# GIFD: A Generative Gradient Inversion Method with Feature Domain Optimization

Hao Fang[1,2] Bin Chen[1,3,4*] Xuan Wang[1,3,4] Zhi Wang[2,3] Shu-Tao Xia[2]
[1]Harbin Institute of Technology, Shenzhen
[2]Tsinghua Shenzhen International Graduate School, Tsinghua University [3]Peng Cheng Laboratory
[4]Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies
190110304@stu.hit.edu.cn, chenbin2021@hit.edu.cn, wangxuan@cs.hit.edu.cn
{wangzhi, xiast}@sz.tsinghua.edu.cn

## Abstract

*Federated Learning (FL) has recently emerged as a promising distributed machine learning framework to preserve clients' privacy, by allowing multiple clients to upload the gradients calculated from their local data to a central server. Recent studies find that the exchanged gradients also take the risk of privacy leakage, e.g., an attacker can invert the shared gradients and recover sensitive data against an FL system by leveraging pre-trained generative adversarial networks (GAN) as prior knowledge. However, performing gradient inversion attacks in the latent space of the GAN model limits their expression ability and generalizability. To tackle these challenges, we propose **G**radient **I**nversion over **F**eature **D**omains (GIFD), which disassembles the GAN model and searches the feature domains of the intermediate layers. Instead of optimizing only over the initial latent code, we progressively change the optimized layer, from the initial latent space to intermediate layers closer to the output images. In addition, we design a regularizer to avoid unreal image generation by adding a small $l_1$ ball constraint to the searching range. We also extend GIFD to the out-of-distribution (OOD) setting, which weakens the assumption that the training sets of GANs and FL tasks obey the same data distribution. Extensive experiments demonstrate that our method can achieve pixel-level reconstruction and is superior to the existing methods. Notably, GIFD also shows great generalizability under different defense strategy settings and batch sizes.*

## 1. Introduction

Federated learning [21, 35] is an increasingly popular distributed machine learning framework, which has been applied in many privacy-sensitive scenarios [19, 33], such as financial services, medical analysis, and recommendation systems.

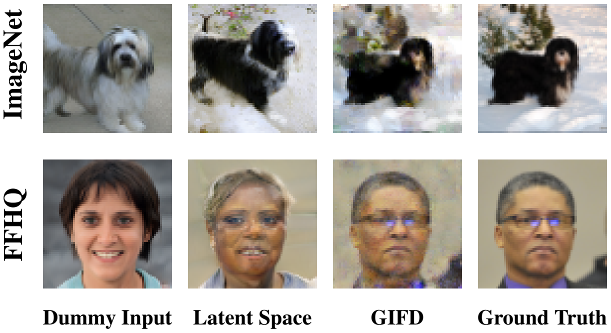


Figure 1: The reconstructed results of our proposed GIFD on ImageNet[6] and FFHQ[17]. The first column contains the randomly initialized images generated by generators. The next two columns show the reconstruction samples of the latent space search and our proposed GIFD,

It allows multiple clients to participate in collaborative learning under the coordination of the central server. The central server aggregates the uploaded gradients calculated from the local data by the end users, rather than the private data. This mechanism resolves the data silos problem and brings privacy benefits to distributed learning. However, a series of recent studies have shown that even the gradients uploaded in FL take the risk of privacy leakage. Zhu *et al.* [40] first formulate it as an optimization problem and design an optimization-based algorithm that reconstructs private data by best matching the dummy gradients with the real gradients. Zhao *et al.* [38] further improve the attack with an extra label restoration step. Geiping *et al.* [9] first achieve ImageNet-level recovery through a well-designed loss function that adds a new regularization and

[*]Corresponding Author

uses a different distance metric. In order to improve the performance on larger batch sizes, Yin *et al.* [34] propose a batch-level label extraction method and assume that certain side-information is available to regularize feature distributions through batch normalization (BN) prior.

It is widely investigated and acknowledged that a pre-trained GAN learned from a public dataset generally captures a wealth of prior knowledge. Recent studies [34, 16, 20] propose to leverage the manifold of GAN as prior information, which provides a good approximation of the natural image space and enhances the attacks significantly. The aforementioned works achieve impressive results in their own scenarios, but most of them rely on strong assumptions, e.g., known labels, BN statistics, and private data distribution, which are actually impractical in the real FL scenario. Therefore, it is hard for most existing methods to recover high-quality private data in a more realistic setting.

In this paper, we advocate a simple and effective solution, Gradient Inversion over Feature Domain (GIFD), to address the challenges of expression ability and generalizability of pre-trained GANs. Recently, it has been shown that rich semantic information is encoded in the intermediate features and the latent space of GANs [2, 30, 26, 5]. Among them, the GAN-based intermediate layer optimization in solving compressed sensing problems achieves great performance [5]. Inspired by these works, We reformulate the GAN inversion as a novel intermediate layer optimization problem by minimizing the gradient matching loss by searching the intermediate features of the generative model. Specifically, our first step is to optimize the latent space and then we optimize the intermediate layers of the generative model successively. During the feature domain optimization stage, we only use part of the generator and the solution space becomes larger, which can easily lead to unreal image generation. To solve this problem, we iteratively project the optimizing features to a small $l_1$ ball centered at the initial vector induced by the previous layer. Finally, we select output images from the layer with the corresponding least gradient matching loss as the final results. The visual comparison in Figure 1 clearly demonstrates the necessity of optimizing the intermediate feature domains.

Another issue unsolved in GAN-based gradient attacks is the flexibility of private data generation under more rigorous and realistic settings. To relax these assumptions, we first investigate an out-of-distribution (OOD) gradient attack scenario, where the private data distribution is significantly different from that of the GAN's training set. The significant result improvement demonstrates the proposed method has excellent generalizability and achieves great performance on OOD datasets. Furthermore, we discuss several common defense strategies in *protection form gradient sharing*[36], including gradient sparsification [28, 1], gradient clipping [10], differential privacy [10], and Soteria (*i.e.*, perturbing the data representations) [29]. These frequently used privacy defense approaches have been confirmed to achieve high resilience against existing attacks by degrading the privacy information carried by the share gradients. Extensive experiments and ablation studies have demonstrated the effectiveness of the GIFD attack.

