Generative Image Reconstruction from Gradients

Ekanut Sotthiwata, Liangli Zhen, Chi Zhang, Zengxiang Li, and Rick Siow Mong Goh

***Abstract*—In this paper, we propose a method, Generative Image Reconstruction from Gradients (GIRG), for recovering training images from gradients in a federated learning setting, where privacy is preserved by sharing model weights and gradients rather than raw training data. Previous studies have shown the potential for revealing clients' private information or even pixel-level recovery of training images from shared gradients. However, existing methods are limited to low-resolution images and small batch sizes or require prior knowledge about the client data. GIRG utilizes a conditional generative model to reconstruct training images and their corresponding labels from the shared gradients. Unlike previous generative model-based methods, GIRG does not require prior knowledge of the training data. Furthermore, GIRG optimizes the weights of the conditional generative model to generate highly accurate "dummy" images instead of optimizing the input vectors of the generative model. Comprehensive empirical results show that GIRG is able to recover high-resolution images with large batch sizes and can even recover images from the aggregation of gradients from multiple participants. These results reveal the vulnerability of current federated learning practices and call for immediate efforts to prevent inversion attacks in gradient-sharing-based collaborative training.**

Keywords*—Inversion attack, deep leakage, data privacy, federated learning

I. Introduction

Federated learning (FL) [1] is a novel paradigm in distributed learning that offers the potential for both privacy and efficiency in training models across multiple organizations. In a centralized FL setting, a central server sends a joint model, also referred to as the collaborative model, to local participants. Upon receipt of the joint model, each participant computes local gradients using their own local dataset. These local gradients are then transferred back to the central server, where they are aggregated to update the joint model. This process is repeated multiple times, allowing for the training of an accurate model without the need for the exchange of private training data. FL enables cooperation among competitive organizations by eliminating the requirement for sharing raw data with other participants, as it is believed that gradients and models can be safely shared while preserving data privacy.

However, recent studies have demonstrated that the shared gradients from each participant and weight parameters of the joint model, represented by M, may contain sensitive information about the training data of each participant. A variety of attacks have been identified that can potentially reveal this sensitive information, including membership inference attack [2, 3], properties inference attack [4], class representative attack [5], and model inversion attack [6–8]. In this study, we focus on the model inversion attack, also known as the reconstruction attack, which aims to reconstruct the training images from the gradients shared by participants. The pioneer reconstruction attacks [6–8] adjust values of pixels of a randomly initialized image or dummy image in a direction that minimizes a specific loss function, as illustrated in Figure 1 (a). It is the difference between the shared gradients from one local participant and the dummy gradients, which are calculated by computing the gradients over the global model using the dummy image as input. If the attacker is able to minimize this distance to zero, the dummy image is transformed from the randomized image to the training image of the local participant. The limitation of these attacks is the lack of capability in reconstructing a large batch of images. The potential reasons are two fold: 1) the increasing number of parameters to be optimized along with the increase of reconstruct images and 2) the ignorance of considering relationships among neighbour pixels. For example, when the attacker reconstructs N images (32×32) on the single shared gradients, the training parameters are directly proportional to the number of reconstructed images ($N \times 32 \times 32$ parameters). These attack methods updated the pixel value without regard to the neighbour pixels. In addition, the pixel relationships are not used crucially by these attacks due to the fact that pixels are updated without regard to the neighbour pixels.

More recently, Jeon *et al.* [9] explored the use of generative models to generate dummy images, which has achieved promising performance. Specifically, they proposed the Gradient Inversion in Alternative Spaces (GIAS) method, which relies on prior knowledge of the generative model to search the optimal parameter values in the latent space deduced by the generative model instead of the ambient input space, as illustrated in Figure 1 (b). In such a way, it aims to speed up the search process and leverage the prior knowledge learned in the pre-trained generator to improve the reconstruction attack. To further improve the performance, GIAS optimizes each trained generator for each latent vector to reconstruct training images after the optimization process of the latent vectors. However, the reliance on prior knowledge of the user data distribution as a requirement for pre-training the generative model in GIAS makes it impractical when this information is unavailable.

This work is supported by the National Research Foundation Singapore under its AI Singapore Programme (Award Number: AISG2-GC-2023-007). (Corresponding author: Liangli Zhen).

E. Sotthiwat is with the National University of Singapore, Singapore 119077, Singapore and the Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore (e-mail: e0431608@u.nus.edu).

L. Zhen, C. Zhang, and R. Goh are with the Institute of High Performance Computing, Agency for Science, Technology and Research (A*STAR), Singapore 138632, Singapore (e-mail: {zhenll, zhang_chi, gohsm}@ihpc.a-star.edu.sg).

Z. Li is with the ENNEW Digital Research Institute, ENN Group, Beijing, China (e-mail: lizengxiang@enn.cn).

Additionally, the adaptation of the generative model to each individual input in GIAS to improve reconstruction attack performance is extremely resource-consuming for a large batch of images.

In this paper, we propose a novel method, named Generative Image Reconstruction from Gradients (GIRG), to reconstruct the training images from the shared gradients using a conditional generative model without requiring prior knowledge of the user's data distribution. By randomly initializing the input latent vectors of the generative model, GIRG optimizes the weight parameters of the generative model to transform the random latent vectors into the reconstructed images that can produce the gradients to align with the shared gradients. It is noteworthy that only a single generative model is employed to reconstruct a batch of images in the inversion attack scenario.

The novelty and main contributions of this work can be summarized as follows:

- A conditional generative model-based approach (GIRG) is proposed for an efficient reconstruction attack that leverages shared gradients from participants. GIRG does not require prior knowledge of the user's data distribution.
- Unlike existing generative model-based methods, such as GIAS, which search for optimal latent vectors, GIRG only needs to train a single generator and optimizes its weight parameters to reconstruct both training images and labels in a unified framework.
- For the first time, GIRG demonstrates the ability to reconstruct a batch of 128 private training images from gradients with sharp and realistic images through the training of a single conditional generative model. Furthermore, GIRG exhibits high performance even when duplicated labels are present in a batch.
- This work is the first study to successfully demonstrate reconstruction attacks on averaged gradients from all participants. The private training images of all participants can be reconstructed without knowledge of the ownership of the reconstructed images, highlighting the threat posed by reconstruction attacks in privacy-preserving FL using homomorphic encryption or multi-party computation.

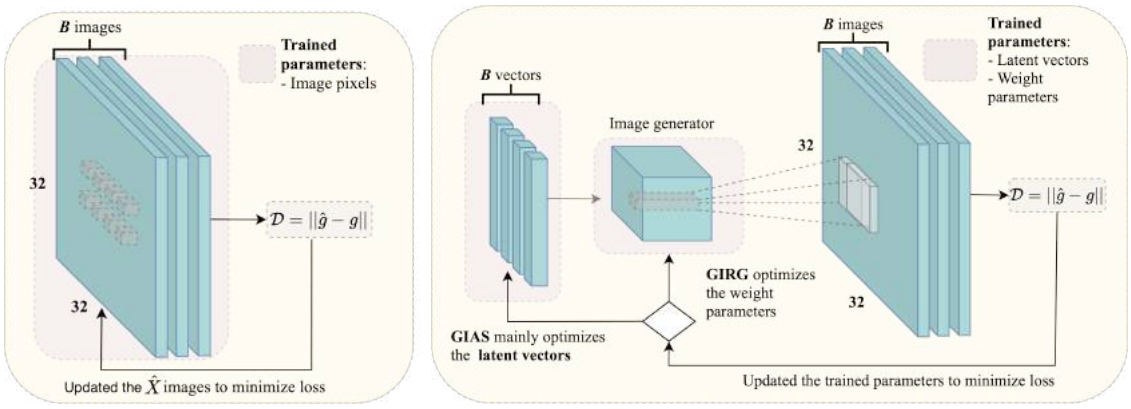
II. Related Work

In this section, the investigation into the various forms of information leakage from local participants in centralized federated learning and the methods utilized by attackers to obtain sensitive information from shared gradients is presented. The focus is on the relevant literature that addresses different attack scenarios and those capable of reconstructing training data. The overview of information leakage in centralized federated learning is discussed, followed by a review of the methodology of reconstruction attacks and an evaluation of their strengths and weaknesses. Lastly, the unique aspects of the current study are highlighted in comparison to existing studies on reconstruction attacks.

