

# Auditing Privacy Defenses in Federated Learning via Generative Gradient Leakage

Zhuohang Li<sup>1</sup>

Jiaxin Zhang<sup>2</sup>

Luyang Liu<sup>3</sup>

Jian Liu<sup>1</sup>

<sup>1</sup>University of Tennessee, Knoxville

<sup>2</sup>Oak Ridge National Laboratory

<sup>3</sup>Google Research

zli96@vols.utk.edu,

zhangj@ornl.gov,

luyangliu@google.com,

jliu@utk.edu

## Abstract

Federated Learning (FL) framework brings privacy benefits to distributed learning systems by allowing multiple clients to participate in a learning task under the coordination of a central server without exchanging their private data. However, recent studies have revealed that private information can still be leaked through shared gradient information. To further protect user's privacy, several defense mechanisms have been proposed to prevent privacy leakage via gradient information degradation methods, such as using additive noise or gradient compression before sharing it with the server. In this work, we validate that the private training data can still be leaked under certain defense settings with a new type of leakage, i.e., Generative Gradient Leakage (GGL). Unlike existing methods that only rely on gradient information to reconstruct data, our method leverages the latent space of generative adversarial networks (GAN) learned from public image datasets as a prior to compensate for the informational loss during gradient degradation. To address the nonlinearity caused by the gradient operator and the GAN model, we explore various gradient-free optimization methods (e.g., evolution strategies and Bayesian optimization) and empirically show their superiority in reconstructing high-quality images from gradients compared to gradient-based optimizers. We hope the proposed method can serve as a tool for empirically measuring the amount of privacy leakage to facilitate the design of more robust defense mechanisms<sup>1</sup>.

## 1. Introduction

Federated Learning (FL) [26, 29, 34] has recently emerged as a new machine learning paradigm that enables multiple clients to collaboratively train a global learning model under the orchestration of a central server. Instead of directly exchanging their private data, each client learns

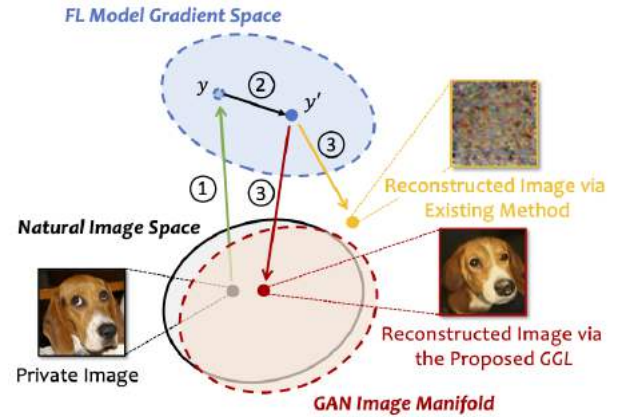


Figure 1. Illustration of data leakage via gradient: ① Client computes gradients on its private data; ② Client applies defense to degrade the computed gradients  $y$ ; ③ Adversary attempts to reconstruct the private image from the shared gradients  $y'$ .

on its local dataset and shares the computed model update or gradient to update the global model. FL places a heavy emphasis on user's data privacy, which has made it particularly suitable for developing machine learning models in privacy-sensitive scenarios such as typing prediction [21], spoken language understanding [16, 20], medical research [4, 8, 41], and financial services [32, 50].

Although FL is designed to structurally encode data minimization principles to protect privacy, recent studies have revealed that, in certain cases, sensitive information can still be leaked through the shared gradients [13, 35, 51, 54, 56]. To further strengthen FL's privacy properties in these cases, several defense strategies have been proposed to *degrade* the gradient information before sharing it with the server, such as differential privacy [14, 48], gradient compression/sparsification [56], and perturbing gradients via data representations [44]. These state-of-the-art privacy defenses have been shown to be effective against existing attacks through modifying the gradient information to degrade its fidelity prior to sharing.

<sup>1</sup>Code is available at: <https://github.com/zhuohangli/GGL>

A natural question is: *Can the aforementioned defenses provide sufficient privacy guarantees to prevent the leakage of sensitive information from the client's private data?* To investigate this, we model the gradient leakage process as an inverse problem, where the goal is to reconstruct the private training data from the client's shared low-fidelity and noisy gradients. Existing methods seek to solve this inverse problem by iteratively solving for the optimal set of data samples that best match the client's shared gradients via an optimization process (e.g., gradient descent [13, 51] or L-BFGS [54, 56]). However, such a problem is ill-posed as there are infinite sets of feasible solutions in the image space and the outcome of the reconstruction may not be a decent natural image. To solve this, existing attacks [13, 51] utilize handcrafted image priors such as total variation [33] to regularize the reconstruction process. Although such prior constraint is relatively effective when there is no defense, we find that it is still not sufficiently tight (i.e., many non-image signals can satisfy this constraint) for reconstructing from low-fidelity and noisy gradient observations, causing existing attacks to falsely return unrealistic images when a defense mechanism is applied (e.g., differential privacy), as illustrated in Figure 1.

In this work, we demonstrate on two image datasets that recovering high-fidelity images from shared gradients is still feasible even under certain defense settings by introducing a new type of leakage, namely Generative Gradient Leakage (GGL). As shown in Figure 1, our method leverages the manifold of the generative adversarial network (GAN) [6, 15, 27] learned from a large public image dataset as prior information, which provides a good proximation of the natural image space. By minimizing the gradient matching loss in the GAN image manifold, our method can find images that are highly similar to the client's private training data with high quality. However, solving such an optimization problem is not trivial as both the gradient operator and the GAN latent space are highly non-linear and non-convex, and the defense methods applied at the client's side also inject noises into the objective function. To resolve this, we design an adaptive loss function against common defenses by considering the underlying gradient transformation and resort to gradient-free optimization methods (e.g., evolution strategies [19] and Bayesian optimization [10]) to search for the global minima within the GAN latent space. We empirically demonstrate that compared with gradient-based optimizers, doing so significantly reduces the chance of converging to a local minimum, leading to a higher quality of reconstructed images as well as improved similarity to the client's private image. We note that the findings made from the chosen defense settings and datasets may not be general in scope. Nevertheless, we expect the proposed method can serve as a means for privacy auditing in FL by showing how much an adversary can learn under a specific defense setting

to assist the future design of privacy mechanisms.

Our main contributions are summarized as follows:

- We propose to solve the inverse problem of gradient leakage in FL under noises and defensive transformations by leveraging the prior information learned from deep generative models.
- We systematically study 4 types of gradient-degradation-based defenses, including additive noise, gradient clipping, gradient compression, and representation perturbation, and design adaptive loss functions by accounting for the underlying gradient transformation.
- To avoid sub-optimal solutions and reveal more private information, we compare different gradient-free optimizers with conventional gradient-based optimizers (e.g., Adam) and experimentally show their superiority for gradient leakage attack in terms of reconstructed image quality and its similarity to the client's private image.
- We demonstrate on two image datasets (i.e., CelebA [31] and ImageNet [9]) that with the proposed GGL, high-resolution images can still be recovered from the shared gradients even with the considered defenses, while existing gradient leakage attacks all fail.

## 2. Related Work

### 2.1. Privacy Leakage via Gradient

The studies on privacy leakage in FL originate from *membership inference*, where a malicious analyst infers whether a specific data sample has been involved in the training set [38]. Moreover, researchers have discovered that the exchanged model updates can be utilized to further infer unintended private information, such as the retrieval of certain *input attributes* [11, 35] (e.g., whether people in the training data wear glasses). Further studies find it is possible to recover class-level [24] or even client-level *data representatives* [47] (i.e., prototypical samples of the private training set) through generative modeling.

