OASIS: Offsetting Active Reconstruction Attacks in Federated Learning

Tre' R. Jeter[†*], Truc Nguyen[‡*], Raed Alharbi[§], and My T. Thai[†]
[†]University of Florida, Gainesville, FL 32611, USA
[‡]National Renewable Energy Laboratory, Golden, CO 80401, USA
[§]Saudi Electronic University, Riyadh, Saudi Arabia
Email: t.jeter@ufl.edu, Truc.Nguyen@nrel.gov, ri.alharbi@seu.edu.sa, mythai@cise.ufl.edu

***Abstract*—Federated Learning (FL) has garnered significant attention for its potential to protect user privacy while enhancing model training efficiency. For that reason, FL has found its use in various domains, from healthcare to industrial engineering, especially where data cannot be easily exchanged due to sensitive information or privacy laws. However, recent research has demonstrated that FL protocols can be easily compromised by active reconstruction attacks executed by dishonest servers. These attacks involve the malicious modification of global model parameters, allowing the server to obtain a verbatim copy of users' private data by inverting their gradient updates. Tackling this class of attack remains a crucial challenge due to the strong threat model. In this paper, we propose a defense mechanism, namely OASIS, based on image augmentation that effectively counteracts active reconstruction attacks while preserving model performance. We first uncover the core principle of gradient inversion that enables these attacks and theoretically identify the main conditions by which the defense can be robust regardless of the attack strategies. We then construct our defense with image augmentation showing that it can undermine the attack principle. Comprehensive evaluations demonstrate the efficacy of the defense mechanism highlighting its feasibility as a solution.**

***Index Terms*—Federated Learning, Privacy, Deep Neural Networks, Reconstruction Attack, Dishonest Servers**

I. Introduction

In recent years, Federated Learning (FL) has developed into a well-respected distributed learning framework that promotes user privacy with high model performance. By design, FL authorizes collaborative training of a global model between millions of users without revealing any of their locally trained, private data. It is an iterative protocol where, in each round, a central server distributes the most up-to-date global model to an arbitrary subset of users that train locally and communicate their model updates back to the server. These model updates include the gradients that are calculated based on the global model and the local training data. The central server then averages these model updates to form a new global model to distribute in the next round.

With a disruptive privacy-centric design, FL has been regarded as an auspicious solution for applying machine learning to the healthcare sector, particularly in scenarios where sharing medical data between different sites is intractable due to strict privacy protection policies such as the Health Insurance Portability and Accountability Act (HIPAA) [1] and General Data Protection Regulation (GDPR) [2]. Numerous studies have proposed FL for medical image analysis, utilizing data such as X-rays, MRIs, and PET scans from different hospital sites while complying with privacy laws [3]–[8]. This innovative approach is not limited to healthcare; FL is also making significant strides in industrial engineering. For instance, in urban environment image sensing, research has shown that FL makes it easier to perform a time-series analysis of industrial environment factors obtained from multiple sensors and unmanned aerial vehicles (UAVs) across different companies while maintaining confidential data privacy [9]–[12]. Beyond these applications, FL is also stimulating advancements in diverse domains such as control systems, autonomous vehicles, and smart manufacturing [13]–[15], showcasing the versatility and broad impact of a privacy-preserving learning framework.

However, the promise of privacy for clients in FL has been constantly challenged [16]. Recent work [17]–[24] has investigated a strong and practical threat model in which the server can be *actively dishonest*, such that it is capable of maliciously modifying the global model before dispatching it to the users. This threat model has instigated several *active* reconstruction attacks in which an FL server can perfectly reconstruct some data points in a users' training data [17], [18], [24]. These attacks exploit a fundamental concept that the gradients in the local model updates sent by users may contain complete and memorized individual data points. These gradients can later be inverted by the server to reveal such data points. As an actively dishonest adversary, the server can strategically manipulate the weights of the global model to maximize the number of individual data points that can be reconstructed from the users' gradients. For that reason, although the training data is said to never leave a users' device, it can still be reconstructed, thereby refuting the claim of privacy-preservation in FL.

Given such a strong adversary, defending against this class of *active* reconstruction attacks is challenging. Until now, a mitigation approach for FL-based attacks focused on obfuscating the gradients via a Differential Privacy (DP) mechanism, such as DPSGD [25], that formally bounds the privacy leakage by adding calibrated noise to the gradients. However, previous work [17], [18] has shown that to prevent an attacker from discerning the content of reconstructed data, the user must add a significant amount of DP noise to the gradients that unfortunately degrades the overall model performance.

In this paper, we propose a new defense, OASIS, to **O**ffset this class of **A**ctive recon**S**truct**I**on attack**S**. As there are different strategies to manipulate the global model for conducting

[*]These authors contributed equally to this work.

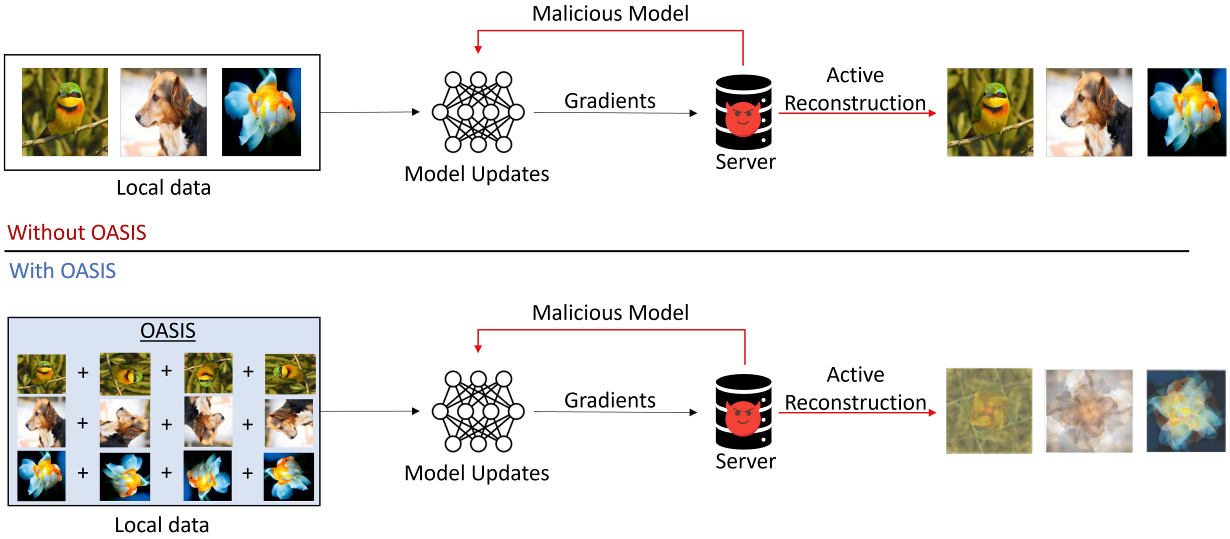


Figure 1: Overview design of OASIS. *Top:* Standard *active* reconstruction attack with malicious model modifications perfectly reconstructing training samples. *Bottom:* OASIS in place with augmented data to defend the *active* reconstruction attacks. The resulting reconstruction is a linear combination of images, effectively hiding the content of training samples. *Note: Rotation is not the only transformation within OASIS.*

this attack, it is imperative to figure out how to tackle the attack in principle so that the defense is robust regardless of the manipulation strategies. We first analyze the attack surface and determine the core vulnerability in the gradient updates that enables the memorization of individual training samples. By doing so, we generalize the existing attacks by discovering the conditions under which a dishonest server can conduct *gradient inversion* to reconstruct users' data. We then intuitively show how to undermine those conditions and mitigate the impact of the attacks.