Our main contributions are summarized as follows:

- We propose GIFD for exploiting pre-trained generative models as data prior to invert gradients by searching the latent space and the intermediate features of the generator successively with $l_1$ ball constraint.
- We show that this optimization method can be used to generate private OOD data with different styles, demonstrating the impressive generalization ability of the proposed GIFD under a more practical situation.
- We systematically evaluate our proposed method compared with the state-of-the-art baselines with the gradient transformation technique under four considered defense strategies.

## 2. Related Work

### 2.1. Gradient-based Attack in FL

In federated learning, the early studies investigate *member inference* [27, 22], where a malicious attacker can determine whether a certain data sample has participated in model training. A similar attack, called *property inference* [8], can reveal the attributes of the samples in the training set. Another powerful attack is *model inversion* [14], which works by training a GAN from local images and the shared gradients to generate samples with the same distribution as the private data. Wang *et al.* [31] then improve the model attack and reconstruct client-level data representatives.

**Gradient Inversion Attacks.** This is a more threatening type of attack where an adversary can fully reconstruct the client's private data samples. The existing attack methods can be characterized into two paradigms [36]: *recursion* and *iteration*-based methods.

Recursion-based attacks. Phong *et al.* [25] first utilized gradients to successfully recover the input data from a shallow perceptron. Fan *et al.* [7] considered networks with convolution layers and solved the problem by converting the convolution layer into a full connection layer. Zhu *et al.* [39] combined forward and backward propagation to transform the problem into solving a system of linear equations. Chen *et al.* [4] then combined optimization problems under different situations and proposed a systematic framework. The recursion-based methods have the following limitations: (1) low-resolution images only; (2) the global model in FL cannot contain pooling layers or shortcut connections; (3) these methods cannot handle mini-batch training; and (4) they heavily depend on gradients, *i.e.*, if gradients are perturbed, most of these methods barely work.

Iteration-based attacks. Zhu *et al.* [40] first formulated the attack as an iterative optimization problem. Attackers restore data samples by minimizing the distance between the shared gradients and the dummy gradients generated by a pair of dummy samples. Zhao *et al.* [38] proposed to extract the label of a single sample from the gradients and further improved the attack. Geiping *et al.* [9] reconstructed higher resolution images from ResNet [13] by changing the distance metric and adding a regularization term. Yin *et al.* [34] primarily focused on larger batch sizes recovery. With strong BN statistics and deep pre-trained ResNet-50 as the global model (larger model generates more gradient information), they successfully revealed some information from partial images at larger batch sizes. Jeon *et al.* [16] fine-tuned the GAN parameter to better utilize image prior and improved the quality of restored images. Hatamizadeh *et al.* [12] extended attacks on Vision Transformers. Considering defense strategies in FL, Li *et al.* [20] proposed a new technique called gradient transformation to deal with the degraded gradients and still revealed private information.

Currently, several strong assumptions are made to help better reconstruct, which are not identical to the realistic FL setting. By nullifying some of these assumptions [15], the reconstruction performance drops significantly.

### 2.2. GAN as prior knowledge

GAN [11] is a deep generative model, which can learn the probability distribution of the images in the training set through adversarial training. A well-trained GAN can generate realistic and high-diversity images. Recent studies show that GAN can be leveraged to solve inverse problems [32], *e.g.* compressed sensing. Yin *et al.* [34] introduced a method that utilizes a pre-trained generative model as an image prior. Jeon *et al.* [16] proposed to search the latent space and parameter space of the generative model in turn, which fully exploits GAN's generation ability to reconstruct images of outstanding quality. A weakness is that it requires a specific generator to be trained for each reconstructed image, which consumes large amounts of GPU memory and inference time. Li *et al.* [20] also adopted the generative model, but only optimized the latent code, which achieves semantic-level reconstruction. Among the GAN-based methods, only Jeon *et al.* [16] really considered the situation when the training data of the generative model and the global model obey different probability distributions.

Inspired by the successful application of Intermediate Layer Optimization (ILO) [5] in compressed sensing, we decide to search the latent space and feature domains of the generative model to achieve pixel-level reconstruction. Meanwhile, we find that our method is superior to the previous methods for OOD data.

## 3. Method

In this section, we first introduce the basic paradigm of gradient inversion attacks. Then, we explain how former methods leverage GAN to achieve better results. Finally, we elaborate on our proposed GIFD, which successively searches the latent space and intermediate feature spaces of the generative model.

### 3.1. Problem Formulation

Given a neural network $f_\theta$ with weights $\theta$ for image classification tasks, and batch-averaged gradients $g$ calculated from a private batch with images $\mathbf{x}^*$ and labels $\mathbf{y}^*$, the attacker attempts to invert the gradients to private data with randomly initialized input tensor $\hat{\mathbf{x}} \in \mathbb{R}^{B\times H\times W\times C}$ and labels $\hat{\mathbf{y}} \in \{0,1\}^{B\times L}$ ($B, H, W, C, L$ being batch size, height, width, number of channels and class number):

$$\hat{\mathbf{x}}^*, \hat{\mathbf{y}}^* = \arg\min_{\hat{\mathbf{x}},\hat{\mathbf{y}}} \mathcal{D}\left(\frac{1}{B}\sum_{i=1}^{B}\nabla\ell(f_\theta(x_i), y_i), g\right), \quad (1)$$

where $\hat{\mathbf{x}} = (x_1, \dots, x_B)$, $\hat{\mathbf{y}} = (y_1, \dots, y_B)$. $\mathcal{D}(\cdot,\cdot)$ is the measurement of distance, *e.g.*, $l_2$-distance [34, 20], negative cosine similarity [9, 16], and $\ell(\cdot,\cdot)$ is the loss function for classification. In the workflow of the algorithm, the attacker generates a pair of random noise $\hat{\mathbf{x}}$ and labels $\hat{\mathbf{y}}$ as parameters, optimized towards the ground truth $\mathbf{x}^*$ and $\mathbf{y}^*$ through minimizing the matching loss between dummy gradients and transmitted gradients.

