GRADIENT INVERSION VIA OVER-PARAMETERIZED CONVOLUTIONAL NETWORKS IN FEDERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

The main premise of federated learning is local clients could upload gradients instead of data during collaborative learning, hence preserving data privacy. But the development of gradient inversion method renders this premise under severe challenges: a third-party could still reconstruct the original training images through the uploaded gradients. While previous works are majorly conducted under relatively low-resolution images and small batch sizes, in this paper, we show that image reconstruction from complex datasets like ImageNet is still possible, even nested with large batch sizes and high resolutions. Success of the proposed method is built upon three key factors: a convolutional network to implicitly create an image prior, an over-parameterized network to guarantee the non-empty of the image generation and gradient matching, and a properly-designed architecture to create pixel intimacy. We conduct a series of practical experiments to demonstrate that the proposed algorithm can outperform SOTA algorithms and reconstruct the underlying original training images more effectively. Source code is available at: (to be released upon publication).

1 INTRODUCTION

Federated learning (FL) (Konečnỳ et al., 2015; 2016; McMahan et al., 2017) provides a distributed paradigm that allows multiple parties to learn a machine learning model in a collaborative way. The main premise of this learning scheme is to allay the concerns related to data privacy and security: users can upload their local gradients instead of the data itself. As a canonical example, hospitals are often keen to train models through such a federated learning system, especially when the medical data contain sensitive patient information.

While this paradigm might provide some safety guarantees at the first glance, a line of recent works (Zhu et al., 2019; Zhao et al., 2020) have begun to question this central property of federated learning: since the gradients are directly linked to the local data, is it possible to reveal the local images through the uploaded gradient? Recent studies (Geiping et al., 2020; Yin et al., 2021; Jeon et al., 2021) provide a positive answer through multiple rounds of gradient matching, and indicate the training images can be revealed after certain iterations. The procedure, generally known as gradient inversion, starts from some random images and then gradually modifies these image pixels to match the uploaded gradient values.

But yet, the success of these works often relies on some strong assumptions: image recovery can be performed under small batch sizes and low image resolutions. For datasets like CIFAR-10, gradient inversion for batch sizes larger than 4 would be very challenging (Zhao et al., 2020). For high resolution and complex datasets like ImageNet, recovering images for batch size larger than 1 would almost be impossible (Yin et al., 2021). As comparison, participants of FL typically use a much larger batch size (*e.g.,* 128 on CIFAR-10 and 16 on ImageNet) for local model training. Inverting gradients into the original images in these cases remains challenging for current algorithms.

Lying ahead of the gradient inversion problem is the challenge of nested gradients. In general, local clients in the FL system will only transmit an *averaged gradient* to the server, instead of gradients for each image. Decoupling these gradients, especially for a large batch size, is clearly non-trivial since random decomposition may only lead the inversion work to a batch of noises. An ideal algorithm

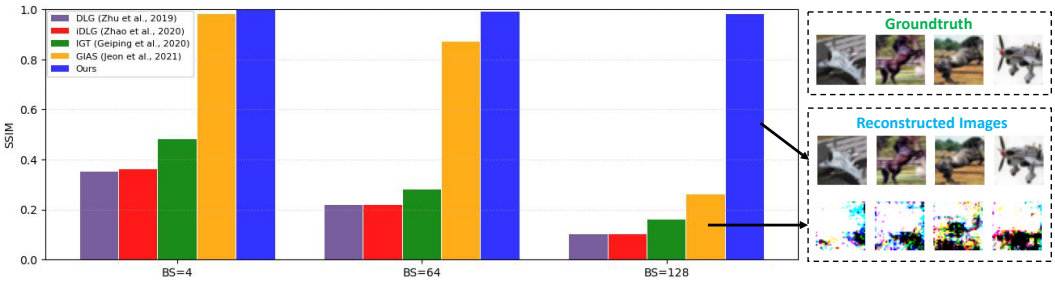

Figure 1: Gradient inversion on CIFAR-10 dataset. Results indicate the structural similarity (SSIM) drops significantly when the batch size increases to 128, except for our proposed method. Sample reconstructed images for GIAS and our method are also plotted for visual comparison to the groundtruth.

would therefore know how to decouple the averaged gradient in a correct way such that each gradient acts as a proxy for some natural image.

The conventional approach for the gradient coupling issue is to introduce extra regularization terms, besides the original gradient matching. For instance, a total variation term on the generated images is considered in Geiping et al. (2020) by penalizing images with high variances and provide an image prior. But we shall address in this paper that these extra regularization terms may change the fundamental properties of the original problem, and we cannot guarantee the truth images would trigger the minimal loss when these regularization terms are involved (See Section 2.2). As such, optimizing as the conventional ways may not lead us to the true images.