In the centralized federated learning setting, the sharing of weights or gradients among participants has been shown to lead to information leakage. To protect the privacy of participants, some may choose to encrypt their updated parameters. However, this still leaves them vulnerable to data poisoning attacks [10–13] and model poisoning attacks [14–17], as the central server cannot differentiate between legitimate and malicious updates. On the other hand, if the parameters are not encrypted, malicious actors can exploit them to obtain sensitive information. For the first types of leakage information, the membership inference attacks [2, 3, 18, 19] are able to distinguish whether the selected data is or is not trained in the centralized federated learning. This leakage is vulnerable due to the fact that the competitive company can use this information to impose on other organizations. Additionally, some studies [4, 20, 21] have shown that observing the updated gradients from participants can result in leakage of sensitive features of the training data, such as eye wear, identity, membership, gender, region, and race. On the other hand, studies using generative adversarial networks (GANs) [5, 22] have demonstrated the ability to generate images representative of the class images from all participants. However, it is important to note that while these attacks can obtain limited information about the training data, the most vulnerable form of attack is the gradient inversion attack, as it allows the attacker to reconstruct the entire training data.

The gradient inversion attack in FL aims to reconstruct the training data from participants by exploiting the shared gradients uploaded to the central server or a malicious agent. The existing gradient inversion attacks [6–9, 23] have already presented the effectiveness of each attack and the information that can be obtained from shared gradients. Pioneer studies of DLG [6] and iDLG [7] demonstrate the possibility of reconstructing training images from the shared gradients. Geiping *et al.* [8] proposed the inverting gradients to reconstruct the private images from shared gradients by using the angles as a loss function and the Adam as an optimizer and added the regularization term as total variation in the loss function. Nevertheless, the quality of inverting gradients is not efficient when the batch size is large or the image resolution is high. As a following-up, Yin *et al.*[23] improved the gradient inversion by adding fidelity and group consistency regularization to the loss function. Moreover, they presented the batch label restoration technique when the labels are not repeated in a single batch. Even though, the existing attacks proposed the new methodologies to improve a gradient inversion attack in FL, all of them [6–8, 23] reconstruct the training images by optimizing the pixel values of the dummy images directly. The limitation of these attacks is the lack of capability in reconstructing a large batch of images.

Recently, Jeon *et al.* [9] introduced the Gradient Inversion in Alternative Spaces (GIAS) method, similar to our proposed approach, which utilizes generative models to generate dummy images. The fundamental concept of GIAS is to explore solutions within the latent space as opposed to the high-dimensional ambient input space of generative models. Additionally, they implement multiple trained generative models to enhance the performance of reconstruction attacks. The experimental results in [9] demonstrate that the generative model surpasses previous gradient inversion attacks based on



(a) (b)

Fig. 1. The comparison of two types of inversion attack approaches: (a) Pixel-wise optimization-based attack approach and (b) Generative model-based attack approach. In (a), it optimizes pixel-values of dummy images directly; In (b), GIAS optimizes the latent vectors and weight parameters of B latent vector-specific generators, while our method (GIRG) optimizes the weight parameters of a single generator only.

pixel-wise techniques. Nevertheless, the accuracy of image reconstruction is compromised when the batch size surpasses 32. Moreover, reconstructing N training images in a single batch requires high memory and computational time as N generative models need to be trained using shared gradients. GIAS also necessitates a trained generator, trained on the approximate data distribution of participants or shared gradients from participants, which may not be feasible in real-world scenarios. Furthermore, the number of training parameters in GIAS increases significantly with the number of training images, hindering its scalability. Despite advancements in existing gradient inversion attacks, there is still a lack of a single attack that can accurately reconstruct all training images in large batch sizes, such as 128 images, without prior knowledge of the training data.

Our proposed method, GIRG, requires only a single generative model for the reconstruction of both the training images and their respective labels within a unified framework. Unlike GIAS, which searches for optimal latent vectors, GIRG optimizes the weight parameters of the generative model solely. This methodology leads to improved reconstruction performance, particularly in scenarios involving a large batch size or high-resolution training images.

III. The Proposed Method

Federated learning is a widely adopted approach in the field of distributed machine learning, which enables collaboration between K participants $(P_1, P_2, \ldots, P_K)$ in the training of a joint model M without exchanging their private data $(X_1, X_2, \ldots, X_K)$. Although private data is not being transferred, the previous reconstruction attacks have demonstrated the potential to reconstruct participants' private data by observing the trained gradients g. However, these reconstruction efforts may face potential failures when the batch size B increases, leading to challenges in accurately reconstructing private data. Additionally, some reconstruction attacks require knowledge of the ground truth labels y or assume no duplicated labels exist. Lastly, some attacks require the pre-trained model, which has been trained on a similar data distribution as prior knowledge to the participants, to reconstruct training data.

This study aims to develop a novel reconstruction attack that operates without the participant's prior knowledge and can accurately reconstruct images using aggregating gradients from multiple participants instead of trained gradients from one participant. To achieve this objective, we need to overcome several key challenges. It should utilize sufficient computational resources to accurately reconstruct data on large batch sizes (e.g., $B = 128$) without relying on prior knowledge, such as pre-trained models or ground truth labels. Furthermore, it must be able to precisely reconstruct labels when duplicated labels are in the training batch. If these challenges can be overcome, the methodology can be applied in real-world scenarios, raising concerns about the vulnerability of current federated learning practices.

A. Generative Image Reconstruction from Gradients

We present a novel method, called Generative Image Reconstruction from Gradients (GIRG), for reconstructing training images X from shared gradients g. An overview of our method is shown in Figure 2, from which we can see that GIRG trains a class-conditional generative model (e.g., a conditional generator G from a Large Scale Generative Adversarial Network (BigGAN) [24]) to generate a dummy image $\hat{X}$ to match the true gradients g iteratively. After many iterations, the generated dummy image $\hat{X}$ can produce the gradients aligned with the shared gradients g, and we can obtain the image $\hat{X}$ that looks as same as the training image X.