From the attack principle, we show that the users can preprocess their training data in a way that prevents the samples from being revealed via gradient inversion, effectively countering this class of *active* reconstruction attacks. A mechanism for such preprocessing is image augmentation [26], [27]. This includes adding augmented versions of an image, such as rotated, flipped, and sheared counterparts to the training data before computing the gradients. By doing this, OASIS aims to have the gradients memorize a *linear combination* of the original image and its augmented versions, instead of memorizing any individual images. As a result, inverting these gradients would reconstruct what appears to be an overlap of multiple images, thereby effectively preventing the server from discerning the content of the reconstructed images, as shown in Figure 1. Since image augmentation is used to improve model generalization [28], we safely maintain the performance of FL with this countermeasure. Our analysis shows that OASIS opens a new approach to protect users' data from gradient inversion without suffering the utility loss as in DP.

Contributions. Our key contributions are as follows:

- We analyze the attack surface and determine the key principle behind gradient inversion that enables *active* reconstruction attacks. From that, we theoretically show how to tackle this class of attack, regardless of how the attacker manipulates the global model parameters.
- Based on the attack principle, we present OASIS as a suite of image augmentations. To our knowledge, this mechanism stands as the first general and scalable defense against *active* reconstruction attacks via gradient inversion by *actively dishonest servers* in FL.
- We thoroughly analyze the effectiveness of OASIS through experiments with respect to attack success rate and augmentation type. We also show how OASIS maintains model performance.

Organization. The paper's structure is as follows: Section II provides a primer on FL, image augmentation, and the main augmentations used. Section III presents the threat model, attack principle, and our OASIS defense. Section IV presents an in-depth experimental analysis and results supporting our defense. Section V discusses related research on reconstruction attacks and existing defenses. Section VI concludes the paper, summarizing our key findings.

II. Preliminaries

In this section, we summarize the FL process while also describing the benefits of image augmentation during training.

A. *Federated Learning*

Depending on how training data is distributed among the participants, there are two main versions of FL: horizontal and vertical. In this paper, we focus on a horizontal setting in which different data owners hold the same set of features but different sets of samples. We denote $f_w : \mathbb{R}^d \to \mathbb{R}^k$ as a k-class neural network model that is parameterized by a set of weights w. The goal of f_w is to map a data point $x \in \mathbb{R}^d$ to a vector of posterior probabilities $f_w(x_i) = \mathcal{Y}$ over k classes.

FL is an iterative learning framework for training a global model f_w on decentralized data owned by N different users $\{u_j\}_{j=1}^N$. A central server coordinates the training of f_w by iteratively aggregating gradients computed locally by the users. Let $t \in [0, T]$ be the current iteration of the FL protocol, and w^t be the set of parameters at iteration t. At iteration $t = 0$, the global w^t is initialized at random by a central server. At every iteration t, a subset of $M < N$ users is randomly selected to contribute to the training. Each of the selected users u_j obtains f_{w^t} from the central server and calculates the gradients G_j^t for f_{w^t} using their local training batch $\mathcal{D}_j$. In specific, $G_j^t = \nabla_{w^t}\mathcal{L}(\mathcal{D}_j, w^t)$ where $\mathcal{L}$ is a loss function. Then, each u_j uploads its gradients to the central server. With a learning rate η, the server averages these gradients to update the global model's parameters as follows:

$$G^t = \frac{1}{M}\sum_{j=1}^{M} G_j^t, \quad w^{t+1} = w^t - \eta G^t \tag{1}$$

The training continues until f_{w^t} converges.

B. Image Augmentation

Image augmentation is a very useful technique in deep learning that allows for the expansion of a training dataset with artificially generated data. Given a dataset of images, augmenting each image using rotation, shearing, or flipping yields a new expanded dataset for training. This added preprocessing helps increase model generalization and avoid overfitting by altering the makeup of data and adding it to the training set [26], [27]. With more data to train, the model is less prone to *memorize* the data, but generalize the pattern between the data. In turn, increased model generalization *tends to* lead to higher model performance. Image augmentation has been widely used in datasets like ImageNet [29] and CIFAR-10 [28].

Our work focuses on three main transformations: rotation, shearing, and flipping. In each augmentation scenario, we consider a 2D image I where $I(i, j)$ denotes the pixel value at coordinates (i, j). Rotation includes tilting an image's pixels by an angle θ. We define major rotation angles as the maximum degrees of each respective quadrant in an x-y coordinate system (i.e., 90°, 180°, and 270°). Minor rotation angles are described as any angle $< 90°$. More formally, an image I' can be constructed from I as follows:

$$I'(i, j) = I(i\cos(\theta) - j\sin(\theta), i\sin(\theta) + j\cos(\theta)) \quad \forall i, j \tag{2}$$

where θ is the angle in which an image is rotated.

Flipping includes reflecting an image on its x-axis (vertical flip) or its y-axis (horizontal flip). A horizontally flipped image I' can be constructed from I as follows:

$$I'(i, j) = I(-i, j) \quad \forall i, j \tag{3}$$

Similarly, a vertically flipped image I' can be constructed from I as follows:

$$I'(i, j) = I(i, -j) \quad \forall i, j \tag{4}$$

Shearing is projecting a point or set of points within an image in a different direction. A sheared image I' can be constructed from I as follows:

$$I'(i, j) = I(i + \mu j, j) \quad \forall i, j \tag{5}$$

where μ is the shear factor controlling the *shearing intensity*.

III. OASIS – A Proposed Defense

This section describes our proposed defense, OASIS, against *active* reconstruction attacks via gradient inversion in FL. In order to devise an effective defense, we analyze the attack surface to determine the core vulnerability of the system and how the attacks exploit it in principle. We then propose OASIS to prevent such exploitation, effectively tackling this class of attacks, regardless of how they are implemented.

A. Generalizing Active Reconstruction Attacks via Gradient Inversion

Threat Model. We examine a server that is dishonest and aims to reconstruct the private data of a targeted user. As discussed in previous work [17], [18], a dishonest server is capable of making malicious modifications to w before dispatching it to the users at any iterations. These modifications can include changing and/or adding model parameters. However, the modification should be minimal to avoid detection.

For this attack, the adversary places a malicious fully-connected layer consisting of n attacked neurons in the neural network model f_w, so that inverting the gradients of these neurons would recover the users' data. Generally, the attack becomes less effective when the layer is placed deeper in the neural network. For the purpose of devising a robust defense, we consider a strong adversary who can place the malicious layer directly right after the input layer. The layer is parameterized by a weight matrix $W \in \mathbb{R}^{n \times d}$ and a bias vector $b \in \mathbb{R}^n$. Denoting $\mathcal{D} = \{x_j \in R^d\}_{j=1}^B$ as the local training data of a targeted user where B is the batch size, the goal of the attack is to reconstruct the data points in $\mathcal{D}$ via the malicious layer. Our defense, OASIS, aims to minimize the quality of reconstruction, regardless of how the malicious layer (W, b) is determined by the adversary.

Attack Vector Analysis. We aim to generalize state-of-the-art *active* reconstruction attacks by deducing their core principle. Suppose that the malicious layer is updated based on one single-input $x_t \in \mathbb{R}^d$, for each neuron i, the gradients of the loss with respect to the weights, and biases will be

$$\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)$$

where $\mathcal{L}_t$ is shorthand for $\mathcal{L}(x_t, (W, b))$. All the gradients

$$\left\{\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)\right\}_{i=1}^n$$

are then uploaded to the server. As shown in [17], [18], [30], with a ReLU activation function, the server can perfectly reconstruct x_t by dividing the gradients as follows:

$$\left(\frac{\partial \mathcal{L}_t}{\partial b_i}\right)^{-1}\frac{\partial \mathcal{L}_t}{\partial W_i} = x_t \tag{6}$$

where i is the index of a neuron that is *activated* by the input x_t and $\frac{\partial \mathcal{L}_t}{\partial b_i} \neq 0$. In other words, knowing the gradients $\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)$ of a particular input sample x_t allows perfect reconstruction of that sample via gradient inversion.

