcal{N}\left(x_t ; \sqrt{1-\beta_t} x_{t-1}, \beta_t \mathbf{I}\right) \tag{1}$$

Where β_t represents the noise variance at each step, ranging from 0 to 1. A linear variance schedule is employed to customize the variance at each step, resulting in x_t being generated at each step of the diffusion process and the entire process is fixed as a Markov chain, as shown below:

$$q\left(x_{1: \mathrm{T}} \mid x_0\right)=\prod_{t=1}^{\mathrm{T}} q\left(x_t \mid x_{t-1}\right) \tag{2}$$

Given this property, we only need to sample and train the t_{th} step of the training process. Based on the first two formulas, we derive the following equation:

$$q\left(x_t \middle| x_0\right)=\mathcal{N}\left(x_t ; \sqrt{\bar{\alpha}_t} x_0,\left(1-\bar{\alpha}_t\right) \mathbf{I}\right) \tag{3}$$

Where $\alpha_t = 1 - \beta_t$, and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$, thus $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{(1-\bar{\alpha}_t)} \varepsilon$, $\varepsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This means that the noise data x_t obtained after adding t times of noise can be directly derived from the original auxiliary data x_0. Since the variance increases linearly, the limit of $\bar{\alpha}_t$ approaches 0 as long as T becomes sufficiently large, resulting in x_{T} being close to standard normally distributed noise.

Backward prediction involves predicting the noise added during forward diffusion. This is accomplished using a U-Net [26] neural network model comprising downsampling blocks, upsampling blocks, and attention blocks. The input is x_t, the added noise is predicted and optimized using $D_{KL}(q\left(x_{t-1} \mid x_t, x_0\right) \parallel p_\theta(x_{t-1} \mid x_t))$, where $p_\theta(x_{t-1} \mid x_t)$ represents the predicted noise distribution and $q\left(x_{t-1} \mid x_t, x_0\right)$ denotes the real posterior distribution. Denoising diffusion probabilistic models (DDPM) experiments [26] have demonstrated that calculating the Mean Squared Error (MSE) loss between the predicted noise and random Gaussian noise yields better optimization results. Thus, during the training phase, we only need to calculate $\bigtriangledown_\theta \left\|\varepsilon-\varepsilon_\theta(x_t, t)\right\|^2$where ε represents the Gaussian noise added from step 0 to step t, and the $\varepsilon_\theta(x_t, t)$ denotes the noise predicted by the generator G_θ based on x_t, t.

To ensure that the training process is guided by the predicted labels, we incorporate labels encoded via the Embedding function of PyTorch into each timestamp. The timestamp t is initially encoded via the position encoding function [41] to a fixed dimension, and then the label is encoded into the same dimension by the Embedding function, which in turn sums the two. This is followed by a time embedding layer at each sampling block of the diffusion model, which is synchronized with the hidden layer feature dimension of the current block through the repeat function. The time embedding layer comprises a SiLU activation layer and a Linear layer. The specific semantics of this approach is to enable the diffusion model to learn the noise distribution added to the predicted target $F_W(x_0)$. Therefore, noise reduction can be carried out during the recovery phase by progressively embedding the target labels into the timestamps. This process ultimately results in the generation of outcomes specific to that target.

To ensure the training of the conditional diffusion model does not over-fit the label information, a certain probability p is introduced. This allows the training to optimize G_θ without the need for label guidance, i.e., $\bigtriangledown_\theta \left\|\varepsilon-\varepsilon_\theta[x_t, (\textit{pos\_encoding}(t)+\textit{Embedding}(F_W(x_0)))]\right\|^2$ or $\bigtriangledown_\theta \left\|\varepsilon-\varepsilon_\theta(x_t, t)\right\|^2$. Consequently, the attack model learns not only from the labels but also from the inherent features and structure of the data. This approach improves the generalization capability of the attack model, thereby ensuring the diversity and authenticity of the generated data. The method for determining the value of probability p is elaborated in Section VI.

To summarize, during the training process, we initially assign each auxiliary data a random timestamp t that falls between 1 and T. Following this, x_t can be computed from Equation (3) after iterative noise addition from x_0 after t steps. Subsequently, the encoded predicted label of x_0 is embedded as the condition into the timestamp. Finally, the MSE loss between the true Gaussian noise in Equation (3) and the noise predicted by the conditional diffusion model is calculated. This is iteratively optimized, with the optimization objective shown below:

$$\begin{gathered}\min\ \mathcal{L}_{\text{MSE}}\left(\varepsilon, \varepsilon_\theta(x_t, t, F_W(x_0))\right)= \\ \left\|\varepsilon-\varepsilon_\theta[x_t, (\textit{pos\_encoding}(t)+\textit{Embedding}(F_W(x_0)))]\right\|^2 \\ \text{s.t. } \operatorname{Pr}[F_W(x_0)=\varnothing]=p\end{gathered} \tag{4}$$

C. The Recovering Phase

This section describes how we recover data for corresponding target labels using the conditional diffusion model trained in Section IV.B.

1) Recovering and correcting target label images: We input target label l, guidance strength ω and standard normally distributed noise image $\mathbf{z}$ into the trained U-Net to predict the noise and gradually denoise over T rounds. In order to sample $x_{t-1} \sim p_\theta(x_{t-1} \mid x_t)$, based on the above available information it is only necessary to calculate:

$$x_{t-1}=\frac{1}{\sqrt{\bar{\alpha}_t}}\left(x_t-\frac{\beta_t}{\sqrt{(1-\bar{\alpha}_t)}} \varepsilon_\theta\left(x_t, t\right)\right)+\sqrt{\beta_t} \mathbf{z} \tag{5}$$

$$\text{, where } \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

To use a target label to guide recovery, according to [21], we need to modify the predicted noise in Eq. (5) as follows:

$$\begin{gathered}\varepsilon_\theta\left(x_t, t\right) \rightarrow \widetilde{\varepsilon}_\theta\left(x_t, t, l\right)= \\ (1+\omega)\, \varepsilon_\theta[x_t, (\textit{pos\_encoding}(t)+\textit{Embedding}(l))]-\omega \varepsilon_\theta(x_t, t)\end{gathered} \tag{6}$$

The semantics of doing so in Eqs. 5,6 is precisely the step-by-step noise prediction based on the input target l and the noise reduction accordingly, with the guided training in the training phase echoing. Where ω represents the strength of the guidance provided by the target labels. As can be learned

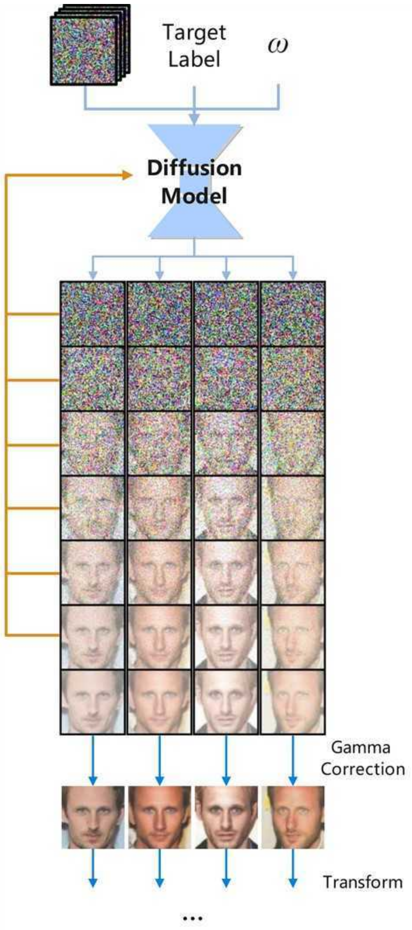

Fig. 3. The first step in the recovery phase is to input noise, target labels, and guidance strength ω to the trained diffusion model and denoise them step by step to obtain the generated image, and eventually correct it.

from Equation 6, if the strength is higher, then the proportion of conditional prediction noise is greater. This results in the generated image features being closer to the target, but at the expense of the quality of the generation.