To initiate the reconstruction process, which is illustrated in Algorithm 1, we first initialized a conditional generator G with weight parameters θ and a latent vector ($Z \in R^{d_z}$ randomly, where d_z denotes the dimensionality of the latent vector). After the initial setup, G transforms the latent vector Z and the class label y into a dummy image $\hat{X} = G(\theta; Z, y)$, with dimensions $d_C \times d_W \times d_H$ and class label $y \in \mathbb{R}^N$. Subsequently,

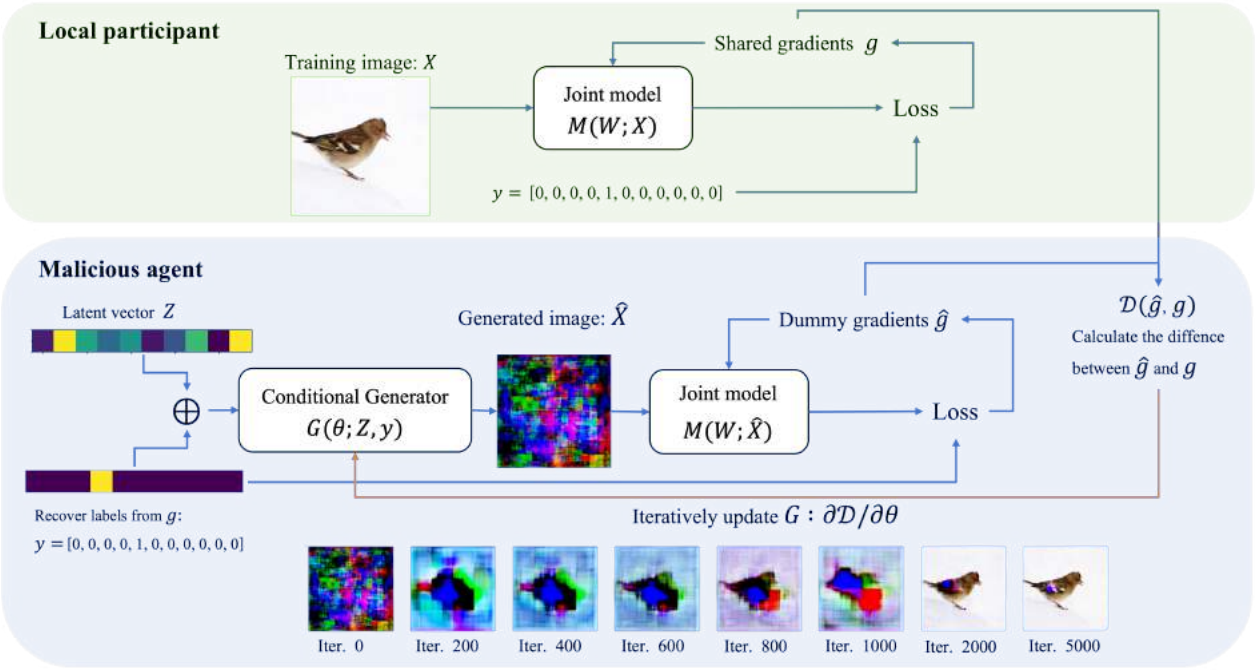


Fig. 2. The framework of our proposed GIRG. It iteratively generates dummy images that can produce the gradients aligned with the shared gradients from the collaborative participant.

forward propagation is performed on the joint model M with weight parameters W by passing the dummy image $\hat{X}$ to M, resulting in the predicted output $\hat{y} = M(W; \hat{X})$. After that, the predicted output $\hat{y}$ and the class label y are utilized to calculate the dummy gradients, represented by $\hat{g}$ as:

$$\hat{g} = \frac{\partial \mathcal{L}(M(W; \hat{X}), y)}{\partial W}, \tag{1}$$

$\mathcal{L}$ is the loss function for the classification task, and we set it as a cross-entropy function in this work.

Then, agent A computes the difference between the dummy gradients $\hat{g}$ and the shared gradients g according to the following equation:

$$\mathcal{D}(\hat{g}, g) = 1 - \frac{\hat{g}^T g}{||\hat{g}|| \cdot ||g||}, \tag{2}$$

where we reshape the shared gradients g and $\hat{g}$ into column vectors for calculation.

Next, the agent A updates the generator by minimizing the loss function $\mathcal{D}$ as:

$$\theta \leftarrow \underset{\theta}{\text{argmin}}\ \mathcal{D}(\hat{g}, g), \tag{3}$$

where $\mathcal{L}$ is the loss function for the classification task, and we set it as a cross-entropy function in this work.

In such a way, after a large number of N training iterations, the conditional generator G can retain the training images' characteristics and generate a dummy image $\hat{X}$ that closely resembles the training image X from the participant P.

Algorithm 1 Reconstructing Single Image from Shared Gradients

Input: M: joint model from the participant or the aggregation server; W: the weight parameters of the joint model; g: the public shared gradients calculated based on the training image X; G: the image generator; θ: the weight parameters of G; y the ground truth label for X;

Output: $\hat{X}$: the reconstructed image.

$Z \leftarrow \mathcal{N}(0, 1)$; $\theta \leftarrow \mathcal{N}(0, 1)$ ▷ Initialize Z and θ randomly

for j $\leftarrow$ 1 to N **do**

  $\hat{X} \leftarrow G(\theta; Z, y)$

  $\hat{g} \leftarrow \partial \mathcal{L}(M(W; \hat{X}), y)/\partial W$

  $\theta \leftarrow \underset{\theta}{\text{argmin}}\ \mathcal{D}(\hat{g}, g)$

end for

return $\hat{X}$

B. Reconstruct Multiple Images on Shared Gradients With GIRG

To extend the capabilities of single-image reconstruction, a generator incorporating multiple latent vectors is utilized to reconstruct multiple images. The number of initialized latent vectors is adjusted from one to a batch of B vectors, without affecting the number of training parameters in GIRG. Subsequently, the B latent vectors are input to the generator G to generate B images, which are used to calculate the dummy gradients $\hat{g}$. The generator G is trained to minimize the difference $\mathcal{L}$ between the dummy $\hat{g}$ and shared gradients

g. Therefore, a modification is made at Line 1 in Algorithm 1 to the following line, represented by

$$Z \leftarrow [\mathcal{N}(0,1) \text{ for } i \text{ in range}(B)]. \tag{4}$$

C. Reconstruct Multiple Images on the Aggregation of Gradients – Averaged Gradients Over All Participants

Traditionally, it has been argued that if the aggregating server A in FL is able to calculate the averaged gradients from all local participants ($\bar{g} = \sum_{l=1}^{k} g_l/k$) without having access to the shared gradients from all participants $[g_1, g_2, \ldots, g_k]$, server A has the knowledge about the gradients $\bar{g}$ only and does not know about the actual value of each shared gradients from participants. Nevertheless, there is increasing concern that the $\bar{g}$ can be used to refer to the training data in distributed participants $\{X_1, X_2, \ldots, X_k\}$. This concern may be explained by the fact that the $\bar{g}$ is equal to the gradients that are computed by data from all participants ($\widetilde{X} = X_1 \cup X_2 \cdots \cup X_k$) when model M does not have the batch normalization as

$$\bar{g} = \sum_{h=1}^{k} \frac{g}{k} = \sum_{l \in \widetilde{X}}^{B \times k} \frac{g_l}{B \times k} \tag{5}$$

and

$$g = \sum_{l=1}^{B} \frac{g_l}{B}. \tag{6}$$

We demonstrate that GIRG is able to reconstruct the training images from all participants, even without knowing their ownership, by using averaged gradients aggregated from each participant. Although we do not know the ownership, this data leakage provides an exciting finding to reconstruct the training images on the averaged gradients instead of the shared gradients from each participant. Prior to the attack, we assume that the label information and number of training images (R = $B \times K$) from all participants are known to the attacker. For inversion attack, we initialize a batch of R latent vectors and input them to generator G. Then, we train the generator G using a slightly modified training objective that minimizes the difference between the dummy gradients $\hat{g}$ and the averaged gradients $\bar{g}$, instead of the shared gradients from each participant g_l. After a large number of N training iterations, the batch of reconstructed images $\hat{X}$ looks like the training images from all participants $\bar{X}$, indicating the possibility of using either the shared gradients g_k from each participant or the averaged gradients from every participant $\bar{g}$ is able to reconstruct the training images X.

D. Label Recovery When Duplicated Labels Exist

Previous studies have demonstrated that prior knowledge of the labels of the images before reconstructing training images simplifies the optimization problem (Equation (3)). It should be noted that existing techniques, such as those described in [7, 23], have limitations in accurately reconstructing batch labels with duplicates.