**Data Reconstruction Attacks.** Recently, Zhu *et al.* [56] demonstrate a more severe type of privacy threat where an attacker can fully restore the client's private data samples by solving for the optimal pair of input and label that best matches the exchanged gradients. A follow-up work [54] improves on this method by proposing a method for analytically extracting the label information. However, these methods are limited to shallow networks trained with low-resolution images. A later study by Geiping *et al.* [13] extends this attack to more realistic scenarios by successfully restoring ImageNet-level high-resolution data from deeper networks (e.g., ResNet [23]) using a magnitude-invariant loss design. Along this direction, a more recent work by Yin *et al.* [51] even achieves image batch reconstruction by utilizing the strong prior encoded in batch normalization

statistics. Despite the improvement, the current research efforts on data reconstruction attacks often assume an ideal setting by targeting a bare-bone FL system without applying any additional privacy-preserving measures or defenses, which contradicts industrial practices.

## 2.2. Privacy Preservation in FL

Existing research efforts for achieving privacy preservation in FL can be generally categorized into *cryptography-based* and *gradient-degradation-based* approaches.

A common type of cryptographic solution is secure multi-party computation (MPC), which aims to have a set of parties to jointly compute the output of a function over their private inputs in a way that only the intended output is revealed to the parties. This can be achieved by designing custom protocols [1, 37], or via secure aggregation schemes such as homomorphic encryption [22] and secret sharing [49]. However, merely relying on MPC isn't sufficient to resist inference attacks over the output [35, 45].

Another line of research seeks to constrain the amount of leaked sensitive information by intentionally sharing degraded gradients. Differential privacy (DP) is the standard way to quantify and limit the privacy disclosure about individual users. DP can be applied at either the server's side (central DP) or the client's side (local DP). In comparison, local DP provides a better notion of privacy as it does not require the client to trust anyone. It utilizes a randomized mechanism to distort the gradients before sharing them with the server [14, 48]. DP offers a worst-case information theoretic guarantee on how much an adversary can learn from the released data. However, for these worst-case bounds to be most meaningful, they typically involve adding too much noise which often reduces the utility of the trained models. In addition to DP, it is demonstrated that performing gradient compression/sparsification can also help to prevent information leakage from the gradients [56]. A most recent work by Sun *et al.* [44] identifies the data representation leakage from gradients as the root cause of privacy leakage in FL and proposes a defense named Soteria, which computes the gradients based on perturbed data representations. It is shown that Soteria can achieve a certifiable level of robustness while maintaining good model utility.

## 3. Methodology

### 3.1. Threat Model

In most existing data leakage attacks [13, 51, 54, 56], the adversary is considered to be an honest-but-curious server and has access to the current FL model as well as the shared gradients. As illustrated in Figure 2a, we further assume that clients apply a privacy defense locally on the gradients computed from their private data, and the adversary can only access the degraded gradients modified by the defense

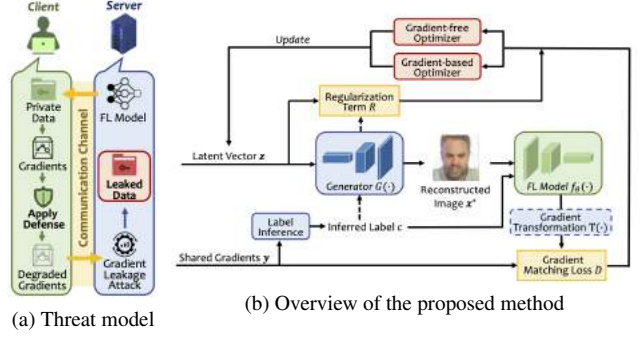


Figure 2. Illustration of the threat model and the proposed method.

mechanism. The adversary's objective is to reveal as much private information as possible from the degraded gradients. The adversary may or may not know the underlying defense strategy adopted by the client. In either case, the adversary could attempt to launch an adaptive attack by directly using this knowledge or by estimating the defense parameters through the observed gradients. Additionally, we assume the adversary can utilize the knowledge extracted from publicly available datasets (disjoint from client's private data) to facilitate and improve the attack.

### 3.2. Background

**Problem Formulation.** The task of reconstructing a training image  $\mathbf{x} \in \mathbb{R}^d$  from its gradients  $\mathbf{y} \in \mathbb{R}^m$  can be formulated as a non-linear inverse problem:

$$\mathbf{y} = F(\mathbf{x}), \quad (1)$$

where  $F(\mathbf{x}) = \nabla_{\theta} \mathcal{L}(f_{\theta}(\mathbf{x}), c)$  is the forward operator that calculates the gradients of the loss  $\mathcal{L}$  provided with  $\mathbf{x}$  and its label  $c$ , along with the FL model  $f_{\theta}$  parameterized by  $\theta$ . When defense is applied at the client's side, the reconstructing problem defined in Equation 1 becomes:

$$\mathbf{y} = \mathcal{T}(F(\mathbf{x})) + \epsilon, \quad (2)$$

where  $\mathcal{T}(\cdot)$  is referred to as the lossy transformation (e.g., compression or sparsification) and  $\epsilon$  means the additive noise (e.g., DP) introduced by the defense algorithm.

**Current Approach and Its Limitation.** Existing methods [13, 51, 56] aim to solve this inverse problem by using image priors in a penalty form:

$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathbb{R}^d}{\operatorname{argmin}} \mathcal{D}(\mathbf{y}, F(\mathbf{x})) + \lambda \omega(\mathbf{x}), \quad (3)$$

where  $\mathcal{D}(\cdot)$  is a distance metric,  $\omega(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$  is the standard image prior (e.g., total variance [2] regularization) and  $\lambda$  is the weight factor. This form has been demonstrated effective for reconstructing images from the actual gradients. However, when reconstructing from a set of low-fidelity and

noisy gradients, such methods would suffer from the limited identification ability of hand-crafted priors, rendering them to return false solutions that are not valid natural images, which is illustrated in Section 4.4.

### 3.3. Generative Gradient Leakage

Motivated by the success of deep generative models for compressed sensing [3, 46], in this work, we aim to leverage a generative model trained on public datasets as the learned natural image prior to ensure the reconstructed image quality. Moreover, to further account for the privacy defenses that produce degraded gradient information, we propose an adaptive attack by estimating the transformation  $\mathcal{T}(\cdot)$  and incorporating it in the optimization process. Specifically, given a well-trained generator  $G(\cdot)$ , we target to solve the following optimization problem:

$$\mathbf{z}^* = \underset{\mathbf{z} \in \mathbb{R}^k}{\operatorname{argmin}} \underbrace{\mathcal{D}(\mathbf{y}, \mathcal{T}(F(G(\mathbf{z}))))}_{\text{gradient matching loss}} + \lambda \underbrace{\mathcal{R}(G; \mathbf{z})}_{\text{regularization}}, \quad (4)$$

where  $\mathbf{z} \in \mathbb{R}^k$  is the latent space of the generative model,  $\mathcal{R}(G; \mathbf{z})$  is a regularization term that penalizes latent vectors which deviate from the prior distribution, and  $\lambda$  is the weight factor. Once the optimal solution  $\mathbf{z}^*$  is obtained, the image can be reconstructed by  $G(\mathbf{z}^*)$ . An overview of the proposed method is provided in Figure 2b. We next describe each component in detail.

**Label Inference.** Given the shared gradients, the adversary can first adopt an analytical method [54] to infer the ground truth label  $c$  associated with the client’s private image  $\mathbf{x}$ . Specifically, for FL models performing classification task over  $n$  classes, the  $i^{\text{th}}$  entry of the gradients with respect to the weights of the final fully-connected (FC) classification layer (denoted as  $\nabla \mathbf{W}_{FC}^i$ ) is given by:

$$\nabla \mathbf{W}_{FC}^i = \frac{\partial \mathcal{L}(f_{\theta}(\mathbf{x}), c)}{\partial z_i} \times \frac{\partial z_i}{\partial \mathbf{W}_{FC}^i}, \quad (5)$$

where  $z_i$  is the  $i^{\text{th}}$  output of the FC layer. Note that computing the second term  $\frac{\partial z_i}{\partial \mathbf{W}_{FC}^i}$  results in the post-activation outputs of the previous layer, which will be always non-negative if activation functions like ReLU or sigmoid are applied. For networks trained with cross-entropy loss on one-hot labels (assuming softmax is applied at the last layer), the first term will be negative if and only if  $i = c$ . Thus the ground truth label can be retrieved by identifying the index of the negative entry of  $\nabla \mathbf{W}_{FC}^i$ . The inferred label will be used for evaluating the FL model training loss  $\mathcal{L}(f_{\theta}(\mathbf{x}), c)$ . For conditional GANs [36], the inferred label will also be used as the class condition.