After T rounds of denoising, we obtain the representative recovered images $G_\theta(\mathbf{z})$ of the target label. However, since diffusion models vary in the samples of data that may be learned for specific labels, this leads to a gap between the results of noise reduction and those generated by traditional diffusion models. However, the purpose of essentially learning the added noise distribution for a specific target is achieved, so it is only necessary to do a secondary correction for the generated image.

We apply gamma correction [27] to the generated image, i.e., $G_\theta(\mathbf{z}) \rightarrow A \cdot G_\theta(\mathbf{z})^\gamma$, adjusting it to match the human eye's perception. Where γ is the gamma factor, and the value of A is usually 1. The graphical representation of the recovery process can be observed in Figure 3. According to [27], γ = 2.2 aligns more closely with the human eye's judgment of brightness and color. As such, the value of γ can be set to approximately 2.2. Section VI.C.4 provides a comparison of the specific impact of different gamma values on the results, and shows that γ = 2.2 is indeed more consistent with human perceptual similarity judgments.

In addition, since the number of recovered images can be large, we need to filter out the most representative k images. In a black-box scenario, our strategy involves performing multiple random transformations and making predictions to select the most robust generated image.

2) Random transformations: Thus, we randomly transform $G_\theta(\mathbf{z})^\gamma$ by randomly cropping and flipping it vertically or horizontally with a certain probability. The transformed image $T(G_\theta(\mathbf{z}))^\gamma$ is then fed into the target model for prediction, and this process is repeated M times.

3) Selecting the top-k robust generated images: We calculate the representative weights $\mathbb{E}[\delta(F_W(T(G_\theta(\mathbf{z})^\gamma)), l)]$ for each image based on the above information, as follows:

$$\mathbb{E}[\delta(F_W(T(G_\theta(\mathbf{z})^\gamma)), l)] = \frac{1}{M}\sum_{i=1}^{M}\delta(F_W(T_i(G_\theta(\mathbf{z})^\gamma)), l) \tag{7}$$

where the δ function returns 1 when its two input values are equal and 0 when they are different. The reason why the δ function compares the predicted labels with the target labels is that the adversaries can only obtain the label information by accessing the target model F_W. Further, using the above equation, we select the $G_\theta(\mathbf{z})$ corresponding to the top-k largest representative weights.

It's crucial to highlight that the target model exhibits varying classification precision for different individuals, which significantly influences the generation of results and the calculation of weights $\mathbb{E}[\delta(F_W(T(G_\theta(\mathbf{z})^\gamma)), l)]$. As depicted in Figure 4, the individuals in (a) display increasing classification precision from top to bottom. To verify the classification robustness for the corresponding individuals, we also performed 100 random transformations in step 2 for the test set and calculated the average precision, which is 13.5%, 25%, 54%, and 58.7% for the four individuals on the graph, respectively. The increase in precision and robustness also provides the target individuals with more auxiliary data that can be used to train the attack model, as shown in the data interval on the left side of Figure 4(a). Because the target model that more accurately extracts the individual's features will assign more auxiliary data that are close to the target.

From the results in (b), it can be observed that when the target model is more accurate in classifying that target and the more data can be used for training, the higher the calculated weights $\mathbb{E}[\delta(F_W(T(G_\theta(\mathbf{z})^\gamma)), l)]$ are, i.e., the generated images are classified under the target class after many random transformations. For example, the first individual has a test set precision of 0%, and the maximum weight calculated for the generated data according to Equation 7 is 0.3784, which means that the rest of the generated data are all below this value. The rest of the individuals have gradually increased weight values as the accuracy rate and robustness increase, and the gap between the maximum weight and the second largest weight is narrowed. This indicates that the quality of the generated data has also been improved, and this characteristic can be identified through qualitative judgment.

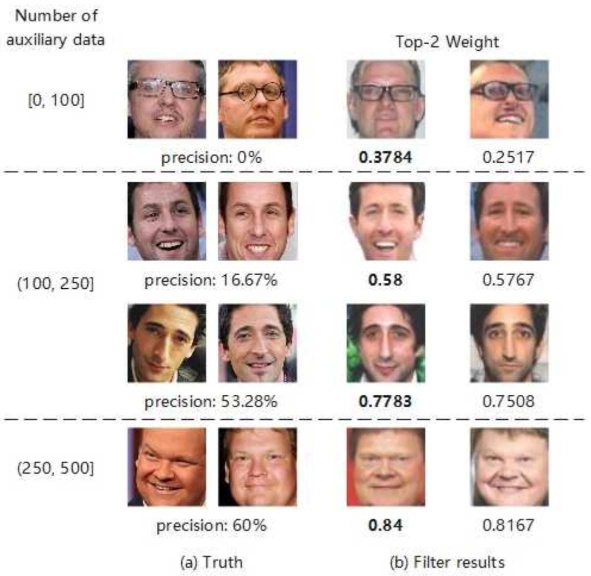

Fig. 4. The impact of variations in the target model's precision rate for classifying different individuals on weight filtering is depicted. (a) represents the true images of an individual, with the target model's test precision rate for each individual. (b) represents the two most optimal results after filtering according to the weights $\mathbb{E}[\delta(F_W(T(G_\theta(\mathbf{z})^\gamma)), l)]$.

V. Discussion of the attack method

A. The Training Phase

While the first and second steps of the training phase serve as a foundation for the subsequent conditional diffusion model training, each stage holds its significance. For instance, the proximity of the auxiliary dataset to the training set and the number of images that can be assigned to each label within it can influence the results (see Section VI.C.3 for details). Moreover, it is intuitively preferable to assign target labels to data that exhibit extreme closeness to the features. Apart from the impact of the dataset, labels play a pivotal role in our approach by guiding the diffusion model toward making noisy predictions for specific targets. The third step constitutes the crux of the training phase and through this process, we can summarize the following advantages of the conditional diffusion model:

1) **Applied to label-only black-box scenarios.** The state-of-the-art approach [12] using conditional GAN requires the computation of gradients by the target model for conditional guidance, but our training process only requires the encoding of predicted labels with embedded timestamps. Therefore, it is not necessary to obtain gradient information with the help of a target model to train a powerful attack model in a label-only scenario. In contrast to white-box attacks, the effectiveness of an attack model is not directly impacted by the architecture and parameters of the target model. Instead, it is determined by the target model's ability to accurately judge the features of the auxiliary data. As such, the more accurately the target model can judge these features, the more effective the attack model training will be. This proposition is verified in Section VI.C.
2) **Stable training process.** In contrast to conditional GAN or GAN, training conditional diffusion models does not require adversarial or sophisticated loss functions. As shown in Equation 4, our training uses a uniform loss function to narrow the $D_{KL}(q(x_{t-1} \mid x_t, x_0) \parallel p_\theta(x_{t-1} \mid x_t))$.
3) **Avoiding MIA overfitting.** As opposed to normal training, MIA requires training the generator on auxiliary data to enable the recovery of the target training set data. The GAN itself suffers from the mode collapse problem [42], and training the generator on the auxiliary data set results in a worse fit. However, the conditional diffusion model is trained for specific labels to learn how to recover the probability distribution of the original data from the noise.