In this study, we propose a novel method for reconstructing the batch labels based on the reconstruction of training images presented in a previous subsection. The key difference between the two methods lies in the utilization of gradients at different layers of the joint model. Instead of minimizing the distance between the dummy gradients and trained gradients at every hidden layer, our method minimizes the distance between the dummy gradients and trained gradients at the last β hidden layers only. The finding from our studies suggests that these layers, which are close to the output layers, contain crucial information regarding the ground truth labels (y). Let W be the weight parameters of the joint model M, β the number of layers utilized for label reconstruction, g_β the gradients at the last β hidden layers from participant P, and B the batch size. Our methodology is initiated with the random generation of latent vectors Z, an untrained generator G, and randomized or dummy labels $\hat{y}$. After the initialized phase, the dummy images $\hat{X}$ are then generated by passing Z and $\hat{y}$ through the generator G. These images are input into the joint model M to obtain the predicted outputs, which are then used in backpropagation to obtain the dummy gradients $\hat{g}$. As previously mentioned, the loss term for label reconstruction is calculated only on the dummy and trained gradients at the last β layers. Subsequently, G and $\hat{y}$ are optimized to minimize the following equation to reconstruct the ground truth labels from the participants:

$$\theta, \hat{y} = \underset{\theta, \hat{y}}{\operatorname{argmin}} \, \mathcal{D}(\hat{g}_\beta, g_\beta). \tag{7}$$

Following multiple iterations of optimization, our proposed method is capable of reconstructing the batch labels from the last β hidden layers. While it is not possible to guarantee 100% accuracy, as demonstrated by previous studies such as [7, 23], our method outperforms prior attacks in terms of accuracy when duplicated labels are present in the batch. This represents a significant improvement over prior methods and enables the reconstruction of training images from local participants without prior knowledge of the labels. Our method thereby enables the reconstruction of training images from local participants without any prior knowledge of the actual labels.

IV. Experimental Study

In this section, we compare the performance of GIRG in reconstructing training images with other state-of-the-art algorithms, including DLG [6], iDLG [7], inverting gradients (IG) [8], and GIAS [9]. The experiments are conducted on two popular datasets: CIFAR-10 $(32 \times 32\ px)$ [25], ImageNet $(224 \times 224\ px)$ [26] and the ChestX-ray dataset from the National Institutes of Health [27] using ResNet architectures. In the experiments, by following the setting in [6], we replace the ReLU activation function with Sigmoid function in ResNet architectures. The Adam optimization algorithm [28]is used for all experiments. Moreover, in GIAS, a pre-trained generator, which is trained on a similar distribution as participants, is used for the inversion attack.

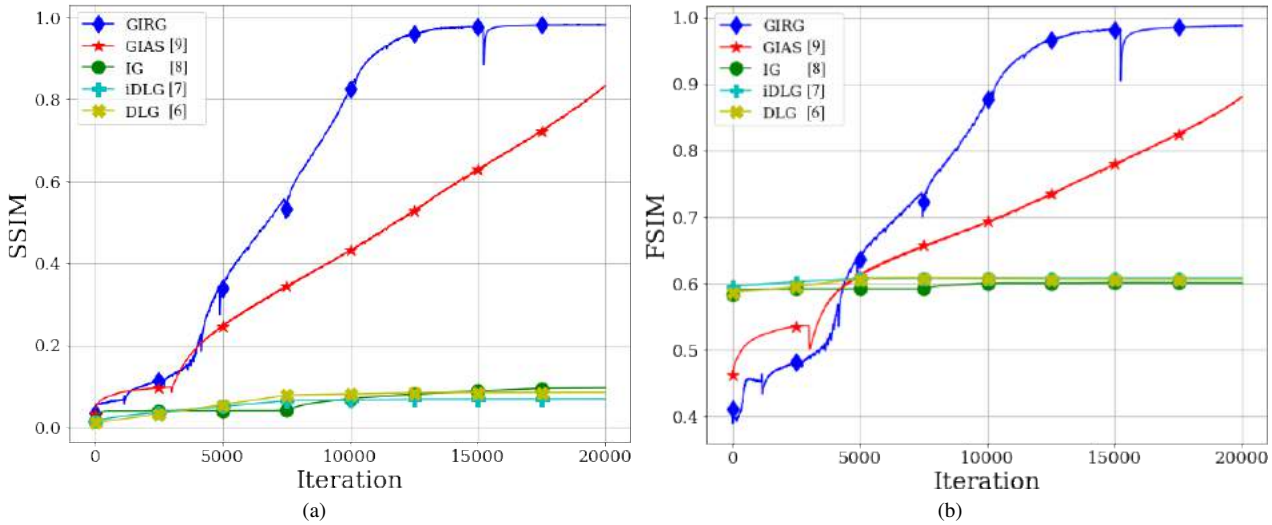
The performance of the reconstruction attacks is evaluated in terms of convergence speed and image quality by using the structural similarity index measure (SSIM) [29] and a feature similarity index for image quality (FSIM) [30]. The

TABLE I. THE MEAN AND STANDARD DEVIATION OF SSIM AND FSIM ON THE CIFAR-10 DATASET OVER RESNET-18 NETWORKS WITH DIFFERENT BATCH SIZE (BS).

	BS = 8		BS = 64		BS = 128	
	SSIM	FSIM	SSIM	FSIM	SSIM	FSIM
GIRG	0.983 ± 0.016	0.987 ± 0.006	**0.980 ± 0.006**	**0.979 ± 0.007**	**0.979 ± 0.009**	**0.979 ± 0.010**
GIAS [9]	**0.999 ± 0.000**	**0.999 ± 0.000**	0.958 ± 0.012	0.975 ± 0.009	0.906 ± 0.044	0.929 ± 0.027
IG [8],	0.403 ± 0.027	0.623 ± 0.006	0.129 ± 0.007	0.588 ± 0.007	0.088 ± 0.002	0.608 ± 0.004
iDLG [7]	0.423 ± 0.022	0.639 ± 0.009	0.138 ± 0.010	0.588 ± 0.007	0.071 ± 0.001	0.619 ± 0.007
DLG [6]	0.322 ± 0.020	0.610 ± 0.013	0.150 ± 0.009	0.584 ± 0.005	0.085 ± 0.007	0.610 ± 0.004

TABLE II. THE MEAN AND STANDARD DEVIATION OF SSIM AND FSIM ON THE CIFAR-10 DATASET OVER RESNET-20 NETWORKS WITH BS = 16.

	GIRG	GIAS [9]	IG [8]	iDLG [7]	DLG [6]
SSIM	**0.945 ± 0.038**	0.657 ± 0.060	0.055 ± 0.004	0.074 ± 0.005	0.069 ± 0.006
FSIM	**0.949 ± 0.025**	0.8 ± 0.03	0.572 ± 0.011	0.573 ± 0.012	0.574 ± 0.014



(a) (b)

Fig. 3. Plotting the mean of SSIM and FSIM from GIRG, GIAS, IG, iDLG, and DLG on the CIFAR-10 dataset over ResNet-18 networks with BS = 128.

experimental results are divided into four parts, focusing on the reconstruction attack on a single image, multiple images, high-resolution images, and the reconstruction attacks on averaged gradients.

