STEALTHY SHIELD DEFENSE: A CONDITIONAL MUTUAL INFORMATION-BASED APPROACH AGAINST BLACK-BOX MODEL INVERSION ATTACKS

Tianqu Zhuang[1*], Hongyao Yu[2*], Yixiang Qiu[1*], Hao Fang[1*], Bin Chen[2#], Shu-Tao Xia[1]
[1]Shenzhen International Graduate School, Tsinghua University, China
[2]Harbin Institute of Technology, Shenzhen, China
{zhuangtq23, qiu-yx24, fang-h23}@mails.tsinghua.edu.cn; yuhongyao@stu.hit.edu.cn;
chenbin2021@hit.edu.cn; xiast@sz.tsinghua.edu.cn; [*]Equal contribution [#]Corresponding author

ABSTRACT

Model inversion attacks (MIAs) aim to reconstruct the private training data by accessing a public model, raising concerns about privacy leakage. Black-box MIAs, where attackers can only query the model and obtain outputs, are closer to real-world scenarios. The latest black-box attacks have outperformed the state-of-the-art white-box attacks, and existing defenses cannot resist them effectively. To fill this gap, we propose Stealthy Shield Defense (SSD), a post-processing algorithm against black-box MIAs. Our idea is to modify the model's outputs to minimize the conditional mutual information (CMI). We mathematically prove that CMI is a special case of information bottlenecks (IB), and thus inherits the advantages of IB—making predictions less dependent on inputs and more dependent on ground truths. This theoretically guarantees our effectiveness, both in resisting MIAs and preserving utility. For minimizing CMI, we formulate a convex optimization problem and solve it via the water-filling method. Adaptive rate-distortion is introduced to constrain the modification to the outputs, and the water-filling is implemented on GPUs to address computational cost. Without the need to retrain the model, our algorithm is plug-and-play and easy to deploy. Experimental results indicate that SSD outperforms existing defenses, in terms of MIA resistance and model's utility, across various attack algorithms, training datasets, and model architectures. Our code is available at `https://github.com/ZhuangQu/Stealthy-Shield-Defense`.

1 INTRODUCTION

Deep neural networks (DNNs) have driven widespread deployment in multiple mission-critical domains, such as computer vision (He et al., 2016), natural language processing (Devlin et al., 2019) and dataset distillation (Zhong et al., 2024b;a). However, their integration with sensitive training data has raised concerns about privacy breaches. Recent studies (Fang et al., 2024b;a; 2025) have explored various attack methods to probe these privacy, such as gradient inversion (Fang et al., 2023; Yu et al., 2024b) and membership inference (Hu et al., 2022). Among the emergent threats, model inversion attacks (MIAs) aim to reconstruct the private training data by accessing a public model, posing the greatest risk (Qiu et al., 2024c). For instance, consider a face recognition access control system with a publicly accessible interface. Through carefully crafted malicious queries, model inversion attackers can infer the sensitive facial images stored in the system, along with the associated user identities.

MIAs are divided into *white-box* and *black-box* (Fang et al., 2024c). White-box attackers know the details of the model, whereas black-box attackers can only query the model and obtain outputs. Black-box MIAs become more threatening than white-box because: **(1) Black-box scenarios are more common.** As models grow larger nowadays, they are mostly stored on servers and can only be accessed online, which are typical black-box scenarios. **(2) Black-box attacks are more powerful.** The latest soft-label attack RLBMI (Han et al., 2023) and hard-label attack LOKT (Nguyen et al., 2024) have outperformed the state-of-the-art white-box attacks. **(3) Existing defenses cannot resist**

black-box attacks effectively. Existing defenses focus on modifying the weights and structure of the model, but black-box attackers only exploit the outputs, and thus are less susceptible.

To address these concerns, we propose Stealthy Shield Defense (SSD), a post-processing algorithm against black-box MIAs. As shown in Figure 1, the idea of SSD is to modify the model's outputs to minimize the conditional mutual information (CMI) (Yang et al., 2024). CMI quantifies the dependence between inputs and predictions when ground truths are given. In Theorem 1, we prove that CMI is a special case of information bottlenecks (IB), and thus inherits the advantages of IB—making predictions less dependent on inputs and more dependent on ground truths. Under this theoretical guarantee, SSD achieves a better trade-off between MIA resistance and model's utility. Without the need to retrain the model, SSD is plug-and-play and easy to deploy.

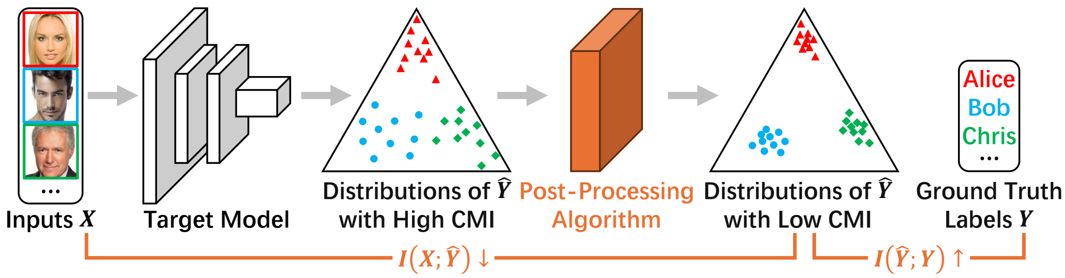

Figure 1: An overview of Stealthy Shield Defense. The probability simplex is a triangle when the number of classes is three. CMI is defined as $\mathcal{I}(X;\hat{Y}|Y)$. According to our Theorem 1, minimizing CMI makes the mutual information $\mathcal{I}(X;\hat{Y})$ minimized and $\mathcal{I}(\hat{Y};Y)$ maximized. As shown by Yang et al. (2024), minimizing CMI makes the outputs more concentrated class-wisely.

The contributions of this paper are:

- We introduce CMI into model inversion defense for the first time, and theoretically prove its effectiveness.
- We propose a post-processing algorithm to minimize CMI without retraining models. In our algorithm, temperature is introduced to calibrate the probabilities and adaptive rate-distortion is introduced to constrain the modification to the outputs. We speed up our algorithm by GPU-based water-filling method as well.
- Our experiments indicate that we outperform all competitors, in terms of MIA-resistance and model's utility, exhibiting good generalizability across various attack algorithms, training datasets, and model architectures.

2 Related Work

2.1 Model Inversion Attacks and Defenses

Model inversion attacks (MIAs) are a serious privacy threat to released models (Fang et al., 2024c). MIAs are categorized as *white-box* (Zhang et al., 2020; Chen et al., 2021; Struppek et al., 2022; Yuan et al., 2023; Qiu et al., 2024a) and *black-box*. We focus on black-box MIAs, where attackers can only query the model and obtain outputs. In this scenario, BREP (Kahla et al., 2022) utilizes zero-order optimization to drive the latent vectors away from the decision boundary. Mirror (An et al., 2022) and C2F (Ye et al., 2023) explore genetic algorithms. LOKT (Nguyen et al., 2024) trains multiple surrogate models and applies white-box attacks to them.

To address the threat of MIAs, a variety of defenses have been proposed. MID (Wang et al., 2021), BiDO (Peng et al., 2022) and LS (Struppek et al., 2024) change the training losses, TL (Ho et al., 2024) freezes some layers of the model, and CA-FaCe (Yu et al., 2024a) change the structure of the model. However, black-box attackers only exploit the outputs, and thus are rarely hindered. The defense against black-box MIAs is still limited.

In this paper, we propose a novel black-box defense based on post-processing, without retraining the model. Experimental results indicate that we outperform the existing defenses.

2.2 Information Bottleneck and Conditional Mutual Information

Tishby et al. (1999) proposed the Information Bottleneck (IB) principle: a good machine learning model should compress the redundant information in inputs while preserving the useful information for tasks. They later highlighted that information is compressed layer-by-layer in DNNs (Tishby & Zaslavsky, 2015; Shwartz-Ziv & Tishby, 2017). Alemi et al. (2017) proposed Variational Information Bottleneck (VIB) to estimate the bounds of IB, and Wang et al. (2021) applied VIB in their Mutual Information-based Defense (MID).

Yang et al. (2024) proposed to use conditional mutual information (CMI) as a performance metric for DNNs, providing the calculation formula and geometric interpretation of CMI. By minimizing CMI, they improve classifiers (Yang et al., 2025) and address class imbalance (Hamidi et al., 2024). By maximizing CMI, they improve knowledge distillation (Ye et al., 2024) and address nasty teachers (Yang & Ye, 2024).