Since private labels can be inferred directly from the gradients [38, 34], the objective function with regularization term can be simplified to the following form:

$$\hat{\mathbf{x}}^* = \arg\min_{\hat{\mathbf{x}}} \mathcal{D}\left(F(\hat{\mathbf{x}}), g\right) + R_{prior}(\hat{\mathbf{x}}), \quad (2)$$

where $F(\hat{\mathbf{x}}) = \frac{1}{B}\sum_{i=1}^{B}\nabla\ell(f_\theta(x_i), y_i)$, $R_{prior}(\hat{\mathbf{x}})$ is prior knowledge regularization (*e.g.*, BN statistics [34]).

Given a pre-trained generative model $G_w(\cdot)$ learning from the public dataset, an intuitive method is to transform the problem into the following form:

$$\mathbf{z}^* = \arg\min_{\mathbf{z}} \mathcal{D}\left(F(G_w(\mathbf{z})), g\right) + R_{prior}(\mathbf{z}; G_w), \quad (3)$$

where $\mathbf{z} \in \mathbb{R}^{B\times k}$ is the latent code of the generative model. By narrowing the search range from $\mathbb{R}^{B\times m}$ ($m = H\times W\times C$) to $\mathbb{R}^{B\times k}$ ($k<<m$), one can reduce the uncertainty in the optimizing process. Based on this, various GAN-based gradient inversion methods [20, 16] are proposed to ensure the quality and fidelity of the generated images.

### 3.2. Gradient Inversion over Feature Domains

First, we formally formulate our optimization objective:

$$\hat{\mathbf{x}}^* = \arg\min_{\hat{\mathbf{x}}} \mathcal{D}\left(\mathcal{T}(F(\hat{\mathbf{x}})), g\right) + \mathcal{R}_{fidty}(\hat{\mathbf{x}}), \quad (4)$$

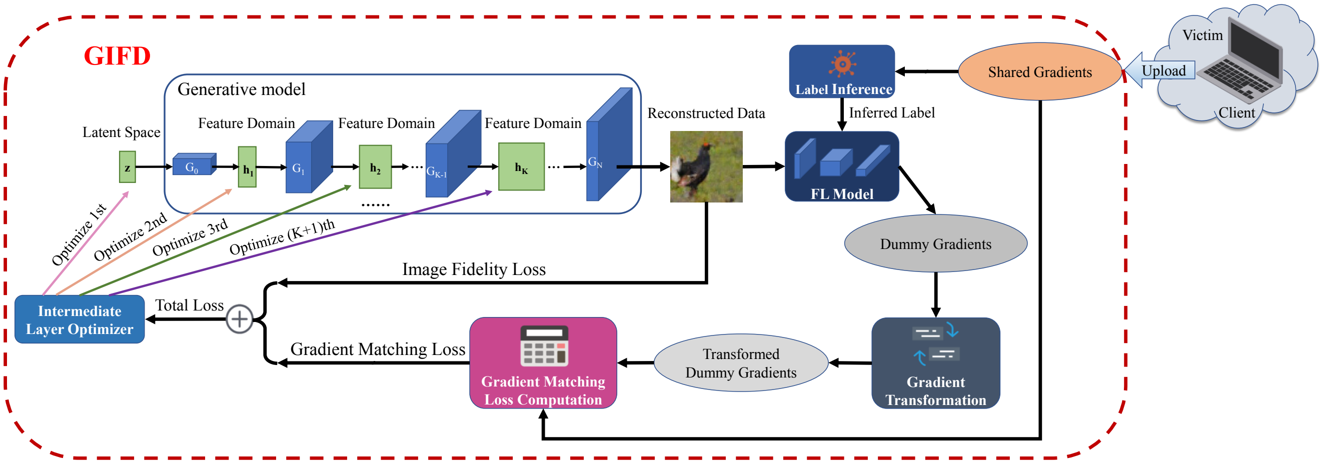


Figure 2: Overview of our proposed GIFD attack. The intermediate layer optimizer minimizes the loss computed from the dummy gradients and the shared gradients from the victim under the image fidelity regularization, to update the latent vector and the intermediate features successively. Finally, the generative model outputs reconstruction data from the layer with the corresponding least gradient matching loss.

where $\hat{\mathbf{x}}$ is generated by $G_w$ or part of $G_w$, $F(\cdot)$ is the batch-averaged gradient operator, $\mathcal{T}(\cdot)$ is the gradient transformation technique we will discuss later. The first term $\mathcal{D}\left(\mathcal{T}(F(\hat{\mathbf{x}})), g\right)$ denotes the gradient matching loss, and the second term $\mathcal{R}_{fidty}(\hat{\mathbf{x}})$ is the image fidelity regularization. To simplify the expression, we solve for the objective function in the following form:

$$\hat{\mathbf{x}}^* = \arg\min_{\hat{\mathbf{x}}} \mathcal{L}_{grad}(\hat{\mathbf{x}}), \qquad (5)$$

where we denote the loss function in (4) by $\mathcal{L}_{grad}(\hat{\mathbf{x}})$. An overview of our method is shown in Figure 2, we next introduce each component in detail.

**Intermediate Layer Optimizer.** This is the core of our algorithm. As the pseudocode described in Algorithm 1, instead of directly optimizing over $\hat{\mathbf{x}}$, we focus on searching the latent space and the intermediate space of the generator in turn, to make the most of the GAN prior.

The first step is to optimize over the randomly initialed latent vector $\mathbf{z}$ using gradient descent with an effective Spherical Optimizer [23]. Once we obtain the optimal $\mathbf{z}^*$, we dissemble the generator $G_w$ into $G_0 \circ G_1 \circ \cdots \circ G_{N-1} \circ G_N$ for intermediate feature optimization. Then, we map optimal latent vector $\mathbf{z}^*$ into intermediate latent representations $\mathbf{h_1^0}$ using $G_0$, *i.e.*, $\mathbf{h_1^0} := G_0(\mathbf{z}^*)$. Next, our algorithm enters the for loop in line 7 of Algorithm 1 and starts to search the intermediate features.

At the pass of loop $i$, we perform the following operations. First, we generate images from intermediate feature $\mathbf{h_i}$ only with the rest part of $G_w$ (*i.e.*, $G_i \circ \cdots \circ G_N$). Then, we use the generated images to compute dummy gradients and optimize over $\mathbf{h_i}$ via minimizing cost function in (4). Considering the intermediate feature searching might lead to unreal images generation, we constrain the searching range to lie within an $l_1$ ball of radius $r[i]$ centered at $\mathbf{h_i^0}$, *i.e.* the term $ball^{r[i]}_{\mathbf{h_i^0}}$ in the line 9 of Algorithm 1. After obtaining the optimal results $\mathbf{h_i^*}$ of the present layer, we generate the initial intermediate representations for the next layer with $G_i$, *i.e.* $\mathbf{h_{i+1}^0} := G_i(\mathbf{h_i^*})$.

