From Split to Share: Private Inference with Distributed Feature Sharing

Zihan Liu[1], **Jiayi Wen**[1], **Shouhong Tan**[1], **Zhirun Zheng**[2], **Cheng Huang**[1]

[1]College of Computer Science and Artificial Intelligence, Fudan University
[2]Department of Artificial Intelligence, Ajou University

Abstract

Cloud-based Machine Learning as a Service (MLaaS) raises serious privacy concerns when handling sensitive client data. Existing Private Inference (PI) methods face a fundamental trade-off between privacy and efficiency: cryptographic approaches offer strong protection but incur high computational overhead, while efficient alternatives such as split inference expose intermediate features to inversion attacks. We propose PrivDFS, a *new paradigm* for private inference that replaces a single exposed representation with *distributed feature sharing*. PrivDFS partitions input features on the client into multiple balanced shares, which are distributed to non-colluding, non-communicating servers for independent partial inference. The client securely aggregates the servers' outputs to reconstruct the final prediction, ensuring that no single server observes sufficient information to compromise input privacy. To further strengthen privacy, we propose two key extensions: PrivDFS-AT, which uses adversarial training with a diffusion-based proxy attacker to enforce inversion-resistant feature partitioning, and PrivDFS-KD, which leverages user-specific keys to diversify partitioning policies and prevent query-based inversion generalization. Experiments on CIFAR-10 and CelebA demonstrate that PrivDFS achieves privacy comparable to deep split inference while cutting client computation by up to *100*$\times$ with no accuracy loss, and that the extensions remain robust against both diffusion-based in-distribution and adaptive attacks.

1 Introduction

Cloud-based inference, enabled by the rise of Machine Learning as a Service (MLaaS) (Ribeiro, Grolinger, and Capretz 2015), allows resource-constrained clients to access powerful AI models through the cloud. However, this deployment model exposes raw client data, such as facial images, medical scans, and other sensitive inputs, to potentially untrusted servers, creating severe privacy risks (Rao et al. 2025; Otroshi-Shahreza, Hahn, and Marcel 2024). These risks highlight the need for Private Inference (PI) techniques that can protect client inputs during inference without sacrificing accuracy (Mann et al. 2024).

Existing PI methods fall into two broad categories. Cryptographic approaches based on Homomorphic Encryption (HE) (Xu et al. 2024; Peng et al. 2023; Cheng et al. 2023) or Secure Multi-Party Computation (SMPC) (Feng et al. 2025; Diaa et al. 2024) provide strong privacy guarantees, but their heavy computational and communication overhead makes them impractical for latency-sensitive inference (Garimella et al. 2023). A more efficient alternative is split inference (Kang et al. 2017; Vepakomma et al. 2018), where the model is split between the client and the cloud so that the client uploads only an intermediate feature representation (often called smashed data) instead of the raw input (Yang et al. 2023; Bakhtiarnia et al. 2023). This improves efficiency but leaves the intermediate features directly exposed, creating a new and critical privacy vulnerability (Liu et al. 2025; Zhu et al. 2025; Singh et al. 2024).

Modern **inversion attacks** (a.k.a reconstruction attacks) exploit these exposed intermediate features to reconstruct private inputs (Fredrikson, Jha, and Ristenpart 2015). These attacks have quickly evolved from early optimization-based methods using image priors (He, Zhang, and Lee 2019; Ulyanov, Vedaldi, and Lempitsky 2018; Rudin, Osher, and Fatemi 1992) to generative-model-based inverters, including GANs (Li et al. 2023; Qiu et al. 2024) and diffusion models (Zhang et al. 2025), which achieve highly accurate reconstructions. Simple defenses that add noise to features (Mireshghallah et al. 2020; Vepakomma et al. 2020; Avella-Medina, Bradshaw, and Loh 2023) are no longer sufficient, as they still rely on a single holistic representation that advanced inverters can exploit. Consequently, **split inference remains fundamentally insecure in the face of modern inversion attacks.**

To overcome this inherent vulnerability, we propose *PrivDFS (Private Inference via Distributed Feature Sharing)*, a framework inspired by the intuition of secret sharing (Shamir 1979) but designed for lightweight, practical deployment rather than strict information-theoretic guarantees. Instead of sending a single intermediate representation, PrivDFS splits client features into multiple balanced shares and distributes them to non-colluding servers. Each server performs inference only on its share, without inter-server communication, and the client aggregates the partial results to obtain the final prediction. *By replacing one exposed representation with multiple isolated shares, PrivDFS fundamentally changes the attack surface: no server alone has enough information to mount a successful inversion. Compared with cryptographic approaches, PrivDFS provides privacy protection with much lower overhead, and unlike traditional split inference, it removes the single point of ex-*

posure by spreading information across isolated servers.