The goal of this work is to avoid using these regularization terms and propose an alternative solution with over-parametered convolutional networks. Design of the algorithm relies on three key understandings and insights into this gradient inversion problem: a convolutional network (also known as the generative network in Jeon et al. (2021)) generates an image prior to avoid gradient matching to noises, an over-parameterized network guarantees the non-empty of the image generation and gradient matching, and a properly-designed architecture creates pixel intimacy to implicitly reduces total variations. Building upon these insights, we propose an over-parameterized network, named as Convolutional Inversion Network (CI-Net), to provide a novel method for gradient matching without the necessity of prior information.

Numerical experiments indicate our proposed algorithm can perform well under more general scenarios, even with large batch sizes and high-resolutions. Figure 1 provides a simple example when increasing the batch size from 4 to 128, where only our proposed algorithm can reconstruct all groundtruth images when the batch size equals 128.

In summary, contributions of this paper are as follows: 1) We raise the issues of introducing additional regularization terms in the previous works: it may change the properties of the optimization function itself and cannot help to decouple the nested gradients; 2) Our understandings to the gradient inversion problem allow us to deviate from this mainstream and propose a novel over-parameterized algorithm that performs well in more general cases, such as large batches and high resolutions; 3) Moreover, the proposed algorithm is designed in a "plug and play" way: we do not require any pre-training, image prior or explicit regularizations, rendering the proposed method more applicable for gradient inversion in federated learning.

2 Problem Formulation

We introduce the problem formulation of gradient inversion in this part, namely how to reconstruct the original images from local gradient information. The potential issues of adding extra regularization terms are discussed when proposing optimization algorithms to solve this formulation.

2.1 Formulation

In the federated learning system, local training data and labels are generally not accessible and a curious server or third-party may only obtain the uploaded local gradient information in most scenarios. Given the uploaded gradient ∇W computed from a minibatch of groundtruth images and labels (x^*, y^*), the goal of gradient inversion is to search for some fake images $(\hat{x}^*, \hat{y}^*)$ that trigger

the minimum gradient matching loss:

$$(\hat{x}^*, \hat{y}^*) = \arg\min_{(\hat{x},\hat{y})} L_{\text{grad}}\left((\hat{x}, \hat{y}); W, \nabla W\right).$$

Following Zhu et al. (2019); Zhao et al. (2020), we assume the batch size and image resolution are known in advance so that the truth images x^* and the fake images $\hat{x}^*$ lie in the same space $\mathbb{R}^{N\times D}$, with N denoting the batch size and D representing the dimension of each individual sample.

Recent works (Zhao et al., 2020; Yin et al., 2021; Dang et al., 2021) find out the groundtruth labels y^* can be explicitly discovered via the last layer information. As such, the above formulation can be simplified to:

$$\hat{x}^* = \arg\min_{\hat{x}} L_{\text{grad}}\left(\hat{x}, W, \nabla W\right). \tag{1}$$

Choice of the gradient matching loss L_{grad} could be

$$L_{\text{grad}}\left(\hat{x}, W, \nabla W\right) := \left\|\nabla_W L(\hat{x}, y^*) - \nabla_W L(x^*, y^*)\right\|^2, \tag{2}$$

if an L_2-norm loss is involved. Alternatively, the gradient matching loss L_{grad} can be the cosine-similarity loss (Geiping et al., 2020):

$$L_{\text{grad}}\left(\hat{x}, W, \nabla W\right) := 1 - \frac{\langle \nabla_W L(\hat{x}, y^*), \nabla_W L(x^*, y^*)\rangle}{\|\nabla_W L(\hat{x}, y^*)\|\|\nabla_W L(x^*, y^*)\|}. \tag{3}$$

Besides this gradient matching loss, a series of recent studies also consider adding extra regularization terms λL_{reg} to the gradient loss, in order to create the simple image prior when reconstructing images. For instance, the work in Geiping et al. (2020) considers the total variation loss $\text{TV}(x)$ as the regularization term, while multiple fidelity and group consistency regularization terms are utilized in Yin et al. (2021). Therefore, the overall loss becomes:

$$L_{\text{sum}}\left(\hat{x}, W, \nabla W\right) = L_{\text{grad}}\left(\hat{x}, W, \nabla W\right) + \lambda L_{\text{reg}}(\hat{x}). \tag{4}$$

2.2 Issues of Regularization Terms

Adding these regularization terms may help to improve the performance in certain scenarios, but we shall highlight one key issue that is often neglected in the previous studies: adding these regularization terms may alter the fundamental properties of the problem itself. To see this, note the groundtruth images will obtain a zero loss in Eq (2) and (3), but there is no guarantee that the underlying original training images will still trigger the minimal loss in Eq (4) when additional terms are involved. As such, when using gradient descent to minimize the summed loss, we cannot ensure the generated data are moving towards groundtruth images as expected. Moreover, as we have shown in Figure 1, these regularization terms may not help to decouple the nested gradients when the batch size is relatively large.