**Gradient Transformation Estimation.** The adversary can further attempt to mitigate the impact of the defense by adopting a similar transformation when evaluating the loss

of reconstructed images. Although the transformation process at the client’s side isn’t directly known to the adversary, the adversary can estimate the parameters of the transformation through the observed gradients. Specifically, we consider the following defensive transformations (i.e.,  $\mathcal{T}(\cdot)$ ):

(1) *Gradient Clipping*: A common technique used in DP studies [14, 48] to restrict the contribution of each individual client. Given a clipping bound  $S$ , gradient clipping transforms the gradients as  $\mathcal{T}_{cli}(\mathbf{y}, S) = \mathbf{y} / \max(1, \frac{\|\mathbf{y}\|_2}{S})$ . In practice, gradient clipping is often done in a layer-wise manner. The adversary can take the  $\ell_2$  norm at each layer of the observed gradients as the estimated clipping bound.

(2) *Gradient Sparsification*: Originally proposed for reducing the communication bandwidth of distributed training [30], gradient sparsification is also reported to be effective for defending against gradient leakage attacks [56]. Specifically, given a pruning rate  $p \in (0, 1)$ , the client first computes a threshold  $\tau \leftarrow p$  of  $|\mathbf{y}|$ , which is then used to produce a mask  $\mathcal{M} \leftarrow |\mathbf{y}| > \tau$ . Finally, the mask is applied to the gradients during the transformation, i.e.,  $\mathcal{T}_{spa}(\mathbf{y}, p) = \mathbf{y} \odot \mathcal{M}$ . This operation is also layer-wise. The adversary can use the percentage of non-zero entries in the observed gradient to estimate its sparsity.

(3) *Representation Perturbation*: The core of the recently proposed Soteria [44] defense is to prevent data leakage by perturbing the representation learned from a single fully-connected layer  $L$  (i.e., the defended layer) to cause maximal reconstruction error. Assume  $f_r : \mathbb{R}^d \rightarrow \mathbb{R}^l$  is the feature extractor before the defended layer that maps  $\mathbf{x} \in \mathbb{R}^d$  to a  $l$ -dimensional data representation  $\mathbf{r} \in \mathbb{R}^l$ . Specifically, the client first evaluates the impact of each entry of the representation by computing  $\{\|r_i(\nabla_{\mathbf{x}} f_r(r_i))^{-1}\|_2 : i \in \{0, 1, \dots, l-1\}\}$ . Given a pruning rate  $p \in (0, 1)$ , the client then prunes the  $p \times l$  elements in  $\mathbf{r}$  with the largest  $\|r_i(\nabla_{\mathbf{x}} f_r(r_i))^{-1}\|_2$  values to get  $\mathbf{r}'$ . Finally, the client computes the gradients on the perturbed representation  $\mathbf{r}'$ . This can be thought as applying a mask only to the gradients of the defended layer:  $\mathcal{T}_{rep}(\mathbf{y}, p) = \mathbf{y} \odot \mathcal{M}_L$ . As this process is deterministic for a given  $\mathbf{x}$  and FL model  $f_{\theta}$ , the adversary can reverse-engineer this mask according to the non-zero entries of the gradients from the defended layer.

**Gradient Matching Loss.** The first term in the objective function (Equation 4) encourages the solver to find images that are contextually similar to the client’s private training images in the generator’s latent space by minimizing the distance between the transformed gradients of the generated images  $\tilde{\mathbf{y}}$  and the observed gradients  $\mathbf{y}$ . We explore the following distance metrics for calculating the gradient matching loss: (1) *Squared  $\ell_2$  norm* [51, 54, 56]:  $\mathcal{D}_1(\mathbf{y}, \tilde{\mathbf{y}}) = \|\mathbf{y} - \tilde{\mathbf{y}}\|_2^2$ ; and (2) *Cosine Distance* [13]:  $\mathcal{D}_2(\mathbf{y}, \tilde{\mathbf{y}}) = 1 - \frac{\langle \mathbf{y}, \tilde{\mathbf{y}} \rangle}{\|\mathbf{y}\|_2 \|\tilde{\mathbf{y}}\|_2}$ . Cosine distance is magnitude-invariant and is equivalent to optimizing the Euclidean distance of two normalized gradient vectors.



Reg.	Grad.	$\mathcal{D}_1$		$\mathcal{D}_2$	
		MSE-I ↓	PSNR ↑	MSE-I ↓	PSNR ↑
$\mathcal{R}_1$		<b>0.0320±0.0173</b>	<b>15.6814±2.6387</b>	0.03671±0.0227	15.3471±3.1093
$\mathcal{R}_2$		0.0337±0.0206	15.5405±2.7090	0.06290±0.0815	14.3249±4.1627

Table 1. Comparison of different loss function configurations.

**Regularization Term.** Optimizing with gradient matching loss alone is likely to produce latent vectors that deviate from the generator’s latent distribution, resulting in unrealistic images with significant artifacts. To avoid this issue, we explore the following loss functions to regularize the latent vector during the optimization process: (1) *KL-based regularization* [28]:  $\mathcal{R}_1(G; \mathbf{z}) = -\frac{1}{2} \sum_{i=1}^k (1 + \log(\sigma_i^2) - \mu_i^2 - \sigma_i^2)$ , where  $\mu_i$  and  $\sigma_i$  denote the element-wise mean and standard deviation. The KL term aims to reduce the Kullback–Leibler divergence (KLD) between the latent distribution and the standard Gaussian distribution  $\mathcal{N}(0, \mathbf{I})$ ; and (2) *Norm-based regularization* [7]:  $\mathcal{R}_2(G; \mathbf{z}) = (\|\mathbf{z}\|_2^2 - k)^2$ , which penalizes latent vectors that are far from the prior distribution.

**Optimization Strategy.** The target inverse problem described in Equation 4 is highly non-linear and non-convex, and thus choosing the right optimization strategy becomes a critical factor for achieving good image reconstruction. Existing data reconstruction attacks are all based on gradient-based optimizers such as L-BFGS [54, 56] and Adam [13, 51]. The outcome of such local optimization strategies highly depends on the choice of initialization and often requires multiple trials to find a decent solution. Moreover, we find that for more complex generative models, gradient-based optimizers are likely to converge to local minima, leading to poor reconstruction results. Inspired by Huh *et al.* [25], besides gradient-based optimizers, we further explore two gradient-free optimization strategies to overcome these issues:

(1) *Bayesian Optimization (BO)* [43]: BO is a global optimization method that can well handle stochastic noise in blackbox functions, which are modeled by a Gaussian process. Vanilla BO scales poorly to high-dimensional problems [43] and thus we adopt a variant of BO, namely, trust region BO (TuRBO) [10], for performing a global search in the high-dimensional latent space of the GAN model.

(2) *Covariance Matrix Adaptation Evolution Strategy (CMA-ES)* [19]: CMA-ES leverages a multivariate normal sampling distribution over the search space. At each step, a stochastic search is performed by drawing samples from that distribution to compute the loss. Evolutionary strategies such as recombination and mutation are used to adaptively update its mean and covariance matrix [18].



Figure 3. Visual comparison of different optimizers. The images on the right are the reconstruction samples produced by three types of optimizers with different random seeds.