B. The Recovering Phase

Unlike previous black-box attacks [18] [15] [17] [19], our recovery phase does not employ any optimization algorithms to emphasize the power of the generator itself. Instead, it relies on sampling and filtering to demonstrate that the generator's ability can directly determine the generated results.

Lastly, through the recovery process, the following advantages of the conditional diffusion model can be summarized:

1) **Generate multiple results for the same label.** The Inversion model [7] [14] of the AE architecture can only generate unique results for a specific label, but the conditional diffusion model can generate a diversity of target results based on random input noise. And without optimization, only the conditional diffusion model can be attacked for a specific label.
2) **Targeted attack without optimization algorithm.** The conditional diffusion model can generate the corresponding target by adjusting the ω without an optimization algorithm. Low strength means that the recovery result may not be close to the target but the image is more realistic, while high strength means that the target features are recovered to the maximum extent even though the image is not realistic. And with the help of the parameter ω and the special generation process of the diffusion model, we believe that future black-box optimization strategies can do better on this basis.

VI. Experiments

In this section, we will evaluate this method not only quantitatively with relevant metrics, but also qualitatively from the perspective of visual inspection to assess the authenticity and compare it with the effect of generators in related work. The specific experimental setup and evaluation metrics are shown below.

A. Experimental Setup

1) Datasets: Our experiments were conducted on two face recognition datasets and one handwritten digits dataset, and the detailed data allocation is shown in Table III:

- **FaceScrub [43].** FaceScrub is a URL dataset with 100,000 images of 530 actors, which contains 265 male

TABLE III
DATA ALLOCATION OF THE CLASSIFIER AND ITS ATTACK MODEL

Classifier		Attack Model
Task	Data	Auxiliary Data
FaceScrub (530 classes)	80% train, 20% test	**Overlap:** CelebA (296 individuals overlapping)
	80% train, 20% test	**Nonoverlap:** CelebA (non-individuals overlapping)
MNIST (10 classes)	50%train, 50% test	**Overlap:** MNIST 20% test data
	80% train, 20% test (labels: 0-4)	**Nonoverlap:** MNIST's other 5 labels (label: 5-9)

actors and 265 female actors. However, since not every URL was available during the writing period, we downloaded a total of 43,149 images for 530 individuals and resized the images to 64 × 64.

- **CelebA [44].** CelebA is a dataset with 202,599 images of 10,177 celebrities from the Internet. We used the same crop as [7] [14] to remove the background of images in this dataset other than faces to reduce the impact on the experiment. There are 296 individuals overlapping in the two datasets, and since we need to discuss the impact of whether there are individuals overlapping in the two datasets on the experiment, we removed a total of 6,878 images of 296 individuals from CelebA and similarly resized the images to 64 × 64.
- **MNIST [10].** A dataset composed of 70,000 handwritten digit images in 10 classes. Each image is resized to 64 × 64.

2) Target Model: We adopt the same target model architecture as [7], which consists of four CNN blocks followed by two fully connected layers. Each CNN block contains a convolutional layer, a batch normalization layer, a max-pooling layer, and a ReLU activation layer. We train this target model on the FaceScrub dataset using the same hyperparameters as [7] and achieve a test set accuracy of 83.82% for face recognition. The model outputs the predicted confidence scores for the input face image, i.e., the probabilities of belonging to each of the 530 possible individuals. However, based on our assumptions in Section 3, we modify the model output to predict the label instead of the confidence scores. Note that we use numeric labels to represent the individuals, which does not affect the experimental results since we only need a single number for the label-only output.

In addition to this, in order to argue the impact of the target model capability for this attack, we used different CNNs as follows: (1) VGG16 adapted from [45]; (2) ResNet-152 adapted from [46].

3) Attack Model and Implementation Details: The attack model used in our method is U-Net, which is composed of a double convolution block, 3 downsampling blocks, 2 bottom double convolution blocks, 3 upsampling blocks, and a convolution layer, where the double convolution block is composed of 2 convolution layers, 2 group normalization layers, and a GELU activation layers in the order of activation layer in the middle and two layers on each side. The downsampling block consists of a max-pooling layer, two convolutional layers, a SiLU activation layer, a linear connection layer, and a self-attentive layer, while the upsampling block differs in that the max-pooling layer is replaced by an inverse convolutional layer. The U-Net is trained for a maximum of 300 iterations, where CelebA is the training set, the batch size is 16, the learning rate is 3e-4 and the last 50 iterations are reduced to 1e-4. Secondly, the MSE loss function, AdamW optimization algorithm with Exponential Moving Average mechanism is used. In addition, the noise step of forward diffusion is 1500 and the variance schedule is linear, where $\beta_0 = 1e-4, \beta_t = 0.02$. Moreover, the gamma factor γ of 2.3, the guidance strength ω of 4, and the probability p of 0.1. Given that the image dimensions of the auxiliary datasets are uniformly 64 × 64, both training phases necessitate 28G of graphics memory, while both recovery phases require 26G of graphics memory. The batch size and the number of parameters in the target model directly influence the memory size in a proportional relationship. Furthermore, the number of auxiliary datasets directly affects the training time. Utilizing CelebA as the auxiliary dataset, we employ two NVIDIA RTX 3090 GPUs and Inter Xeon Platinum 8350C CPUs to complete a training round in an average duration of 33 minutes, with the generation of 48 images taking an average of 4 minutes. When employing MNIST as the auxiliary dataset, a training round with identical equipment takes an average of 5 minutes, with the total time for training and sampling amounting to 1.5 days.

4) Comparison of Attack Methods: Considering that none of the current generators in black-box MIA can do what the methods in this paper do in the FaceScrub task, we selected two white-box attack methods and one black-box attack method as baselines for fair comparison: Generative Model Inversion (**GMI**) [9], Pseudo Label-Guided Model Inversion Attack (**PLG**) [12], and Learning-Based Model Inversion (**LB-MIA**) [7]. Due to the low complexity of handwritten digital images (a single channel and most of the pixel values are 0), LB-MIA can do equally well, so for the MNIST task, only comparisons are made with this. All three methods employ a generator for attack purposes, with GMI and PLG utilizing a generator from GAN and LB-MI employing an inversion model from an AE framework. GMI necessitates gradient descent to generate representative data for the target label, thus we did not modify the original training and recovery stages. PLG can generate representative data by inputting the target label and latent vector, so we adopted the same selection strategy as our own method. However, the original strategy of assigning top-n images to each individual proved unreliable under a label-only setting during the training stage. Consequently, we used all auxiliary data for training and trained under a white-box setting to facilitate a fairer comparison between our diffusion model and GAN trained under a white-box setting. Lastly, in order to enable LB-MIA to generate the most representative data for the target label, we directly encoded the label as a corresponding one-hot vector to train the inversion model. All methods, including our own, trained their respective attack models on the same auxiliary dataset and performed model inversion attacks on the same target model trained on the same training set.