Therefore, in this paper, we shall deviate from this mainstream of introducing extra regularization terms, and focus on designing proper network architectures to consider the gradient matching loss L_{grad}.

3 Existence and Uniqueness of Solutions

Let us first consider the existence and uniqueness of solutions in Eq (2) and (3) before marching towards any practical algorithms.

3.1 Existence and Uniqueness

Existence of an optimal solution is apparently trivial, since the groundtruth images trigger zero loss and act as one optimal solution. But the real question is whether these images are the only solution for the above loss functions. Otherwise, we may face the same issue as alluded to earlier: if there are other solutions also obtaining zero loss, we cannot guarantee the optimization algorithm will lead us to the groundtruth.

Unfortunately, the answer is negative in general: we cannot guarantee the uniqueness of solution. To see this, consider the following simple example of a 1-layer neural network.

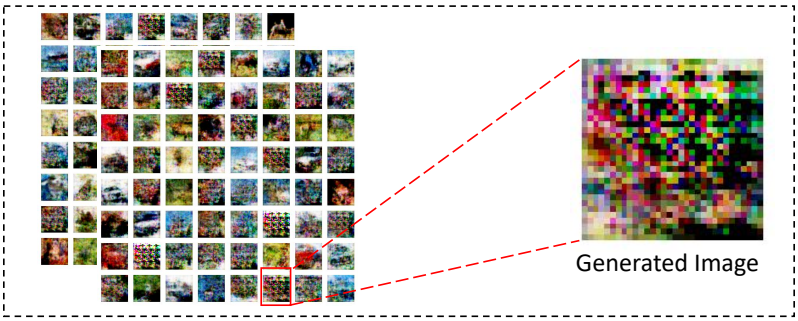

Figure 2: The figure demonstrates the mathematical optimal solution may not be practically feasible, when using gradient inversion to reconstruct 128 images from the CIFAR-10 dataset. A convolutional model from Jeon et al. (2021) is trained for 50k iterations with signed gradient descent, obtaining a mean loss of 5.09×10^{-5}. Despite the small loss, the generated images are highly blurred.

Proposition 1. *Consider the gradient inversion problem on the 1-layer neural network and a mini-batch data of N samples. Eq (2) and (3) obtain a zero loss when the generated images $\hat{x} = [\hat{x}_1, \hat{x}_2, \cdots, \hat{x}_N]$ satisfy:*

$$N \cdot \nabla W = \hat{x} \cdot \hat{P}, \tag{5}$$

where $\hat{P}$ refers to a matrix defined by the prediction probabilities $\{\hat{p}_{i,j}\}$ and the one-hot encoding for the labels y^*.

The above equation defines the constraints that the generated images $\hat{x}$ must satisfy in order to obtain a zero loss. Note these images can be considered as some free variables living in the space of $\mathbb{R}^{N \times D}$, with D denoting the dimensions of each independent sample $\hat{x}_i$. For any uploaded ∇W, we can always find a proper N such that $\dim(\hat{x}) > \dim(\nabla W)$ so that the uniqueness of solution cannot be guaranteed.

This 1-layer neural network can be extended to the more general cases where the gradient constraints are not sufficient enough to derive a unique solution. A typical case would therefore be training local gradients with a large number of local images. Optimizing towards (2) or (3) in this case cannot guarantee us to reach the true images as we expected [1].

3.2 Mathematical Non-Uniqueness vs Practical Non-Uniqueness

The mathematical non-uniqueness nevertheless rains on our parade, since directly optimizing the gradient matching may not lead us to the groundtruth as we expected. But a close look into this set could provide more insights into the problem.

Let us first define the optimal set of (1) as

$$\hat{X}_{\text{grad}} := \left\{ \hat{x}^* \in \mathbb{R}^{N \times D} \mid \arg\min_{\hat{x}^*} L_{\text{grad}}\left(\hat{x}, W, \nabla W\right) \right\}. \tag{6}$$

The existence proof renders this set $\hat{X}_{\text{grad}}$ non-empty for all cases, whereas the non-uniqueness property allows it to contain more than one solution, or even infinite solutions in certain cases. But yet, a close look renders most of the solutions actually "not practically feasible". To see this, consider the case in Figure 2, where we numerically test the gradient inversion performance on 128 CIFAR-10 images. Despite the closeness of gradient matching (loss $\approx \mathcal{O}(10^{-5})$), the obtained results in fact consist of a mixture of natural images and some blurring images that can hardly be reckoned as the true solutions. The underlying reason is the gradient matching step is often performed on a pixel-by-pixel level, while properties of natural images, such as smoothness of neighbourhood pixels, are not well addressed in the gradient inversion problem.