However, in practical FL, when the malicious layer is updated based on a batched input $\mathcal{D} = \{x_j\}_{j=1}^{B}$ where $B > 1$, all derivatives are summed over the batch dimension. In particular, the gradients of the malicious layer that the server receives will instead be:

$$\left\{ \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) \right\}_{i=1}^{n}$$

When the server performs the same inversion computation as Equation (6) on this summed gradient, it will reconstruct

$$\left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right)^{-1} \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i} \right)$$

which is proportional to a linear combination of the samples that activated neuron i. The coefficient for each sample in the linear combination depends on how much the sample contributes to the loss $\mathcal{L}$. Reconstructing such a combination may not be able to reveal the content of each individual input sample, thereby hindering the impact of the attack.

To circumvent the problem of summed gradients, the CAH attack proposed by [17] chooses the parameters for (W, b) that maximize the likelihood that each attacked neuron is activated by only one sample in the batch. The rationale behind this is that if i is activated only by one data point x_t, then

$$\left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) = \left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i} \right)$$

since $\frac{\partial \mathcal{L}_j}{\partial W_i} = 0$ and $\frac{\partial \mathcal{L}_j}{\partial b_i} = 0$ for data points $x_j (j \neq t)$ that do not activate the neuron i. After obtaining $\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)$, the server can reconstruct x_t by Equation (6).

On the other hand, [18] proposes the RTF attack in which the reconstruction can be carried out by considering the difference between two successive neurons' gradients, with respect to some specific parameters (W, b). Specifically, the server can strategically choose (W, b) so that, given the gradients

$$\left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right)$$

of neuron i and

$$\left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_{i+1}}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_{i+1}} \right)$$

of neuron $i + 1$, the difference between them can reveal the gradients $\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)$ of a particular sample x_t that activates neuron i. With this, Equation (6) can perfectly reconstruct that sample x_t.

From this analysis, we can observe the *underlying principle of these attacks*: as long as the gradients $\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)$ of one individual sample x_t can be extracted from the summed gradients

$$\left\{ \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) \right\}_{i=1}^{n}$$

with $\frac{\partial \mathcal{L}_t}{\partial b_i} \neq 0$, that sample x_t can be perfectly reconstructed by gradient inversion via Equation (6). Therefore, the attack strategies specifically involve choosing (W, b) that optimizes the chance of extraction, thus improving reconstruction quality.

Defense Intuition. By this principle, to effectively defend against such attacks, it is essential to prevent the leaking of any individual data points' gradients from the summed gradients, regardless of how the parameters (W, b) are chosen. With this in mind, we establish the following proposition:

Proposition 1. *Given a sample $x_t \in \mathcal{D}$, if there exists an $x'_t \in \mathcal{D}$ such that x_t and x'_t activate the same set of neurons in the malicious layer, then the adversary cannot extract*

$$\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i} \right)$$

with $\frac{\partial \mathcal{L}_t}{\partial b_i} \neq 0$ from

$$\left\{ \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) \right\}_{i=1}^{n}$$

Proof. There are two cases in which the adversary is able to obtain $\left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i}\right)$ with $\frac{\partial \mathcal{L}_t}{\partial b_i} \neq 0$ from

$$\left\{ \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) \right\}_{i=1}^{n}$$

(1) There exists an $i \in \{1, 2, ..., n\}$ s.t.

$$\left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) = \left(\frac{\partial \mathcal{L}_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i} \right)$$

This means that the neuron i is activated only by x_t, thus contradicting the fact that x_t and x'_t activate the same set of neurons.

(2) There exists a subset $D \subseteq \mathcal{D} \setminus x_t$ such that the adversary can determine

$$\left(\sum_{x_j \in D} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{x_j \in D} \frac{\partial \mathcal{L}_j}{\partial b_i} \right)$$

and

$$\left(\sum_{x_j \in D \cup x_t} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{x_j \in D \cup x_t} \frac{\partial \mathcal{L}_j}{\partial b_i} \right)$$

from

$$\left\{ \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) \right\}_{i=1}^{n}$$

To be able to obtain

$$\left(\sum_{x_j \in D \cup x_t} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{x_j \in D \cup x_t} \frac{\partial \mathcal{L}_j}{\partial b_i} \right)$$

it must be that x_t activates neuron i. This also means that x'_t activates neuron i (since x_t and x'_t activate the same set of neurons) and that $x'_t \in D$. But in order to get

$$\left(\sum_{x_j \in D} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{x_j \in D} \frac{\partial \mathcal{L}_j}{\partial b_i} \right)$$

there must be a neuron that is activated by samples in D, which includes x'_t, and is not activated by x_t. This contradicts the fact that x_t and x'_t activate the same set of neurons. □

Intuitively, suppose that for every $x_t \in \mathcal{D}$, we find a data point x'_t such that x_t and x'_t always activate the same set of neurons, and then we add x'_t to $\mathcal{D}$. From Proposition 1, it can be inferred that the best that the attacker can do is extracting

$$\left(\frac{\partial \mathcal{L}_t}{\partial W_i} + \frac{\partial \mathcal{L}'_t}{\partial W_i}, \frac{\partial \mathcal{L}_t}{\partial b_i} + \frac{\partial \mathcal{L}'_t}{\partial b_i} \right)$$

from the summed gradients

$$\left\{ \left(\sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial W_i}, \sum_{j=1}^{B} \frac{\partial \mathcal{L}_j}{\partial b_i} \right) \right\}_{i=1}^{n}$$

Hence, it could only reconstruct a linear combination of x_t and x'_t. If the linear combination does not reveal the content of x_t, then the proposed defense is successful.

B. *Image Augmentation as a Defense*

From the previous attack principle and defense intuition, we devise a robust defense mechanism as follows. For every $x_t \in \mathcal{D}$, we find a set of data points X'_t such that x_t and every $x' \in X'_t$ activate the same set of neurons. Then, we construct a new local training dataset:

$$\mathcal{D}' = \mathcal{D} \cup \bigcup_{t=1}^{B} X'_t \qquad (7)$$

If $\mathcal{D}$ is labeled then the data points in X'_t are given the same label as x_t. The user will use $\mathcal{D}'$ instead of $\mathcal{D}$ for the FL process, so that an *active* reconstruction attack can only reconstruct a linear combination of x_t and $x' \in X'_t$. This mechanism is illustrated in Figure 1. The defense is considered effective if it satisfies two conditions: (1) using $\mathcal{D}'$ does not heavily reduce the training performance, and (2) a linear combination of x_t and $x' \in X'_t$ does not reveal the content of x_t.

To find X'_t that activates the same set of neurons as x_t, we propose using image augmentation [27] where X'_t contains the transformations of x_t, such as rotation, shearing, or flipping. As noted in [28], image augmentation can be used to teach a model about invariances in the data domain. For that reason, training with image augmentation makes the model invariant to the transformations of images. In other words, the model should exhibit similar behavior (i.e., similar patterns of neuron activations) given different transformations of an image. As a result, x_t and images in X'_t are *more likely* to activate the same set of neurons. Our experiments in Section 4, especially Figures 7-12, further support this claim by showing that the reconstructed image is a linear combination of the transformed and the original, which is caused by x_t and X'_t activating the same set of neurons.

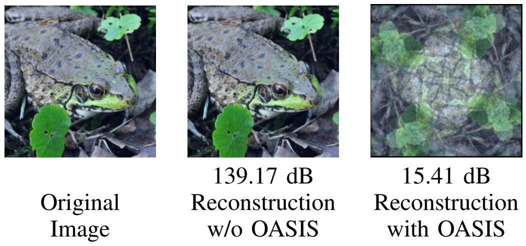

Original Image
139.17 dB Reconstruction w/o OASIS
15.41 dB Reconstruction with OASIS

Figure 2: Example visual representation of PSNR values. Images with lower PSNR *tend to* have worse reconstruction quality compared to images with higher PSNR.