A. Reconstruct Multiple Image Reconstruction on Shared Gradients.

To illustrate the efficiency of our method, firstly, we perform five experiments on the CIFAR-10 validation dataset over the ResNet-18 architecture. We evaluate the image quality of the reconstructed images by calculating the mean and standard deviation of SSIM and FSIM between the ground-truth images and the reconstructed images. A mean score close to 1 for SSIM and FSIM indicates high similarity between the reconstructed images and ground-truth images. Table I reports the experimental results on ResNet-18 with different batch sizes. We find that traditional methods such as DLG, iDLG, and IG perform poorly when the batch size is 64 and 128, as evidenced by mean scores of SSIM less than 0.2. These results demonstrate that those attacks cannot reconstruct images successfully when the participant trained a joint model with large batch size. On the other hand, GIAS and GIRG, which incorporate generators in their reconstruction process, can perform well. For instance, their mean score of SSIM and FSIM is greater than 0.9. From the results, we can see that GIRG outperforms previous attacks when the batch size is 64 and 128. However, our method's performance is slightly less optimal than that of GIAS when the batch size is 8. This may be due to the fact that GIAS employs pre-trained generators, which can leverage prior knowledge of the data distribution to improve reconstruction accuracy, especially when the batch size is small.

In contrast, the training process for GIAS becomes complicated when the batch size is large due to the increase in parameters to be optimized. However, increasing batch size in GIRG does not increase the optimization parameters, thus no significant impact on the image quality of the reconstructed images using GIRG. These results highlight the effectiveness of our proposed method, GIRG.

*1) **Convergence of Different Attack Methods**:* Previous attack methodologies have demonstrated that the reconstruction attack on the large batch size required more attack iterations to successfully reconstruct private images. The possible explanation is that the training parameters in their methodologies

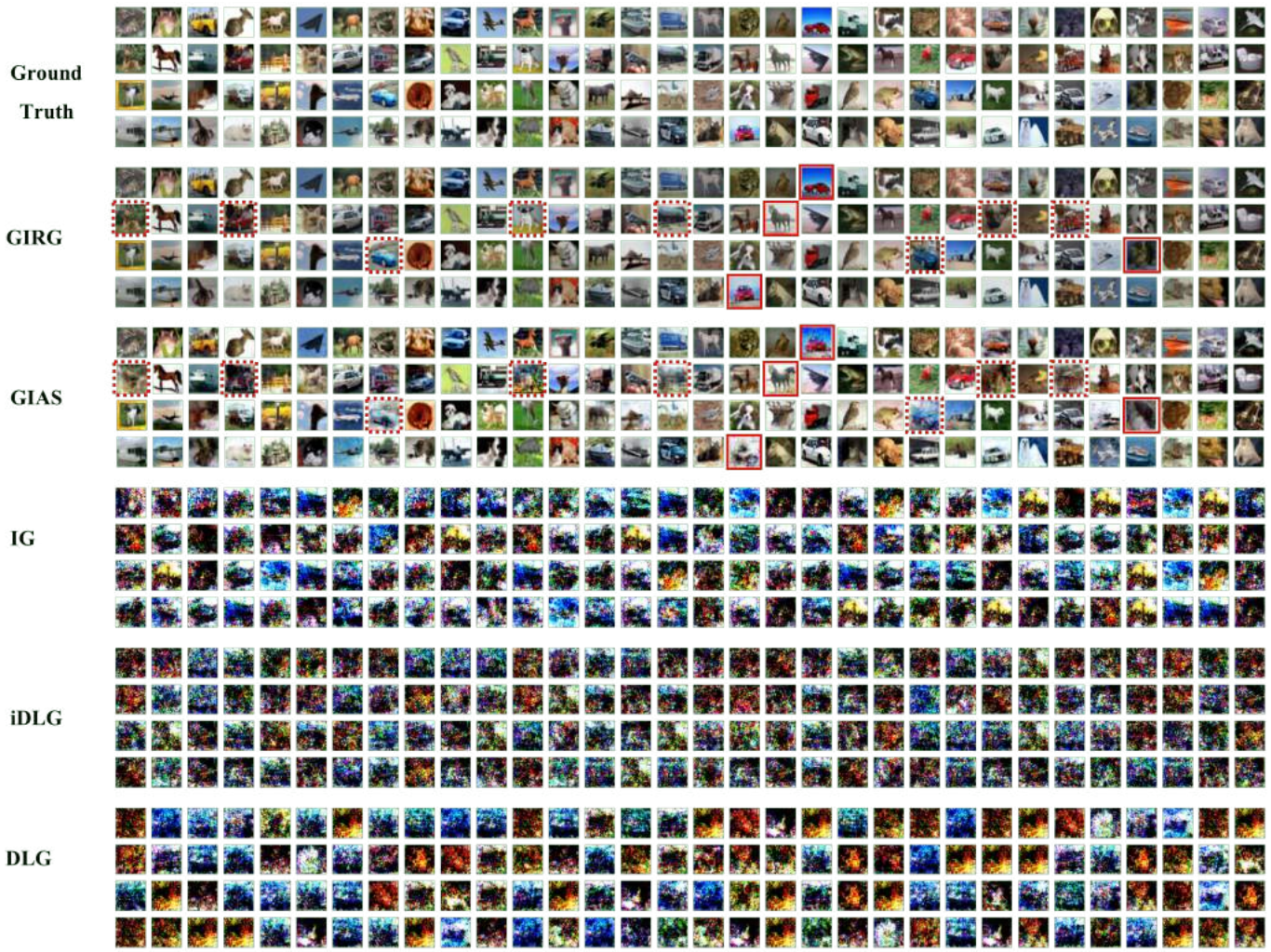


Fig. 4. A batch of 128 reconstructed images obtained by GIRG, GIAS, IG, iDLG, and DLG on the CIFAR-10 dataset over the ResNet-18 network (The order of reconstructed images is realigned for observing the quality of reconstructed images).

increased linearly with batch size. For example, the pixels of dummy images in DLG, iDLG, and inverting gradients are training parameters. Although GIAS performs better than DLG, iDLG, and inverting gradients, the training parameters of GIAS, which are the parameters of generators times the number of images, are still increased significantly with batch size.

To evaluate the convergence speed, we perform attacks on the CIFAR-10 dataset with ResNet-18 and measure SSIM and FSIM on every attack iteration when the batch size is 128. The mean and standard deviation of SSIM and FSIM of multiple attacks are calculated and plotted in Figure 3, from which we can see that GIRG outperforms previous attacks and is able to converge in 20,000 iterations. In contrast, GIAS, which performs better than DLG, iDLG, and inverting gradients, still needs to train multiple generators with more attack iterations to improve the image quality. Furthermore, we observe a significant decrease in GIRG around the 15,000th iteration. One potential explanation for this phenomenon is the existence of an alternative set of weights for the generator that produces generated images distinct from the ground-truth images, despite the dummy gradients closely resembling the ground-truth gradients. To illustrate, the generated images may share similarities in semantics with the ground-truth images, yet exhibit variations in their stylistic attributes and result in lower SSIM and FSIM scores.

After examining the convergence speed, we also demonstrate 128 reconstructed images from all attacks in Figure 4. The reconstructed images from GIAS and GIRG are reordered to match the order with ground truth images for a detailed explanation. From the figure, we can observe that the previous attacks without a generator are not able to accurately reconstruct the training images from participants. The potential reason is that the pixel-wise attacks, such as DLG and iDLG, are iteratively adjusting individual pixels within synthetic images. They suffer from a significant increase in the total

number of trainable parameters as the batch size is augmented and result in a low performance in inversion attack. Comparing the performance between GIAS and GIRG, we match a set of unsuccessfully reconstructed images from GIAS with the successfully reconstructed image from GIRG and draw the red boxes around the images. This figure has two styles of red boxes: 1) the red dash boxes are drawn to demonstrate the images that are not able to reconstruct accurately, and 2) the straight red boxes, which are the most exciting finding in this experiment, are drawn to demonstrate the impact of using the trained generator for reconstruction attacks in GIAS. For example, as seen from the red car with the blue sky reconstructed image in the first row of GIAS, the red car in this image is from the bottom row of GIRG, and the blue sky is from the first row of GIRG. For the following two examples, the horse image in GAIS has two heads, and the cat is transformed into a black and white cat. A plausible explanation is that the trained generator tried to reconstruct the images based on training data distribution and parallelly minimize the difference between dummy gradients and trained gradients. In summary, these results suggest that using GIRG is able to reconstruct a large batch size accurately without generating new images from a pre-trained generator like GIAS.