Although PrivDFS effectively blocks basic inversion attacks, its protection weakens when adversaries possess auxiliary data or can launch repeated adaptive queries. To address these stronger threats, we define *a three-level threat model* based on data access and attack capability, and strengthen PrivDFS with two complementary extensions. **PrivDFS-AT** leverages adversarial training with a diffusion-model proxy attacker (Wang et al. 2024; Zhang, Rao, and Agrawala 2023; Zhang et al. 2025) to produce inversion-resistant feature partitions, while **PrivDFS-KD** diversifies feature-sharing policies across users so that an inversion model trained for one policy does not transfer to others. Together, these two extensions provide complementary protection: PrivDFS-AT hardens the shared features against learned inversion, and PrivDFS-KD prevents adaptive query attacks from transferring across users, extending the baseline framework to withstand stronger adversaries.

Contributions. Our main contributions are:

- We propose PrivDFS, a new paradigm for private inference that replaces a single exposed representation with distributed feature sharing across non-colluding servers, fundamentally reshaping the attack surface.
- PrivDFS breaks the core limitation of split inference: privacy no longer depends on a deep, expensive client model. Even with a very shallow client (one lightweight layer), PrivDFS achieves the same privacy level as deep-split baselines while *cutting client FLOPs by up to 100× and preserving accuracy.*
- To withstand stronger adversaries, we extend the framework with PrivDFS-AT (diffusion-guided adversarial training that learns inversion-resistant features) and PrivDFS-KD (user-specific sharing policies that block cross-user transfer of adaptive attacks).
- Extensive experiments on CIFAR-10 and CelebA, conducted under a three-level threat model, show that PrivDFS and its extensions deliver state-of-the-art privacy–efficiency–accuracy trade-offs. Specifically, they preserve high accuracy while substantially lowering reconstruction fidelity (SSIM↓/PSNR↓, LPIPS↑), and remain *robust against diffusion-based in-distribution attacks and adaptive query attacks.*

2 Threat Model

We consider a general inference scenario where a client computes intermediate features locally and transmits them to multiple remote servers for further processing. We assume an honest-majority setting in which fewer than half of the servers can be compromised; an adversarial server acts alone without colluding with the others and attempts to reconstruct the client's private input x from the transmitted representation $z = M_c(x)$.

Adversary Knowledge and Capability. We assume a gray-box adversary: it has full knowledge of the architecture and parameters of the server-side model M_s under its control but does not have access to the client-side parameters. The adversary can issue queries to the client-side encoder M_c and, by collecting input-representation pairs from such queries (He, Zhang, and Lee 2019), trains an inversion model $\mathcal{A}$ that approximates the inverse mapping, i.e., $\mathcal{A}(M_c(x)) \approx x$.

Adversary Resources. The primary factor distinguishing adversary resources is the quality and quantity of the auxiliary dataset available for training $\mathcal{A}$ (Yeom et al. 2018). We therefore define three levels:

- **Level 1 (Similar-distribution).** The adversary can only gather a limited dataset from a related but not identical distribution.
- **Level 2 (In-distribution).** The adversary can gather a limited dataset drawn from exactly the same distribution as the target, enabling more accurate inversion.
- **Level 3 (Unbounded).** The adversary has access to an unrestricted, large-scale in-distribution dataset, representing a worst-case upper bound.

In practice, mechanisms such as query rate limiting, anti-crawling, and anti-distillation (Juuti et al. 2019; Lee et al. 2019) substantially restrict the data that an adversary can collect, making the limited-data assumption in Levels 1 and 2 a realistic and commonly adopted threat model.

3 Proposed Framework: PrivDFS

3.1 Framework Overview

PrivDFS tackles a core challenge in private inference: *how to split features so that each server sees only fragments that are useless on their own, while their combination still enables accurate predictions without imposing heavy cost on the client.* This challenge arises because naive feature partitioning almost invariably breaks down due to three fundamental factors: *Privacy leakage:* retaining excessive information in each branch leaves the intermediate representations highly susceptible to inversion attacks; *Accuracy loss:* overly aggressive or unstructured partitioning destroys essential predictive signals and severely degrades model performance; *Efficiency-robustness trade-off:* a partitioning module that is too complex imposes prohibitive computational cost on resource-limited clients, whereas an oversimplified module can be easily modeled and inverted by an adaptive adversary. These conflicting objectives make effective feature partitioning a fundamentally non-trivial design problem. PrivDFS addresses this problem with a principled *Distributed Feature Sharing (DFS)* module that jointly determines *where to partition, what information each share retains, and how to transform the representations,* achieving a balanced integration of privacy, accuracy, and client efficiency rather than a one-dimensional trade-off.