In this paper, we theoretically prove that CMI is a special case of IB and thus inherits the advantages of IB. Furthermore, we propose a novel model inversion defense based on CMI.

3 Preliminary

3.1 Notation

Let $f\colon \mathbb{X} \to \mathbb{Y}$ be a neural classifier, $X \in \mathbb{X}$ be the input to f, $Y \in \mathbb{Y}$ be the ground truth label, $\hat{Y} \in \mathbb{Y}$ be the label predicted by f, and $Z \in \mathbb{Z}$ be the intermediate feature in f. Note that $Y \to X \to Z \to \hat{Y}$ is a Markov chain. Let $\mathcal{P}$ be the probability function and, for brevity, let $\mathcal{P}(x) \coloneqq \mathcal{P}\{X = x\}$, $\mathcal{P}(y) \coloneqq \mathcal{P}\{Y = y\}$, $\mathcal{P}(x, \hat{y}|y) \coloneqq \mathcal{P}\{X = x, \hat{Y} = \hat{y} \mid Y = y\}$, etc.

Let $\Delta^{\mathbb{Y}}$ be the probability simplex with $|\mathbb{Y}|$ vertices. Let $\boldsymbol{f}(x) \in \Delta^{\mathbb{Y}}$ be the output from the softmax layer of f when x is input to f, and $f_{\hat{y}}(x)$ be the $\hat{y}$-th component of $\boldsymbol{f}(x)$, $\hat{y} \in \mathbb{Y}$. Note that $f(x) = \arg\max_{\hat{y} \in \mathbb{Y}} f_{\hat{y}}(x)$.

3.2 Model Inversion Attacks

Let $D \subseteq \mathbb{X} \times \mathbb{Y}$ be the dataset learned by f. MIAs aim to reconstruct $\hat{D}$ as close to D as possible. According to the access to f, MIAs are categorized as:

Hard-label: Attackers can query any $x \in \mathbb{X}$ and obtain $f(x) \in \mathbb{Y}$.

Soft-label: Attackers can query any $x \in \mathbb{X}$ and obtain $\boldsymbol{f}(x) \in \Delta^{\mathbb{Y}}$.

White-box: Attackers know the details of f.

Hard-label and soft-label, collectively called *black-box*,[1] are what we aim to defend against.

3.3 Defense via Mutual Information

Wang et al. (2021) proposed Mutual Information-based Defense (MID). The mutual information between X and $\hat{Y}$ is defined as

$$\mathcal{I}(X; \hat{Y}) \coloneqq \sum_{x \in \mathbb{X}} \sum_{\hat{y} \in \mathbb{Y}} \mathcal{P}(x, \hat{y}) \log \frac{\mathcal{P}(x, \hat{y})}{\mathcal{P}(x)\mathcal{P}(\hat{y})}. \tag{1}$$

$\mathcal{I}(X; \hat{Y})$ quantifies the dependence between X and $\hat{Y}$. They minimize it to prevent attackers from obtaining the information about D. However, minimizing $\mathcal{I}(X; \hat{Y})$ hurts the model's utility. Especially, $\mathcal{I}(X; \hat{Y}) = 0$ iff X and $\hat{Y}$ are independent, in which case f is immune to any attack but useless at all.

[1]Some literature refer to *hard-label* as *label-only*, and *soft-label* as *black-box*.

As an alternative, they introduced information bottlenecks (IB), which is defined as

$$\mathcal{I}(X;Z) - \lambda \cdot \mathcal{I}(Z;Y), \tag{2}$$

where $\lambda > 0$. They use (2) as a regularizer to train f, minimizing $\mathcal{I}(X;Z)$ to resist MIAs while maximizing $\mathcal{I}(Z;Y)$ to preserve model's utility.

4 METHODOLOGY

4.1 DEFENSE VIA CONDITIONAL MUTUAL INFORMATION

We aim to resist black-box MIAs, so we still focus on $\hat{Y}$ rather than Z. Furthermore, we observe that all MIA algorithms target one fixed label during attacking. Formally, let

$$D^y \coloneqq \{x \in \mathbb{X} \mid (x,y) \in D\}$$

be the sub-dataset whose ground truth label is y. Given $y \in \mathbb{Y}$, all MIA algorithms aim to reconstruct $\hat{D}^y$ as close to D^y as possible. Against their intention, we propose to minimize

$$\mathcal{I}(X;\hat{Y}|Y=y) \coloneqq \sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}} \mathcal{P}(x,\hat{y}|y)\log\frac{\mathcal{P}(x,\hat{y}|y)}{\mathcal{P}(x|y)\mathcal{P}(\hat{y}|y)}. \tag{3}$$

$\mathcal{I}(X;\hat{Y}|Y=y)$ quantifies the dependence between X and $\hat{Y}$ when $Y=y$. We minimize it to prevent attackers from obtaining the information about D^y. Minimizing (3) on each $y \in \mathbb{Y}$ is equivalent to minimizing the conditional mutual information (CMI), which is defined as

$$\mathcal{I}(X;\hat{Y}|Y) \coloneqq \sum_{y\in\mathbb{Y}} \mathcal{P}(y)\cdot\mathcal{I}(X;\hat{Y}|Y=y). \tag{4}$$

Theorem 1. *CMI is a special case of information bottlenecks (IB) when $Z=\hat{Y}$ and $\lambda = 1$, i.e.*

$$\mathcal{I}(X;\hat{Y}|Y) = \mathcal{I}(X;\hat{Y}) - \mathcal{I}(\hat{Y};Y).$$

Our proof is provided in Appendix A. Theorem 1 proves that CMI inherits the benefits of IB, including two aspects:

- Minimizing $\mathcal{I}(X;\hat{Y})$ to compress the redundant information in inputs, as well as decreasing the dependence between inputs and predictions. This helps to resist MIAs as shown in MID (Wang et al., 2021).
- Maximizing $\mathcal{I}(\hat{Y};Y)$ to preserve the useful information for tasks, as well as increasing the dependence between predictions and ground truths. This helps to improve model's utility obviously.

The $\mathcal{I}(X;Z)$ in (2) is challenging to calculate because the input space $\mathbb{X}$ and feature space $\mathbb{Z}$ are both high-dimensional. Previous work had to estimate the variational bounds of IB (Tishby et al., 1999; Tishby & Zaslavsky, 2015; Alemi et al., 2017; Shwartz-Ziv & Tishby, 2017). Fortunately, as a special case of IB, CMI can be calculated and minimized directly, as described in the next section.

4.2 MINIMIZE CMI VIA POST-PROCESSING

Previous work used CMI as a regularizer and minimized it during training models (Yang et al., 2024; Hamidi et al., 2024; Yang et al., 2025). Unlike them, we propose to minimize CMI via post-processing.

CMI can be calculated as follows:

$$
\begin{aligned}
\mathcal{I}(X;\hat{Y}|Y) &= \sum_{y\in\mathbb{Y}}\mathcal{P}(y)\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(x,\hat{y}|y)\log\frac{\mathcal{P}(x,\hat{y}|y)}{\mathcal{P}(x|y)\mathcal{P}(\hat{y}|y)}, && \text{by definitions (3-4)},\\
&= \sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\sum_{y\in\mathbb{Y}}\mathcal{P}(x,\hat{y},y)\log\frac{\mathcal{P}(\hat{y}|x,y)}{\mathcal{P}(\hat{y}|y)},\\
&= \sum_{x\in\mathbb{X}}\mathcal{P}(x)\sum_{y\in\mathbb{Y}}\mathcal{P}(y|x)\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(\hat{y}|x,y)\log\frac{\mathcal{P}(\hat{y}|x,y)}{\mathcal{P}(\hat{y}|y)},\\
&= \sum_{x\in\mathbb{X}}\mathcal{P}(x)\sum_{y\in\mathbb{Y}}\mathcal{P}(y|x)\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(\hat{y}|x)\log\frac{\mathcal{P}(\hat{y}|x)}{\mathcal{P}(\hat{y}|y)}, && \text{by Markov } Y\to X\to\hat{Y}.
\end{aligned}
$$

Based on the above mathematical transformation, minimizing $\mathcal{I}(X;\hat{Y}|Y)$ is equivalent to minimizing $\sum_{y\in\mathbb{Y}}\mathcal{P}(y|x)\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(\hat{y}|x)\log\frac{\mathcal{P}(\hat{y}|x)}{\mathcal{P}(\hat{y}|y)}$ for each x input to f. However, this objective function is too complex to optimize. For simplicity, we sample $y\in\mathbb{Y}$ with the probability $\mathcal{P}(y|x)$ and minimize $\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(\hat{y}|x)\log\frac{\mathcal{P}(\hat{y}|x)}{\mathcal{P}(\hat{y}|y)}$ instead, which is equivalent to the original objective in terms of mathematical expectation. Next, we find a way to calculate $\mathcal{P}(\hat{y}|x)$ and $\mathcal{P}(\hat{y}|y)$.