Dataset	Metric	Adam		BO		CMA-ES	
		Mean	Std.	Mean	Std.	Mean	Std.
CelebA	MSE-I ↓	<b>0.0427</b>	0.0025	0.0813	0.0131	0.0708	<b>0.0008</b>
	PSNR ↑	<b>13.6965</b>	0.2593	10.9455	0.6816	11.4989	<b>0.0533</b>
	LPIPS ↓	<b>0.1435</b>	<b>0.0083</b>	0.2162	0.0328	0.2136	0.0133
	MSE-R ↓	<b>0.0003</b>	<b>0.0001</b>	0.0012	0.0003	0.0015	0.0022
ImageNet	MSE-I ↓	0.5918	0.1955	<b>0.2648</b>	0.0181	0.2667	<b>0.0119</b>
	PSNR ↑	2.4433	1.3565	<b>5.7783</b>	0.2992	5.7420	<b>0.1988</b>
	LPIPS ↓	0.7983	0.0280	0.6166	0.0590	<b>0.5736</b>	<b>0.0209</b>
	MSE-R ↓	0.1051	0.0703	0.0035	0.0005	<b>0.0018</b>	<b>0.0002</b>

Table 2. Quantitative comparison of different optimizers.

## 4. Experiments

### 4.1. Experimental Setup

**FL Tasks & Datasets.** We evaluate our method on two FL tasks: (1) *Gender Classification*: Binary gender classification performed on the CelebFaces attributes dataset (CelebA) [31] with images of size  $32 \times 32$ ; and (2) *Image Classification*: 1000-class image classification on the ImageNet ILSVRC 2012 dataset [9] with images of size  $224 \times 224$ . The FL model for all tasks adopts the ResNet-18 [23] architecture with randomly initialized weights. We consider the case where the client performs one local step with batch size = 1 to compute the gradients.

**Implementation.** For CelebA dataset, we use the training set containing 162k images to train a DCGAN [40] on the Wasserstein loss with gradient penalty [17], while the rest images are reserved for evaluation. For experiments on ImageNet dataset, we use a pretrained BigGAN [6] released by the authors [5]. Note that the FL task is performed on the evaluation set which is disjoint from the GAN training set. We use the gradients computed from the FL model after applying defenses to conduct reconstruction.

**Evaluation Metrics.** Besides qualitative visual comparison, we use the following metrics for quantitative evaluation of the similarity between the target image and the reconstructed image: (1) *Mean Square Error - Image Space (MSE-I ↓)*: the pixel-wise MSE between the target image and the reconstructed image; (2) *Peak Signal-to-Noise Ra-*

Table 3. Quantitative comparison of GGL with state-of-the-art methods under various defenses.

Dataset	Attack	Additive Noise [44, 56]				Gradient Clipping [14, 48]				Gradient Sparsification [56]				Soteria [44]			
		MSE-I ↓	PSNR ↑	LPIPS ↓	MSE-R ↓	MSE-I ↓	PSNR ↑	LPIPS ↓	MSE-R ↓	MSE-I ↓	PSNR ↑	LPIPS ↓	MSE-R ↓	MSE-I ↓	PSNR ↑	LPIPS ↓	MSE-R ↓
CelebA	DLG [56]	0.6479	1.8843	0.8197	0.0021	0.2097	6.7831	0.7375	0.0326	0.3335	4.7679	0.7986	0.0155	0.3624	4.4069	0.8007	0.0285
	iDLG [54]	0.6261	2.0329	0.8209	0.0025	0.1960	7.0762	0.7280	0.0326	0.3301	4.8124	0.8035	0.0162	0.3269	4.8553	0.8036	0.0396
	IG [13]	0.4880	3.1151	0.8260	0.0097	<b>0.0543</b>	<b>12.6517</b>	0.2998	<b>0.0003</b>	0.4103	3.8687	0.7975	0.0113	0.3441	4.6326	0.8008	0.0316
	GI [51]	0.5738	2.4116	0.8302	0.0023	0.1790	7.4701	0.7142	0.0322	0.2958	5.2888	0.7775	0.0163	0.3179	4.9768	0.7991	0.0409
	<b>GGL</b>	<b>0.0780</b>	<b>11.0766</b>	<b>0.1906</b>	<b>0.0010</b>	0.0760	11.1902	<b>0.1670</b>	0.0015	<b>0.0768</b>	<b>11.1466</b>	<b>0.1620</b>	<b>0.0007</b>	<b>0.0968</b>	<b>10.1434</b>	<b>0.2561</b>	<b>0.0007</b>
ImageNet	DLG [56]	0.7438	1.2852	0.9353	0.0049	0.3809	4.1912	0.9798	2.1610	0.4432	3.5336	0.8907	0.0075	0.5990	2.2253	0.9195	0.5415
	iDLG [54]	0.7352	1.3359	0.9392	0.0041	0.3699	4.3190	0.9473	1.8810	0.4357	3.6077	0.8935	0.0077	0.6089	2.1542	0.9198	0.5425
	IG [13]	0.3081	5.1120	0.8677	0.4490	<b>0.1432</b>	<b>8.4386</b>	0.7476	0.0214	0.2993	5.2376	0.8805	0.0501	0.3683	4.3373	0.8700	0.5057
	GI [51]	0.6593	1.8090	0.9448	0.0031	0.3702	4.3154	0.9451	1.8807	0.4404	3.5611	0.8889	0.0072	0.6235	2.0511	0.9169	0.5792
	<b>GGL</b>	<b>0.2686</b>	<b>5.7089</b>	<b>0.5915</b>	<b>0.0018</b>	0.2230	6.5163	<b>0.5592</b>	<b>0.0015</b>	<b>0.2141</b>	<b>6.6920</b>	<b>0.5170</b>	<b>0.0017</b>	<b>0.2484</b>	<b>6.0477</b>	<b>0.5685</b>	<b>0.0022</b>

*tio* (PSNR ↑): The ratio of the maximum squared pixel fluctuation and the MSE between the target image and the reconstructed image; (3) *Learned Perceptual Image Patch Similarity* (LPIPS ↓) [52]: the perceptual image similarity between the target image and the reconstructed image measured by a VGG network [42], and (4) *MSE - Representation Space* (MSE-R ↓): the MSE between the target image and the reconstructed image measured in the learned representation space, i.e., the feature vector before the final classification layer [44]. Note that “↓” means the lower the metric the higher relative image quality, while “↑” represents the higher the metric the higher image quality.

## 4.2. Choice of Loss Function

We first evaluate the performance of different loss function configurations. We randomly select 10 images from the evaluation set of the CelebA dataset and measure the mean and standard deviation of the MSE-I and PSNR scores between the original images and their reconstructions using Adam optimizer. From results presented in Table 1 we observe that using squared  $\ell_2$  norm ( $\mathcal{D}_1$ ) for computing the gradient matching loss with KLD as the regularization term ( $\mathcal{R}_1$ ) yields the best reconstructed image quality. Therefore, hereinafter we use this loss configuration for analyzing the impact of different optimizers and defenses.

## 4.3. Choice of Optimization Strategy

We next study the impact of different optimizers on the reconstruction results. We randomly select images from the CelebA and ImageNet dataset to compute the reconstruction and repeat the experiment by varying its random seed. The numbers of updates are set to 2500, 1000, and 800 for Adam, BO, and CMA-ES, respectively. We summarize the results in Table 2 and provide visualization of the reconstruction samples in Figure 3. We find that the gradient-based and gradient-free optimizers show similar performance on the CelebA dataset, with Adam performing slightly better both visually and statistically. However, on the ImageNet dataset, the gradient-based Adam optimizer fails to recover any useful information from the gradients other than the class label. Moreover, its reconstruction results are highly dependent on the initialization. The gradient-free optimizers (BO and CMA-ES), on the other

hand, are still able to find samples that resemble the original private image and are more resilient to different initialization conditions. The reason causing this performance difference is twofold: (1) the images in the CelebA dataset are well-aligned, while the ImageNet dataset has a more heterogeneous data distribution; and (2) the generator used for generating high-resolution ImageNet data has a deeper and more complex structure, which makes it hard for gradient-based optimizers to find a projection in its latent space. Based on this observation, we choose to use CMA-ES as the optimizer for conducting experiments under various defense settings.