B. Evaluation Metrics

In this section, we conduct a comprehensive evaluation of the generated data in terms of its accuracy and similarity to target individuals, as well as its realism from a visual perspective. Our evaluation approach is more extensive and closely approximates that of a white-box attack relative to related black-box attacks. The specific evaluation metrics are detailed below.

Attack Accuracy (Attack Acc). This metric is employed to quantitatively evaluate the ability of generated images to accurately identify target individuals. To this end, we trained an evaluation model with a distinct architecture from the target model and higher test accuracy to serve as a proxy for human judgment. Attack accuracy is determined by calculating the percentage of k generated images classified as the target label and averaging the results over 530 individuals. The evaluation model, a ResNet-18 [46] trained using the same training set as the target model, achieved a test set accuracy of 93.03%. The evaluation model accuracy and allocation in all tasks are shown in Table IV.

K-Nearest Neighbor Distance (KNN Dist). KNN Dist is the shortest feature distance from a reconstructed image to the real private training images for a given target. This metric is used to evaluate the similarity at the feature level. Furthermore, the feature distance is measured by the l_2 distance between two images when projected onto the feature space, i.e., the output of the penultimate layer of the evaluation classifier. It is important to note that different evaluation models produce different feature dimensions, resulting in variations in the value of this metric across models. However, it is sufficient to compare the magnitude of the metric within the same evaluation model.

Frechet Inception Distance (FID). FID [47] is commonly used in the work of GAN to evaluate the generated images. FID score measures feature distances between real and fake images, and lower values indicate better image quality and diversity. However, there is inaccuracy in the evaluation of this metric due to the unsuitability of Inception v3 for extracting face features and the inaccuracy of FID calculations based on a small number of generated images [40]. In contrast to white-box attacks, black-box attacks basically do not use this metric.

Learned Perceptual Image Patch Similarity (LPIPS). LPIPS [29] is a metric that measures the perceptual similarity between two images by calculating the similarity between activations of image blocks within a predefined network. This metric has been demonstrated to closely align with human perception and is employed in our experiment to evaluate perceptual similarity. AlexNet [48], which serves as the default predefined network, performs optimally as a forward metric and closely approximates human perception. However, this does not imply that VGG is inferior in terms of matching human perception; when used for optimization, VGG more closely resembles traditional perceptual loss. Consequently, both networks are utilized in our evaluation.

TABLE IV
ALLOCATION OF THE TARGET MODEL AND EVALUATION MODEL.

Target model			Evaluation model	
Task	Model	Accuracy	Model	Accuracy
FaceScrub	CNN	83.82%	Resnet-18	93.03%
	VGG16	88.22%		
	IR152	89.68%		
MNIST	Resnet-18 (Overlap)	99.31%	CNN	99.5%
	CNN (Nonoverlap)	99.86%	Resnet-18	99.94%

C. Experimental Results

1) Compare overlapping vs. non-overlapping auxiliary sets: We study for the first time whether the auxiliary set overlaps with the target training set on the impact of the attack. CelebA was chosen as the auxiliary set due to its greater distributional differences with FaceScrub compared to other face datasets, as established in previous work [11] [12] [18], and demonstrates that the selection of our dataset is the most challenging. While all previous studies assumed no individual overlap between the auxiliary and training sets by default, it is plausible for an adversary to obtain data for a small number of target individuals in real-world scenarios. Thus, it is meaningful to examine the effect of individual overlap on experimental results. As shown in Table V, our method achieves comparable results to those obtained with overlap even in the absence of individual overlap. However, there is a 2% accuracy degradation between GMI and LB-MIA on these two auxiliary sets. The difference in accuracy between our method and the latest white-box attack PLG is approximately 1%, which can be attributed to PLG's reduction of generative image prediction and pseudo-labeling loss through the use of the target model during GAN training and optimization of the generator via gradient descent. But, in the non-overlapping case, the attack accuracy of our method is comparable to that of the PLG effect and even surpasses PLG under the top five accuracy evaluation. Furthermore, high accuracy is not the sole measure of a generative image's effectiveness. In terms of feature distance and perceived similarity, our method yields the best results, producing recovered target image features that are more similar and realistic. From a quantitative evaluation perspective, our results ensure that recovered images closely resemble target individuals, with an average of 56.13% of filtered results accurately identifying targets and 84.54% of results indicating the presence of targets within the top five classes predicted by the model. Qualitative evaluation, as shown in Figure 5, further demonstrates that our methods generate more accurate and realistic results that closely resemble target individuals.

In addition to the recovery of face training set representative samples, we have also evaluated the MNIST dataset to demonstrate the generalization ability of this attack model. Due to the low complexity of this data, the inversion model capability of LB-MIA is sufficiently comparable to ours, so

TABLE V
QUANTITATIVE EVALUATION AND ATTACK PERFORMANCE COMPARISON ON VARIOUS METHODS UNDER WHETHER FACESCRUB AND CELEBA DATASETS OVERLAP OR NOT. ↑ AND ↓ RESPECTIVELY SYMBOLIZE THAT HIGHER AND LOWER SCORES GIVE BETTER ATTACK PERFORMANCE.

Dataset	Scenario	Method	Attack Acc_top1↑	Attack Acc_top5↑	KNN_Dist↓	FID↓	LPIPS_Alex↓	LPIPS_VGG↓
Overlap	White-Box	GMI	31.32%	58.88%	1105.4401	**106.5101**	0.2421	0.4288
		PLG	**57.88%**	**83.58%**	1057.9677	153.9841	0.2524	0.4371
	Black-Box	LB-MIA	6.60%	18.49%	1334.4986	260.5981	0.3107	0.4332
		Ours	54.15%	83.38%	**951.6782**	112.1237	**0.2057**	**0.4151**
Nonoverlap	White-Box	GMI	30.80%	58.66%	1108.2195	**102.7768**	0.2408	0.4264
		PLG	**56.19%**	82.94%	1043.8163	131.2630	0.2333	0.4295
	Black-Box	LB-MIA	4.34%	13.77%	1383.2693	242.1953	0.3160	0.4384
		Ours	56.13%	**84.54%**	**959.3208**	109.5714	**0.2078**	**0.4165**

TABLE VI
QUANTITATIVE EVALUATION AND ATTACK PERFORMANCE COMPARISON ON VARIOUS METHODS UNDER WHETHER MNIST DATASETS OVERLAP OR NOT. ↑ AND ↓ RESPECTIVELY SYMBOLIZE THAT HIGHER AND LOWER SCORES GIVE BETTER ATTACK PERFORMANCE.

Dataset	Method	Attack Acc_top1↑	Attack Acc_top5↑ (top2 in Nonoverlap)	KNN Dist↓	FID↓	LPIPS_Alex↓	LPIPS_VGG↓
Overlap	LB-MIA	70%	100%	4274.1179	254.0187	0.4066	0.3582
	Ours	**100%**	**100%**	**2634.9470**	**222.9036**	**0.2220**	**0.2954**
Nonoverlap	LB-MIA	40%	60%	308.2348	278.2425	0.4047	0.4090
	Ours	**76%**	**100%**	**146.9648**	**238.6635**	**0.2816**	**0.3391**

we only performed an equivalent comparison of the black-box generator for each metric. The quantitative evaluation is shown in Table VI, where our method achieves an attack accuracy of 100% when there is an overlapping in the dataset and is the best in all similarity metrics. The qualitative evaluation is shown in Fig. 6(a). When it is non-overlapping in the dataset, i.e., the target model is a 5-classification model, we evaluate the attack accuracy of top1 and top2, and again achieve the best in all metrics, with qualitative evaluation shown in Fig. 6(b). It should be noted that LB-MIA will only generate a unique image, and since the MNIST task is simpler than faces, generating multiple images would result in large differences in the FID metric. Therefore, we filtered only one of the most robust images in our method for evaluation in order to make a fair comparison.