*2) **Scalability on Large Batch Size**:* To investigate the scalability of reconstruction attacks on large batch size, we conduct experiments to compare the execution time between GIRG, GIAS, IG, iDLG, and iDLG when batch sizes varied from 8, 64, and 128. Fig 5 presents the execution time of attacks when varied the batch size from 8 to 128. As shown in Fig 5, GIAS has a higher execution time for reconstructing training images than GIRG. Moreover, the results appear to confirm that the gap between the execution time of GIAS and GIRG keeps increasing along with the batch size. Please note that we do not include the time cost of training GIAS in this figure. These results demonstrate that training multiple generators require high execution times for reconstructing training images from shared gradients. As seen in Fig. 5, the execution time of the pixel-wise attack is shorter than that of the generator attacks (GIRG and GIAS). However, we also need to note that DLG and iDLG are unable to reconstruct the image accurately. The experimental results in this subsection indicate that using a single BigGAN generative model from GIRG is promising for inversion attack in terms of the image quality in the reconstructed images, convergence speed, and scalability.

*3) **Performance on the ResNet-20 architecture**:* In the following subsection, we evaluate the performance of peer methods and GIRG on the CIFAR-10 dataset using a modified ResNet-20 with BS = 16. This architecture is more practical for evaluating reconstruction attacks since it is deeper and has fewer training parameters than ResNet-18, which is used in the previous section. Table II compares the performance of the reconstruction attacks using the mean and standard deviation of SSIM and FSIM, and the experiments are conducted in five runs. The results show that attacks using the generator (GIRG and GIAS) significantly outperform those based on optimizing pixel values directly (i.e., DLG, iDLG, and IG). However, no significant difference is found among the reconstruction

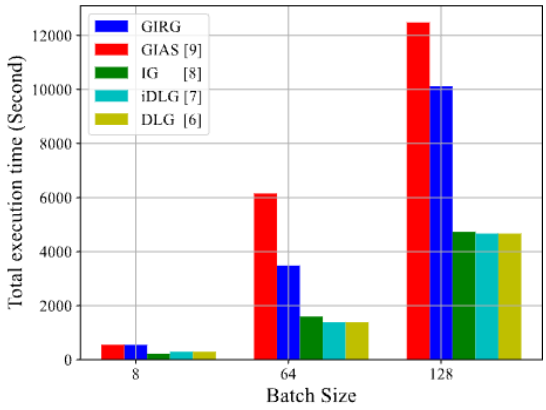


Fig. 5. Comparison of the execution time cost of GIRG, GIAS, IG, iDLG, and DLG when conducting an attack on the CIFAR-10 dataset over ResNet-18.

Fig. 6. A batch of the reconstructed images obtained by GIRG on the ImageNet over ResNet-18.

attacks using pixel values. The results, as shown in Table II, indicate that the GIRG is still able to accurately reconstruct the training image by seeing the mean score of SSIM and FSIM is greater than 0.9. In contrast, the performance of GIAS drops significantly when the training images are trained on ResNet-20 instead of ResNet-18. The results in this section indicate that the proposed method (GIRG) can handle varying batch sizes and deliver high performance even with deep model architectures and limited training parameters.

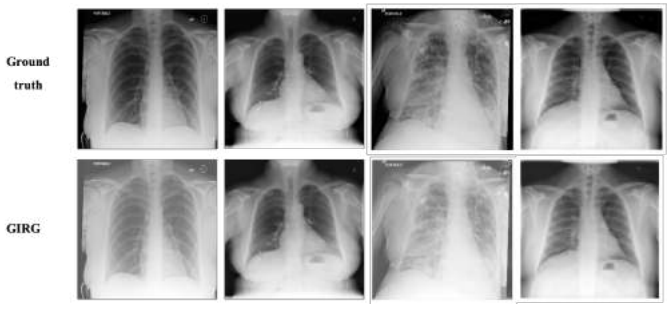


Fig. 7. A batch of the reconstructed images obtained by GIRG on the Chest X-ray dataset over ResNet-18.

B. Reconstruct High-Resolution Images

In the previous section, we have already demonstrated a reconstruction attack on low-resolution images in the CIFAR-10 dataset. To confirm the effectiveness of GIRG on the high-resolution images, we conduct attacks on ImageNet and the ChestX-ray dataset, which have 256×256 pixels, over ResNet-18 when batch size is one. The generator in this experiment is a randomly initialized BigGAN generator, and the maximal attack iteration is 25,000. Hence, we used GIRG to reconstruct a single image from the ImageNet and demonstrated the reconstructed images in Figure 6 and 7. From this figure, GIRG is able to reconstruct the characteristic of training images from participant servers without prior knowledge about their training data. Interestingly, our experimental results demonstrate that the localization of high-resolution images is not changed like in previous study [8]. A possible explanation for this might be that we transform the activation function in ResNet-18 from the ReLU function to the sigmoid function. Even though GIRG is not able to reconstruct the correct pixel value in the first and second images, the malicious agent has already obtained the characteristics of training images and their tones. In summary, these results show that GIRG is able to reconstruct low or high-resolution images from single shared gradients with a high success attack rate.

C. Reconstruct Multiple Images from an Averaged Gradients from All Participants

TABLE III. MEAN AND STANDARD DEVIATION OF SSIM AND FSIM ON THE CIFAR-10 DATASET USING RESNET-18 NETWORKS WHEN BS = 8 WITH AVERAGED GRADIENTS AND VARYING NUMBER OF PARTICIPANTS (#P).

	#P = 8		# P = 16	
	SSIM	FSIM	SSIM	FSIM
GIRG	**0.858 ± 0.047**	**0.899 ± 0.028**	**0.76 ± 0.058**	**0.853 ± 0.027**
GIAS	0.752 ± 0.105	0.843 ± 0.052	0.502 ± 0.064	0.743 ± 0.019

As mentioned in the previous section, in order to protect their training images, participants prefer to encrypt their trained gradients before uploading to the central server. However, even if participants encrypt their individual gradients, the central server can still reconstruct the training images from all participants without knowing who the images belong to by observing averaged gradients instead of individual gradients from participants. In this study, we investigate the effectiveness of existing reconstruction attacks and GIRG on this scenario to observe the quality of the reconstructed image from $\bar{g}$. Specifically, we assume that there are four participants, and each participant has eight images randomly selected from the CIFAR-10 dataset. After each participant uploaded their encrypted gradients using MPC for privacy protection, the central server averaged the encrypted gradients from four participants without knowing the actual value of gradients from each participant. To reconstruct all training images, the malicious agent uses existing attacks and GIRG to acquire the reconstructed images, as shown in Figure 8. Our findings reveal that our method could reconstruct every ground truth image from all participants, and GIAS is able to reconstruct all training images except in the red box.

In addition, it is important to evaluate the effectiveness of reconstruction attacks as the number of participants in a collaborative training scenario increases. In order to investigate the scalability of reconstruction attacks, the GIRG and GIAS algorithms are selected to perform experiments only since the pixel-wise optimization-based attack algorithms cannot reconstruct accurately. For the experiment setting, a batch of eight images from the CIFAR-10 dataset is trained on ResNet-18, with the number of participants increased to 8 and 16. Table III reports the performance of GIRG and GIAS averaged from five runs of experiments. From this table, we can see that GIRG is able to maintain the mean of SSIM and FSIM above 0.75 when the number of participants is increased to 8 or 16. However, GIAS cannot perform well when the number of participants is 16, with the mean SSIM decreasing to almost 0.5. The potential reason is that GIAS is not able to accurately reconstruct when the batch size increased. These results provide further support for the hypothesis that GIAS is not able to reconstruct well when the batch size is large.