This example illustrates a typical phenomenon when the underlying batch size is relatively large and gradient matching may give us more than one solution. Decomposing the averaged gradient

[1] The non-uniqueness problem was also demonstrated in Zhu & Blaschko (2020), where the authors showed that there could exist a pair of different data having the same gradient even for a large network.

arbitrarily may lead to a very large set of mathematical feasible solutions, but most of these decoupled gradients may only result in blurring pictures that are more like noises. The set of practically feasible solutions is actually much smaller when solving the optimization problem in Eq (2) or (3).

4 THE PROPOSED METHOD

The reduction of practically feasible solutions motivates us to find a new gradient inversion solution that emphasizes the similarity to natural images, instead of simply addressing the gradient matching issue. In this part, we present *three key pillars* to find a proper solution in gradient inversion: a convolutional architecture, an over-parameterization setting and a properly designed network to create pixel intimacy.

4.1 CONVOLUTIONAL METHOD

As alluded to earlier, a gradient inversion algorithm should decompose the averaged gradient into proper proxies for "natural images" before inverting them into multiple pictures. But this is clearly non-trivial since there would be infinite solutions to decompose the averaged gradient. Adding regularization terms like total variation may alter the fundamental properties of the problem and does not perform well in practical cases.

Convolutional networks, on the other hand, are proved to have an image prior that favours natural images over high-frequency noises. In Ulyanov et al. (2018), the authors showed that the structure of a convolutional network itself is sufficient to capture a great deal of low-level image statistics prior to any learning. Given a perturbed natural image, the convolutional network could first learn a clean solution before fitting to the noisy groundtruth. This also explained why simple architectures like convolutional generators can generate high-fidelity images in the generative adversarial network (GAN) studies (Goodfellow et al., 2020; Radford et al., 2015).

Such a priority over clean images motivates us to consider the convolutional generative network as the backbone when performing gradient inversion, instead of optimizing pixel values independently. A pure convolutional method also avoids the potential biases arising from regularization terms like total variation (Dosovitskiy et al., 2015).

Specifically, we require the convolutional model G to take a latent vector z_0 as its input and generate a batch of images $\hat{x} = G(z_0, \theta)$ to satisfy the gradient matching constraint. Note the main difference to the conventional algorithms is now we generate images $\hat{x}$ from a convolutional model instead of updating them directly to satisfy the gradient matching loss, as in (Zhu et al., 2019; Zhao et al., 2020; Yin et al., 2021).

From the mathematical aspect, the following intersection theorem indicates that our solution space is actually narrowed down.

Theorem 2. *Given a latent vector z_0 and a convolutional model G, define its generative model space as $\hat{X}_G := \{\hat{x} \in \mathbb{R}^{N\times D} \mid \hat{x} = G(z_0, \theta), \theta \in \Theta\}$. For the gradient inversion problem, we consider a proper solution space as the intersection of the gradient matching space and the model generation space, namely*

$$\hat{X}_\Lambda := \hat{X}_{grad} \cap \hat{X}_G. \tag{7}$$

Let us make a few remarks here.

- We expect the model space $\hat{X}_{\text{G}}$ to contain images more close to natural images, while the gradient matching space $\hat{X}_{\text{grad}}$ requires images (including noisy images) to satisfy the matching requirement. A direct translation of the above theorem is that the solution should not only satisfy the gradient matching requirement, but also should be (or similar to) natural images.
- For conventional gradient inversion algorithms (Zhu et al., 2019; Zhao et al., 2020), solutions would only lie in $\hat{x} \in \hat{X}_{\text{grad}}$, whereas in the above theorem we have

  $$\hat{X}_\Lambda := \hat{X}_{\text{grad}} \cap \hat{X}_{\text{G}} \subset \hat{X}_{\text{grad}}.$$

  Hence, the solution space is narrowed down.
- We can also vary the input z_0 to expand the model space $\hat{X}_{\text{G}}$, but throughout this paper, we shall select a fixed z_0 and only update θ for convenience.