As shown in line 4, 11, 12, 13, 18 of Algorithm 1, we hope to utilize the gradient matching loss as valid information to guide us to select the output images. More specifically, we choose the output images from the layer with the corresponding least gradient matching loss among all the searched intermediate layers as the final output. Although less loss doesn't always mean better image quality, our strategy still outperforms specifying a fixed layer's output.

With all the efforts above, we encourage the optimizer to explore the intermediate space with rich information, to generate more diverse and high-fidelity images, while limiting the solution space within a $l_1$ ball around the manifold induced by the previous layer in order to avoid overfitting and guarantee the realism of the generated images. Furthermore, our approach is easy to implement as it is not tied to any specific GAN architecture and only requires a pretrained generative model.

**Labels Extraction.** Specifically, consider a network parameterized by W for classification task over $n$-classes using cross-entropy loss function, when the training data is a single image, the ground truth label $c$ can be accurately inferred [38] through:

$$c = i, \quad \text{s.t. } \nabla \mathrm{W}_{\mathbf{FC}}^{\mathbf{i}\,\top} \cdot \nabla \mathrm{W}_{\mathbf{FC}}^{\mathbf{j}} \le 0, \ \forall j \neq i, \qquad (6)$$

where we denote the gradient vector w.r.t. the weights (de-

**Algorithm 1** Pseudocode of our proposed GIFD

```
Input: $G_w$: a pre-trained generative model; $f_\theta$: the global model in FL; $g$: shared gradients; $K$: the index of the last intermediate layer to optimize; $r[1 \dots K]$: radius of $l_1$ ball in each intermediate layer; $B$: batch size;
Output: Reconstructed images via GIFD attack;
Initial latent code $\mathbf{z} := (z_1, \dots, z_B)$ with random noise
// Latent space search
$\mathbf{z}^* \leftarrow \arg\min_{\mathbf{z}} \mathcal{L}_{grad}(G_w(\mathbf{z}))$
Set $\hat{\mathbf{x}}^* := G_w(\mathbf{z}^*)$, $loss_{min} = \mathcal{D}(\mathcal{T}(F(G_w(\mathbf{z}^*))), g)$
Dissemble $G_w$ into $G_0 \circ G_1 \circ \dots \circ G_{N-1} \circ G_N$
Set $\mathbf{h_1^0} := G_0(\mathbf{z}^*)$
for $i \leftarrow 1$ to $K$ do
    //Intermediate layers search with $l_1$-ball constraint
    $\mathbf{h_i^*} \leftarrow argmin_{\mathbf{h_i} \in ball_{\mathbf{h_i^0}}^{r[i]}} \mathcal{L}_{grad}(G_i \circ \dots \circ G_N(\mathbf{h_i}))$
    $loss_i = \mathcal{D}(\mathcal{T}(F(G_i \circ \dots \circ G_N(\mathbf{h_i^*}))), g)$
    if $loss_i < loss_{min}$ then
        $\hat{\mathbf{x}}^* := G_i \circ \dots \circ G_N(\mathbf{h_i^*})$
        $loss_{min} = loss_i$
    end if
    // Generate features of the next intermediate layer as the initial vector to optimize
    $\mathbf{h_{i+1}^0} := G_i(\mathbf{h_i^*})$
end for
Return results: $\hat{\mathbf{x}}^*$
```

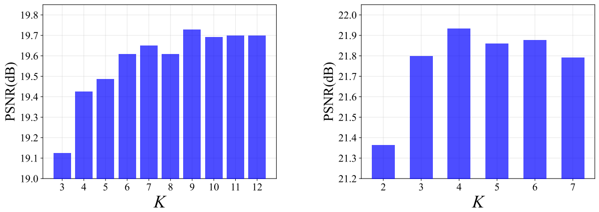
noted as $\mathrm{W}_{\mathbf{FC}}^{\mathbf{i}}$) connected to the $i_{th}$ logit in the classification layer (*i.e.*, the output layer) by $\nabla \mathrm{W}_{\mathbf{FC}}^{\mathbf{i}}$. Hence, we can identify the ground-truth label via the index of the negative gradients. [34] further extends to support batch-level label extraction with high accuracy, while assuming non-repeating labels in the batch. The inferred labels are used to compute dummy gradients and as the class conditions for conditional GANs, which greatly enhances our attack.

**Image Fidelity Regularization.** Intuitively, it is challenging to restore data only from the shared gradients, as gradients are only a non-linear mapping form of the original data. It is therefore worth using some strong priors as an approximation of natural images:

$$\mathcal{R}_{fidty}(\hat{\mathbf{x}}) = \alpha_{\ell_2} \mathcal{R}_{\ell_2}(\hat{\mathbf{x}}) + \alpha_{TV} \mathcal{R}_{TV}(\hat{\mathbf{x}}), \tag{7}$$

where the first term is the $l_2$ norm of the images [34] with scaling factor $\alpha_{\ell_2}$, which encourages the algorithm to solve for a solution that is preferably sparse. Since neighboring pixels of natural images are likely to have close values, we add the second term [9] $\mathcal{R}_{TV}(\hat{\mathbf{x}})$ to penalize total variation of $\hat{\mathbf{x}}$ with scaling factor $\alpha_{TV}$.

**Gradient Transformation.** In order to mitigate the effects of defense strategies, we adopt the adaptive attack [20] by estimating transformation from received gradients and incorporating it into the optimization process, *i.e.*, $\mathcal{T}(\cdot)$ in (4). Specifically, we can infer three defense strategies: (1) *Gradient clipping*; (2) *Gradient sparsification*; and (3) *Soteria*.



(a) BigGAN

(b) StyleGAN2

Figure 3: Comparison of PSNR mean on BigGAN and StyleGAN2 under different values of hyper-parameter $K$ (*i.e.*, the last intermediate layer to optimize). Notably, the figures exclude the results where the corresponding values are below the starting point of the y-axis.