We consider $\mathcal{P}(\hat{y}|x)=f_{\hat{y}}(x)$ according to the design of neural classifiers. Note that

$$
\begin{aligned}
\mathcal{P}(\hat{y}|y) &= \sum_{x\in\mathbb{X}}\mathcal{P}(x,\hat{y}|y)=\sum_{x\in\mathbb{X}}\mathcal{P}(x|y)\mathcal{P}(\hat{y}|x,y)=\sum_{x\in\mathbb{X}}\mathcal{P}(x|y)\mathcal{P}(\hat{y}|x)=\sum_{x\in\mathbb{X}}\mathcal{P}(x|y)f_{\hat{y}}(x),\\
&= \mathbb{E}_{X|Y=y}[f_{\hat{y}}(X)], \qquad \hat{y},y\in\mathbb{Y}.
\end{aligned}
$$

By expressing $\mathcal{P}(\hat{y}|y)$ as a mathematical expectation, we can estimate it with the sample mean. Note that the samples in D^y are i.i.d. with $X|Y=y$, so we consider[2]

$$
\mathcal{P}(\hat{y}|y)\approx\operatorname*{mean}_{x'\in D^y}f_{\hat{y}}(x'), \qquad \hat{y},y\in\mathbb{Y}.
$$

Let $\boldsymbol{q}^y\coloneqq\operatorname*{\mathbf{mean}}_{x'\in D^y}\boldsymbol{f}(x')$ and $q^y_{\hat{y}}$ be the $\hat{y}$-th component of $\boldsymbol{q}^y$, $\hat{y}\in\mathbb{Y}$. We have

$$
\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(\hat{y}|x)\log\frac{\mathcal{P}(\hat{y}|x)}{\mathcal{P}(\hat{y}|y)}\approx\sum_{\hat{y}\in\mathbb{Y}}f_{\hat{y}}(x)\log\frac{f_{\hat{y}}(x)}{q^y_{\hat{y}}}=\mathrm{KL}(\boldsymbol{f}(x)||\boldsymbol{q}^y),
$$

where KL is the Kullback-Leibler divergence, a binary convex function.

To minimize $\mathrm{KL}(\boldsymbol{f}(x)||\boldsymbol{q}^y)$, we fix $\boldsymbol{q}^y$ for simplicity and modify $\boldsymbol{f}(x)$. Let $\boldsymbol{p}\in\Delta^{\mathbb{Y}}$ be the modified output, and then our objective is $\mathrm{KL}(\boldsymbol{p}||\boldsymbol{q}^y)$. To preserve the model's utility, we add constrain $\|\boldsymbol{p}-\boldsymbol{f}(x)\|_1\le\varepsilon$ where $\varepsilon>0$ is the distortion bound.

In rate-distortion theory (Shannon, 1959), minimizing mutual information under bounded distortion constraint is for signal compression. If a signal has less information, it is easier to compress, and a stricter distortion bound can be applied. Inspired by their work, we introduce the normalized Shannon entropy to quantify the information in $\boldsymbol{f}(x)$, which is defined as

$$
\bar{\mathcal{H}}(x)\coloneqq\frac{-1}{\log|\mathbb{Y}|}\sum_{\hat{y}\in\mathbb{Y}}f_{\hat{y}}(x)\log f_{\hat{y}}(x).
$$

Smaller $\bar{\mathcal{H}}(x)$ implies less information in $\boldsymbol{f}(x)$, and a stricter distortion bound can be applied. So we constraint $\|\boldsymbol{p}-\boldsymbol{f}(x)\|_1\le\varepsilon\cdot\bar{\mathcal{H}}(x)$ to further control the distortion. Note that the old constraint $\|\boldsymbol{p}-\boldsymbol{f}(x)\|_1\le\varepsilon$ still holds due to the property of $0\le\bar{\mathcal{H}}(x)\le 1$. This practice is called *adaptive rate-distortion*.

[2]We use the validation set as D^y in practice, because neural networks tend to overfit the training samples, leading to inaccurate estimates.

To determine the sampling probability $\mathcal{P}(y|x)$, a simple idea is to consider

$$\mathcal{P}(y|x) \approx \mathcal{P}(\hat{y}|x) = f_{\hat{y}}(x) \text{ for } y = \hat{y} \in \mathbb{Y}.$$

But Guo et al. (2017) have demonstrated that it is inaccurate for modern neural networks. Inspired by their work, we introduce *temperature mechanism* to calibrate it.

Our defense is summarized as Algorithm 1. Note that the $\boldsymbol{q}^y, y \in \mathbb{Y}$ can be calculated and stored in advance, which helps to reduce the computational cost and protect privacy[3].

Algorithm 1: Our post-processing to minimize CMI.

Input: original output $\boldsymbol{f}(x)$, temperature T, distortion bound ε, validation set D.
Output: modified output $\boldsymbol{p}$.
$y \leftarrow$ Sample in $\mathbb{Y}$ with the probability of **softmax**$(\frac{\boldsymbol{f}(x)}{T})$;
$\boldsymbol{q}^y \leftarrow \underset{x' \in D^y}{\textbf{mean}}\, \boldsymbol{f}(x')$;
$\bar{\mathcal{H}}(x) \leftarrow \frac{-1}{\log |\mathbb{Y}|} \sum_{\hat{y} \in \mathbb{Y}} f_{\hat{y}}(x) \log f_{\hat{y}}(x)$;
Solve the convex optimization problem and return the optimal $\boldsymbol{p}$:

$$\begin{aligned} &\min \mathrm{KL}(\boldsymbol{p} || \boldsymbol{q}^y), \\ &\text{s.t. } \|\boldsymbol{p} - \boldsymbol{f}(x)\|_1 \leq \varepsilon \cdot \bar{\mathcal{H}}(x), \\ &\quad \boldsymbol{p} \in \Delta^{\mathbb{Y}}. \end{aligned} \tag{5}$$

(5) is a convex optimization problem that can be solved by optimizers. Furthermore, we provide an efficient solution in Appendix C and evaluate its time cost in Appendix D. Without the need to retrain the model, our algorithm is plug-and-play and easy to deploy.

5 Experiment

5.1 Experiment Settings

Datasets. Following the previous work of MIAs, we use FaceScrub (Ng & Winkler, 2014) and CelebA (Liu et al., 2015) as private datasets. FaceScrub consists of 530 identities. CelebA contains 10177 identities and we only take 1000 identities with the most images (Kahla et al., 2022). All images are cropped and resized to 64×64 pixels. We use 80% of the data as the training set, and 10% as the validation and test sets. The validation set is used to select the model and adjust the hyperparameters of the defenses.

Models. For target models, we employ VGG-16 (Simonyan & Zisserman, 2014) and IR-152 (He et al., 2016), both of which are trained with different defense methods. We select FaceNet (Cheng et al., 2017) as the evaluation model.

Model inversion attacks. We focus on four state-of-the-art black-box MIAs, including BREP (Kahla et al., 2022), Mirror (An et al., 2022), C2FMI (Ye et al., 2023) and LOKT (Nguyen et al., 2024). We attack the first 100 classes in the private dataset, reconstructing 5 images for each class. For BREP and LOKT attacks, we use the FFHQ (Karras et al., 2020) dataset to train GANs and surrogate models under official settings. For Mirror and C2FMI, we adopt the pre-trained 256×256 GANs with FFHQ prior provided by (Karras et al., 2020). The generated images will be center-cropped to 176×176 and then resized to 64×64.

Metrics. To measure the MIA robustness and model's utility, we consider the following metrics:

- **Attack Accuracy.** The metric is used to imitate a human to determine whether reconstructed images correspond to the target identity or not. Specifically, we employ an evaluation model trained on the same dataset as the target model to re-classify the reconstructed images. We compute the top-1 and top-5 classification accuracies, denoted as $Acc@1$ and $Acc@5$, respectively.

[3]If the owner of the model and the executor of the post-processing are different, the owner only needs to provide the $\boldsymbol{q}^y, y \in \mathbb{Y}$ instead of D, protecting the privacy of the owner.