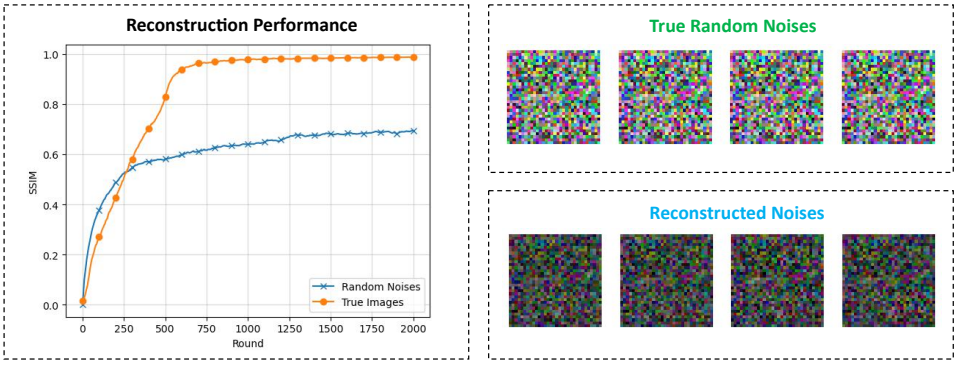

Figure 3: Left is the gradient inversion on both true images and random noises. The true random noises and the reconstructed noises from gradient inversion are plotted in the right figures for visual comparison.

Boiling down to the gradient inversion problem itself, we numerically test such an image prior by requiring the same convolutional network to perform gradient inversion on 4 images from the CIFAR-10 dataset and 4 random noises as the fake images. Figure 3 indicates that the groundtruth can be easily recovered for natural images, while the same convolutional network fails to obtain high-fidelity random noises. The reconstructed noises, ranging from -0.16 to 0.62, are statistically different from the original fake images (normalized to [0,1]).

4.2 Over-Parameterization

But apparently, not every convolutional network G can act as our backbone. For instance, in Figure 2, we follow the previous work (Jeon et al., 2021) to adopt a convolutional generator from DCGAN (Radford et al., 2015) for gradient inversion, but its performance is clearly not satisfactory. The question here is we need to ensure $\hat{X}_{\mathrm{G}}$ is sufficiently large so that the intersection (7) is non-empty. Denoting the parameters of G as $P(G)$, the following proposition provides a guarantee for the non-emptiness of intersection.

Proposition 3. *There exists a number N_0 such that when $P(G) > N_0$, we have $\hat{X}_\Lambda \neq \emptyset$.*

From the theoretical aspect, N_0 can be set as a sufficiently large number so that the generative space $\hat{X}_{\mathrm{G}}$ is expanded to the whole space, namely $\hat{X}_{\mathrm{G}} = \mathbb{R}^{N\times D}$. Intersection in this case is always non-empty as the groundtruth images satisfies the gradient constraints $x^* \in \hat{X}_{\text{grad}}$. Empirically, while the linear-independent constraints of ∇W is case-by-case, a safe choice of N_0 is to require the parameter number of G to be larger than the original model $F(x, W)$.

The over-parameterization requirement of $P(G) > P(F)$ may be somehow counter-intuitive at the first glance: the training parameters now exceeds the constraints and we no longer have the uniqueness guarantee. Under-parameterization, on the one hand, may help to render the solution to be unique. But we shall argue here that the under-parameterization will generally obtain a non-zero

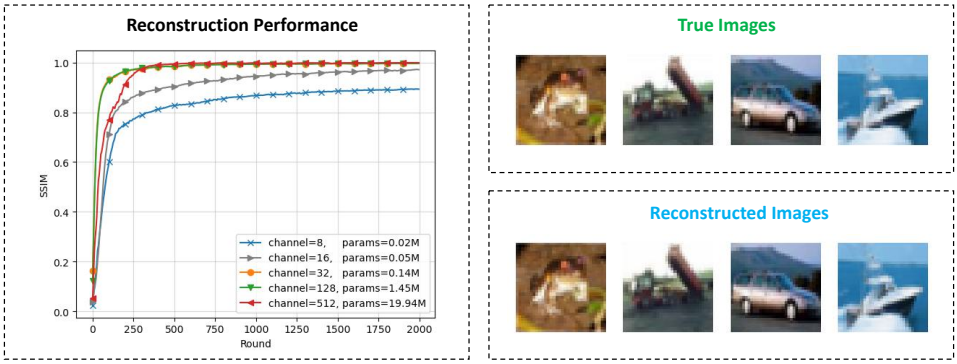

Figure 4: Left figure shows the gradient inversion on a convolutional model with various channels. Right figure plots the true and reconstructed images when using a highly over-parameterized model (channel=512).