Combined with the discussion in Section V and the results in Table V, it can be substantiated that the strategy of employing the generator in GAN as the attack model is subpar to our results in terms of comprehensive evaluation. This is even in the white-box condition due to the unstable training issue posed by the intricate loss function inherent to GAN. Furthermore, since PLG will utilize the loss between the predicted label of the generative image and the target label to optimize the generator, it results in feature similarity and sensory similarity that is not superior to our method, even though the ultimate result emphasizes ensuring attack accuracy. The training of the conditional diffusion model, which is optimized with the loss between the prediction noise and the actual noise, is more stable compared to GAN. Owing to the guidance of predicted labels, it can guarantee the accuracy and structural authenticity of the image. Moreover, a limitation of the attack model in LB-MIA is that it only generates a single image for each label, which reduces the fault tolerance of the attack. However, our trained attack model can generate multiple possible images in label-only scenarios and filter them with the assistance of the target model.

Based on the above analysis, it can be observed that both KNN Dist and FID have limitations. For instance, a review of Table VI reveals that KNN Dist exhibits significant variability in results under different evaluation models, which poses challenges for comparative experimental evaluations within the field. Furthermore, since the model used for evaluating the KNN Dist was trained on the same training set as the target model, and the FID was only calculated based on the pre-trained Inception v3, the FID is not more reliable than the KNN Dist in different scenarios. Consequently, FID is gradually being phased out for evaluations within the MIA field. On the other hand, the LPIPS evaluation metric, which is based on a pre-trained model, quantifies the perceptual similarity between images. As demonstrated in Table V, VI, and Figure 5, LPIPS exhibits greater generalizability than KNN Dist, i.e., it maintains a consistent value domain, thereby facilitating comparative experimental evaluations within the field. In comparison to FID, LPIPS proves to be more reliable. For example, referring to Table V, it can be observed that the results generated by our method outperform GMI in terms of attack accuracy, feature distance, and qualitative assessment. However, GMI outperforms all methods under the FID. Referring to Table VI, it can be seen that our results exhibit a larger gap in attack accuracy and feature distance compared to LB-MIA, but FID does not accurately reflect this. In such instances, the quantitative value of LPIPS provides a more

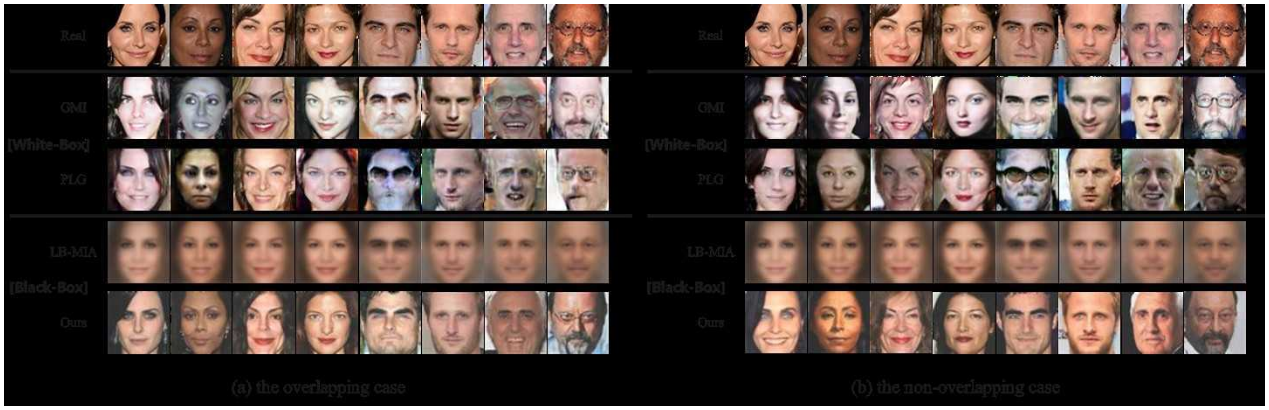

Fig. 5. Qualitative evaluation and attack performance comparison on various methods under whether FaceScrub and CelebA datasets overlap or not.

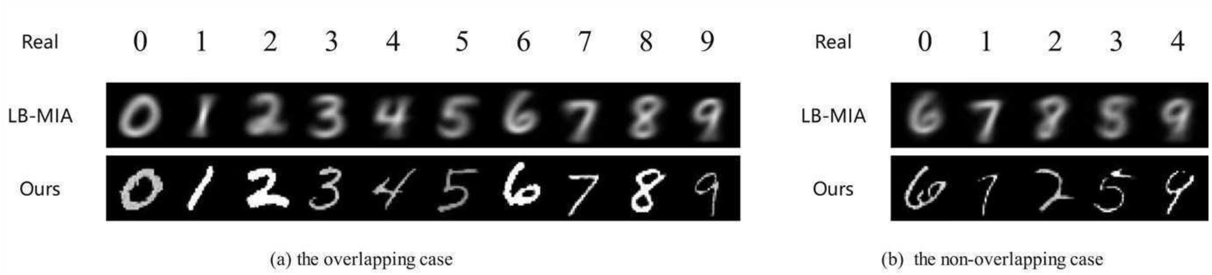

Fig. 6. Qualitative evaluation and attack performance comparison on various methods under whether MNIST datasets overlap or not.

accurate representation of the effect gap between different methods.

2) Evaluate attack performance for same target label: According to the results presented in Table V, it can be observed that the generator trained using a white-box attack compromises generation quality in favor of improved accuracy. This attack relies on the optimization algorithm employed during the recovery phase, with the essence of the optimization study being equivalent to replacing the recovery phase optimization algorithm in GMI for both black-box and white-box attacks. However, this may prove challenging in real-world scenarios. Our focus is on assessing the security risk posed by a powerful attack generator, as opposed to previous work that concentrated on studying optimization algorithms. As shown in Figure 7, our method is capable of generating results without the need for optimization and filtering, yielding recovery results that are more realistic and closely resemble target individuals compared to those generated by other methods. In addition, it can also be observed from the recovery results for the same label that the results recovered by our method are more compact. Moreover, such a comparison provides a fairer representation of the capabilities of diffusion models relative to GANs.

3) Examine the impact of auxiliary data quantity on attack: Our study analyzes both the generator and the auxiliary set and compares the results obtained with different quantities of data assigned to the target individual within the auxiliary set. As shown in Figure 8, our findings indicate an overall upward trend in attack accuracy as the number of auxiliary sets increases. In contrast to PLG, our approach demonstrates a consistent upward trend and surpasses the other three methods in attack accuracy when provided with a substantial quantity of auxiliary data. Specifically, with over 750 auxiliary data, top-1 accuracy reaches 68.18%, significantly surpassing the attack accuracy of PLG. Top-5 accuracy attains 84.82% and 91.06% for quantities ranging from 500 to 750 and above, respectively, both exceeding that of PLG.