Overall, it is apparent from this table that averaged gradients without any encryption leak the private training data even though individual gradients were encrypted by each participant. However, the quality of reconstructed images is lower compared to reconstructing 16 images on single gradients. This inconsistency may be explained by the fact that there is a batch normalization in the ResNet-18 architecture, leading to a slight difference between the actual gradients $\bar{g}$ and dummy gradients$g$. Nevertheless, this finding raised concern about averaged gradients from all participants because our attack is able to steal training data from all participants without knowledge of the trained gradients of each participant. Therefore, participants must encrypt their trained gradients and the averaged gradients to protect their training images from GIRG.

In summary, the experimental results demonstrate the importance of evaluating the effectiveness of reconstruction attacks as the number of participants in collaborative training increases. The findings suggest that GIRG is a more scalable algorithm than GIAS, and that encryption is essential to protect private training data in federated learning.

D. Reconstruct Multiple Images from Shared Gradients without Prior Label Knowledge

To assess the impact of prior labels, we conduct an experiment comparing reconstructed images when prior labels were either known or unknown. As shown in Fig 9, when when the reconstructed labels from shared gradients are not the same as the original labels, our approach is unsuccessful in reconstructing the images. The reason behind this is that the labels significantly impact the direction of dummy gradients $\hat{g}_h$. Therefore, this experiment demonstrated the importance of an accurate reconstructed label approach.

We further compare our approach to existing reconstructed label approaches after evaluating the effect of prior labels. This experiment is conducted for five runs on the CIFAR-10 dataset using ResNet-18 and ResNet-18 (zhu) with a batch

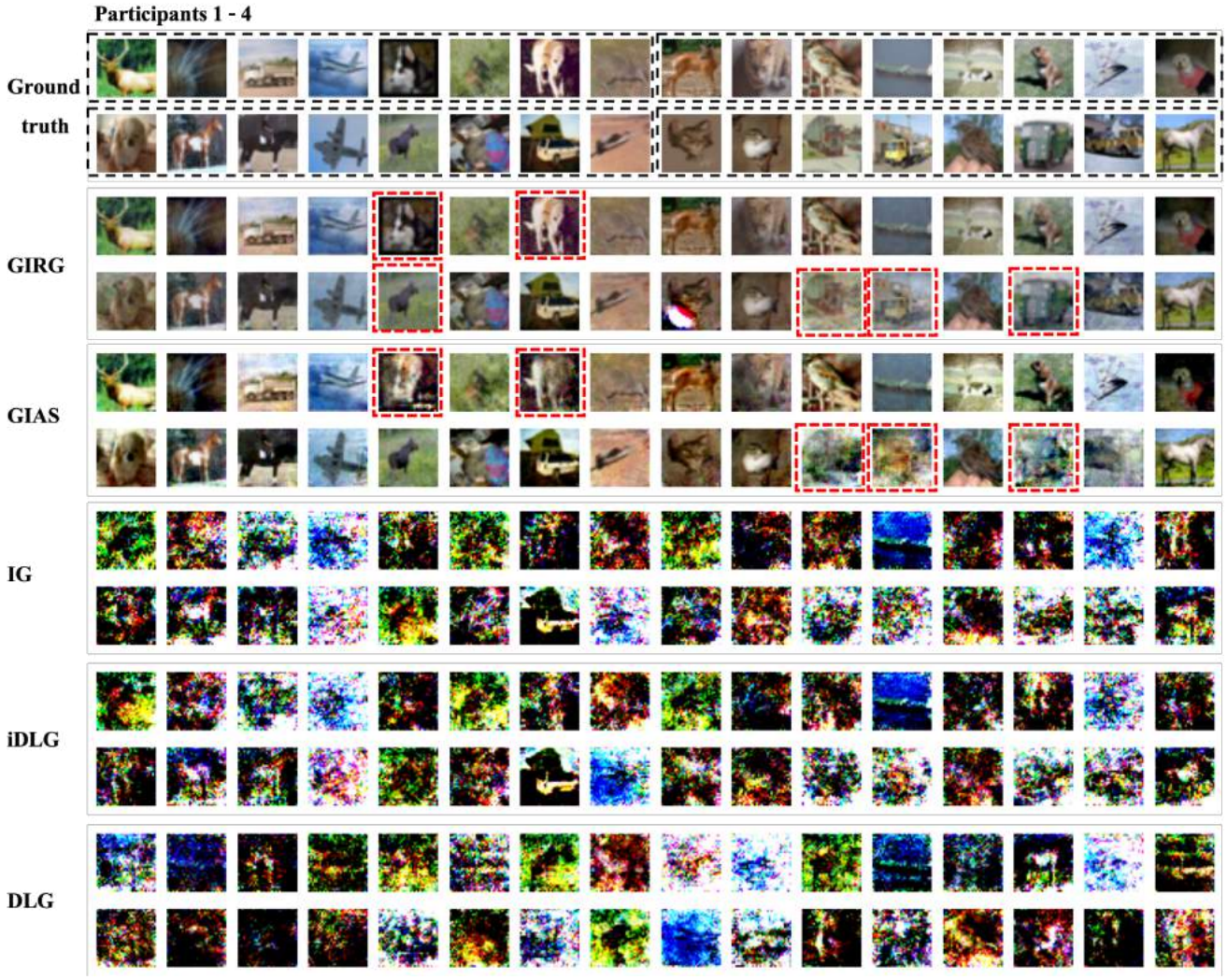
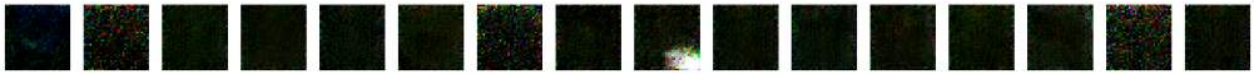


Fig. 8. A batch of 32 reconstructed images obtained by using averaged gradients, from GIRG, GIAS, Inverting Gradients, iDLG, and DLG on the CIFAR-10 dataset over ResNet-18 (The order of reconstructed images is realigned for observing the quality of reconstructed images).

(a) 16 Training images from one participant

(b) Reconstructed images when accuracy of reconstructed labels is equal 100%



(c) Reconstructed images when accuracy of reconstructed labels is 95 %

Fig. 9. A batch of 16 reconstructed images by GIRG on CIFAR-10 over ResNet-18 without knowing the ground truth of image labels.

size of 16. To evaluate the impact of the duplicated label, we chose images from only 5 classes. In each experiment, we restart DLG and our approach five runs. For GIRG, we set the β to 2 for ResNet-18 and all layers for ResNet-18 (Zhu). We assess the quality of reconstructed labels based on accuracy and the number of experiments that achieved 100% accuracy in reconstructing the labels. The results are presented in Table IV, from which we can see that our method and DLG are both able to reconstruct the labels accurately when the model architecture is modified as ResNet-18 (Zhu). Nevertheless, our method significantly outperforms peer methods when the model is not modified as ResNet-18 in terms of label reconstruction accuracy. The reason is that every gradient from the modified network is significant in improving the success rate of label reconstruction. For iDLG, it does not perform well because the batch size exceeds the number of labels in the CIFAR-10 dataset.