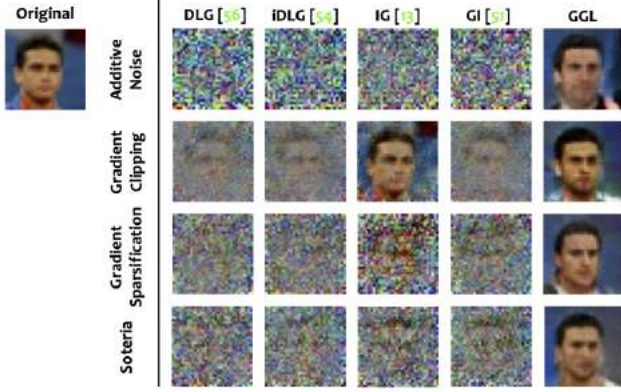
## 4.4. Comparison with Existing Gradient Leakage Attacks Under Defenses

**Attack Baselines.** We compare our method with several state-of-the-art attack methods: (1) *Deep Leakage from Gradients* (DLG) [56]: gradient leakage attack with  $\ell_2$  gradient matching loss and L-BFGS optimizer; (2) *Improved Deep Leakage from Gradients* (iDLG) [54]: improved DLG attack with label inference; (3) *Inverting Gradients* (IG) [13]: gradient leakage attack with cosine distance as loss and total variation as prior, optimized using Adam; and (4) *GradInversion* (GI) [51]: gradient leakage attack with  $\ell_2$  gradient matching loss and Adam optimizer.

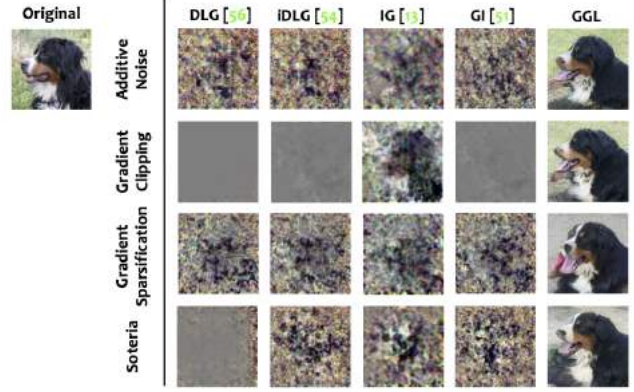
We implemented these attacks following the code repositories released by the authors [12, 53, 55]. In our implementation of GI, we consider a stricter scenario where the batch normalization statistics are unknown to the adversary. For the second-order-based DLG and iDLG attacks, we use the L-BFGS optimizer to conduct 300 iterations of optimization on the CelebA dataset and 1,200 iterations on the ImageNet dataset to reconstruct the data. As for the first-order-based IG and GI attacks, we use the Adam optimizer with an initial learning rate of 0.1 and conduct 8,000 iterations of optimization on CelebA and 24,000 iterations on ImageNet. The performance of several existing methods is highly varying according to different random seeds. To mitigate this, each attack is given 4 trials and the best result with the lowest loss is selected as its final reconstruction.

### Defense Scheme.

Following prior studies [44, 56], we choose a relatively strict defense setting for conducting evaluation: (1) *Additive*



(a) CelebA (32 × 32 pixels)



(b) ImageNet (224 × 224 pixels)

Figure 4. Comparison of the reconstruction results with attack baselines on the CelebA & ImageNet datasets under various privacy defenses.

*Noise* [44, 56]: inject a Gaussian noise  $\varepsilon \sim \mathcal{N}(0, \sigma^2 \mathbf{I})$  to the gradients with  $\sigma = 0.1$ ; (2) *Gradient Clipping* [14, 48]: clip the values of the gradients with a bound of  $S = 4$ ; (3) *Gradient Sparsification* [56]: perform magnitude-based pruning on the gradients to achieve 90% sparsity; and (4) *Soteria* [44]: gradients are generated on the perturbed representation with a pruning rate of 80%.

**Results.** Table 3 compares the performance of the proposed method GGL with other gradient leakage attack methods. Our general observation is that existing attack methods struggle to reconstruct a realistic image with the present of any privacy defense mechanism, while the proposed GGL is able to synthesize high quality images that are similar to the original ones, with the measured PSNR  $> 10.1$  on the CelebA dataset and  $> 5.7$  on ImageNet dataset across all scenarios. One exception is that we find the gradient clipping operation has a very low effect on the IG attack. This is because clipping to  $\ell_2$  norm only changes the magnitude of the gradients and does not affect the angular information (i.e., direction). Therefore, though gradient clipping increases the reconstruction error for attacks based on the Euclidean distance between gradients, it will not affect the IG attack which utilizes the magnitude-invariant cosine distance for computing its gradient matching loss. Clipping to  $L_\infty$  norm instead would address this issue, however, it is not adopted by existing DP mechanisms as it will result in a poor  $\ell_2$  bound. We also notice that comparing to gradient sparsification, reconstructing from the gradients produced from the perturbed data representation using the Soteria defense would result in higher MSE in both the image space and the representation space, especially on the ImageNet dataset. Despite this, such defense can still be bypassed by our adaptive attack.

From the visualization results in Figure 4, we can see that except for the IG attack in the case of gradient clipping, the reconstructed image of existing attacks does not



Figure 5. Reconstruction results against the Soteria [44] defense on the ImageNet dataset: (top) original image and its (bottom) reconstruction by GGL.

reveal much information about the original image. We also observe that on the CelebA dataset, the proposed method GGL isn't able to reconstruct the exact face of the person in the original image when defenses are applied, yet it successfully reveals several key attributes including gender, hair style, hair color, skin color, head posture, and even the background color. Even on the more challenging ImageNet dataset, our method can still produce a high quality reconstruction that reveals the composition of the original image under these defenses. More samples on the ImageNet dataset against the Soteria defense is presented in Figure 5.

**Combining Clipping and Noise Addition.** In addition, we also evaluate our attack against the combination of multiple defense mechanisms. Figure 6 compares the reconstruction results under 3 defense settings: additive noise with  $\sigma = 0.1$ , gradient clipping with  $S = 4$ , and simultaneously applying gradient clipping and additive noise (i.e., the privacy defense used in local and distributed DP). We observe that the high-resolution image can still be reconstructed under these defenses, and combining gradient clipping and additive noise would lead to a relatively worse re-



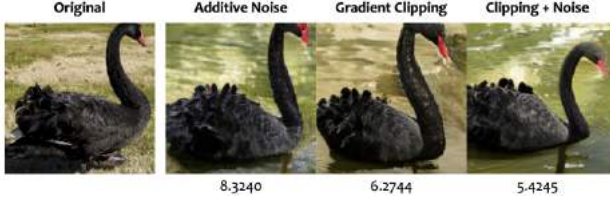


Figure 6. Illustration of combined defense: (left) original image and its (right) reconstruction by GGL. The PSNR with respect to the original image is shown below each reconstructed image.

construction with the lowest PSNR. We thus believe this attack can also be used as an auditing measurement for local differential privacy.

#### 4.5. Impact of Defense Parameter

We next apply the Soteria [44] defense on the CelebA dataset as a case study to investigate the impact of different defense parameters. We use the attack baselines and the proposed GGL to generate reconstructions as we vary the pruning rate from 0% to 80%, and summarize the results in Figure 7. The authors reported in their original paper [44] that the DLG [56] and IG [13] attack can tolerate the Soteria defense with a pruning rate up to 40% on the CIFAR10 dataset. Differently, we observe that on the CelebA dataset, defense with a low pruning rate of 10% would already impose a significant impact on the reconstruction results of these attacks. This is perhaps because the Soteria defense mainly affects the fully-connected layer that produces class-level data representation. Different from CIFAR10, the class-wise label of the CelebA dataset does not directly reveal contextual information about the subject (e.g., the identity of the person). Instead, it only encodes very coarse-grained information (i.e., gender) and thus can be more susceptible to perturbations. In other words, privacy information that is entangled with the class label is more likely to be leaked through gradients. Nevertheless, the proposed method can still reliably recover the profile of the person from the remaining gradients regardless of the pruning rate.