## 4. Experiments

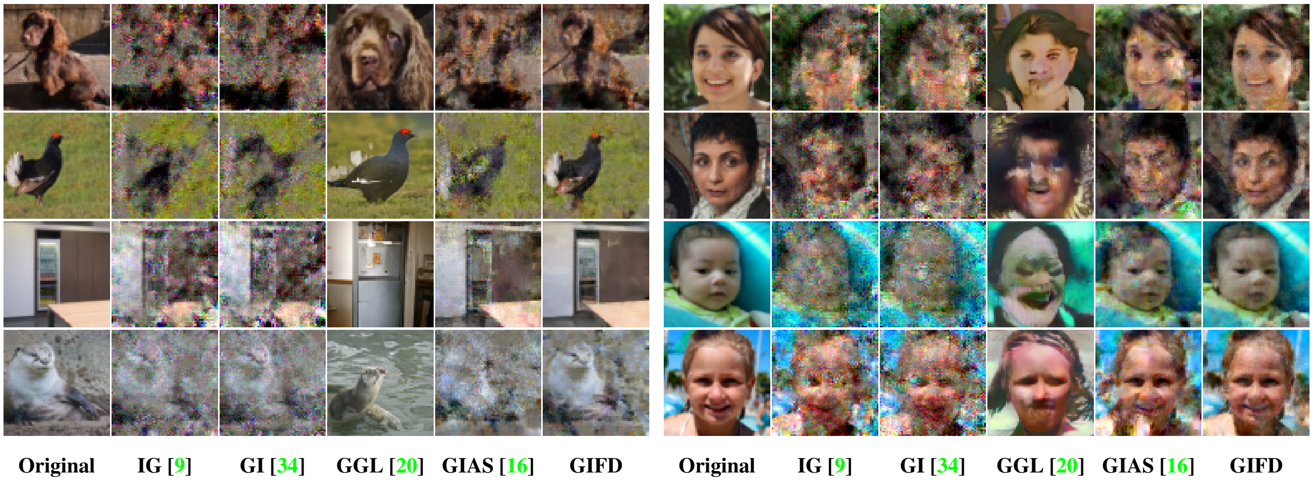
To validate the effectiveness of GIFD in improving attack performance, we conduct experiments on two widely used GANs in a range of scenarios. We evaluate our method for the classification task on the validation set of ImageNet ILSVRC 2012 dataset[6]) and 10-class (using age as label) FFHQ [17] at 64 × 64 pixels. For the generative model, we use a pre-trained BigGAN [3] for ImageNet and a pre-trained StyleGAN2 [17] for FFHQ. We use a randomly initialized ResNet-18 as the FL model, and choose negative cosine similarity as distance metric $\mathcal{D}(\cdot)$. We use the default $B = 1$ at one local step. Then we conduct experiments with larger $B$ and compare the performance of different methods. Our code is available at https://github.com/ffhibnese/GIFD.

**Implementation details**. According to its specific structure, we split BigGAN into $G_0$ to $G_{12}$ with 12 intermediate feature domains, and StyleGAN2 into $G_0$ to $G_7$ with 7 intermediate feature domains. We ensure that the intermediate features lie in the $l_1$ ball through Project Gradient Descent (PGD) [24]. Motivated by the fact that a stepwise optimization over the noises in StyleGAN2 yields better reconstructions [5] for compressed sensing, we gradually allow to optimize more noises as we move to deeper intermediate layers and make them lie inside the $l_1$ ball as well. For more details about experiments, please refer to the Appendix.

**Evaluaion Metrics**. We compute the following quantitative metrics to measure the discrepancy between reconstructed images and ground truth: (1) PSNR (Peak Signal-to-Noise Ratio), (2) LPIPS [37] (Learned Perceptual Image Patch Similarity), (3) SSIM (Similarity Structural Index Measure), and (4) MSE (Mean Square Error) between reconstruction and private images.

Table 1: Comparison of GIFD with state-of-the-art methods on every 1000th image of the ImageNet and FFHQ validation set. We calculate the average value of metrics on reconstructed images.

| Metric | ImageNet | | | | | FFHQ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | IG [9] | GI [34] | GGL [20] | GIAS [16] | **GIFD** | IG [9] | GI [34] | GGL [20] | GIAS [16] | **GIFD** |
| PSNR↑ | 17.0756 | 16.5109 | 13.3885 | 17.4923 | **20.0534** | 15.3523 | 14.9485 | 15.1335 | 20.1799 | **21.3368** |
| LPIPS↓ | 0.3078 | 0.3297 | 0.3678 | 0.2536 | **0.1559** | 0.4172 | 0.4503 | 0.2009 | 0.1266 | **0.1023** |
| SSIM↑ | 0.2908 | 0.2673 | 0.1251 | 0.3381 | **0.4713** | 0.2272 | 0.2044 | 0.2453 | 0.5379 | **0.5768** |
| MSE↓ | 0.0223 | 0.0258 | 0.0553 | 0.0236 | **0.0141** | 0.0311 | 0.0343 | 0.0339 | 0.0121 | **0.0098** |



(a) ImageNet (BigGAN)

(b) FFHQ (StyleGAN2)

Figure 4: Qualitative results of different methods on ImageNet and FFHQ.

### 4.1. Decide Which Layer to End

In order to further improve the quality of output images, we need to carefully handle the parameter $K$ in Algorithm 1. Actually, we find that there is a trade-off between underfitting and over-fitting about the choice of $K$. When $K$ is small, we only search the first few intermediate features of the generative model and do not fully utilize the rich information encoded in the intermediate space. As a result, the quality of the generated images does not meet our expectations. On the contrary, when $K$ is large, we excessively search the deeper layers and generate images that have less cost, but a larger discrepancy with the original images. Therefore, we randomly select images (disjoint from our main experimental data) from the validation set of ImageNet and FFHQ to study the impact of $K$ and try to select the best final layer. As shown in Figure 3, when $K = 9$ and $K = 4$ are used for BigGAN and StyleGAN2 respectively, we obtain results with the largest PSNR. Hence, we use this configuration for conducting all the experiments.