As shown in Figure 1, inference under PrivDFS proceeds in three main stages. First, a lightweight client encoder M_c^{enc} extracts a compact representation from the raw input. This representation is then processed by the DFS module, which generates N *balanced and obfuscated* feature shares co-designed with the server models to satisfy three key properties: (i) each share is deliberately incomplete and non-invertible, ensuring that no single server can reconstruct the input; (ii) when combined, the shares recover a representation that retains task-relevant information for accurate inference; and (iii) the transformation remains lightweight for

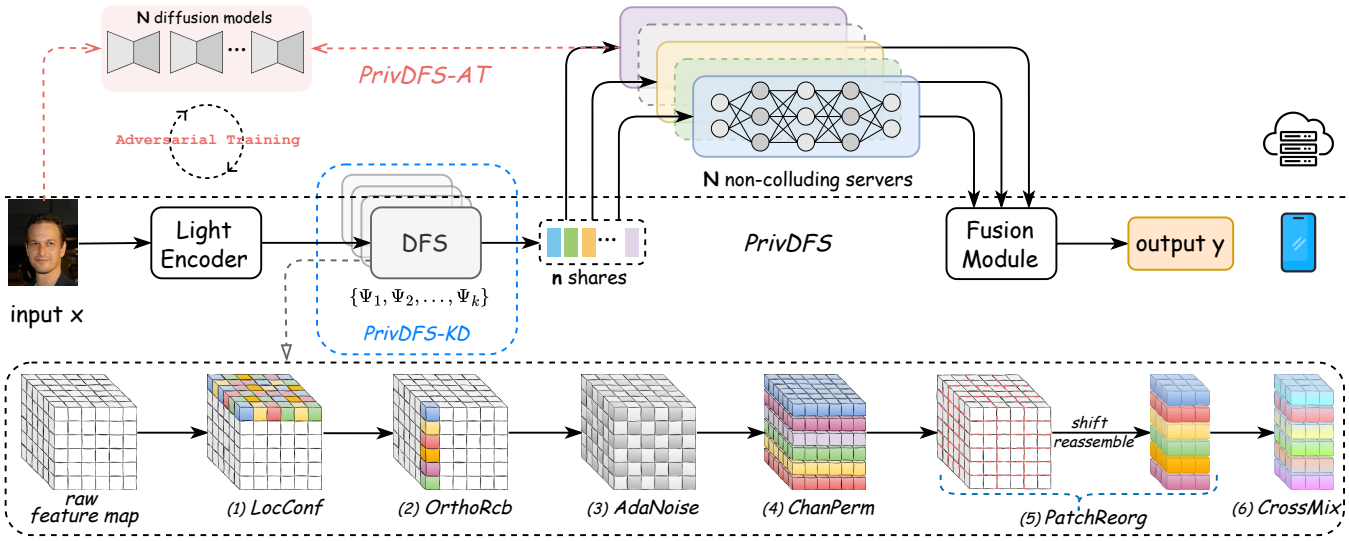

Figure 1: The overall framework of PrivDFS, PrivDFS-AT, and PrivDFS-KD.

resource-constrained clients while structured to resist inversion. Each share is transmitted to a distinct non-colluding server, where a sub-model $M_{S,i}$ performs partial inference and returns a partial output. Finally, the client aggregates these partial outputs using a lightweight fusion module M_c^{fus} to produce the final prediction. Throughout this process, the raw inputs and final predictions remain confined to the client boundary, and the servers receive only fragmented, obfuscated feature shares.

This framework enforces privacy by *distribution rather than perturbation*: information is not erased by random perturbation but deliberately fragmented across multiple servers. Each individual share is noisy and semantically incomplete, making it of little value to an adversary, while their collective fusion restores the full predictive representation needed for accurate inference. This separation of information flow fundamentally changes the attack surface compared to traditional split inference. As our experiments confirm, PrivDFS renders reconstruction and label inference attacks ineffective, whether attempted by a single server or by a small group of colluding servers, while its fused outputs achieve a new state of the art in balancing privacy, accuracy, and client efficiency.