- **Feature Distance.** The feature is extracted from the second-to-last layer of the model. This distance metric measures the average l_2 distance between the features of reconstructed images and the nearest private images. Consistent with previous research, we use both the evaluation model and a pre-trained FaceNet (Schroff et al., 2015) to generate the features. The corresponding feature distances are denoted as σ_{eval} and σ_{face}. A lower feature distance indicates a closer semantic similarity between the reconstructed images and private samples.
- **Test Accuracy.** The top-1 classification accuracy on the private test set. This metric is used to evaluate the utility of the target model with defense.
- **Prediction Bias.** This metric is used to quantify the modification to the predicted probability vectors by defense methods. We take the L_1 distance between the outputs with and without defense. Avg L_1 is the average over private test samples, and Max L_1 is the largest one. Lower values of both suggest that the defense method causes less modification to the outputs.

All experiments are conducted by MIBench (Qiu et al., 2024b).

5.2 Comparison with Previous State-of-the-art Defenses

In this section, we evaluate the robustness of our defense by comparing it against an undefended model and prior state-of-the-art defenses, including MID (Wang et al., 2021), BiDO (Peng et al., 2022), LS (Struppek et al., 2024) and TL (Ho et al., 2024). We adhere to the official configurations for each defense method, and the corresponding hyperparameters are detailed in Appendix B.

We evaluate the MIA robustness under various black-box MIAs, including both soft-label and hard-label attacks. We conduct experiments on different target models and private datasets to demonstrate that our approach performs effectively across diverse scenarios.

For soft-label attacks, we compare our method with previous defense strategies under the Mirror and C2FMI attacks. The attack results are listed in Table 1. We can observe that our SSD achieves significant improvements over existing defense strategies, especially when the attack has a strong performace. Specifically, under the Mirror attack against IR-152 trained on the FaceScrub dataset, our method reduces the attack accuracy from 52.4% to 19.4%, achieving a 3.6% greater reduction compared to the previous SOTA method TL. For C2FMI attacks against VGG16 models trained on the FaceScrub dataset, our method reduces the attack accuracy to approximately $1/9$ of that without defense, which is only a quarter of the accuracy achieved under the TL defense.

Table 1: MIA robustness against soft-label attacks.

Model Dataset	Defense	Mirror				C2FMI			
		$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
IR-152 CelebA	**None**	10.0%	18.8%	2526	1.31	3.6%	8.0%	2521	1.36
	MID	9.0%	17.6%	2448	1.23	0.2%	0.4%	2382	1.56
	BiDO	4.8%	11.4%	**2758**	1.17	0.8%	3.8%	**2598**	1.31
	LS	3.2%	7.8%	2602	1.33	1.4%	4.2%	2536	1.39
	TL	6.6%	14.4%	2613	1.27	2.6%	7.0%	2528	1.37
	SSD	**1.2%**	**3.0%**	2527	**1.56**	**0%**	**0.4%**	2377	**1.67**
IR-152 FaceScrub	**None**	52.4%	74.6%	1893	0.79	27.0%	49.8%	1952	0.98
	MID	43.6%	63.4%	2067	0.86	3.0%	9.6%	**2754**	1.44
	BiDO	27.6%	53.0%	2132	0.99	14.2%	24.4%	2242	1.20
	LS	33.4%	56.6%	2153	0.88	21.8%	46.8%	2022	1.02
	TL	23.0%	47.2%	2155	0.95	6.8%	16.8%	2191	1.23
	SSD	**19.4%**	**28.2%**	**2415**	**1.31**	**2.0%**	**6.4%**	2517	**1.49**
VGG-16 FaceScrub	**None**	8.0%	15.0%	2577	0.78	23.8%	37.0%	2315	0.93
	MID	6.4%	12.2%	2627	0.79	18.4%	31.8%	2239	0.93
	BiDO	11.4%	21.0%	2530	0.79	10.6%	19.2%	2552	0.94
	LS	10.2%	18.4%	2526	0.75	17.0%	29.2%	2424	0.95
	TL	6.8%	12.0%	2624	**0.88**	10.4%	17.6%	2602	1.03
	SSD	**5.6%**	**10.6%**	**2665**	0.80	**8.8%**	**15.2%**	**2681**	**1.07**

In hard-label scenarios with BREP and LOKT attacks, we provided a quantitive results in Table 2. Note that LOKT is the SOTA black-box attack method. It demonstrates very high attack performance across various kinds of settings. While previous defenses only showed limited defensive capabilities, our SSD almost completely defeats this attack. Especially in the attack against IR-152 with FaceScrub dataset, without any defense, LOKT showed an attack accuracy of up to 83.0%. However, our defense method reduce it to only 1.8%, making it almost impossible to launch a successful attack. Moreover, our defense largely enhance the feature distance σ_{face} from 0.66 to 1.53, which indicate that our defense method make the attack failed to capture the privacy characteristics.

Table 2: MIA robustness against hard-label attacks.

Model Dataset	Defense	BREP				LOKT			
		$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
IR-152 CelebA	**None**	7.2%	24.4%	1654	0.95	51.6%	74.4%	1469	0.85
	MID	12.6%	28.8%	1973	1.28	29.8%	51.0%	1713	1.04
	BiDO	13.0%	30.6%	1670	1.03	48.4%	66.8%	1551	0.95
	LS	15.6%	40.0%	1584	0.97	52.0%	73.6%	1489	0.88
	TL	10.2%	27.2%	1643	1.05	56.4%	74.6%	1510	0.92
	SSD	**0.4%**	**1.6%**	**2362**	**1.61**	**0.2%**	**1.0%**	**2321**	**1.54**
IR-152 FaceScrub	**None**	32.8%	56.6%	2161	1.00	83.0%	93.2%	1488	0.66
	MID	34.0%	51.0%	2178	1.06	54.0%	74.4%	1856	0.82
	BiDO	24.2%	39.4%	2235	1.07	59.8%	77.6%	1694	0.77
	LS	22.8%	45.8%	2384	1.07	60.0%	77.6%	1748	0.74
	TL	14.2%	27.2%	2353	1.15	62.6%	78.2%	1682	0.73
	SSD	**3.4%**	**7.0%**	**2622**	**1.51**	**1.8%**	**4.4%**	**2694**	**1.53**
VGG-16 FaceScrub	**None**	33.6%	56.6%	2327	0.94	93.8%	98.0%	1359	0.57
	MID	37.4%	58.2%	2249	0.90	82.4%	92.8%	1526	0.60
	BiDO	30.4%	51.8%	2349	0.96	78.8%	87.4%	1567	0.63
	LS	29.6%	49.0%	2402	0.94	78.2%	88.6%	1573	0.65
	TL	29.0%	47.8%	2381	0.98	58.2%	74.0%	1771	0.71
	SSD	**9.8%**	**15.0%**	**2586**	**1.45**	**12.6%**	**21.4%**	**2370**	**1.18**

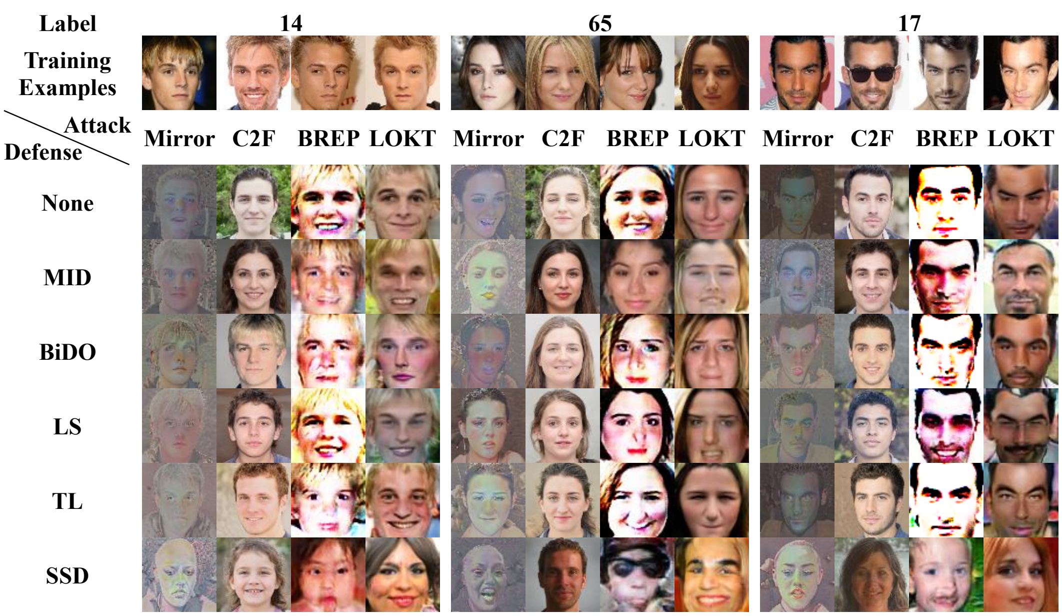

Figure 2: Visual comparison of reconstructed images using various black-box attack methods against an IR-152 model trained on CelebA, evaluated under different defense strategies. The top row displays the images of the target class from the private train dataset for reference.