### 4.2. Comparison with the State-of-the-art Attacks

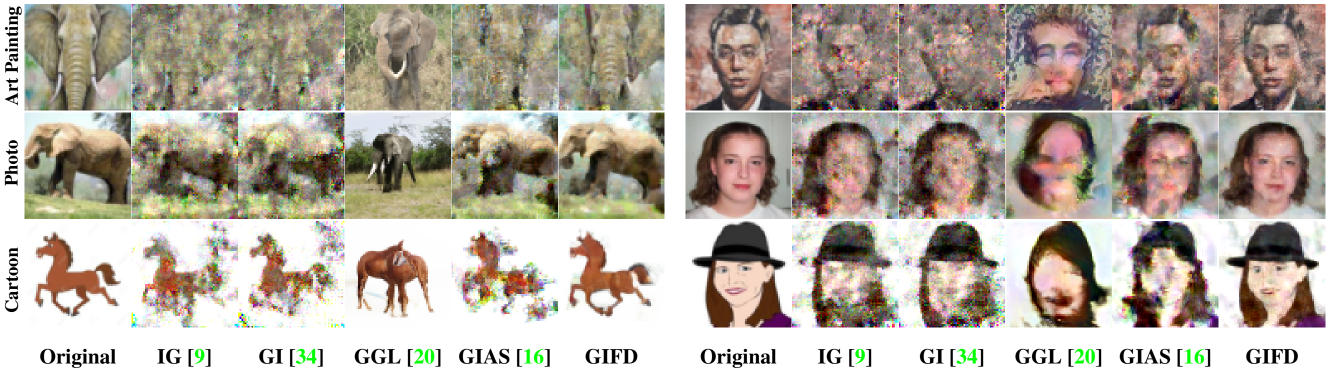
Next, we compare our proposed GIFD with existing methods and provide qualitative and quantitative results. We consider the following four state-of-the-art baselines: (1) *Inverting Gradients (IG)* by Geiping *et al.* [9]; (2) *Grad-Inversion (GI)* by Yin *et al.* [34]; (3) *Gradient Inversion in Alternative Spaces (GIAS)* by Jeon *et al.* [16]; and (4) *Generative Gradient Leakage (GGL)* by Li *et al.* [20].

In real application scenarios, a vast majority of FL systems do not transmit the BN statistics computed from private data [15]. Based on this fact, all the experiments do not use the strong BN prior proposed by [34]. Since the randomly initialized values of vectors will greatly affect the reconstruction results, we conduct 4 trials for every attack and select the result with the least gradient matching loss. The ablation study is conducted in the Appendix.

**Experiment Results.** By observing the results in Table 1, we demonstrate that our method consistently achieves great improvement compared to the competing methods for gradient inversion attacks. Especially in the ImageNet dataset with BigGAN, our method has nearly 2.5dB and 0.1 improvements in average PSNR and LPIPS values respectively. As the visualization comparison shown in Figure 4, under a more practical setting, most existing methods struggle to recover meaningful and high-quality images even at $B = 1$, while our method reveals significant information

Table 2: Comparision of GIFD with state-of-the-art baselines on OOD data of different styles.

| Datset | Method | Art Painting | | | | Photo | | | | Cartoon | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | PSNR↑ | LPIPS↓ | SSIM↑ | MSE↓ | PSNR↑ | LPIPS↓ | SSIM↑ | MSE↓ | PSNR↑ | LPIPS↓ | SSIM↑ | MSE↓ |
| ImageNet* | IG [9] | 18.3476 | 0.2286 | 0.3870 | 0.0172 | 15.6647 | 0.3575 | 0.2409 | 0.0325 | 15.8766 | 0.3183 | 0.3970 | 0.0288 |
| | GI [34] | 17.4681 | 0.2625 | 0.3445 | 0.0203 | 15.2700 | 0.3888 | 0.2201 | 0.0346 | 15.3905 | 0.3112 | 0.3926 | 0.0327 |
| | GGL [20] | 12.8011 | 0.3639 | 0.1356 | 0.0571 | 12.9246 | 0.3159 | 0.1507 | 0.0667 | 11.0315 | 0.3294 | 0.2832 | 0.0895 |
| | GIAS [16] | 17.2804 | 0.2774 | 0.3346 | 0.0227 | 20.4539 | 0.1724 | 0.4913 | 0.0111 | 19.0247 | 0.1862 | 0.5740 | 0.0149 |
| | **GIFD** | **19.3311** | **0.1700** | **0.4503** | **0.0151** | **21.9281** | **0.1137** | **0.5765** | **0.0082** | **22.8055** | **0.1030** | **0.6970** | **0.0067** |
| FFHQ* | IG [9] | 15.9020 | 0.3856 | 0.2736 | 0.0273 | 17.7422 | 0.3043 | 0.3398 | 0.0174 | 14.7029 | 0.3118 | 0.3213 | 0.0358 |
| | GI [34] | 16.2990 | 0.3537 | 0.2917 | 0.0259 | 18.5540 | 0.2388 | 0.3808 | 0.0147 | 15.0097 | 0.3232 | 0.3201 | 0.0331 |
| | GGL [20] | 14.2833 | 0.2514 | 0.1982 | 0.0435 | 15.5001 | 0.2309 | 0.2513 | 0.0302 | 12.3590 | 0.2556 | 0.2322 | 0.0624 |
| | GIAS [16] | 18.4619 | 0.1912 | 0.4424 | 0.0172 | 19.6763 | 0.1615 | 0.4885 | 0.0123 | 15.3798 | 0.2250 | 0.3837 | 0.0338 |
| | **GIFD** | **19.8847** | **0.1534** | **0.4979** | **0.0120** | **21.3981** | **0.1148** | **0.5446** | **0.0098** | **17.4005** | **0.1634** | **0.4614** | **0.0220** |



(a) ImageNet* (BigGAN)

(b) FFHQ* (StyleGAN2)

Figure 5: Visual comparison of different methods on ImageNet* and FFHQ*.

about the private data and achieves pixel-level reconstruction on both two datasets.

The GAN-based methods (i.e. GGL, GIAS, GIFD) generally achieve better results than the GAN-free methods (i.e. GI, IG) on the FFHQ dataset. This indicates that the special data distribution of human-face can be more easily learned by the generative model so that the gain from the GAN prior is larger. We also observe that the GAN-based method GGL, which only optimizes the latent code and does not fully exploit the GAN prior, yields unsatisfactory results and performs even worse than the GAN-free methods [9, 34] on the ImageNet dataset, which again verifies the necessity of searching intermediate layers.