Figure 8 (c-e) shows that the feature distance of recovery results for each method gradually decreases as the quantity increases, with our method achieving the best evaluation in terms of similarity. Furthermore, since both KNN Dist and Attack Acc are computed by the same evaluation model. Theoretically, only the fully connected layer of the evaluation model affects the results of both. By examining (a, b, d), it can be observed that KNN Dist exhibits a trend similar to the attack accuracy. For instance, both PLG and our results show a decreasing trend when the number is in the interval (500,750] in (b), which can also be visualized in (d). Compared to the trends presented by FID and KNN Dist, the evaluation of LPIPS is more objective. We consider a quantitative trend presentation for the qualitative evaluation and demonstrate the trend that the quality of the generation progressively improves as the number increases. Comparative analysis reveals that, with an equivalent amount of data, our approach attains a superior recovery level.

4) Examine the impact of different target models on attack: The predictive ability of the target model directly determines the effectiveness of this attack. For example, if a face recognition model can better determine the closest training set individual with that feature from the auxiliary data, then the

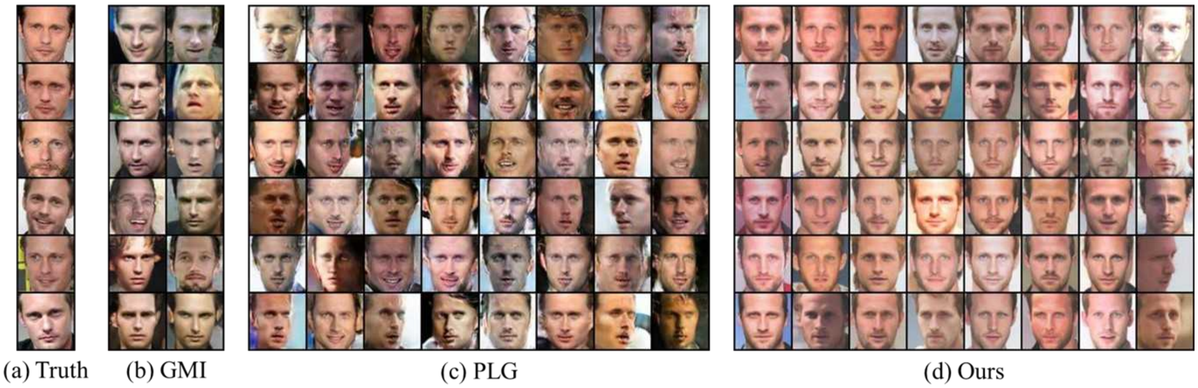
(a) Truth (b) GMI (c) PLG (d) Ours

Fig. 7. Qualitative evaluation and attack performance comparison on various methods under the same target label *when there is no individual overlap*. (a) represent the real target images, (b) shows the GMI attack result, (c) displays the PLG attack result, and (d) presents the attack result of our method. Unlike (b), (c, d) are recovered directly from input noise and a label without optimization and filtering.

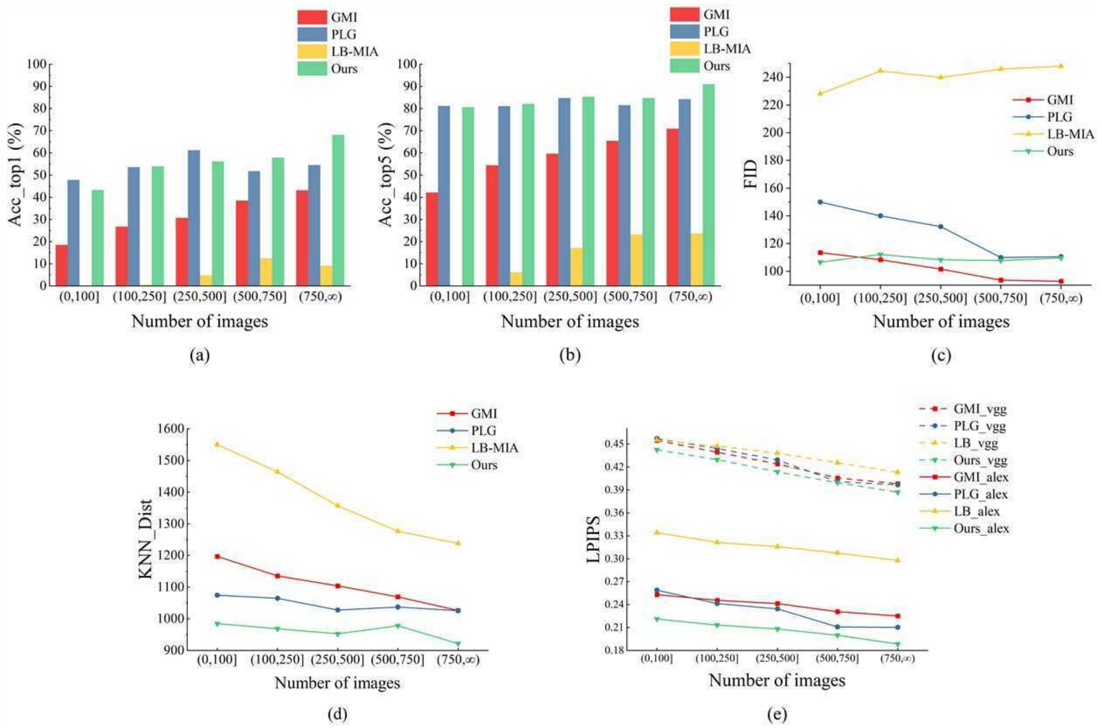

Fig. 8. Quantitative evaluation and attack performance comparison on various numbers of auxiliary images *when there is no individual overlap*. (a, b) represent the effect of quantity on accuracy, (c, d) represent the effect of quantity on similarity, and (e) represent the effect of quantity on similarity and realism.

features guided by the same label in training will be closer and the attack will be more effective. In this section, we only discuss the impact of different target model capabilities on this method. Because the effect of different model architectures on white-box attacks is not the purpose of this paper's discussion. In addition, there is no need to repeat the LB-MIA experiment because is too ineffective. As shown in Table IV and VIII, as the target model prediction capability increases, the evaluation of each metric is better, and the accuracy of the top-5 attack on IR152 reaches 93.76%.

When the target model is CNN and VGG16, metrics such as attack accuracy, KNN Dist, and FID can reflect some degree of effect enhancement. However, the evaluation by LPIPS indicates that the result may not show significant improvement from the perspective of human-eye sensory similarity. When the target model is IR152, which possesses the strongest capability, the attack only has an effect on sensory similarity. The enhancement of the attack effect can also be observed from the FID result.

5) Ablation study: We conducted further analysis on the effect of the gamma factor γ and guidance strength ω on our experimental results. As depicted in Table VII our findings confirm that an increase in γ positively impacts attack accuracy and KNN distance, where $\gamma = 1$ indicates the absence of image correction. However, for perceptual similarity, the evaluated results exhibit a decreasing and then increasing trend and the γ equal of 2.2 is the most consistent with human perceptual judgment, necessitating a trade-off in the value of

TABLE VII
QUANTITATIVE EVALUATION AND EFFECT OF GAMMA FACTOR γ AND GUIDANCE STRENGTH ω ON ATTACK PERFORMANCE.