## 5. Discussion

**Limitation.** Although the image prior captured by the GAN model can help restore the missing information from the degraded gradients for better image reconstruction, at the same time the output image distribution is also constrained by the GAN latent space, rendering it hard to faithfully reconstruct out-of-distribution image samples. Figure 8 shows two examples of attempting to reconstruct out-of-distribution ImageNet images under the Soteria defense [44]: in Figure 8a, the orientation of the object reconstructed image is changed from the original image; and

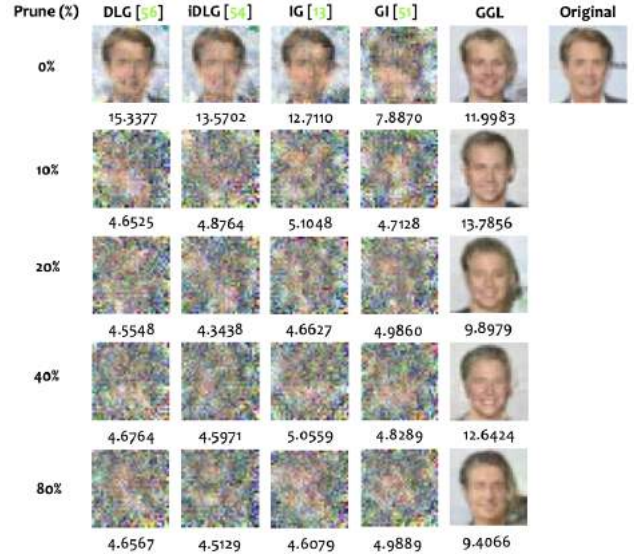


Figure 7. Reconstruction results under the Soteria [44] defense with varying pruning rates on the CelebA dataset. The PSNR with respect to the original image is shown below each reconstructed image.

Figure 8b, the reconstruction result is missing important semantics (e.g., the person) that is not well-represented in its class (i.e., Bernese mountain dog). These phenomena can potentially be improved by jointly optimizing the class condition [25] or relaxing the generator [39].

**Analysis of Loss Landscape and Potential Defense.** To investigate the reconstruction problem under the constraint of a generative model, we use the latent vector returned by GGL as the central point and choose two directions to visualize the loss landscape of the gradient matching loss as well as the LPIPS loss between the original image and the image generated by the BigGAN model by sampling in the latent space. The visualization results are presented in Figure 9, where Figure 9a shows the loss landscape observed by the adversary if only the gradient information is accessible, and Figure 9b shows the ground truth loss landscape measured by the LPIPS score assuming the original image is known. We have the following two observations: (1) the surface of the gradient matching loss is non-convex and contains several local minima; and (2) there exists an inconsistency between the ground truth and the observed loss surface, i.e., the image found by optimizing the gradient matching loss doesn't provide the most similar visual result. However, as showed in our experiments, such a level of inconsistency isn't sufficient to provide privacy guarantees as the suboptimal result with minimized gradient matching loss still leaks a considerable amount of information about the original image. This hints us that applying transformations to the gradients to reform the gradient matching loss so that its landscape is no longer in line with the ground truth LPIPS loss





Figure 8. Reconstruction results of out-of-distribution image samples: (left) original image and its (right) reconstruction by GGL.

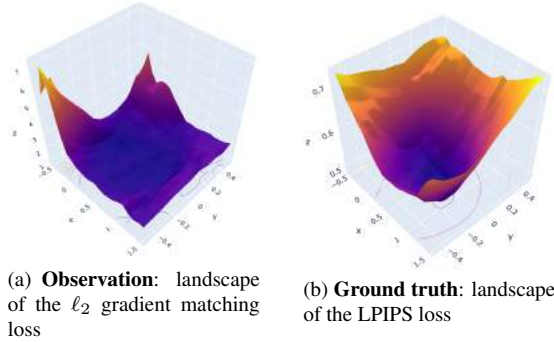


Figure 9. Visualization of the loss landscapes.

can help to effectively achieve privacy preservation against generative gradient leakage attacks.

## 6. Conclusion

This work presents Generative Gradient Leakage (GGL), an approach that utilizes a generative model to extract prior information from public datasets to improve image reconstruction from degraded gradients produced by privacy defenses. Our experimental results on two image classification datasets show that with the learned image prior, the proposed method is more resilient to the perturbations and lossy transformations applied to the gradients and is still able to reconstruct high-fidelity images that reveal information about the original images when existing attacks all fail. We hope the proposed method can serve as an analysis tool for empirical privacy auditing to help facilitate the future design of privacy defenses.

## Acknowledgement

The authors would like to thank Peter Kairouz from Google Research for his valuable feedback on the paper. This work is supported in part by NSF CNS-2114161, ECCS-2132106, CBET-2130643, the Science Alliance’s StART program, and the GCP credits provided by Google Cloud. This work is also supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Applied Mathematics program; and by the Artificial Intelligence Initiative at the Oak Ridge National Laboratory (ORNL). ORNL is operated by UT-

Battelle, LLC., for the U.S. Department of Energy under Contract DEAC05-00OR22725.

## References

- [1] Nitin Agrawal, Ali Shahin Shamsabadi, Matt J Kusner, and Adrià Gascón. Quotient: two-party secure neural network training and prediction. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1231–1247, 2019. 3
- [2] Amir Beck and Marc Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE transactions on image processing*, 18(11):2419–2434, 2009. 3
- [3] Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G Dimakis. Compressed sensing using generative models. In *International Conference on Machine Learning*, pages 537–546. PMLR, 2017. 4
- [4] Theodora S Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International journal of medical informatics*, 112:59–67, 2018. 1
- [5] Andrew Brock and Alex Andonian. BigGAN PyTorch Implementation. <https://github.com/ajbrock/BigGAN-PyTorch>. Accessed: 2021-11-09. 5
- [6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2018. 2, 5, 12
- [7] Dingfan Chen, Ning Yu, Yang Zhang, and Mario Fritz. Ganleaks: A taxonomy of membership inference attacks against generative models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*, pages 343–362, 2020. 5
- [8] Olivia Choudhury, Yoonyoung Park, Theodoros Salonidis, Aris Gkoulalas-Divanis, Issa Sylla, et al. Predicting adverse drug reactions on distributed health data using federated learning. In *AMIA Annual symposium proceedings*, volume 2019, page 313. American Medical Informatics Association, 2019. 1
- [9] 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. 2, 5
- [10] David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. *Advances in Neural Information Processing Systems*, 32:5496–5507, 2019. 2, 5, 12
- [11] 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 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 619–633, 2018. 2
- [12] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting Gradients Implemen-