Visualization results of the reconstructed images with different defenses under different black-box attacks are shown in Fig. 2. Compared to previous approaches, our SSD produces reconstructed images that deviate more significantly from the private images, demonstrating its effectiveness in increasing the challenge for attackers to extract sensitive visual features and thereby enhancing privacy protection.

Table 3: Evaluation results on model's utility.

Defense	IR-152 & CelebA			IR-152 & FaceScrub			VGG-16 & FaceScrub		
	↑Acc	↓Avg L_1	↓Max L_1	↑Acc	↓Avg L_1	↓Max L_1	↑Acc	↓Avg L_1	↓Max L_1
None	94.2%	0	0	98.6%	0	0	97.9%	0	0
MID	88.9%	0.44	1.93	96.5%	0.32	1.96	95.1%	0.36	1.78
BiDO	88.2%	0.37	1.96	94.0%	0.58	1.95	94.3%	0.27	1.90
LS	90.1%	0.37	1.99	94.9%	0.18	1.96	94.9%	0.19	1.88
TL	89.1%	0.35	1.84	95.3%	0.33	1.97	94.5%	0.15	1.96
SSD	**90.3%**	**0.15**	**0.95**	**96.7%**	**0.06**	**0.94**	**96.3%**	**0.05**	**0.74**

The evaluation results for the target model's utility are presented in Table 3. The results indicate that our SSD holds the best utility, outperforming all competitors across different metrics, training datasets and model structures. According to our bounded distortion constraint, our Max $L_1 \leq \varepsilon$ always holds strictly, where the competitors' are close to the maximum of 2. In particular, our Avg L_1 is only 1/5 to 1/2 of the competitors'.

5.3 Ablation Studies

In this section, we conduct ablation experiments to explore the effects of the temperature and distortion bound in our SSD. The target model is IR-152 trained on FaceScrub. The results are shown in Figure 3.

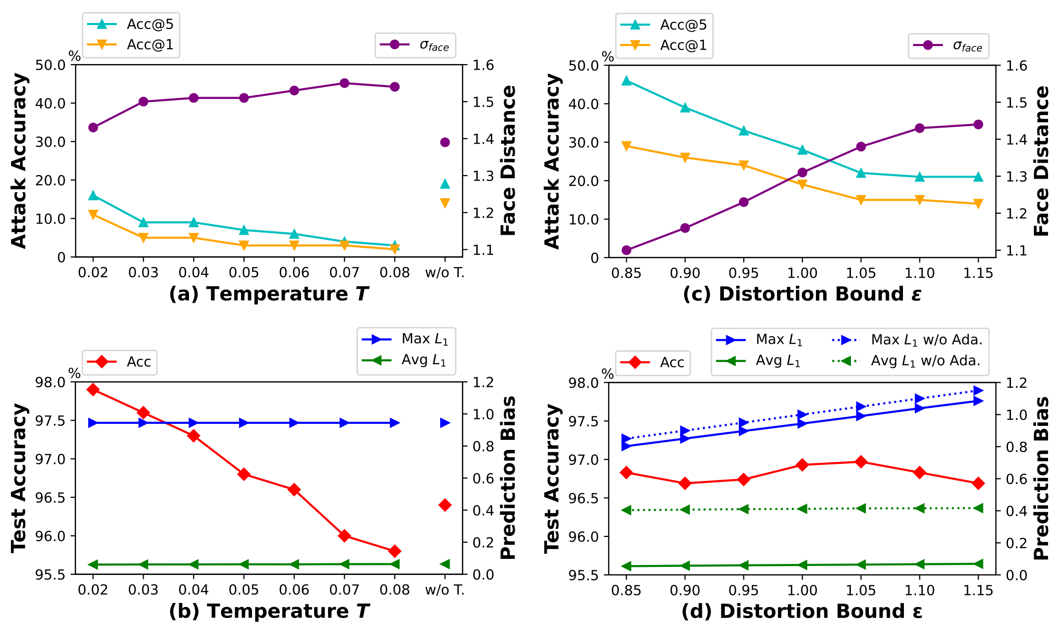

Figure 3: Ablation Study on temperature T and distortion bound ε.

Figure (a)(b) show the results on temperature T, where the attack accuracy is measured on BREP. It can be seen that as the temperature T rises, our MIA robustness becomes stronger. This is because the sampling probability in Algorithm 1 is closer to the uniform distribution, which makes it easier to return misleading labels to hard-label attackers. However, high temperature impairs the model's utility. In particular, the "w/o T." in Figure (a)(b) represents the case without temperature mechanism. In that case, neither MIA robustness nor model's utility is good, which demonstrates the necessity of introducing a temperature mechanism.

For the distortion bound, the results are displayed in Figure (c)(d). The attack accuracy is measured on Mirror. As the distortion bound goes up, our defense can make more modifications to the output, resulting in better MIA robustness. It can be seen that relaxing the distortion bound mainly affects the maximum distortion Max L_1, while having almost no effect on the average distortion Avg L_1. Especially, without the adaptive mechanism, our Avg L_1 would become as high as other defenses. This demonstrates the necessity of introducing the adaptive mechanism.

6 Conclusion

In contrast to previous researches on model inversion defense with focus on white-box attacks, we conduct a specific study on black-box attacks. Specifically, we investigate the impact of conditional mutual information (CMI) and develop a CMI-based defense strategy. We conduct our defense in the post-processing stage, instead of re-training the model. Our method modify the model output by reducing the dependence between model inputs and outputs. To further reduce the modifications to outputs, we introduce an adaptive rate-distortion framework and optimize it by water-filling method. Experimental results demonstrate that our defense method achieves state-of-the-art (SOTA) performance against black-box attacks. We hope that our findings will help shift attention towards robust defense mechanisms in black-box settings and inspire further research in this area.

7 Acknowledgement

This work is supported in part by National Natural Science Foundation of China (62171248, 62301189), Peng Cheng Laboratory (PCL2023A08), and Shenzhen Science and Technology Program (KJZD20240903103702004, JCYJ20220818101012025, RCBS20221008093124061, GXWD20220811172936001).

References

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations (ICLR)*, 2017.

Shengwei An, Guanhong Tao, Qiuling Xu, Yingqi Liu, Guangyu Shen, Yuan Yao, Jingwei Xu, and Xiangyu Zhang. Mirror: Model inversion for deep learning network with high fidelity. In *Network and Distributed System Security Symposium (NDSS)*, 2022.

Si Chen, Mostafa Kahla, Ruoxi Jia, and Guo-Jun Qi. Knowledge-enriched distributional model inversion attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 16178–16187, 2021.

Yu Cheng, Jian Zhao, Zhecan Wang, Yan Xu, Karlekar Jayashree, Shengmei Shen, and Jiashi Feng. Know you at one glance: A compact vector representation for low-shot learning. In *ICCVW*, pp. 1924–1932, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Hao Fang, Bin Chen, Xuan Wang, Zhi Wang, and Shu-Tao Xia. Gifd: A generative gradient inversion method with feature domain optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4967–4976, 2023.

Hao Fang, Jiawei Kong, Bin Chen, Tao Dai, Hao Wu, and Shu-Tao Xia. Clip-guided generative networks for transferable targeted adversarial attacks. In *European Conference on Computer Vision*, pp. 1–19. Springer, 2024a.

Hao Fang, Jiawei Kong, Wenbo Yu, Bin Chen, Jiawei Li, Shutao Xia, and Ke Xu. One perturbation is enough: On generating universal adversarial perturbations against vision-language pre-training models. *arXiv preprint arXiv:2406.05491*, 2024b.

Hao Fang, Yixiang Qiu, Hongyao Yu, Wenbo Yu, Jiawei Kong, Baoli Chong, Bin Chen, Xuan Wang, and Shu-Tao Xia. Privacy leakage on dnns: A survey of model inversion attacks and defenses. *ArXiv*, 2024c.