Factor	Value	Attack Acc_top1↑	Attack Acc_top5↑	KNN_Dist↓	FID↓	LPIPS_Alex↓	LPIPS_VGG↓
γ	1	38.32%	69.17%	1050.698	124.9308	0.2470	0.4325
	2	54.29%	83.6%	965.8551	**109.1064**	0.2081	**0.4161**
	2.1	55.33%	83.96%	963.0374	109.1552	0.2077	**0.4161**
	2.2	55.9%	84.26%	961.1259	109.1554	**0.2076**	0.4164
	2.3	56.13%	84.54%	959.3208	109.5714	0.2078	0.4165
	2.4	56.24%	**84.76%**	958.8945	109.5858	0.2081	0.4168
	2.5	**56.45%**	84.72%	**958.3543**	109.9230	0.2086	0.4172
ω	1	33.89%	63.89%	1009.7654	**94.3645**	**0.2024**	**0.4078**
	2	41.53%	74.72%	984.6321	98.3713	**0.2024**	0.4082
	4	55.00%	**84.72%**	**945.4405**	108.7672	0.2060	0.4185
	6	56.25%	83.75%	955.6436	119.0539	0.2134	0.4257
	8	**57.92%**	84.44%	966.1134	136.4142	0.2233	0.4412
p	0	44.4%	76.67%	**768.6025**	119.4697	0.2068	**0.4171**
	0.1	**55.00%**	**84.72%**	**945.4405**	**108.7672**	**0.2060**	0.4185
	0.2	42.36%	74.17%	955.4660	115.9034	0.2153	0.4267
	0.3	42.78%	71.25%	962.8414	115.9907	0.2145	0.4273

TABLE VIII
QUANTITATIVE EVALUATION AND EFFECT OF DIFFERENT TARGET MODELS ON ATTACK PERFORMANCE. ↑ AND ↓ RESPECTIVELY SYMBOLIZE THAT HIGHER AND LOWER SCORES GIVE BETTER ATTACK PERFORMANCE.

Target model	Attack Acc_top1↑	Attack Acc_top5↑	KNN Dist↓	FID↓	LPIPS _Alex↓	LPIPS _VGG↓
CNN	56.13	84.54	959.3208	109.5714	0.2078	0.4165
VGG16	66.6	90.36	893.884	108.2697	0.2075	0.4128
IR152	**74.8**	**93.76**	**855.1517**	**101.0693**	**0.2015**	**0.4042**

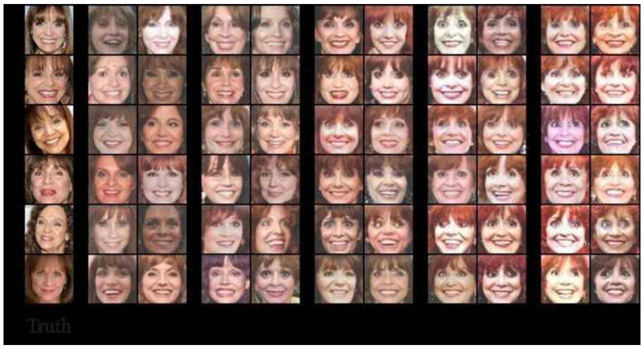

Fig. 9. Qualitative evaluation and effect of guidance strength ω on attack performance.

gamma. Second, as the value of ω increases, there is an initial improvement followed by a decline in both the evaluation of top-5 attack accuracy and feature distance, while perceptual similarity progressively becomes worse. Since when $\omega = 0$ indicates unconditional guidance generation, it is not explicitly evaluated in the results. As shown in Figure 9, excessively high values of ω result in a degradation of image quality but in favor of target features. However, as can be observed from the figure, a lower KNN Dist does not necessarily imply a more effective attack. Similarly, under the influence of γ and ω, the evaluation of FID, when compared to that shown by LPIPS, reveals that the results of LPIPS are more closely aligned with the qualitative judgment of the human eye. This aids in the selection of hyper-parameters and the evaluation of the results. So, in order to effectively balance accuracy, realism, and similarity, it is crucial to select an appropriate level of guidance strength with reference to the LPIPS. For instance, when ω equals 4, the overall evaluation results exhibit greater balance.

Furthermore, we delve into the impact of the probability p of training without label guidance on the results. This hyper-parameter is introduced as it allows for the learning of the data's features and structure with a certain probability, thereby ensuring that the attack model does not over-fit the label information. When p is 0, indicating that the attack model is entirely guided by labels, it is observed that the feature distance of KNN Dist is evaluated to be 768.6025, which is the lowest. However, this is not the optimal choice in terms of attack accuracy evaluation. When p exceeds 0.2, a diminishing effect can be observed through the evaluation of all indicators. Therefore, for the best effect, p should take a value between 0 and 0.2.

VII. LIMITATIONS AND FUTURE WORK

Despite achieving favorable experimental results in our proposed setting, our method has limitations when considered alongside current related works.

1) Training the conditional diffusion model takes longer compared to other methods due to the multiple rounds of iterations required to learn the noise distribution. Research on accelerating diffusion model training is ongoing and this limitation may be overcome in the future.
2) We focus on developing a robust and practical attack model in label-only scenarios. However, secondary optimizations, based on the results produced by this model, may offer more opportunities for enhancement than the outcomes of existing generators. For instance, the optimization of the initial input noise or the images in

the noise reduction path. We believe this is a direction deserving of further investigation in future research.

Furthermore, designing defense strategies against label-only MIAs will be a challenging direction. We believe that future "model-centric" defense strategies [22] [23] should effectively leverage powerful deep learning models available today. "Data-centric" defense strategies [25] [49] need to trade off the prediction accuracy of the target model for training and non-training sets.

In conclusion, it is imperative for future AI research to prioritize the protection of privacy knowledge acquired by models while enhancing their utility. This is particularly relevant in light of recent developments in model inversion attacks and the rapid advancement of AI technology.

VIII. Conclusion

We develop a novel label-only model inversion attack method utilizing a conditional diffusion model, capable of recovering representative data for a specific target label in the training set, given that the target model predicts only the label for the input. The attack model is trained on an auxiliary public dataset and uses the predicted label of corresponding auxiliary data as a condition to guide the training of the diffusion model. This allows the adversary to input standard normally distributed noise and the target label into the conditional diffusion model during the recovery phase, generating data with a pre-defined guidance strength representing that label in the training set without optimization. Experimental results demonstrate that our method generates more accurate, realistic, and similar data compared to generators in related work. Future work will focus on exploring more efficient optimization algorithms based on such a high-quality generator and investigating defense methods that balance model privacy and usability.

Acknowledgments

This work was supported by the "Pioneer" and "Leading Goose" R&D Program of Zhejiang (Grant No.2023C03203, 2023C03180, 2022C03174), and the Key Science and Technology Research Project of Chongqing Municipal Education Commission, KJZD-K202300601.

References

[1] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proceedings of the 2017 ACM SIGSAC Conference on computer and communications security*, 2017, pp. 587–601.

[2] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction apis." in *USENIX security symposium*, vol. 16, 2016, pp. 601–618.

[3] N. Papernot, P. McDaniel, A. Sinha, and M. P. Wellman, "Sok: Security and privacy in machine learning," in *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2018, pp. 399–414.

[4] Y. He, G. Meng, K. Chen, X. Hu, and J. He, "Towards security threats of deep learning systems: A survey," *IEEE Transactions on Software Engineering*, vol. 48, no. 5, pp. 1743–1770, 2020.

[5] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, "Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing," in *23rd {USENIX} Security Symposium ({USENIX} Security 14)*, 2014, pp. 17–32.

[6] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, 2015, pp. 1322–1333.