- tation. <https://github.com/JonasGeiping/invertinggradients>. Accessed: 2021-11-09. **6**
- [13] 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, 2, 3, 4, 5, 6, 8, 13**
  - [14] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017. **1, 3, 4, 6, 7, 11**
  - [15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. **2**
  - [16] Filip Granqvist, Matt Seigel, Rogier van Dalen, Áine Cahill, Stephen Shum, and Matthias Paulik. Improving on-device speaker verification using federated learning with privacy. In *Interspeech*, pages 4328–4332, 2020. **1**
  - [17] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017. **5, 12**
  - [18] Nikolaus Hansen. The cma evolution strategy: a comparing review. *Towards a new evolutionary computation*, pages 75–102, 2006. **5**
  - [19] Nikolaus Hansen. The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016. **2, 5**
  - [20] Andrew Hard, Kurt Partridge, Cameron Nguyen, Niranjan Subrahmanya, Aishanee Shah, Pai Zhu, Ignacio Lopez Moreno, and Rajiv Mathews. Training keyword spotting models on non-iid data with federated learning. *arXiv preprint arXiv:2005.10406*, 2020. **1**
  - [21] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018. **1**
  - [22] Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith, and Brian Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*, 2017. **3**
  - [23] 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. **2, 5**
  - [24] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. Deep models under the gan: information leakage from collaborative deep learning. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 603–618, 2017. **2**
  - [25] Minyoung Huh, Richard Zhang, Jun-Yan Zhu, Sylvain Paris, and Aaron Hertzmann. Transforming and projecting images into class-conditional generative networks. In *European Conference on Computer Vision*, pages 17–34. Springer, 2020. **5, 8, 12**
  - [26] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019. **1**
  - [27] 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*, pages 8110–8119, 2020. **2**
  - [28] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. **5**
  - [29] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020. **1**
  - [30] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *International Conference on Learning Representations*, 2018. **4**
  - [31] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. **2, 5**
  - [32] Guodong Long, Yue Tan, Jing Jiang, and Chengqi Zhang. Federated learning for open banking. In *Federated learning*, pages 240–254. Springer, 2020. **1**
  - [33] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015. **2**
  - [34] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. **1**
  - [35] 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. **1, 2, 3**
  - [36] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. **4**
  - [37] Payman Mohassel and Yupeng Zhang. Secureml: A system for scalable privacy-preserving machine learning. In *2017 IEEE symposium on security and privacy (SP)*, pages 19–38. IEEE, 2017. **3**
  - [38] Milad Nasr, Reza Shokri, and Amir Houmansadr. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *Proceedings of the IEEE symposium on security and privacy (SP)*, pages 739–753. IEEE, 2019. **2**
  - [39] Xingang Pan, Xiaohang Zhan, Bo Dai, Dahua Lin, Chen Change Loy, and Ping Luo. Exploiting deep generative prior for versatile image restoration and manipulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. **8**
  - [40] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. **5, 12**

- [41] Adam Sadilek, Luyang Liu, Dung Nguyen, Methun Kamruzzaman, Stylianos Serghiou, Benjamin Rader, Alex Ingerman, Stefan Mellem, Peter Kairouz, Elaine O Nsoesie, et al. Privacy-first health research with federated learning. *NPJ digital medicine*, 4(1):1–8, 2021. **1**
- [42] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. **6**
- [43] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. *Advances in neural information processing systems*, 25, 2012. **5**
- [44] 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. **1, 3, 4, 6, 7, 8, 11, 12, 13**
- [45] Stacey Truex, Nathalie Baracaldo, Ali Anwar, Thomas Steinke, Heiko Ludwig, Rui Zhang, and Yi Zhou. A hybrid approach to privacy-preserving federated learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pages 1–11, 2019. **3**
- [46] Dave Van Veen, Ajil Jalal, Mahdi Soltanolkotabi, Eric Price, Sriram Vishwanath, and Alexandros G Dimakis. Compressed sensing with deep image prior and learned regularization. *arXiv preprint arXiv:1806.06438*, 2018. **4**
- [47] 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 *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pages 2512–2520. IEEE, 2019. **2**
- [48] Wenqi Wei, Ling Liu, Yanzhao Wut, Gong Su, and Arun Iyengar. Gradient-leakage resilient federated learning. In *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)*, pages 797–807. IEEE, 2021. **1, 3, 4, 6, 7, 11**
- [49] Guowen Xu, Hongwei Li, Sen Liu, Kan Yang, and Xiaodong Lin. Verifynet: Secure and verifiable federated learning. *IEEE Transactions on Information Forensics and Security*, 15:911–926, 2019. **3**
- [50] Wensi Yang, Yuhang Zhang, Kejiang Ye, Li Li, and Cheng-Zhong Xu. Ffd: A federated learning based method for credit card fraud detection. In *International conference on big data*, pages 18–32. Springer, 2019. **1**
- [51] 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. **1, 2, 3, 4, 5, 6, 13**
- [52] 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. **6**
- [53] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Improved Deep Leakage from Gradients Implementation. <https://github.com/PatrickZH/Improved-Deep-Leakage-from-Gradients>. Accessed: 2021-11-09. **6**

- [54] Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. idlg: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610*, 2020. **1, 2, 3, 4, 5, 6**
- [55] Ligeng Zhu and Song Han. Deep Leakage from Gradients Implementation. <https://github.com/mit-han-lab/dlg>. Accessed: 2021-11-09. **6**
- [56] Ligeng Zhu and Song Han. Deep leakage from gradients. In *Federated learning*, pages 17–31. Springer, 2020. **1, 2, 3, 4, 5, 6, 7, 8, 11, 13**

## A. Additional Reconstruction Samples

Due to page limit, we only include the reconstruction results under the Soteria [44] defense in our main paper (Figure 5) for additional visualization samples on the ImageNet dataset. Here we present the full results under all 4 considered defenses (i.e., additive noise [44, 56] with  $\sigma = 0.1$ , gradient clipping [14, 48] with  $S = 4$ , gradient sparsification [56] with a pruning rate of 90%, and Soteria [44] with a pruning rate of 80%) in Figure 10. We observe that our method is able to reconstruct high-quality images from gradients in all these considered cases regardless of the type of defense.

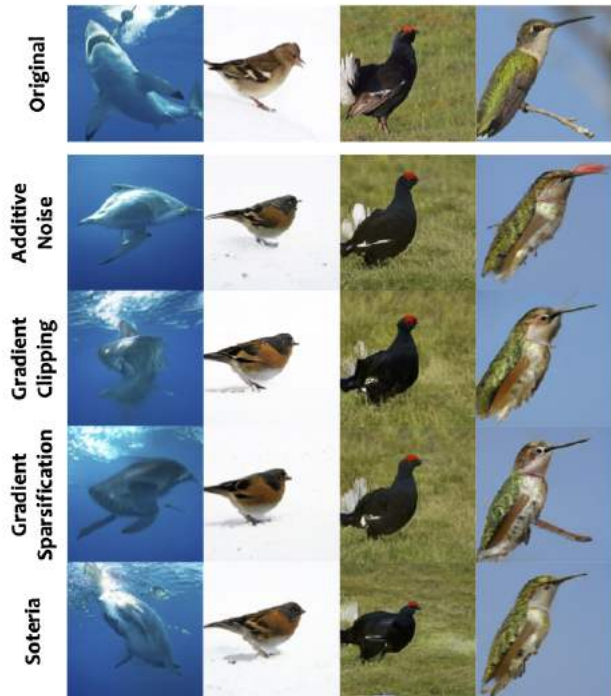


Figure 10. Reconstruction results under various defenses on the ImageNet dataset: (first row) original images and (the rest of rows) their reconstructions by GGL under various defenses.



## B. Implementation Details

**Optimization Configuration.** We use the following configuration for the explored optimizers: (1) *Adam*: initial learning rate  $lr = 0.1$ ,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ . On the CelebA dataset, we use a step learning rate decay at step 937, 1562, and 2189, by a factor of  $\gamma = 0.1$ . On the ImageNet dataset, the learning rate is linearly warmed-up from 0 during the first 125 iterations and gradually reduced to 0 in the last 625 iterations using cosine decay; (2) *BO*: We use the *TurBO-1* algorithm [10] with 256 initial points, batch size = 10, lower bound =  $-2$ , upper bound = 2, and automatic relevance determination (ARD) kernel for the Gaussian process; and (3) *CMA-ES*: we use random initialization with batch size = 50. We set  $\lambda = 0.1$  for experiments on the CelebA dataset. On the ImageNet dataset, for algorithms that do not innately support bound constraints, we apply the *tanh* function to achieve the bound.

**GAN Configuration.** For the CelebA dataset, we train a DCGAN [40] with a latent dimension of 128 with its detailed structure presented in Figure 11. Specifically, we use the Wasserstein distance with the loss weight set to 10 for the gradient penalty [17]. The GAN model is trained for 100 epochs using Adam optimizer with a learning rate of 0.0001 and a batch size of 64. For the ImageNet dataset, we use a pre-trained BigGAN [6] with a latent dimension of 128 and output image size of  $256 \times 256$ . The output image is further rescaled to  $224 \times 224$  for computing the FL task.