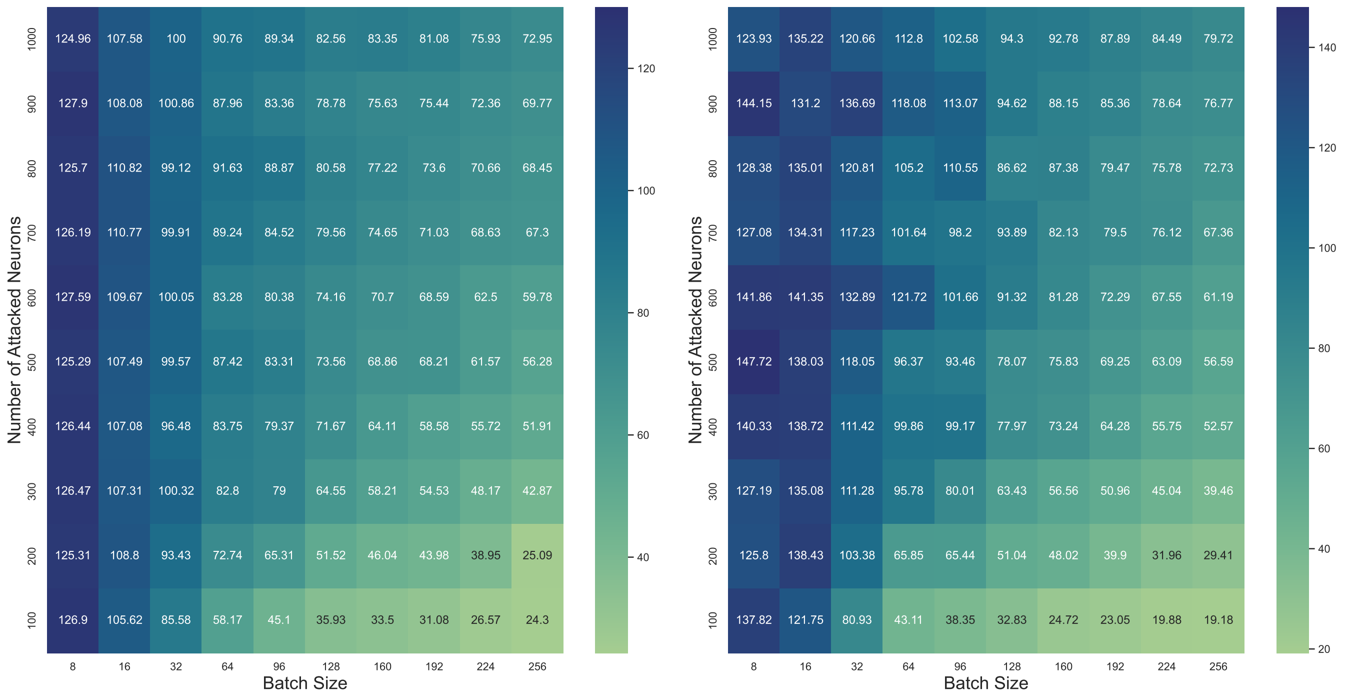
Furthermore, using image augmentation as a defense also satisfies the above-mentioned two conditions. First, using image augmentation maintains the training performance as it was originally designed to improve model generalization and reduce overfitting. Second, as we shall demonstrate in Section 4, a linear combination of an image x_t and its transformations yields an unrecognizable image, thereby protecting the original content of x_t.

IV. Experimental Analysis

This section evaluates the performance of our defense with various experiments to shed light on how OASIS can *offset* state-of-the-art *active* reconstruction attacks while still maintaining the model training performance.

A. *Experimental Settings*

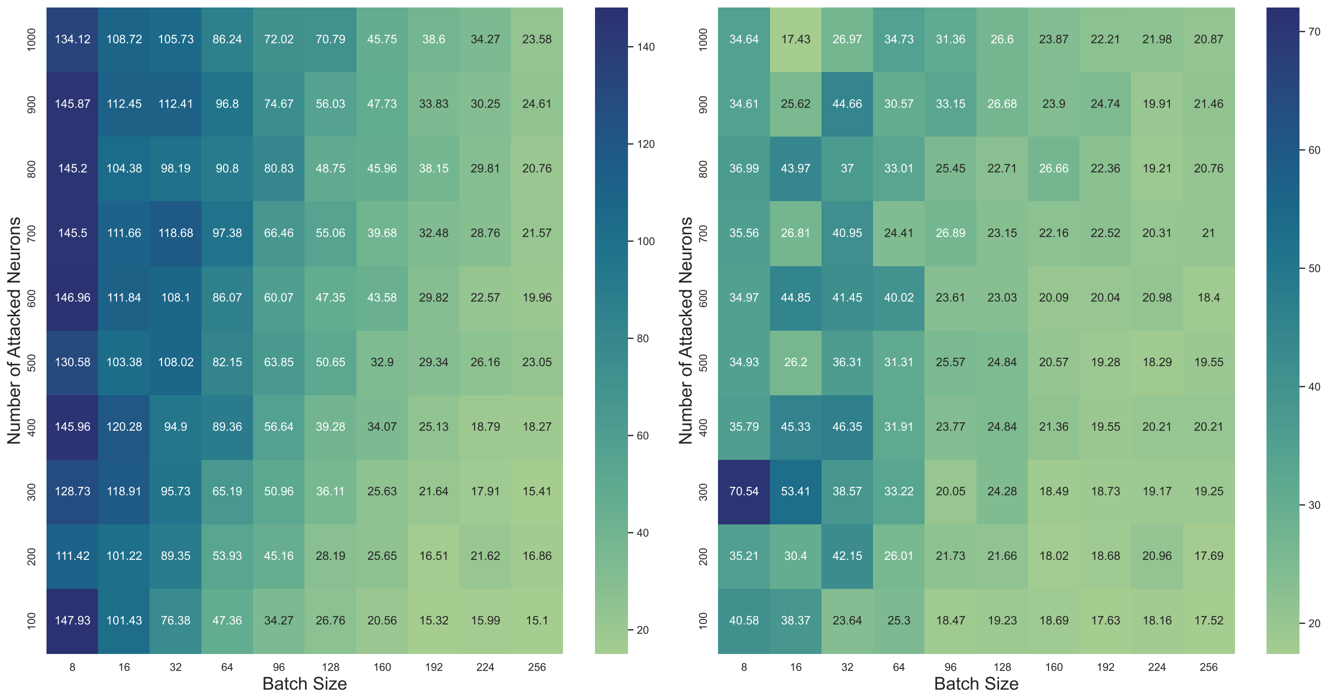
We conduct two state-of-the-art *active* reconstruction attacks, namely *Robbing the Fed (RTF)* [18] and *Curious Abandon Honesty (CAH)* [17], against our OASIS defense on two datasets ImageNet [31] and CIFAR100 [32]. For these attacks, we adopt the implementation from https://github.com/JonasGeiping/breaching. To capture how OASIS mitigates the success rate of the attacks, similar to previous work [18], [30], we use the *Peak Signal-to-Noise Ratio (PSNR)* value to measure the quality of a reconstructed image with respect to the original image. Higher PSNR values indicate better reconstruction quality, thus higher attack success rates. Figure 2 illustrates a visual representation of PSNR values. Our goal is to minimize the PSNR values of reconstructed images. Furthermore, we visually compare the reconstructed images when using OASIS against their respective original images to demonstrate how OASIS protects the content of the dataset. Finally, we measure model performance for each augmentation method on each dataset. OASIS is expected to impose a negligible trade-off on the performance of training models.