[7] Z. Yang, J. Zhang, E.-C. Chang, and Z. Liang, "Neural network inversion in adversarial setting via background knowledge alignment," in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 225–240.

[8] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *2017 IEEE symposium on security and privacy (SP)*. IEEE, 2017, pp. 3–18.

[9] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, and D. Song, "The secret revealer: Generative model-inversion attacks against deep neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 253–261.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[11] S. Chen, M. Kahla, R. Jia, and G.-J. Qi, "Knowledge-enriched distributional model inversion attacks," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 16 178–16 187.

[12] X. Yuan, K. Chen, J. Zhang, W. Zhang, N. Yu, and Y. Zhang, "Pseudo label-guided model inversion attack via conditional generative adversarial network," *arXiv preprint arXiv:2302.09814*, 2023.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[14] T. Zhu, D. Ye, S. Zhou, B. Liu, and W. Zhou, "Label-only model inversion attacks: Attack with the least information," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 991–1005, 2022.

[15] A. Dionysiou, V. Vassiliades, and E. Athanasopoulos, "Exploring model inversion attacks in the black-box setting," *Proceedings on Privacy Enhancing Technologies*, vol. 1, pp. 190–206, 2023.

[16] D. Ye, H. Chen, S. Zhou, T. Zhu, W. Zhou, and S. Ji, "Model inversion attack against transfer learning: Inverting a model without accessing it," *arXiv preprint arXiv:2203.06570*, 2022.

[17] S. Yoshimura, K. Nakamura, N. Nitta, and N. Babaguchi, "Model inversion attack against a face recognition system in a black-box setting," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2021, pp. 1800–1807.

[18] M. Kahla, S. Chen, H. A. Just, and R. Jia, "Label-only model inversion attacks via boundary repulsion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 15 045–15 053.

[19] G. Han, J. Choi, H. Lee, and J. Kim, "Reinforcement learning-based black-box model inversion attacks," *arXiv preprint arXiv:2304.04625*, 2023.

[20] S. An, G. Tao, Q. Xu, Y. Liu, G. Shen, Y. Yao, J. Xu, and X. Zhang, "Mirror: Model inversion for deep learning network with high fidelity," in *Proceedings of the 29th Network and Distributed System Security Symposium*, 2022.

[21] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[22] T. Wang, Y. Zhang, and R. Jia, "Improving robustness to model inversion attacks via mutual information regularization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 13, 2021, pp. 11 666–11 673.

[23] X. Peng, F. Liu, J. Zhang, L. Lan, J. Ye, T. Liu, and B. Han, "Bilateral dependency optimization: Defending against model-inversion attacks," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2022, pp. 1358–1367.

[24] Z. Yang, L. Wang, D. Yang, J. Wan, Z. Zhao, E.-C. Chang, F. Zhang, and K. Ren, "Purifier: defending data inference attacks via transforming confidence scores," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 9, 2023, pp. 10 871–10 879.

[25] L. Struppek, D. Hintersdorf, and K. Kersting, "Be careful what you smooth for: Label smoothing can be a privacy shield but also a catalyst for model inversion attacks," *arXiv preprint arXiv:2310.06549*, 2023.

[26] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[27] R. W. G. Hunt, *The reproduction of colour*. John Wiley & Sons, 2005.

[28] L. Struppek, D. Hintersdorf, A. D. A. Correia, A. Adler, and K. Kersting, "Plug & play attacks: Towards robust and flexible model inversion attacks," *arXiv preprint arXiv:2201.12179*, 2022.

This article has been accepted for publication in IEEE Transactions on Information Forensics and Security. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIFS.2024.3372815

[29] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[30] I. W. P. Consortium, "Estimation of the warfarin dose with clinical and pharmacogenetic data," *New England Journal of Medicine*, vol. 360, no. 8, pp. 753–764, 2009.

[31] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, pp. 81–106, 1986.

[32] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[33] P. Baldi, "Autoencoders, unsupervised learning, and deep architectures," in *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings, 2012, pp. 37–49.

[34] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies–a comprehensive introduction," *Natural computing*, vol. 1, pp. 3–52, 2002.

[35] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[36] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *International conference on machine learning*. PMLR, 2015, pp. 2256–2265.

[37] C. Jarzynski, "Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach," *Physical Review E*, vol. 56, no. 5, p. 5018, 1997.

[38] J. R. Norris, *Markov chains*. Cambridge university press, 1998, no. 2.

[39] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[40] J. Zheng, "Targeted image reconstruction by sampling pre-trained diffusion model," *arXiv preprint arXiv:2301.07557*, 2023.

[41] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, vol. 30, 2017, pp. 6000–6010.

[42] D. Bau, J.-Y. Zhu, J. Wulff, W. Peebles, H. Strobelt, B. Zhou, and A. Torralba, "Seeing what a gan cannot generate," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4502–4511.

[43] H.-W. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *2014 IEEE international conference on image processing (ICIP)*. IEEE, 2014, pp. 343–347.

[44] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3730–3738.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[47] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[48] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[49] S. Chen, F. Kang, N. Abhyankar, M. Jin, and R. Jia, "Data-centric defense: Shaping loss landscape with augmentations to counter model inversion," in *Proceedings of the 40th International Conference on Machine Learning*. PMLR, 2023.

Rongke Liu received B.Eng from Hangzhou Dianzi University in 2024. He is currently pursuing the Ph.D. degree with the School of Computer Science, Nanjing University of Aeronautics and Astronautics. His research interests include artificial intelligence security and privacy protection.

Dong Wang Dong Wang received her Ph.D. degree in Communication and Information System from Wuhan University, China in 2021. From 2019 to 2021, she was a visiting Ph.D. student at the University of Notre Dame, USA. She is currently a lecturer of the School of Cyberspace, Hangzhou Dianzi University. Her research interests include privacy computing, differential privacy, and artificial intelligence security.

Yizhi Ren received the Ph.D. degree in computer software and theory from the Dalian University of Technology, China, in 2011. From 2008 to 2010, he was a Research Fellow at Kyushu University, Japan. He is currently a Full Professor with the School of Cyberspace, Hangzhou Dianzi University, China. His research interests include data security, privacy-preserving, and artificial intelligence security.

Zhen Wang received the B.Sc., M.Eng., and Ph.D. degrees in software engineering from the Dalian University of Technology, Dalian, China, in 2007, 2009, and 2016, respectively. From 2014 to 2016, he was a Research Fellow with Nanyang Technological University, Singapore. He is currently an Associate Professor with the School of Cyberspace, Hangzhou Dianzi University, Hangzhou, China. His current research interests include network security, artificial intelligence security, complex networks, and algorithmic game theory.

Kaitian Guo received the B.Eng degree from Nanjing University in 2022. He is currently pursuing the Master's degree with the School of Cyberspace, Hangzhou Dianzi University. His research interests are in the fields of artificial intelligence security and privacy protection.

Qianqian Qin received the B.Eng degree from Wuhan University of Science and Technology in 2022. She is currently pursuing the Master's degree with the School of Cyberspace, Hangzhou Dianzi University. Her research interests are in the fields of artificial intelligence security and privacy protection.

Xiaolei Liu received the Ph.D. degree and M.S. degree in software engineering from the University of Electronic and Technology of China (UESTC). He is an Associate Research Fellow in Institute of Computer Application, China Academy of Engineering Physics. His research interests include system security, artificial intelligence security, and privacy protection.