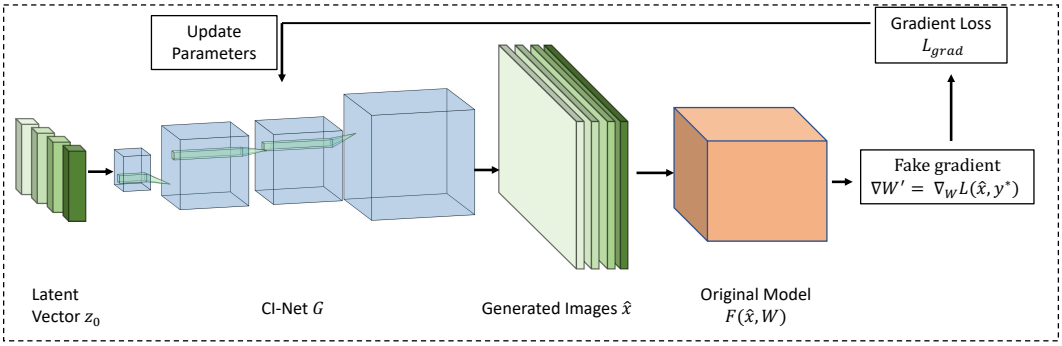

Figure 5: Gradient Inversion with CI-Net.

loss and the obtained figures may not be close to the groundtruth. By adding more parameters, we expect the parameter number to exceed the constraint number so that a minimal loss is incurred. When the optimal solutions for over-parameterization may be non-unique, the previous convolutional image prior comes to rescue: among all possible solutions, it prioritizes natural images.

For numerical validations, we conduct an ablation study in Figure 4 by varying the parameter numbers of the same convolutional network architecture. The original model F is made of 2 convolutional layers and 1 linear output layer, containing 0.33M parameters. Results indicate that an under-parameterized G (channel=8 and 16) could lead to poor performance, whereas an over-parameterized network model (channel=128) obtains a structural similarity index measure value (SSIM, Wang et al. (2004)) of almost 1. Note even for the highly over-parameterized case where channel equals 512 and P(G) $\approx$ 60 P(F), obtaining high-fidelity images is still possible.

4.3 Pixel Intimacy

The last issue is how to choose a practical over-parameterized convolutional architecture. Natural images possess a series of properties that distinguish them different from random noises, where frequent pixel jumps are generally not possible. To address this property, we consider the progressive-growing network (Karras et al., 2017) as our underlying model but tailor its architecture to fit to the gradient inversion problem (details in Appendix B.1). Specifically, an image core, usually starting from 4×4 pixels, is first generated and then progressively grows to the targeted resolution. A key step here is the interpolation when upscaling image resolution: it inserts extra pixels by considering neighbourhood values. This interpolation step allows the new pixels to be intimately related to its neighbouring, hence creating an implicit regularization on the total variation.

4.4 Summary

To this end, we can now propose an over-parameterized convolutional algorithm to generate images before fitting to the gradient matching requirement. The network, named as the convolutional inversion network (CI-Net), is built upon the above understandings and insights for the gradient inversion problem itself. We depict how to utilize such a network in Figure 5: firstly, an over-parameterized convolutional CI-Net G will take a random vector z_0 as its input and generate some images $\hat{x}$; the fake gradient $\nabla W'$ is then compared to the true gradient to update the parameters of our network $G(z_0, \theta)$. Details of the above procedure is depicted in Alg 1 in Appendix B.

More importantly, the proposed method is designed in a "plug and play" way: we do not require any prior information, pre-training or regularization. Gradient inversion can be applied directly on the untrained network, without the necessities to know potential data distribution or fitting other data beforehand. This allows the above algorithm to be more applicable for general FL settings, where local data are totally invisible.

5 Experiments

With these designs, we can now proceed to the practical validations on real-world datasets. The focus of this part is to test algorithm performance on large batch size and high resolution images.