Hao Fang, Xiaohang Sui, Hongyao Yu, Jiawei Kong, Sijin Yu, Bin Chen, Hao Wu, and Shu-Tao Xia. Retrievals can be detrimental: A contrastive backdoor attack paradigm on retrieval-augmented diffusion models. *arXiv preprint arXiv:2501.13340*, 2025.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, 2017.

Shayan Mohajer Hamidi, Renhao Tan, Linfeng Ye, and En-Hui Yang. Fed-it: Addressing class imbalance in federated learning through an information- theoretic lens. In *IEEE International Symposium on Information Theory (ISIT)*, 2024.

Gyojin Han, Jaehyun Choi, Haeil Lee, and Junmo Kim. Reinforcement learning-based black-box model inversion attacks. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Sy-Tuyen Ho, Koh Jun Hao, Keshigeyan Chandrasegaran, Ngoc-Bao Nguyen, and Ngai-Man Cheung. Model inversion robustness: Can transfer learning help? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54 (11s):1–37, 2022.

Mostafa Kahla, Si Chen, Hoang Anh Just, and Ruoxi Jia. Label-only model inversion attacks via boundary repulsion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738, 2015.

Hong-Wei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *2014 IEEE international conference on image processing (ICIP)*, pp. 343–347, 2014.

Bao-Ngoc Nguyen, Keshigeyan Chandrasegaran, Milad Abdollahzadeh, and Ngai-Man Man Cheung. Label-only model inversion attacks via knowledge transfer. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

Xiong Peng, Feng Liu, Jingfeng Zhang, Long Lan, Junjie Ye, Tongliang Liu, and Bo Han. Bilateral dependency optimization: Defending against model-inversion attacks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2022.

Yixiang Qiu, Hao Fang, Hongyao Yu, Bin Chen, MeiKang Qiu, and Shu-Tao Xia. A closer look at gan priors: Exploiting intermediate features for enhanced model inversion attacks. In *ECCV*, 2024a.

Yixiang Qiu, Hongyao Yu, Hao Fang, Wenbo Yu, Bin Chen, Xuan Wang, Shu-Tao Xia, and Ke Xu. Mibench: A comprehensive benchmark for model inversion attack and defense. *arXiv preprint arXiv:2410.05159*, 2024b.

Yixiang Qiu, Hongyao Yu, Hao Fang, Wenbo Yu, Bin Chen, Xuan Wang, Shu-Tao Xia, and Ke Xu. Mibench: A comprehensive benchmark for model inversion attack and defense. *arXiv preprint arXiv:2410.05159*, 2024c.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Claude E. Shannon. *Coding Theorems for a Discrete Source With a Fidelity Criterion*. 1959.

Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *ArXiv*, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Lukas Struppek, Dominik Hintersdorf, Antonio De Almeida Correira, Antonia Adler, and Kristian Kersting. Plug & play attacks: Towards robust and flexible model inversion attacks. In *ICML*, 2022.

Lukas Struppek, Dominik Hintersdorf, and Kristian Kersting. Be careful what you smooth for: Label smoothing can be a privacy shield but also a catalyst for model inversion attacks. In *International Conference on Learning Representations (ICLR)*, 2024.

Naftali Tishby and Noga Zaslavsky. Deep learning and the information bottleneck principle. In *IEEE Information Theory Workshop (ITW)*, 2015.

Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. In *Annual Allerton Conference on Communication, Control, and Computing (Allerton Conference)*, 1999.

Tianhao Wang, Yuheng Zhang, and Ruoxi Jia. Improving robustness to model inversion attacks via mutual information regularization. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

En-hui Yang and Linfeng Ye. Markov knowledge distillation: Make nasty teachers trained by self-undermining knowledge distillation fully distillable. In *European Conference on Computer Vision (ECCV)*, 2024.

En-Hui Yang, Shayan Mohajer Hamidi, Linfeng Ye, Renhao Tan, and Beverly Yang. Conditional mutual information constrained deep learning: Framework and preliminary results. In *IEEE International Symposium on Information Theory (ISIT)*, 2024.

En-Hui Yang, Shayan Mohajer Hamidi, Linfeng Ye, Renhao Tan, and Beverly Yang. Conditional mutual information constrained deep learning for classification. *IEEE Transactions on Neural Networks and Learning Systems (TNNLS)*, 2025.

Ziqi Yang, Lijin Wang, Da Yang, Jie Wan, Ziming Zhao, Ee-Chien Chang, Fan Zhang, and Kui Ren. Purifier: Defending data inference attacks via transforming confidence scores. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 10871–10879, 2023.

Linfeng Ye, Shayan Mohajer Hamidi, Renhao Tan, and En-Hui Yang. Bayes conditional distribution estimation for knowledge distillation based on conditional mutual information. In *International Conference on Learning Representations (ICLR)*, 2024.

Zipeng Ye, Wenjian Luo, Muhammad Luqman Naseem, Xiangkai Yang, Yuhui Shi, and Yan Jia. C2fmi: Corse-to-fine black-box model inversion attack. *IEEE Transactions on Dependable and Secure Computing (TDSC)*, 2023.

Hongyao Yu, Yixiang Qiu, Hao Fang, Bin Chen, Sijin Yu, Bin Wang, Shu-Tao Xia, and Ke Xu. Calor: Towards comprehensive model inversion defense. *arXiv preprint arXiv:2410.05814*, 2024a.

Wenbo Yu, Hao Fang, Bin Chen, Xiaohang Sui, Chuan Chen, Hao Wu, Shu-Tao Xia, and Ke Xu. Gi-nas: Boosting gradient inversion attacks through adaptive neural architecture search. *arXiv preprint arXiv:2405.20725*, 2024b.

Xiaojian Yuan, Kejiang Chen, Jie Zhang, Weiming Zhang, Nenghai Yu, and Yang Zhang. Pseudo label-guided model inversion attack via conditional generative adversarial network. In *AAAI*, 2023.

Yuheng Zhang, Ruoxi Jia, Hengzhi Pei, Wenxiao Wang, Bo Li, and Dawn Song. The secret revealer: Generative model-inversion attacks against deep neural networks. In *CVPR*, 2020.

Xinhao Zhong, Bin Chen, Hao Fang, Xulin Gu, Shu-Tao Xia, and En-Hui Yang. Going beyond feature similarity: Effective dataset distillation based on class-aware conditional mutual information. *arXiv preprint arXiv:2412.09945*, 2024a.

Xinhao Zhong, Hao Fang, Bin Chen, Xulin Gu, Tao Dai, Meikang Qiu, and Shu-Tao Xia. Hierarchical features matter: A deep exploration of gan priors for improved dataset distillation. *arXiv preprint arXiv:2406.05704*, 2024b.

A PROOF OF THEOREM 1

$$
\begin{aligned}
&\mathcal{I}(X;\hat{Y}|Y) \\
=&\sum_{y\in\mathbb{Y}}\mathcal{P}(y)\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(x,\hat{y}|y)\log\frac{\mathcal{P}(x,\hat{y}|y)}{\mathcal{P}(x|y)\mathcal{P}(\hat{y}|y)}, && \text{by definitions (3-4),}\\
=&\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\sum_{y\in\mathbb{Y}}\mathcal{P}(x,\hat{y},y)\log\frac{\mathcal{P}(\hat{y}|x,y)}{\mathcal{P}(\hat{y}|y)},\\
=&\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\sum_{y\in\mathbb{Y}}\mathcal{P}(x,\hat{y},y)\log\frac{\mathcal{P}(\hat{y}|x)}{\mathcal{P}(\hat{y}|y)}, && \text{by Markov chain } Y\to X\to\hat{Y},\\
=&\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\sum_{y\in\mathbb{Y}}\mathcal{P}(x,\hat{y},y)\log\left(\frac{\mathcal{P}(x,\hat{y})}{\mathcal{P}(x)}\Big/\frac{\mathcal{P}(\hat{y},y)}{\mathcal{P}(y)}\right),\\
=&\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\sum_{y\in\mathbb{Y}}\mathcal{P}(x,\hat{y},y)\log\left(\frac{\mathcal{P}(x,\hat{y})}{\mathcal{P}(x)\mathcal{P}(\hat{y})}\Big/\frac{\mathcal{P}(\hat{y},y)}{\mathcal{P}(\hat{y})\mathcal{P}(y)}\right),\\
=&\sum_{x\in\mathbb{X}}\sum_{\hat{y}\in\mathbb{Y}}\mathcal{P}(x,\hat{y})\log\frac{\mathcal{P}(x,\hat{y})}{\mathcal{P}(x)\mathcal{P}(\hat{y})}-\sum_{\hat{y}\in\mathbb{Y}}\sum_{y\in\mathbb{Y}}\mathcal{P}(\hat{y},y)\log\frac{\mathcal{P}(\hat{y},y)}{\mathcal{P}(\hat{y})\mathcal{P}(y)},\\
=&\mathcal{I}(X;\hat{Y})-\mathcal{I}(\hat{Y};Y), && \text{by definition (1).}
\end{aligned}
$$

B THE HYPERPARAMETERS FOR EACH DEFENSE

Table 4: The hyperparameters for each defense.