(a) ImageNet

(b) CIFAR100

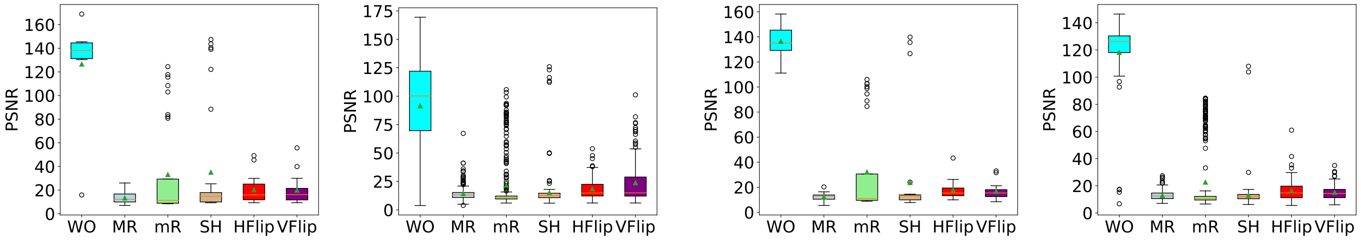
Figure 3: Average PSNR over the images reconstructed by the RTF attack w.r.t the batch size and the number of attacked neurons on ImageNet and CIFAR100.



(a) ImageNet

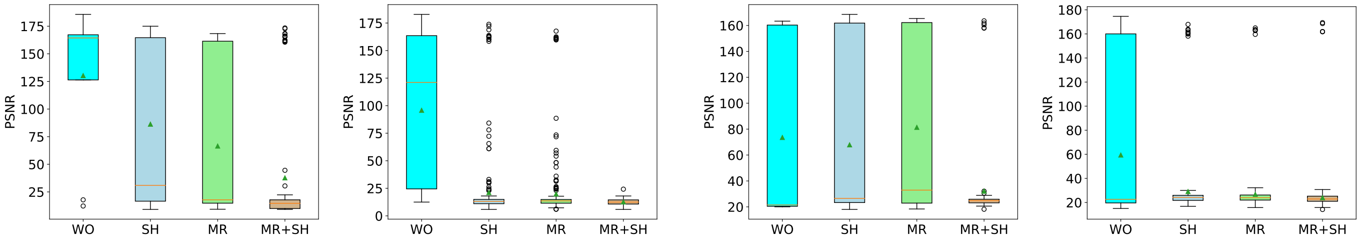
(b) CIFAR100

Figure 4: Average PSNR over the images reconstructed by the CAH attack w.r.t the batch size and the number of attacked neurons on ImageNet and CIFAR100.



(a) ImageNet. *Left:* $(B, n) = (8, 900)$. *Right:* $(B, n) = (64, 800)$ (b) CIFAR100. *Left:* $(B, n) = (8, 500)$. *Right:* $(B, n) = (64, 600)$

Figure 5: PSNR values of images reconstructed by the RTF attack w.r.t different transformations and different batch sizes on ImageNet and CIFAR100. The green triangle denotes the average PSNR over all reconstructed images. *(WO = Without OASIS, MR = Major Rotation, mR = Minor Rotation, SH = Shearing, HFlip = Horizontal Flip, and VFlip = Vertical Flip)*



(a) ImageNet. *Left:* $(B, n) = (8, 100)$. *Right:* $(B, n) = (64, 700)$ (b) CIFAR100. *Left:* $(B, n) = (8, 300)$. *Right:* $(B, n) = (64, 600)$

Figure 6: PSNR values of images reconstructed by the CAH attack w.r.t different transformations and different batch sizes on ImageNet and CIFAR100. The green triangle denotes the average PSNR over all reconstructed images. *(WO = Without OASIS, SH = Shearing, MR = Major Rotation, and MR + SH = Major Rotation + Shearing)*

For a fair evaluation, the attacks are first configured to have the highest success rate. As discussed in the threat model in Section III-A, the malicious layer is appended right after the input layer. Furthermore, the attack performance depends on the number of attacked neurons n, and the batch size B. Generally, it is straightforward that the reconstruction attacks perform worse with larger batch sizes. We experiment with two batch sizes: $B = 8$ for evaluating against strong attacks, and $B = 64$ for a more realistic training configuration. We conduct preliminary experiments to find the hyperparameters that result in the strongest attacks. Specifically, we test the attacks with various batch sizes and numbers of attacked neurons, and report the average PSNR value over the images reconstructed by RTF and CAH in Figures 3 and 4, respectively. As previously stated, the reconstruction attacks perform worse with larger batch sizes, and that behavior is illustrated in Figures 3 and 4. For each batch size, we choose the number of attacked neurons n that yields the highest average PSNR.

As can be seen in Figure 3, the RTF attack's optimal settings for ImageNet with a batch of 8 occur with 900 attacked neurons yielding an average PSNR value of 127.9 dB. The optimal settings for a batch of 64 occur with 800 attacked neurons yielding an average PSNR value of 91.63 dB. For CIFAR100, we see the optimal settings for a batch of 8 and 64 are 500 and 600 attacked neurons yielding average PSNR values of 147.72 dB and 121.72 dB, respectively.

We test for the optimal settings of the CAH attack in a similar manner in Figure 4. For ImageNet, a batch of 8 with 100 attacked neurons produces an average PSNR value of 147.93 dB and a batch of 64 with 700 attacked neurons produces an average PSNR value of 97.38 dB. CIFAR100 was treated the same as before. A batch of 8 along with 300 attacked neurons results in an average PSNR value of 70.54 dB while a batch of 64 with 600 attacked neurons yields an average PSNR value of 40.02 dB.

OASIS Implementation. As for constructing $\mathcal{D}'$ in Equation 7, we test with various methods of image augmentation, including rotation, shearing, and flipping, and observe how each of them impacts the performance of OASIS. We describe how the transformations are implemented as follows. In the case of major rotation, every image in $\mathcal{D}$ was rotated three different times at angles of 90°, 180°, and 270°, following Equation 2, to generate three transformed images for $\mathcal{D}'$. For minor rotation, we rotate each image three different times at angles of $30°, 45°, 60°$.

For flipping, we conduct both horizontal and vertical flipping using Equations 3 and 4, respectively. In regard to shearing, we follow Equation 5 and shear every image in $\mathcal{D}$ with three different shear factors of 0.55, 1.0, and 0.9 to generate three transformed images for $\mathcal{D}'$. Each transformation is implemented with the official PyTorch Vision library[1] and the Kornia library[2].

[1] *https://github.com/pytorch/vision.git*

[2] *https://github.com/kornia/kornia.git*

B. *OASIS Defensive Performance*

Figure 5 depicts the effectiveness of our defense in regard to reducing the reconstruction quality of the RTF attack. Five transformations are used in this experiment, and it can be seen from Figure 5 that each of them substantially reduce the PSNR values of the reconstructed images across all testing scenarios. Specifically, without OASIS, most of the images reconstructed by the RTF attack have PSNR ranging from 130 dB to 145 dB at batch size 8, indicating perfect reconstruction. Major rotation is the most robust transformation such that by adding rotations at major angles to each image in $\mathcal{D}$, the resulting reconstruction by RTF only yields PSNR from 15 dB to 20 dB. Thus, the content of each image in $\mathcal{D}$ remains hidden.

To understand how the major rotation can invalidate the RTF attack, we note that the activation of attacked neurons in RTF depends on a scalar quantity of the input, such as the average of pixel values [18]. Major rotation imposes minimal change to this quantity (it does not change the average of pixel values). Hence, using this transformation for building X_t' ensures that x_t and X_t' activate the same set of neurons, for all $x_t \in \mathcal{D}$. Furthermore, as we shall see in Section IV-C, a linear combination of an image and its rotations yields an unrecognizable image. We also note that flipping does not change the average of pixel values either, however, this transformation does not necessarily result in unrecognizable reconstruction (as shown later in Section IV-C), thus its PSNR is slightly higher than that of major rotation.

Figure 6 illustrates the performance of OASIS against the CAH attack. With batch size 64, we observe a similar result as the previous experiment against RTF in which the major rotation keeps the PSNR of reconstructed images low. However, for batch size 8, the major rotation fails to prevent many images from being perfectly reconstructed. The same behavior is exhibited through shearing. The core issue here is that these transformations alone are not enough to prevent several $x_t \in \mathcal{D}$ from being the sole activation of certain attacked neurons in CAH, thus the content of those x_t is revealed through reconstruction.