Type	Kernel	Stride	Output
FC			8192
BN1D			8192
DeConv2D	$2 \times 2$	$2 \times 2$	256
BN2D			256
DeConv2D	$2 \times 2$	$2 \times 2$	128
BN2D			128
DeConv2D	$2 \times 2$	$2 \times 2$	3
(a) Generator			
Type	Kernel	Stride	Output
Conv2D	$3 \times 3$	$2 \times 2$	128
Conv2D	$3 \times 3$	$2 \times 2$	256
Conv2D	$3 \times 3$	$2 \times 2$	512
FC			1
(b) Discriminator			

Figure 11. GAN structure for the CelebA dataset.

## C. Loss Landscape Analysis

**Comparison with GAN Inversion.** In our attack, we consider the private image to be unknown and the adversary attempts to reconstruct the image from the shared gradient information using a pre-trained GAN. However, such reconstruction is constrained by the generator’s fitting abil-

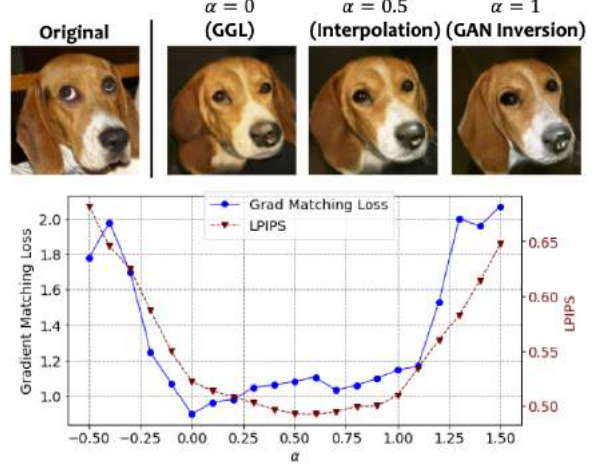


Figure 12. Comparison of image reconstructed by our method and GAN inversion.

ity. GAN inversion technique which inverts a given image to the GAN’s latent space can serve as a means for testing the upper bound of the image quality reconstructed from GAN. To evaluate, we compare the reconstructed image from gradients using our method and the inverted image using GAN inversion technique [25]. To compare the information provided by gradient information with the information provided by the original image, we further visualize the gradient matching loss and the LPIPS loss in the GAN latent space. Specifically, we plot the loss functions by interpolating between the latent vectors found by the proposed GGL ( $\mathbf{z}_1$ ) and GAN inversion ( $\mathbf{z}_2$ ):  $\mathbf{z}(\alpha) = (1-\alpha)\mathbf{z}_1 + \alpha\mathbf{z}_2$ . From the results presented in Figure 12 we observe that (1) the latent vector found by our method does yield the lowest gradient matching loss on this line; (2) compared to the gradient information, the information provided by the original image can better guide the optimization process in the GAN latent space: the latent vector found by GAN inversion produces a better image quality (lower LPIPS) than the solution found by our method; and (3) the latent vector with the lowest gradient match loss doesn’t result in the best image quality/similarity (measured by LPIPS).

**Different Defenses.** We next analyze how each defense mechanism affects the loss landscape. We extend the visualization to a 2D surface by adding a second random direction vector  $\boldsymbol{\eta}$  (normalized according to  $\mathbf{z}_2 - \mathbf{z}_1$ ):  $\mathbf{z}(\alpha, \beta) = \mathbf{z}_1 + \alpha(\mathbf{z}_2 - \mathbf{z}_1) + \beta\boldsymbol{\eta}$ . Figure 13 shows the visualized loss surface under different defense settings. We can see that additive noise and gradient sparsification do not have much impact on the geometric landscape of the gradient matching loss, whereas gradient clipping and Soteria [44] clearly deform the gradient matching loss surface, rendering it hard for the adversary to find a good reconstruction under such defenses. However, by applying the adaptive transformation at the adversary’s side, such deforma-

tion can be greatly mitigated and thereby enables the adversary to reconstruct high-quality images even with the presence of these defenses.

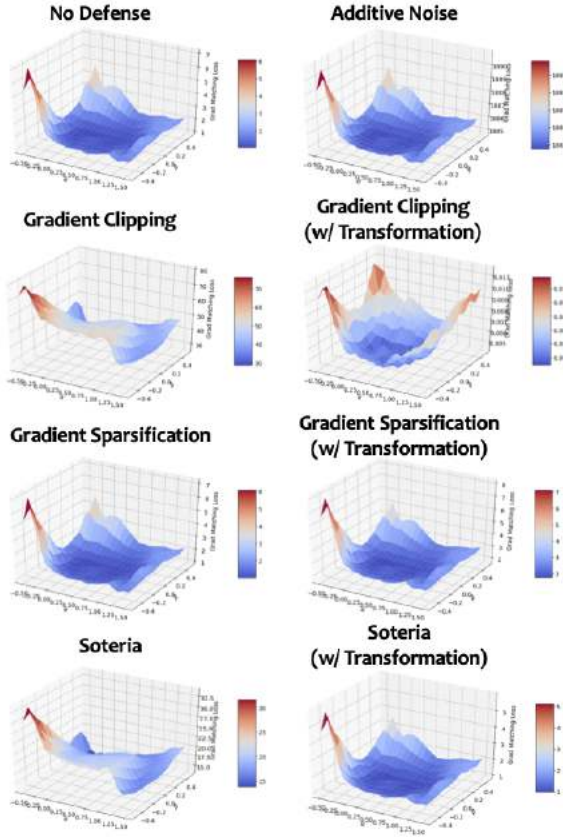


Figure 13. Visualization of observed loss landscapes under various defense settings. The bottom 3 rows compare the loss surface with (right) and without (left) applying adaptive transformation at the adversary’s side.

## D. Larger Batch Sizes or Multiple Local Steps

Recovering high-resolution batch data with multiple local steps remains a major challenge in this line of research. Most existing studies [13, 56] only work on small images ( $32 \times 32$ px) for batch size  $> 1$ . Currently, the only study that accounts for local steps  $> 1$  is IG [13], but it only works on a single ImageNet image. The only study that can work on batched full-size ImageNet images ( $224 \times 224$ px) is GI [51], which supports up to 48 images with local step = 1. However, it can only reveal limited information from partial images of the batch, and it assumes that the BatchNorm (BN) statistics (mean and std.) of the target batch is jointly provided with the gradients and only works for specially pre-trained large ResNet-50 model (larger model provides more gradient information).

Differently, we seek to investigate the privacy leakage under various defense strategies. We show that even with batch size = 1 and local step size = 1, existing methods

still failed to reconstruct the input under defenses, while our method can reveal a good amount of visual information.

To investigate the generalizability of GGL, we conducted additional experiments on batched ImageNet images ( $224 \times 224$ px) and with multiple local steps, with the results presented in Figure 14 and Figure 15, respectively. We can see that GGL can still restore a decent amount of visual information under these settings. The proposed GGL can be further strengthened with additional prior information (e.g., BN statistics).



Figure 14. Image reconstruction with batch size = 4: (1st row) original images, (2nd row) reconstructions by GGL w/o defense, and (3rd row) reconstructions by GGL w/ Soteria [44] defense.



Figure 15. Reconstruction by GGL with multiple local steps.



Figure 16. Reconstruction of *in-the-wild* images: (1st row) images from Google Images and (2nd row) their reconstructions by GGL.

## E. Recovering In-the-wild Data

We target the practical scenario where the attacker can utilize all public-accessible data as prior information to launch the attack. Thus we chose to use CelebA and ImageNet for evaluation as they are all Internet-based datasets and are easy to access as an attacker. We also used the disjoint dataset so that the images used for testing haven’t been used for GAN training. To investigate the performance of GGL under the scenario where the testing image is not from the GAN training distribution, we conducted additional experiments to recover *in-the-wild* images (i.e., arbitrary images from the search results in Google Images with appropriate cropping/resizing). From the results in Figure 16, we can see that GGL can still reveal a reasonable amount of visual information even if the testing images are not from the GAN training distribution.