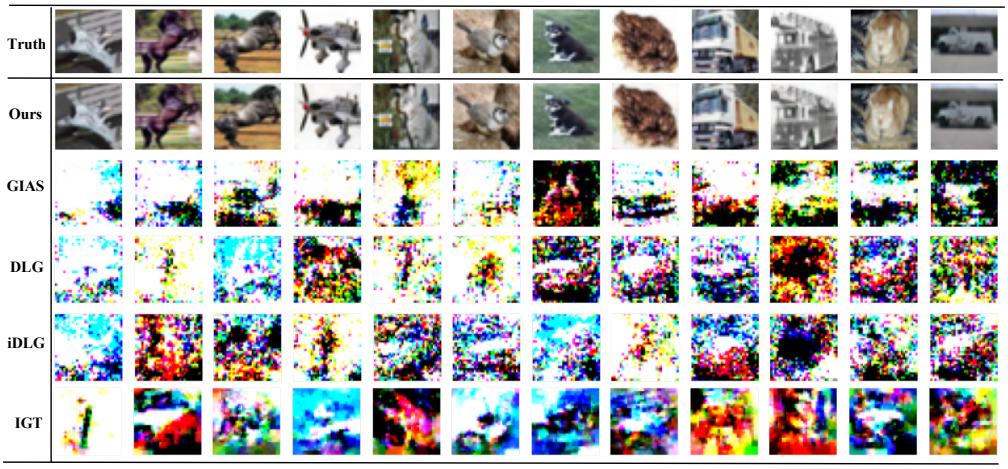

Figure 6: Sample reconstructed images on CIFAR-10 dataset, when batch size equals 128.

5.1 PREPARATIONS

We consider the image classification task on CIFAR-10 (resolution: 32×32) and ImageNet (resolution: 256×256). A Resnet-18 network He et al. (2016) is utilized as the original model F, with its activation function replaced by sigmoid as Zhu et al. (2019). All experiments are conducted on Nvidia-A100 with 40GB GPU memory. A line of recent gradient inversion algorithms, DLG Zhu et al. (2019), iDLG Zhao et al. (2020), IGT Geiping et al. (2020) and GIAS Jeon et al. (2021)), are reproduced based on their original source codes for performance comparison. Specifically, this work is most related to the pioneering generative method GIAS, but the differences lie in three aspects: 1) we do not require any image prior or pre-training; 2) we reveal the over-parameterization is one of the key factors, whereas the convolutional model in Jeon et al. (2021) could be under-parameterized (e.g., model in Figure 4); 3) we design a specific network for the gradient inversion problem instead of using any existing architectures.

5.2 CIFAR-10 EXPERIMENT

In this experiment, algorithms are required to decode an averaged gradient computed from 128 CIFAR-10 images through inverse engineering. The goal is to address the large batch size issue in gradient inversion. For validations, the whole process is repeated on 3 ResNet-18 models generated from different seeds. Four image quality assessment metrics are then applied to measure the similarities between the obtained images and the groundtruth.

Algorithm	SSIM↑	FSIM↑	PSNR↑	LPIPS (VGG)↓
DLG Zhu et al. (2019)	0.10 ± 0.01	0.58 ± 0.01	6.13 ± 0.06	0.65 ± 0.01
iDLG Zhao et al. (2020)	0.10 ± 0.01	0.57 ± 0.01	6.03 ± 0.01	0.61 ± 0.01
IGT Geiping et al. (2020)	0.16 ± 0.01	0.59 ± 0.01	8.03 ± 0.26	0.61 ± 0.01
GIAS Jeon et al. (2021)	0.26 ± 0.11	0.66 ± 0.06	11.05 ± 2.85	0.59 ± 0.06
Ours	$\mathbf{0.98 \pm 0.01}$	$\mathbf{0.98 \pm 0.01}$	$\mathbf{31.40 \pm 0.14}$	$\mathbf{0.03 \pm 0.01}$

Table 1: Algorithm performance of gradient inversion on CIFAR-10 data, when batch size equals 128. SSIM and FSIM have maximum value 1, and LPIPS has minimum value 0.

Table 1 summarizes the overall performance for all algorithms. Results indicate that all algorithms except for the proposed method fail to obtain proper decompositions in the gradient inversion process and obtains very low SSIM and PSNR value. In contrast, the proposed algorithm can successfully reconstruct the groundtruth images in Figure 1, obtaining high-fidelity images with an SSIM value of 0.98.

5.3 IMAGENET EXPERIMENT

We repeat the above experiments on the ImageNet dataset to address the high-resolution problem. As such, we do not scale down the images (e.g., Jeon et al. (2021)) but keep the original image

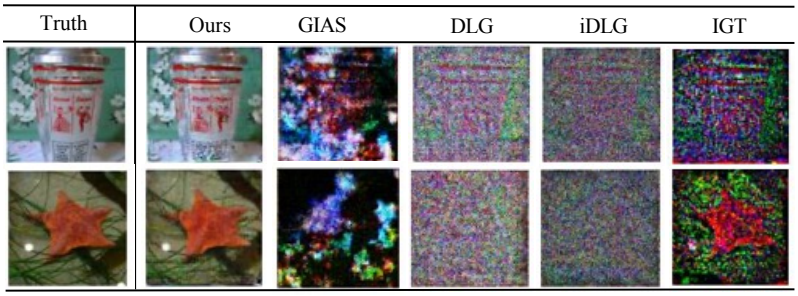

Figure 7: Sample reconstructed images on ImageNet dataset, when batch size equals 24.

resolution for better simulations. A batch size of 24 images is selected to generate an averaged gradient for each algorithm to perform gradient inversion.