To tackle this issue, we attempt to integrate multiple transformations to increase the likelihood that x_t and some images in X_t' activate the same set of neurons in the malicious layer. In other words, the set X_t' is constructed by more than one transformation. As shown in Figure 6, we experiment with integrating the two most robust transformations: major rotation and shearing. This integration is able to render the reconstruction by CAH unrecognizable with low PSNR. Specifically, with ImageNet (Figure 6a), it significantly decreases the PSNR of reconstructed images from above 125 dB to below 25 dB. The same effect is also exhibited with CIFAR100 (Figure 6b).

C. *Visual Reconstructions*

We visually demonstrate the resulting reconstruction from the attacks. The goal is to show that, with our OASIS defense, the attacks indeed reconstruct a linear combination of an image and its transformations, effectively confirming the claims in Section III-B. Moreover, it shows that the linear combination yields the reconstructed image unrecognizable, protecting the content of the input images.

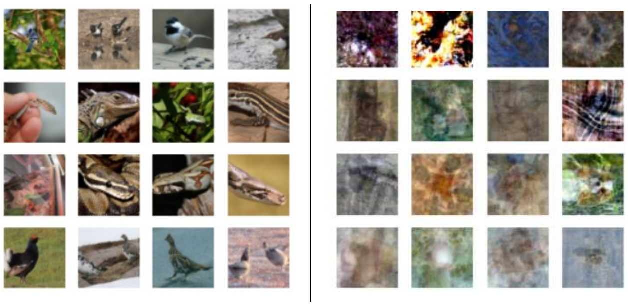


Figure 7: *Left:* Raw input images. *Right:* Reconstruction result with major rotation.

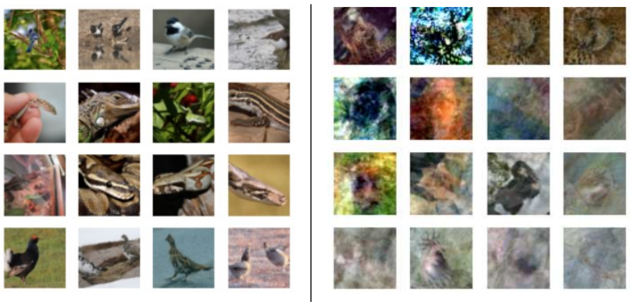


Figure 8: *Left:* Raw input images. *Right:* Reconstruction result with minor rotation.

Rotation. Figures 7 and 8 illustrate the reconstruction from the RTF attack with major rotation and minor rotation being used as augmentations from OASIS, respectively. We can see that the reconstructed images are an overlap of the original images and their respective rotations. As previously discussed in Figure 5, major rotation is the most effective transformation with the lowest PSNR for reconstruction, and we can see in Figure 7 that the reconstructed images are unrecognizable. Although the reconstruction with minor rotation has higher PSNR, Figure 8 shows that it is still challenging to discern the original images from the reconstructed ones.

Shearing. Figure 9 presents the reconstruction from the RTF attack with shearing being used as augmentation for OASIS.

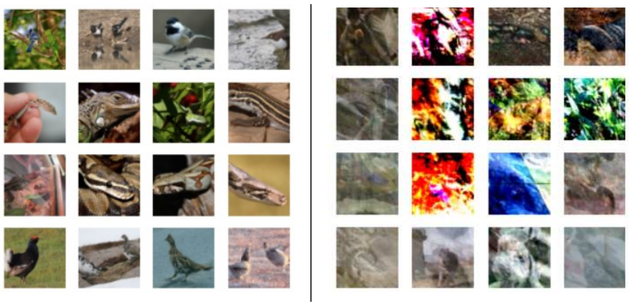


Figure 9: *Left:* Raw input images. *Right:* Reconstruction result with shearing.

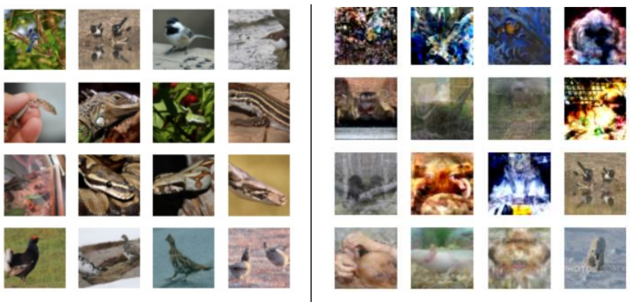


Figure 10: *Left:* Raw input images. *Right:* Reconstruction result with horizontal flipping.

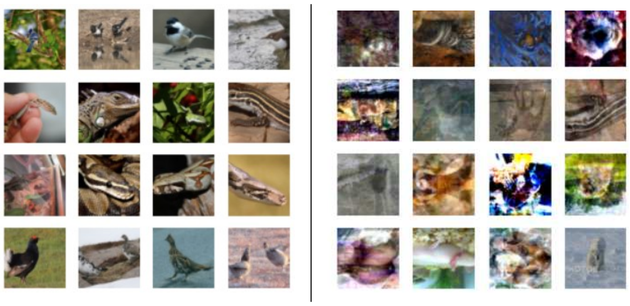


Figure 11: *Left:* Raw input images. *Right:* Reconstruction result with vertical flipping.

We can see that the original image and its sheared version overlap one another in the reconstruction, thereby hindering the attacker from making out the original. This also explains the low PSNR of shearing in Figure 5.

Flipping. Figures 10 and 11 illustrate the reconstruction from the RTF attack with horizontal flipping and vertical flipping being used as augmentation for OASIS, respectively. We can see that they did not defend as well against the attack compared to rotation and shearing. A linear combination of an original image and its horizontally or vertically flipped version only generates a reflection of the original, thus the original image is still revealed in the reconstruction. Figures 10 and 11 show that some images are reflected in the reconstruction. This means that flipping, when used alone, is not the best suited transformation to defend against this class of attacks. However, using flipping in combination with a strong transformation such as rotation or shearing may yield better results.

Integrating Major Rotation and Shearing. As previously discussed in Figure 6, an integration of multiple transformations is needed to counter the CAH attack. Figure 12 illustrates the reconstruction from CAH when both major rotation and shearing are used in OASIS. It can be seen that all the reconstructed images are unrecognizable and it is impossible to identify any original image from them. This behavior is consistent with the results in Figure 6.

In summary, major rotation and an integration of major rotation and shearing result in the strongest defense against the RTF and CAH attacks, respectively. Additionally, OASIS has been shown to be scalable as it maintains low PSNR on reconstructed images for both small and large batch sizes. We further note that it is not trivial to extract the original image from such an overlap of multiple transformed images without any prior knowledge about certain characteristics of the original image. Although the server might know about certain augmentations being used as a defense, it does not know the specific parameters of the transformations (e.g., shearing intensity). Previous research has shown that, even with a mild blurry image, it is very challenging to practically reconstruct the original image without knowing the blurring kernel and padding [33], while our defense uses far more complicated and multiple transformations.

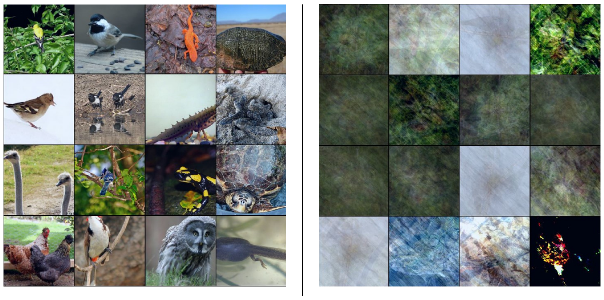


Figure 12: *Left:* Raw input images. *Right:* Reconstruction result with an integration of major rotation and shearing.