We note that the performance of GIAS with BigGAN is worse than with StyleGAN2. One reason is that the data of ImageNet is more diverse. More importantly, with such a large number of parameters in BigGAN, the solution space for the GAN parameter search process becomes larger and presents a great challenge, *i.e.*, GIAS is more susceptible to the scale of GAN. In contrast, GIFD chooses to optimize the intermediate features and then avoids this problem, hence achieving faithful reconstruction on both two GANs, demonstrating the excellent versatility of our method.

### 4.3. Out of Distribution Data Recovery

We then consider a more practical scenario where the training sets of the GAN model and the FL task obey different data distributions. Considering the difficulty and feasibility of gradient attack tasks, we define the OOD data as having the same label space, but quite different feature distributions. Hereinafter, we denote the OOD data of ImageNet and FFHQ by ImageNet* and FFHQ* respectively.

PAC [18] dataset is a widely used benchmark for domain generalization with four different styles, *i.e.*, Art Painting, Cartoon, Photo, and Sketch. In order to achieve our OOD setting, we manually select data with three different styles (*i.e.*, Art Painting, Cartoon, Photo) from the validation set of PACS. For each style in ImageNet*, we select 15 images of guitar, elephant and horse in total. For FFHQ*, we select 15 images for each style and crop them to obtain the face images. We present visual comparison and quantitative results in Figure 5 and Table 2.

**Experiment Results.** As shown in Table 2, the experiment results demonstrate our significant improvement over the baseline methods. For instance, our method has nearly 3.8dB improvement in average PSNR upon GIAS for Cartoon in ImageNet*. Compared with other styles, the GAN-based methods perform best on Photo, whose domain characteristics are similar to the training sets of GANs. We also note that for Art in ImageNet*, the GAN-based methods except GIFD perform even worse than the GAN-free ones, which implies that here the gain from GAN is minor and even brings negative effects to them.

Generally, the other GAN-based methods preserve more pre-trained knowledge from ImageNet or FFHQ, thus struggling to generate images similar to ground truth with different styles. In contrast, our method augments the generative ability of the GAN models and enlarges the diversity of the output space, hence achieving outstanding performance. Thus, with our proposed GIFD, we are able to safely relax the assumption that the datasets of the generative model and FL have to obey the same feature distribution.

## 4.4. Attacks under Certain Defense Strategies

Next, we consider attacking a more robust and secure FL system with defense strategies. In order to make a fair comparison, we equip all the baselines with the well-designed gradient transformation technique mentioned before to mitigate the impact of defense.

We consider a relatively strict defense setup as the previous work [18]: (1) *Gaussian Noise* with standard deviation 0.1; (2) *Gradient Clipping* with a clip bound of 4; (3) *Gradient Sparsification* in a sparsity of 90; and (4) *Soteria* with a pruning rate of 80%.

Table 3: PSNR mean of different methods under different defense strategies.

| Method | Defense Strategies | | | |
|---|---|---|---|---|
| | Noise [10] | Clipping [10] | Sparsification [1] | Soteria [29] |
| IG [9] | 11.0654 | 16.4418 | 12.0760 | 9.1941 |
| GI [34] | 10.0818 | 12.5387 | 12.1691 | 10.1831 |
| GGL [20] | 12.7640 | 12.7930 | 12.6810 | 12.8433 |
| GIAS [16] | 12.5397 | 17.9384 | 15.1745 | 16.8151 |
| GIFD | **13.2558** | **18.8983** | **16.0240** | **18.3205** |

(a) ImageNet

| Method | Defense Strategies | | | |
|---|---|---|---|---|
| | Noise [10] | Clipping [10] | Sparsification [1] | Soteria [29] |
| IG [9] | 11.2766 | 18.1382 | 12.0077 | 9.8334 |
| GI [34] | 10.4968 | 12.4146 | 12.1849 | 10.0843 |
| GGL [20] | **14.8982** | 15.6669 | 14.9123 | 15.1798 |
| GIAS [16] | 12.1276 | 20.4726 | 16.7005 | 20.4283 |
| GIFD | 13.7118 | **21.2861** | **17.3253** | **21.1545** |

(b) FFHQ

**Experiment Results.** We present experiment results in Table 3 compared to related methods. In general, with the underlying gradient transformation and the fully exploited GAN image prior, GIFD is still able to invert a degraded gradient observation to generate high-quality images or reveal private information, especially in cases of clipping and Soteria. One exception is that GGL takes the lead on FFHQ when applying additive noise operation. This is because the gradient information is seriously corrupted by the added high-variance Gaussian noise and is no more enough for pixel-level reconstruction. However, GGL only searches the latent space and with GAN's powerful generative capability, it can still produce well-formed images with clear facial contour, which can give a fair result in the metrics even though they are quite different from the original ones. This also indicates that adding Gaussian noise is indeed an effective defense method against related attacks when the variance exceeds a certain threshold.

## 4.5. Performance of Larger Batch Sizes

We then increase the batch size and observe the results of each algorithm. Notably, we assume that no duplicate labels in each batch and infer the labels from the received gradients [34]. We present the results on ImageNet in Table 4, see Appendix for results on FFHQ.

Table 4: PSNR mean of different methods for different batch sizes on ImageNet.

| Method | Batch Size | | | | | |
|---|---|---|---|---|---|---|
| | 1 | 2 | 4 | 8 | 16 | 32 |
| IG[9] | 17.4634 | 15.2417 | 14.3744 | 13.6599 | 13.1545 | 12.0795 |
| GI[34] | 17.4373 | 14.7293 | 14.0947 | 13.3001 | 12.7842 | 11.8767 |
| GGL[20] | 12.7511 | 12.8903 | 13.1875 | 12.6001 | 11.8027 | 11.0896 |
| GIAS[16] | 17.1401 | 16.1683 | 15.5894 | 15.2130 | 14.4462 | 13.6080 |
| **GIFD** | **20.6217** | **16.7542** | **16.4272** | **15.4889** | **14.6500** | **13.8106** |

**Experiment Results.** We find that the proposed GIFD achieves a steady improvement over previous methods at any batch size. The numerical results also show that the performance of all methods generally degrades as the batch size increases, implying that the reconstruction at large batch sizes is still a significant challenge.