Table 2 indicates such a high-resolution and large batch causes severe challenges to all the algorithms, where the best SSIM value for our competitors is only 0.04. Performance of the proposed algorithm still beats the rest algorithm, but is also affected by the data complexity. Figure 7 plots the reconstructed results for the first 2 images, and the rest are presented in Appendix D. Visual results illustrate the proposed algorithm can still reconstruct the original image, but pictures are relatively blurred compared to the groundtruth.

Algorithm	SSIM↑	FSIM↑	PSNR↑	LPIPS (VGG)↓
DLG Zhu et al. (2019)	0.01 ± 0.00	0.45 ± 0.01	5.40 ± 0.05	0.84 ± 0.01
iDLG Zhao et al. (2020)	0.01 ± 0.00	0.47 ± 0.01	6.09 ± 0.04	0.83 ± 0.01
IGT Geiping et al. (2020)	0.04 ± 0.01	0.53 ± 0.01	7.90 ± 0.22	0.77 ± 0.01
GIAS Jeon et al. (2021)	0.04 ± 0.02	0.54 ± 0.04	8.03 ± 0.77	0.78 ± 0.07
Ours	$\mathbf{0.52 \pm 0.06}$	$\mathbf{0.77 \pm 0.03}$	$\mathbf{19.64 \pm 1.05}$	$\mathbf{0.49 \pm 0.04}$

Table 2: Algorithm performance of gradient inversion on ImageNet, when batch size equals 24.

5.4 Extension to Larger Sizes

The above findings can be extended to an even larger batch, and we also numerically test on 256 CIFAR-10 images and 32 ImageNet pictures. Note these are the maximum batch sizes that our GPU memory can support. Results in Table 3 are consistent with our previous conclusions and the proposed algorithm continuously generate high-fidelity images on the CIFAR-10 dataset. For space limitations, results on ImageNet are provided in Appendix D.

Batch Size	SSIM↑	FSIM↑	PSNR↑	LPIPS (VGG)↓
64	1.00 ± 0.00	1.00 ± 0.00	33.72 ± 0.05	0.01 ± 0.00
128	0.98 ± 0.01	0.98 ± 0.01	31.40 ± 0.14	0.03 ± 0.01
256	0.98 ± 0.01	0.99 ± 0.01	34.11 ± 0.13	0.02 ± 0.01

Table 3: Performance of CI-Net, with various batch sizes.

6 Summary

In this paper, we propose a new convolutional network named CI-Net to perform gradient inversion attack in federated learning. The three key elements to the network are: a convolutional architecture, an over-parameterization requirement and a properly designed growing model. Such a network is required to generate images and then adjust its parameters to fit the true gradients, instead of the conventional ways on the pixel level. We conduct a series of practical experiments to demonstrate that the proposed algorithm can outperform SOTA algorithms and reconstruct the underlying original training images more effectively, even with large batch sizes and high resolutions.

REFERENCES

Trung Dang, Om Thakkar, Swaroop Ramaswamy, Rajiv Mathews, Peter Chin, and Françoise Beaufays. Revealing and protecting labels in distributed training. *Advances in Neural Information Processing Systems*, 34:1727–1738, 2021.

Alexey Dosovitskiy, Thomas Brox, et al. Inverting convolutional networks with convolutional networks. *arXiv preprint arXiv:1506.02753*, 4(2):3, 2015.

Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Proceedings of the Advances in Neural Information Processing Systems*, 33:16937–16947, 2020.

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

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

Jinwoo Jeon, Kangwook Lee, Sewoong Oh, Jungseul Ok, et al. Gradient inversion with generative image prior. *Advances in Neural Information Processing Systems*, 34:29898–29908, 2021.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Jakub Konečnỳ, Brendan McMahan, and Daniel Ramage. Federated optimization: Distributed optimization beyond the datacenter. *arXiv preprint arXiv:1511.03575*, 2015.

Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pp. 1273–1282. PMLR, 2017.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9446–9454, 2018.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

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*, pp. 16337–16346, 2021.

Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. iDLG: Improved deep leakage from gradients. *CoRR*, 2020. URL `https://arxiv.org/abs/2001.02610`.

Junyi Zhu and Matthew Blaschko. R-gap: Recursive gradient attack on privacy. *arXiv preprint arXiv:2010.07733*, 2020.

Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Proceedings of the Advances in Neural Information Processing Systems*, 32, 2019.