These results suggest that the prior label information is critical for reducing the search space and impacting the success rates of reconstruction attacks. Our method can perform well in reconstructing the labels from shared gradients.

TABLE IV. ACCURACY (%) AND NUMBER OF COMPLETE LABEL RECONSTRUCTIONS WITH 100% PRECISION (# CLR) ON CIFAR-10 USING RESNET-18 WITH A BATCH SIZE OF 32 IN 25 RUNS.

	ResNet-18		ResNet-18 (Zhu)	
	Accuracy	# CLR	Accuracy	# CLR
GIRG	**92.25**	**3**	**100**	**25**
iDLG [15]	66.25	0	63.75	0
DLG [14]	82.25	0	**100**	**25**

V. DISCUSSION

We propose a reconstruction attack that is capable of reconstructing a large batch of training images (BS $= 128$) from the shared gradients of local participants without prior knowledge, as well as reconstructing all training images using averaged gradients from all participants. The important finding is that GIRG can reconstruct every image from local participants without knowing ownership of images, as it uses averaged gradients instead of shared gradients from local participants. This finding demonstrates that while the shared gradients are protected by MPC, averaged gradients are not, which allows a malicious agent to reconstruct every training image from all participants who joined the training. Furthermore, we find that label knowledge is crucial for reconstruction attacks, as demonstrated in our experimental study. When the label accuracy is less than 100%, the reconstruction attack cannot reconstruct the corresponding image accurately. Therefore, we propose a new method for reconstructing the labels from shared gradients, which can achieve 100% label accuracy, when the attack is restarted multiple times in our experiments.

The vulnerability of federated learning to reconstruction attacks has raised significant concerns for privacy and security. While reconstruction attacks have been demonstrated on modified ResNet models, it is feasible to conduct them in the real world. Specifically, in the mobile application, most users or participants do not know which model is being used for the recommendation system, so these types of attacks are able to reconstruct the images without the awareness of users. As a result, central servers may claim that they use the slightly modified novel models to do a recommendation system and they are able to reconstruct data by using GIRG. This highlights the fact that encryption of gradients is crucial for protecting the training data and ensuring the success of federated learning in real-world scenarios.

In this study, we make the assumption that local participants are required to upload gradients after each iteration. However, in certain real-world applications, participants may locally update both their weights and gradients during each iteration, only transmitting the gradients to the central server at the end of each epoch, which spans multiple iterations. In such cases, our proposed method may not yield the desired effectiveness. Additionally, an examination of Table IV indicates that the accuracy of label reconstruction using GIRG does not consistently achieve 100%, potentially resulting in a significant reduction in the success rate of the inversion attack.

In our future research, we plan to investigate the inversion attack using diffusion models [31] and its application in scenarios where participants can locally update both their weights and gradients during each iteration, subsequently transmitting the gradients to the central server after each epoch. Furthermore, we will explore alternative and more effective approaches to label reconstruction.

VI. CONCLUSION

In this paper, we proposed Generative Image Reconstruction from Gradients (GIRG) as a novel approach for reconstructing training images and labels from trained gradients. GIRG has been shown to effectively reconstruct high-resolution images, large batch sizes of images, and multiple images using a single randomly initialized generator, without requiring prior knowledge such as batch normalization or approximate data distribution. Comparing GIRG with current state-of-the-art reconstruction attacks reveals that GIRG performs comparably with GIAS when the batch size is small, but outperforms GIAS with larger batch sizes. Additionally, GIRG converges faster and uses fewer training times on large batch sizes. This study also provides evidence that the averaged gradients from all participants can be used to recover the training images without knowing the owner of the images, highlighting the sensitivity of trained gradients as private information that must be protected to safeguard training data. Overall, GIRG represents a promising avenue for image reconstruction that could have important implications for improving the privacy and security of machine learning applications.

REFERENCES

[1] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *CoRR*, vol. abs/1610.05492, 2016. [Online]. Available: https://arxiv.org/pdf/1610.05492

[2] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE, 2017, pp. 3–18.

[3] M. Nasr, R. Shokri, and A. 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*. IEEE, 2019, pp. 739–753.

[4] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE, 2019, pp. 691–706.

[5] B. Hitaj, G. Ateniese, and F. 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*, 2017, pp. 603–618.

[6] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proceedings of the Advances in Neural Information Processing Systems*, 2019, pp. 14 774–14 784.

[7] B. Zhao, K. R. Mopuri, and H. Bilen, "iDLG: Improved deep leakage from gradients," *CoRR*, vol. abs/2001.02610, 2020. [Online]. Available: https://arxiv.org/pdf/2001.02610

[8] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients - how easy is it to break privacy in federated learning?" in *Proceedings of the International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2020.

[9] J. Jeon, K. Lee, S. Oh, J. Ok *et al.*, "Gradient inversion with generative image prior," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 34, pp. 29 898–29 908, 2021.

[10] S. Alfeld, X. Zhu, and P. Barford, "Data poisoning attacks against autoregressive models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, 2016.

[11] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and T. Goldstein, "Poison frogs! targeted clean-label poisoning attacks on neural networks," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 31, 2018.

[12] T. Gu, B. Dolan-Gavitt, and S. Garg, "BadNets: Identifying vulnerabilities in the machine learning model supply chain," *CoRR*, vol. abs/1708.06733, 2017. [Online]. Available: https://arxiv.org/abs/1708.06733

[13] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in *International Conference on Machine Learning*, 2012.

[14] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," *Proceedings of the Advances in Neural Information Processing Systems*, vol. 30, 2017.

[15] C. Fung, C. J. Yoon, and I. Beschastnikh, "Mitigating sybils in federated learning poisoning," *CoRR*, vol. abs/1808.04866, 2018. [Online]. Available: https://arxiv.org/abs/1808.04866

[16] C. Xie, K. Huang, P.-Y. Chen, and B. Li, "Dba: Distributed backdoor attacks against federated learning," in *International Conference on Learning Representations*, 2019.

[17] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*. PMLR, 2020, pp. 2938–2948.

[18] J. Zhang, J. Zhang, J. Chen, and S. Yu, "Gan enhanced membership inference: A passive local attack in federated learning," in *Proceedings of the IEEE International Conference on Communications*. IEEE, 2020, pp. 1–6.

[19] J. Chen, J. Zhang, Y. Zhao, H. Han, K. Zhu, and B. Chen, "Beyond model-level membership privacy leakage: an adversarial approach in federated learning," in *Proceedings of the International Conference on Computer Communications and Networks*. IEEE, 2020, pp. 1–9.

[20] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1322–1333.

[21] X. Luo, Y. Wu, X. Xiao, and B. C. Ooi, "Feature inference attack on model predictions in vertical federated learning," in *Proceedings of the International Conference on Data Engineering*, 2021, pp. 181–192.

[22] Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in *Proceedings of the IEEE Conference on Computer Communications*. IEEE, 2019, pp. 2512–2520.

[23] H. Yin, A. Mallya, A. Vahdat, J. M. Alvarez, J. Kautz, and P. Molchanov, "See through gradients: Image batch recovery via gradinversion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 337–16 346.

[24] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4401–4410.

[25] A. Krizhevsky, V. Nair, and G. Hinton, "The CIFAR-10 dataset," Canadian Institute for Advanced Research (CIFAR), 2009. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

[26] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Ieee, 2009, pp. 248–255.

[27] X. Wang, Y. Peng, L. Lu, Z. Lu, M. Bagheri, and R. M. Summers, "Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2097–2106.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the International Conference on Learning Representations*, 2015.

[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[30] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011.

[31] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, W. Zhang, B. Cui, and M.-H. Yang, "Diffusion models: A comprehensive survey of methods and applications," *ACM Computing Surveys*, vol. 56, no. 4, 2024.