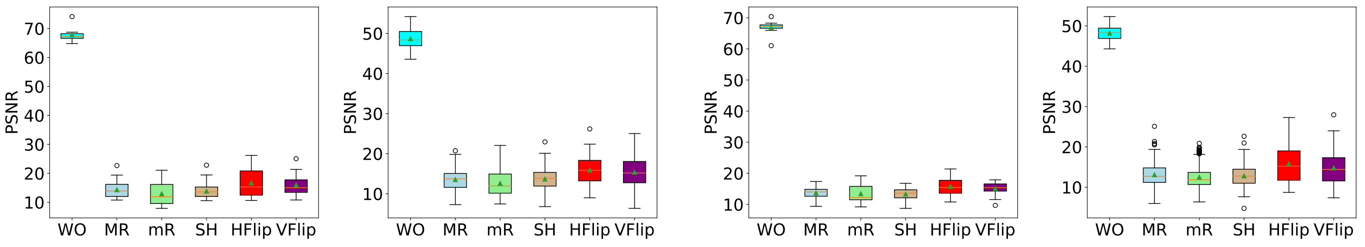
D. Gradient Inversion Attack on Linear Models.

In addition to the RTF [18] and CAH [17] attacks, we evaluate our OASIS defense against a reconstruction attack on linear models that was discussed in [18], [30]. The attack assumes a very restrictive setting where the model is a single-layer and is trained with a logistic regression loss function. Furthermore, the images in each training batch $\mathcal{D}$ are assumed to have unique labels. As users upload their local model updates, the server simply inverts the gradient of each neuron to reconstruct the training images.

Figure 13 illustrates the effectiveness of our OASIS defense in reducing the reconstruction quality of this attack. Since this is a single-layer model, adding transformed images to the training batch guarantees that x_t and X'_t activate the same neuron, for all $x_t \in \mathcal{D}$. Hence, each reconstructed image will be a linear combination of x_t and X'_t. Moreover, such a linear combination hides the content of the original image (as discussed in Section IV). Therefore, Figure 13 shows that all five transformations yield reconstruction with low PSNR for both datasets and both batch sizes. We can also see that rotation and shearing have better defensive performance than flipping, corroborating our findings in Section IV.

E. Impact of OASIS on Model Performance

We measure the effect of using OASIS on the training models as it alters the input dataset for training. For this experiment, we train ResNet-18 models [34] on ImageNet and CIFAR100, then compare the final testing accuracies when training with and without OASIS. In particular, when training with OASIS, we replace each training batch $\mathcal{D}$ with $\mathcal{D}'$ as



(a) ImageNet. **Left:** $B = 8$. **Right:** $B = 64$

(b) CIFAR100. **Left:** $B = 8$. **Right:** $B = 64$

Figure 13: PSNR values of images reconstructed by the gradient inversion attack on linear models w.r.t different transformations and different batch sizes on ImageNet and CIFAR100. The green triangle denotes the average PSNR over all reconstructed images. *(WO = Without OASIS, MR = Major Rotation, mR = Minor Rotation, SH = Shearing, HFlip = Horizontal Flip, and VFlip = Vertical Flip)*

mentioned in Section III-B. The result for each transformation is shown in Table I.

For ImageNet [31], we extract a subset of 10 classes: tench, English springer, cassette player, chain saw, church, French horn, garbage truck, gas pump, golf ball, and parachute[3]. Then, we evaluate the model performance on classifying those 10 classes. Using our ResNet-18 architecture, we train for 100 epochs with an Adam optimizer at a learning rate of 0.001 and weight decay of 10^{-5}.

With regard to CIFAR100 [32], we use its original classification task with 100 classes. Again, using our ResNet-18 architecture, we train for 120 epochs with an Adam optimizer at a learning rate of 0.001 and weight decay of 10^{-2}.

TABLE I: Comparing model accuracy (%) when training with and without OASIS

Transformation	Dataset	
	ImageNet	CIFAR100
Major Rotation	92.6	74.3
Minor Rotation	92.6	74.1
Shearing	95.4	73.7
Horizontal Flip	94.0	75.1
Vertical Flip	94.8	74.3
Major Rotation + Shearing	90.9	74.6
Without OASIS	94.8	75.2

Across all the transformations, OASIS does not impose any major degradation on the model accuracy. The accuracy is still maintained over 90% on ImageNet, and drops *at most* 1.5% on CIFAR100. The reason for this is that image augmentation methods are originally developed for improving the generalization and reducing overfitness of ML models. From this, the claims made in Section IV-A are confirmed.

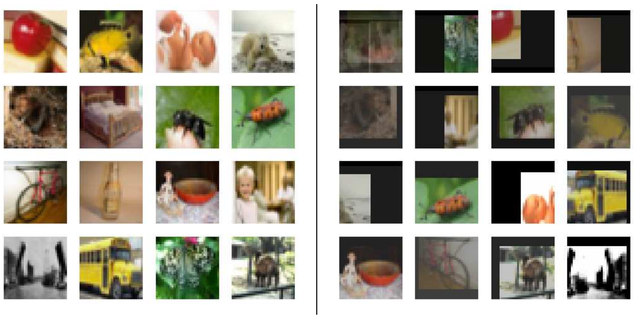


Figure 14: Reconstruction result of RTF against the defense in [41]. The content of the original images is revealed in the reconstruction. *Left:* Raw input images. *Right:* Reconstruction.

V. Related Work

Data Reconstruction Attacks. Reconstruction attacks have been one of the main topics of interest in ML security and privacy. Over the decade, various kinds of reconstruction attacks have been proposed, including class-wise representation-based attacks [35]–[37] and optimization-based attacks [38]–[40]. However, in the context of FL, most of these attacks are not able to exploit the full capability of dishonest servers. Recent work [17], [18] devises a new class of *active* reconstruction attacks that has been shown to significantly outperform prior attacks by having the dishonest server manipulate the global model parameters to its advantage. For that reason, this new class of attack remains a critical and practical threat for FL. Our work focuses on devising a general defense that effectively protects user data against these attacks. From analyzing the underlying principle of gradient inversion, our defense OASIS is designed to minimize reconstruction quality.

Current Defenses. Presently, there is no existing defense that can defend against *active* reconstruction attacks via *dishonest servers*. In general, previous defenses utilize a threat model with an honest-but-curious server that is substantially weaker than our threat model which includes an actively dishonest server. Several defense mechanisms have been proposed to tackle data reconstruction attacks in general, but they remain ineffective in countering the versions presented in this paper. Through gradient compression and sparsification methods, the work in [37], [38] *pruned* gradients with negligible magnitudes to zero. Nonetheless, even in a case where the majority of the gradients are pruned, data extracted is still recognizable [17].

Gao et al. [41] leverage image augmentation in their proposed defense, but it can only tackle optimization-based

[3] https://github.com/fastai/imagenette

attacks. In particular, the defense replaces each image in the dataset with a transformed image so that the objective function of the attacks becomes more difficult to solve. However, it fails to counter the *active* reconstruction attacks since their principle (Section III-A) still applies: if an attacked neuron is activated only by one transformed image, the image would be reconstructed. To support this claim, we conduct an experiment in which we launch the RTF attack [18] against this defense and illustrate the resulting reconstruction in Figure 14 (we adopt the implementation of [41] from https://github.com/gaow0007/ATSPrivacy). As can be seen, the reconstruction reveals the content of the original input images. Therefore, defenses against optimization-based reconstruction attacks are not robust against these *active* reconstruction attacks if they do not address the attack principle of gradient inversion.

In [17], [18], the authors evaluate the use of DP as a defense, and show that it imposes a major degradation on the model accuracy and the reconstructed images are still recognizable. Our OASIS defense is proven to effectively counter this new class of attacks as it tackles the core attack principle. Moreover, OASIS imposes minimal impact on model performance.