Defense	IR-152 & CelebA	IR-152 & FaceScrub	VGG-16 & FaceScrub
MID	$\beta = 0.005$	$\beta = 0.01$	$\beta = 0.02$
BiDO	$\lambda_x = 0.001, \lambda_y = 0.01$	$\lambda_x = 0.002, \lambda_y = 0.02$	$\lambda_x = 0.002, \lambda_y = 0.02$
LS	$\alpha = -0.05$	$\alpha = -0.1$	$\alpha = -0.1$
TL	Freeze the first 50% of the layers.		
SSD	$T = 0.03, \varepsilon = 1$	$T = 0.05, \varepsilon = 1$	$T = 0.3, \varepsilon = 1$

C OUR WATER-FILLING ALGORITHM TO OPTIMIZE (5)

For brevity, let $\boldsymbol{q} \coloneqq \tilde{\boldsymbol{q}}^y$, $\boldsymbol{f} \coloneqq \boldsymbol{f}(x)$, and $\varepsilon \coloneqq \varepsilon \cdot \bar{\mathcal{H}}(x)$. The problem (5) is restated as

$$
\begin{aligned}
&\min \mathrm{KL}(\boldsymbol{p}||\boldsymbol{q}),\\
&\text{s.t. } \|\boldsymbol{p}-\boldsymbol{f}\|_1 \le \varepsilon,\\
&\boldsymbol{p}\in\Delta^{\mathbb{Y}}.
\end{aligned} \tag{6}
$$

Note that Kullback-Leibler divergence is a metric. $\mathrm{KL}(\boldsymbol{p}||\boldsymbol{q}) \ge 0$ always holds and $\mathrm{KL}(\boldsymbol{p}||\boldsymbol{q}) = 0$ iff $\boldsymbol{p} = \boldsymbol{q}$. Trivially, when $\|\boldsymbol{q}-\boldsymbol{f}\|_1 \le \varepsilon$, the optimal solution is $\boldsymbol{p} = \boldsymbol{q}$.

When $\|\boldsymbol{q}-\boldsymbol{f}\|_1 > \varepsilon$, the optimal $\boldsymbol{p}$ must be between $\boldsymbol{f}$ and $\boldsymbol{q}$ due to the properties of KL, i.e.

$$\text{Either } f_i \le p_i \le q_i \text{ or } f_i \ge p_i \ge q_i, \text{ for each } i\in\mathbb{Y}. \tag{7}$$

Furthermore, due to $\boldsymbol{f},\boldsymbol{p}\in\Delta^{\mathbb{Y}}$, there must be

$$\sum_{i\in\mathbb{Y}:f_i<q_i} p_i - f_i = \sum_{i\in\mathbb{Y}:f_i>q_i} f_i - p_i = \frac{\varepsilon}{2}. \tag{8}$$

In the following we consider the case $f_i < q_i$ (another is symmetric). Assuming that $f_i < q_i$ iff $i \in \{1, 2, \ldots, n\}$, a semi-problem of (6) is

$$\begin{aligned} \min \ & \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}, \\ \text{s.t.} \ & \sum_{i=1}^{n} p_i - f_i = \frac{\varepsilon}{2}, \\ & p_i \geq f_i, \qquad i = 1, 2, \ldots, n. \end{aligned} \tag{9}$$

Introducing Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^n_{\geq 0}$ and $v \in \mathbb{R}$, the KKT conditions are

$$(p_i - f_i)\lambda_i = 0, \tag{10}$$

$$1 + \log \frac{p_i}{q_i} - v - \lambda_i = 0, \tag{11}$$

where $i = 1, 2, \ldots, n$. Eliminating $\lambda_i \geq 0$ yields

$$(p_i - f_i)\left(1 + \log \frac{p_i}{q_i} - v\right) = 0, \tag{12}$$

$$1 + \log \frac{p_i}{q_i} \geq v. \tag{13}$$

When $v > 1 + \log \frac{f_i}{q_i}$, (13) implies $p_i > f_i$, and (12) implies $p_i = q_i \exp(v - 1)$.

When $v \leq 1 + \log \frac{f_i}{q_i}$, $p_i > f_i$ implies $\left(1 + \log \frac{p_i}{q_i} - v\right) > 0$ that against (12), so $p_i = f_i$.

In summary, the optimal solution is

$$p_i = \begin{cases} q_i \exp(v - 1) & v > 1 + \log \dfrac{f_i}{q_i} \\ f_i & \text{other} \end{cases}, \qquad i = 1, 2, \ldots, n, \tag{14}$$

where v is determined by the constraint $\sum_{i=1}^{n} p_i - f_i = \frac{\varepsilon}{2}$.

Let $w \coloneqq \exp(v - 1) \in \mathbb{R}_{>0}$ and (14) is simplified to

$$p_i = \max(f_i, wq_i), \qquad i = 1, 2, \ldots, n. \tag{15}$$

We propose Algorithm 2 to calculate (15) efficiently. Our algorithm is known as "water-filling", because w is like a rising water level and $\frac{\varepsilon}{2}$ is like the maximum volume of water. Its time complexity is $O(n \log n)$ due to the sorting at the beginning.

Algorithm 2: Water-filling on CPU.

Input: f_i, q_i for $i = 1, 2, \ldots, n$.
Output: p_i for $i = 1, 2, \ldots, n$.
Reindex f_i, q_i so that $\frac{f_1}{q_1} \leq \frac{f_2}{q_2} \leq \ldots \leq \frac{f_n}{q_n}$;
$i \leftarrow 1$;
$f_{\text{sum}} \leftarrow 0$;
$q_{\text{sum}} \leftarrow 0$;
while $q_{sum} \frac{f_i}{q_i} - f_{sum} < \frac{\varepsilon}{2}$ **do**
  $i \leftarrow i + 1$;
  $f_{\text{sum}} \leftarrow f_{\text{sum}} + f_i$;
  $q_{\text{sum}} \leftarrow q_{\text{sum}} + q_i$;
end
$w \leftarrow \dfrac{f_{\text{sum}} + \frac{\varepsilon}{2}}{q_{\text{sum}}}$;
Reindex f_i, q_i back to the original;
return $\max(f_i, wq_i)$ for $i = 1, 2, \ldots, n$;

Algorithm 3: Water-filling on GPU.

Input: PyTorch tensors $\boldsymbol{f}, \boldsymbol{q}$ of size n.
Output: PyTorch tensor $\boldsymbol{p}$ of size n.
Reindex $\boldsymbol{f}, \boldsymbol{q}$ by torch.sort($\frac{\boldsymbol{f}}{\boldsymbol{q}}$);
$\boldsymbol{f}_{\text{sum}} \leftarrow \boldsymbol{f}$.cumsum();
$\boldsymbol{q}_{\text{sum}} \leftarrow \boldsymbol{q}$.cumsum();
mask $\leftarrow \boldsymbol{q}_{\text{sum}} \frac{\boldsymbol{f}}{\boldsymbol{q}} - \boldsymbol{f}_{\text{sum}} < \frac{\varepsilon}{2}$;
$i \leftarrow$ **mask**.argmax();
$w \leftarrow \dfrac{\boldsymbol{f}_{\text{sum}}[i] + \frac{\varepsilon}{2}}{\boldsymbol{q}_{\text{sum}}[i]}$;
Reindex $\boldsymbol{f}, \boldsymbol{q}$ back to the original;
return torch.max($\boldsymbol{f}, w\boldsymbol{q}$);

To further speed up, we also propose Algorithm 3, a GPU-based water-filling. Specifically, we manage to eliminate the loop and branch in Algorithm 2, making it completely sequential and suitable for GPUs. By utilizing the operators of PyTorch tensors, we fully leverage the parallelism capabilities of GPUs.