# 5. Conclusion

We propose GIFD, a powerful gradient inversion attack that can generalize well in unseen OOD data scenarios. We leverage the GAN prior via optimizing the feature domain of the generative model to generate stable and high-fidelity inversion results. Through extensive experiments, we demonstrate the effectiveness of GIFD with two widely used pre-trained GANs on two large datasets in a variety of more practical and challenging scenarios. To alleviate the proposed threat, one possible defense strategy is utilizing gradient-based adversarial noise as a novel privacy mechanism to provide confused inversion.

We hope this paper can inspire some new ideas for future work and make contributions to the gradient attacks under more realistic scenarios. We also hope that our work can shed light on the design of privacy mechanisms, to enhance the security and robustness of FL systems.

## 6. Acknowledgments

This work is supported in part by the National Natural Science Foundation of China under grant 62171248, Guangdong Basic and Applied Basic Research Foundation under grant 2021A1515110066, Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies (2022B1212010005), and Shenzhen Science and Technology Program under Grant JCYJ20220818101012025, JCYJ20220818101014030, RCBS20221008093124061, GXWD20220811172936001, RCYX20200714114523079.

## References

[1] Alham Fikri Aji and Kenneth Heafield. Sparse communication for distributed gradient descent. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 440–445, Copenhagen, Denmark, sep 2017. Association for Computational Linguistics. 2, 8

[2] David Bau, Jun Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B Tenenbaum, William T Freeman, and Antonio Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. In *International Conference on Learning Representations*. International Conference on Learning Representations, ICLR, 2019. 2

[3] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. 5

[4] Cangxiong Chen and Neill D. F. Campbell. Understanding training-data leakage from gradients in neural networks for imageclassifications. In *NeurIPS 2021 Workshop Privacy in Machine Learning*, 2021. 2

[5] Giannis Daras, Joseph Dean, Ajil Jalal, and Alex Dimakis. Intermediate layer optimization for inverse problems using deep generative models. In *International Conference on Machine Learning*, pages 2421–2432. PMLR, 2021. 2, 3, 5

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1, 5

[7] Lixin Fan, Kam Woh Ng, Ce Ju, Tianyu Zhang, Chang Liu, Chee Seng Chan, and Qiang Yang. Rethinking privacy preserving deep learning: How to evaluate and thwart privacy attacks. In *Federated Learning*, pages 32–50. Springer, 2020. 2

[8] Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pages 619–633, 2018. 2

[9] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in Neural Information Processing Systems*, 33:16937–16947, 2020. 1, 3, 5, 6, 7, 8

[10] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017. 2, 8

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3

[12] Ali Hatamizadeh, Hongxu Yin, Holger R Roth, Wenqi Li, Jan Kautz, Daguang Xu, and Pavlo Molchanov. Gradvit: Gradient inversion of vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10021–10030, 2022. 3

[13] 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*, pages 770–778, 2016. 3

[14] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep models under the gan: information leakage from collaborative deep learning. In *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pages 603–618, 2017. 2

[15] Yangsibo Huang, Samyak Gupta, Zhao Song, Kai Li, and Sanjeev Arora. Evaluating gradient inversion attacks and defenses in federated learning. *Advances in Neural Information Processing Systems*, 34:7232–7241, 2021. 3, 6

[16] Jinwoo Jeon, Kangwook Lee, Sewoong Oh, Jungseul Ok, et al. Gradient inversion with generative image prior. *Advances in Neural Information Processing Systems*, 34:29898–29908, 2021. 2, 3, 6, 7, 8

[17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 1, 5

[18] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017. 7, 8

[19] Li Li, Yuxi Fan, Mike Tse, and Kuo-Yi Lin. A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854, 2020. 1

[20] Zhuohang Li, Jiaxin Zhang, Luyang Liu, and Jian Liu. Auditing privacy defenses in federated learning via generative gradient leakage. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10132–10142, 2022. 2, 3, 5, 6, 7, 8

[21] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 1

[22] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*, pages 691–706. IEEE, 2019. 2

[23] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 2437–2445, 2020. 4

[24] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003. 5

[25] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, Lihua Wang, and Shiho Moriai. Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Transactions on Information Forensics and Security*, 13(5):1333–1345, 2018. 2

[26] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9243–9252, 2020. 2

[27] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE, 2017. 2

[28] Nikko Ström. Scalable distributed dnn training using commodity gpu cloud computing. In *Interspeech 2015*, 2015. 2

[29] Jingwei Sun, Ang Li, Binghui Wang, Huanrui Yang, Hai Li, and Yiran Chen. Soteria: Provable defense against privacy leakage in federated learning from representation perspective. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9311–9319, 2021. 2, 8

[30] Ayush Tewari, Mohamed Elgharib, Florian Bernard, Hans-Peter Seidel, Patrick Pérez, Michael Zollhöfer, and Christian Theobalt. Pie: Portrait image embedding for semantic control. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020. 2

[31] Zhibo Wang, Mengkai Song, Zhifei Zhang, Yang Song, Qian Wang, and Hairong Qi. Beyond inferring class representatives: User-level privacy leakage from federated learning. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 2512–2520. IEEE, 2019. 2

[32] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[33] Liu Yang, Ben Tan, Vincent W Zheng, Kai Chen, and Qiang Yang. Federated recommendation systems. *Federated Learning: Privacy and Incentive*, pages 225–239, 2020. 1

[34] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M Alvarez, Jan Kautz, and Pavlo Molchanov. See through gradients: Image batch recovery via gradinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16337–16346, 2021. 2, 3, 5, 6, 7, 8

[35] Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. A survey on federated learning. *Knowledge-Based Systems*, 216:106775, 2021. 1

[36] Rui Zhang, Song Guo, Junxiao Wang, Xin Xie, and Dacheng Tao. A survey on gradient inversion: Attacks, defenses and future directions. *arXiv preprint arXiv:2206.07284*, 2022. 2

[37] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 5

[38] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. idlg: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610*, 2020. 1, 3, 4

[39] Junyi Zhu and Matthew B. Blaschko. R-{gap}: Recursive gradient attack on privacy. In *International Conference on Learning Representations*, 2021. 2

[40] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019. 1, 3