VI. Conclusion

In this paper, we have revealed the key principle behind active reconstruction attacks in Federated Learning (FL) and have theoretically shown how to tackle this class of attacks. With machine learning foundations in data preprocessing, we have proposed OASIS, a novel method to augment images in a way such that an *actively* dishonest server is unable to memorize individual gradient parameters, but a linear combination of an image and its augmented counterparts. In doing so, we *offset* the active reconstruction attacks, rendering reconstructions unrecognizable. To address FL's promise of maintaining model performance, we also demonstrate that the expansion of a labeled dataset through augmentation preserves and, in some cases, improves model performance. From our evaluation, OASIS stands as a general, viable, and scalable solution to truly promote and reinforce the guarantees of FL. Although the use of image augmentation makes OASIS confined to the image domain, we note that the attack principle that we uncover in Section III-A is not limited to any data types. Future work will focus on finding alternative methods besides image augmentation to implement an effective defense for tabular and textual data.

Acknowledgement

This work was supported in part by the National Science Foundation under grants CNS-2140477, CNS-1935923, and ITE-2235678, and by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government (MOTIE) (RS-2023-00303559, Study on developing cyber-physical attack response system and security management system to maximize real-time distributed resource availability).

This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

References

[1] W. Moore and S. Frye, "Review of hipaa, part 1: history, protected health information, and privacy and security rules," *Journal of nuclear medicine technology*, vol. 47, no. 4, pp. 269–272, 2019.

[2] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, no. 3152676, pp. 10–5555, 2017.

[3] G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, "Secure, privacy-preserving and federated machine learning in medical imaging," *Nature Machine Intelligence*, vol. 2, no. 6, pp. 305–311, 2020.

[4] M. Adnan, S. Kalra, J. C. Cresswell, G. W. Taylor, and H. R. Tizhoosh, "Federated learning and differential privacy for medical image analysis," *Scientific reports*, vol. 12, no. 1, p. 1953, 2022.

[5] X. Li, Y. Gu, N. Dvornek, L. H. Staib, P. Ventola, and J. S. Duncan, "Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: Abide results," *Medical Image Analysis*, vol. 65, p. 101765, 2020.

[6] S. Silva, B. A. Gutman, E. Romero, P. M. Thompson, A. Altmann, and M. Lorenzi, "Federated learning in distributed medical databases: Meta-analysis of large-scale subcortical brain data," in *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*, pp. 270–274, IEEE, 2019.

[7] M. Jiang, Z. Wang, and Q. Dou, "Harmofl: Harmonizing local and global drifts in federated learning on heterogeneous medical images," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, pp. 1087–1095, 2022.

[8] D. Stripelis, J. L. Ambite, P. Lam, and P. Thompson, "Scaling neuroscience research using federated learning," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pp. 1191–1195, IEEE, 2021.

[9] B. Hu, Y. Gao, L. Liu, and H. Ma, "Federated region-learning: An edge computing based framework for urban environment sensing," in *2018 ieee global communications conference (globecom)*, pp. 1–7, IEEE, 2018.

[10] Y. Liu, J. Nie, X. Li, S. H. Ahmed, W. Y. B. Lim, and C. Miao, "Federated learning in the sky: Aerial-ground air quality sensing framework with uav swarms," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9827–9837, 2020.

[11] P. Tam, S. Math, C. Nam, and S. Kim, "Adaptive resource optimized edge federated learning in real-time image sensing classifications," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 10929–10940, 2021.

[12] Y. Gao, L. Liu, B. Hu, T. Lei, and H. Ma, "Federated region-learning for environment sensing in edge computing system," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 4, pp. 2192–2204, 2020.

[13] T. T. Huong, T. P. Bac, D. M. Long, T. D. Luong, N. M. Dan, B. D. Thang, K. P. Tran, *et al.*, "Detecting cyberattacks using anomaly detection in industrial control systems: A federated learning approach," *Computers in Industry*, vol. 132, p. 103509, 2021.

[14] T. Zeng, O. Semiariy, M. Chen, W. Saad, and M. Bennis, "Federated learning on the road autonomous controller design for connected and autonomous vehicles," *IEEE Transactions on Wireless Communications*, 2022.

[15] I. Kevin, K. Wang, X. Zhou, W. Liang, Z. Yan, and J. She, "Federated transfer learning based cross-domain prediction for smart manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 6, pp. 4088–4096, 2021.

[16] T. Nguyen and M. T. Thai, "Preserving privacy and security in federated learning," *IEEE/ACM Transactions on Networking*, vol. 32, no. 1, pp. 833–843, 2024.

[17] F. Boenisch, A. Dziedzic, R. Schuster, A. S. Shamsabadi, I. Shumailov, and N. Papernot, "When the curious abandon honesty: Federated learning is not private," in *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*, pp. 175–199, IEEE, 2023.

[18] L. H. Fowl, J. Geiping, W. Czaja, M. Goldblum, and T. Goldstein, "Robbing the fed: Directly obtaining private data in federated learning with modified models," in *International Conference on Learning Representations*, 2021.

[19] T. Nguyen, P. Thai, J. Tre'R, T. N. Dinh, and M. T. Thai, "Blockchain-based secure client selection in federated learning," in *2022 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, pp. 1–9, IEEE, 2022.

[20] D. Pasquini, D. Francati, and G. Ateniese, "Eluding secure aggregation in federated learning via model inconsistency," in *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2429–2443, 2022.

[21] J. Tre'R and M. T. Thai, "Privacy analysis of federated learning via dishonest servers," in *2023 IEEE 9th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, pp. 24–29, IEEE, 2023.

[22] T. Nguyen, P. Lai, K. Tran, N. Phan, and M. T. Thai, "Active Membership Inference Attack under Local Differential Privacy in Federated Learning," in *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, vol. 206, pp. 5714–5730, PMLR, Apr. 2023. ISSN: 2640-3498.

[23] M. Vu, T. Nguyen, T. Jeter, and M. T. Thai, "Analysis of privacy leakage in federated large language models," in *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics* (S. Dasgupta, S. Mandt, and Y. Li, eds.), vol. 238 of *Proceedings of Machine Learning Research*, pp. 1423–1431, PMLR, 02–04 May 2024.

[24] M. N. Vu, J. Tre'R, R. Alharbi, and M. T. Thai, "Active data reconstruction attacks in vertical federated learning," in *2023 IEEE International Conference on Big Data (BigData)*, pp. 1374–1379, IEEE, 2023.

[25] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

[26] D. Ho, E. Liang, X. Chen, I. Stoica, and P. Abbeel, "Population based augmentation: Efficient learning of augmentation policy schedules," in *International Conference on Machine Learning*, pp. 2731–2741, PMLR, 2019.

[27] D. Yarats, I. Kostrikov, and R. Fergus, "Image augmentation is all you need: Regularizing deep reinforcement learning from pixels," in *International Conference on Learning Representations*, 2020.

[28] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation strategies from data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 113–123, 2019.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems* (F. Pereira, C. Burges, L. Bottou, and K. Weinberger, eds.), vol. 25, Curran Associates, Inc., 2012.

[30] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients-how easy is it to break privacy in federated learning?," *Advances in Neural Information Processing Systems*, vol. 33, pp. 16937–16947, 2020.

[31] 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*, pp. 248–255, Ieee, 2009.

[32] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," 2009.

[33] D. Ren, K. Zhang, Q. Wang, Q. Hu, and W. Zuo, "Neural blind deconvolution using deep priors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3341–3350, 2020.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[35] 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 *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pp. 2512–2520, IEEE, 2019.

[36] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep models under the gan: information leakage from collaborative deep learning," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, pp. 603–618, 2017.

[37] J. Sun, A. Li, B. Wang, H. Yang, H. Li, and Y. Chen, "Soteria: Provable defense against privacy leakage in federated learning from representation perspective," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9311–9319, 2021.

[38] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," *Advances in neural information processing systems*, vol. 32, 2019.

[39] B. Zhao, K. R. Mopuri, and H. Bilen, "idlg: Improved deep leakage from gradients," *arXiv preprint arXiv:2001.02610*, 2020.

[40] 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*, pp. 16337–16346, 2021.

[41] W. Gao, S. Guo, T. Zhang, H. Qiu, Y. Wen, and Y. Liu, "Privacy-preserving collaborative learning with automatic transformation search," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 114–123, 2021.