D EXPERIMENTS ON COMPUTATIONAL COST

We quantitatively demonstrate the efficiency of our post-processing Algorithm 1 by experiments. The target models, training sets, and defense settings are consistent with Table 4. We take a batch with 512 test samples and let the model infer 100 times on it. We record the time cost by torch.profiler, an official tool provided by PyTorch. We exclude the time for I/O (i.e. the time from disk to memory, and from CPU to GPU), and only include the time for forward propagation on GPU. Our experiment is conducted on one NVIDIA GeForce RTX 3090. The results are in Table 5.

Table 5: The time cost of our post-processing algorithm.

	IR-152 & CelebA	IR-152 & FaceScrub	VGG-16 & FaceScrub
Time without defense	18.63 s	17.70 s	5.65 s
Time with our defense	19.22 s	18.16 s	6.07 s
Percent of increased time	3.1%	2.5%	7.4%

It can be seen that we only increase the time by 2.5% to 7.4%. The higher percent on VGG is due to the shallower model structure. In absolute terms, modifying 512 predictions for 100 times only needs 0.5 seconds. If we take the I/O time into account, the percents will be small enough to be ignored.

We further investigate the relationship between $|\mathbb{Y}|$ and the time cost of our Algorithm 3. We generate $\boldsymbol{s} \in \mathbb{R}^{|\mathbb{Y}|} \sim N(\boldsymbol{0}, \boldsymbol{I})$ and let $\boldsymbol{r} \leftarrow \text{softmax}(10\boldsymbol{s})$. It is observed that the $\boldsymbol{r}$ generated in this way is close to the real probability distributions. We use these $\boldsymbol{r}$ to simulate the real $\boldsymbol{f}(x)$ and $\boldsymbol{q}^y$, and let our GPU-based water-filling to find the optimal solution $\boldsymbol{p}$. We take a batch with 256 pairs $(\boldsymbol{f}(x), \boldsymbol{q}^y)$ and solve in parallel. The time costs are shown in Table 6.

Table 6: The relationship between $|\mathbb{Y}|$ and the time cost of our GPU-based water-filling.

$\lvert\mathbb{Y}\rvert$	10^1	10^2	10^3	10^4	10^5	10^6
Time	131 ms	132 ms	143 ms	163 ms	249 ms	1301 ms

It shows that even when $|\mathbb{Y}|$ reaches a million, solving 256 convex optimization problems only takes 1.3 seconds. We believe that at this point, our post-processing will not be the performance bottleneck, but the slow inferring and massive parameters of the target model will be.

E EXPERIMENTS UNDER RLB ATTACK

We evaluate the all defenses' MIA robustness against RLB (Han et al., 2023), a SOTA soft-label attack method. All settings are consistent with Tables 1-4, where the target model is IR-152 and the private dataset is CelebA. The first 10 classes of CelebA are attacked and each class reconstructed 5 images. The results are shown in Table 7.

Table 7: The MIA robustness of all defense under RLB attack.

	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
No Defense	32%	64%	2006	0.77
MID	30%	48%	2088	0.84
BiDO	16%	28%	2254	0.94
LS	12%	34%	2204	0.85
TL	22%	34%	2107	0.82
SSD (ours)	**8%**	**12%**	**2480**	**1.26**

It can be seen that our defense has the best MIA robustness against RLB. The models' utility and defenses' settings are consistent with the Tables 3-4, which shows that we also preserve the best model's utility.

F EXPERIMENTS ON HIGH RESOLUTION

To adapt to high resolution, we choose Mirror as the attacker. The prior distribution is StyleGAN2 trained on FFHQ with a resolution of 1024×1024. The generated images are center-cropped to 800×800, resized to 224×224, and inputted to the target model. The target model is ResNet-152, and the evaluation model is Inception-v3. The first 10 classes of FaceScrub are attacked, and for each class, we reconstruct 5 images. The attack results are shown in Table 8 and the models' utility are shown in Table 9. Although models are more vulnerable on high resolution, our defense still achieves the best MIA robustness, with a good utility.

Table 8: The MIA robustness of all defenses under Mirror attack on high resolution.

	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
No Defense	70%	94%	195	0.84
MID	62%	90%	183	0.76
BiDO	66%	86%	194	0.90
LS	48%	82%	202	0.87
TL	58%	92%	191	0.80
SSD (ours)	**42%**	**66%**	**211**	**1.13**

Table 9: The target models' utility and defenses' settings on high resolution.

	$\uparrow$ Acc	$\downarrow$ Avg L_1	$\downarrow$ Max L_1	Settings
No Defense	98.5%	0	0	–
MID	96.7%	0.30	**1.97**	$\beta = 0.005$
BiDO	96.3%	0.09	1.99	$\lambda_x = 0.15, \lambda_y = 1.5$
LS	96.5%	0.11	1.99	$\alpha = -0.01$
TL	96.7%	0.19	1.99	First 70% layers
SSD (ours)	**96.9%**	**0.07**	1.98	$T = 1, \varepsilon = 20$

G DISCUSSION ON ADAPTIVE ATTACKS

In this section we discuss adaptive attacks, where attackers are aware of our defense and take targeted actions.

Firstly, we believe that launching adaptive attacks in black-box scenarios is unrealistic, because attackers don't know the target model, and naturally don't know its defense strategy. If they were to guess the defense strategy based on the model's behavior, they would need to consume a large number of queries.

Step back and consider, if attackers know our defense, their best strategy is:

1. Query the same x repeatedly and count the frequency of different outputs.
2. Estimate our sampling probability $\mathcal{P}(y|x)$ by the frequency they count.
3. Infer our true prediction $\mathcal{P}(\hat{y}|x)$ by the $\mathcal{P}(y|x)$ they estimate and the temperature T (assuming they know).

If an online server detects such pattern of queries, it can block them. Step back and consider again, we propose a memory-free and low-cost improvement to block such adaptive attacks:

Design a hash function $h : \mathbb{X} \to \mathbb{N}$, where $\mathbb{X}$ is the input space and $\mathbb{N}$ is the set of integers. When users/attackers query x, we take $h(x)$ as the random seed for sampling, ensuring same-input-same-output. However, attackers can add subtle perturbations to x, therefore our h needs to be robust. For example, it can be

$$h(x) := \sum_{i=1}^{m} \lfloor k \cdot z_i(x) \rfloor, \tag{16}$$

where $\boldsymbol{z}(x) \in \mathbb{R}^m$ is the penultimate layer feature in target model, and k is the sensitivity coefficient. Note that $\boldsymbol{z}(x)$ are commonly used to evaluate the similarity between two images, i.e., the closer the two $\boldsymbol{z}(x)$ are, the more similar the two x look. The larger k is, the more numerically sensitive h is, and the more random our defense is.

How to evaluate and improve h is a new and interesting topic, worth studying deeply in the future.

H COMPARISON ON PURIFIER DEFENSE

Purifier is a black-box defense against membership inference attacks and may have the effect of resisting MIAs (Yang et al., 2023). We reproduce Purifier, setting $\lambda = 0.01$ and $k = 1$. We use the validation set as the reference set and swap the first and second labels if the L2 distance < 0.0001.

The comparisons on Purifier are aligned with the main experiments in our paper. The target model is IR-152 and the GANs are trained on FFHQ. The first 100 classes in FaceScrub are attacked and each reconstructs 5 images. The results are shown in Table 11, 12, 13. The target model's utility is listed in Tabel 10.

Table 10: The target model's utility.

	$\uparrow Acc$	$\downarrow AvgL_1$	$\downarrow MaxL_1$
None	98.4%	0	0
Purifier	96.0%	0.14	2.00
SSD	**96.5%**	**0.06**	**0.95**

Table 11: The MIA robustness against C2F.

	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
None	28.0%	47.4%	1949	1.01
Purifier	3.8%	7.6%	**2655**	1.47
SSD	**1.8%**	**3.6%**	2518	**1.49**

Table 12: The MIA robustness against BREP.

	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
None	37.4%	54.2%	2150	1.00
Purifier	38.6%	56.8%	2140	1.00
SSD	**2.6%**	**3.8%**	**2650**	**1.55**

Table 13: The MIA robustness against Mirror.

	$\downarrow Acc@1$	$\downarrow Acc@5$	$\uparrow \delta_{eval}$	$\uparrow \delta_{face}$
None	69.2%	89.0%	1752	0.77
Purifier	62.4%	78.4%	1915	0.99
SSD	**22.8%**	**34.6%**	**2272**	**1.21**