3.2 Module Design: DFS and Fusion

DFS Module. DFS progressively transforms intermediate features into fragments that are uninformative in isolation yet predictive when fused. Rather than applying naive perturbations, DFS follows a *structured pipeline* where each stage targets a specific leakage dimension and builds upon the previous one. The pipeline addresses three complementary sources of leakage. **Spatial Structure (S):** Local textures and global layouts in convolutional features allow attackers to align activations with spatial positions or reassemble patch arrangements. **Channel Semantics (C):** Semantically coherent concepts often concentrate in a few channels, making them highly class-discriminative and exploitable for inversion or label inference. **Deterministic Patterns (D):** Strong, repetitive activations (e.g., edges or salient regions) form predictable patterns that can be learned as stable input–feature mappings. DFS mitigates these leakage pathways in sequence: suppressing local spatial cues, dissolving channel semantics, breaking deterministic patterns, and finally fragmenting context and adding redundancy for robust fusion. **(1) Localized Confusion (LocConf).** The pipeline begins by attenuating fine-grained spatial details that make features amenable to inversion. A depthwise convolution followed by a nonlinearity diffuses each activation into its neighborhood, breaking the one-to-one correspondence between activations and input patches while preserving coarse structure. **(2) Orthogonal Recombination (OrthoRcb).** Once local cues are blurred, a 1×1 orthogonal convolution globally mixes channel information, dispersing class-discriminative signals so that no single branch retains a coherent semantic representation. **(3) Adaptive Noise Injection (AdaNoise).** With spatial and channel cues weakened, the next step injects variance-scaled Gaussian noise, reducing the predictability of regular high-magnitude activations while leaving broad patterns intact. **(4) Channel Permutation (ChanPerm).** A random permutation of channels further disrupts any stable channel-semantic alignment that might remain after mixing and noise. **(5) Patch Reorganization (PatchReorg).** The fifth stage fragments global spatial context: the feature map is divided into patches that are cyclically shifted and reassembled across channels so that each branch only sees fragmented, context-free pieces. **(6) Cross-branch Mixing (CrossMix).** Finally, a lightweight mixing matrix introduces controlled redundancy, allowing the fused output to recover predictive power while keeping each individual branch incomplete. Table 1 summarizes how each stage targets spatial, channel, and deterministic leakage and how they complement one another to achieve privacy without sacrificing accuracy.

Fusion Module. Despite the heavy obfuscation applied

Table 1: Impact of each DFS stage on the three leakage dimensions (S, C, D) and on redundancy (R). ▲ = primary effect, ○ = secondary effect.

Stage	S	C	D	R	Impact
LocConf	▲	○	○	○	Reduce fine detail
OrthoRcb	○	▲	○	○	Disperse channel semantics
AdaNoise	○	○	▲	○	Randomize patterns
ChanPerm	○	▲	▲	○	Remove fixed mapping
PatchReorg	▲	○	▲	○	Disrupt global layout
CrossMix	○	○	○	▲	Enable robust fusion

to individual branches, the information distributed across them remains largely complementary with only limited redundancy. Consequently, the fusion module does not require a deep or complex architecture: a shallow multi-layer perceptron followed by a softmax is sufficient to re-aggregate these signals. This is because DFS is designed to redistribute task-relevant information rather than perturb it, i.e., each branch captures a different aspect of the input while retaining some redundancy for robustness. When these partially overlapping features are aggregated, the fused representation recovers the full discriminative structure with minimal computation. In other words, the burden of information separation lies entirely in DFS, so fusion becomes a simple low-capacity operation, avoiding additional overhead and reducing the risk of overfitting or leakage.

Together these stages ensure that *no single share contains enough coherent structure or semantics to support reliable inversion*, and their fusion can reconstruct a representation usable for prediction. This progressive design systematically dismantles exploitable information, making each share insufficient for reconstruction while keeping the client-side computation lightweight.

3.3 Hardening via Adversarial Training

PrivDFS protects sensitive inputs by fragmenting intermediate features so that each server receives only a partial representation. Its privacy basically relies on the situation that reconstructing the input from a single share is intrinsically hard in the absence of prior knowledge. However, when an adversary has access to a large amount of in-distribution data, this case may no longer hold: the adversary can learn a strong prior over the input distribution and train specialized inversion models for each branch, effectively turning an under-determined reconstruction problem into a learnable mapping. In such settings, fragmentation alone is insufficient, and the feature representations themselves should be made resistant to inversion. Thus, we introduce **PrivDFS-AT**, a hardened variant of PrivDFS that retains the same distributed feature-sharing architecture but augments it with an adversarial training regularization.

Core Idea. PrivDFS-AT introduces an adversarially regularized learning objective that anticipates reconstruction attacks. For each feature branch, a conditional diffusion model acts as an adaptive attacker trained to recover the input from that share. The defender is then optimized in a minimax formulation to preserve task-relevant information while actively suppressing reconstruction fidelity. The adversarial pressure drives the encoder to discover representations whose predictive content is decoupled from any structure that could support inversion, yielding features that are inherently resistant to strong in-distribution priors.

Training Pipeline. Starting from a pretrained PrivDFS model G (defender), PrivDFS-AT alternates between two stages: (1) *Adversary update.* For each branch i, a conditional diffusion model D_i is trained to reconstruct the input x from its feature share $s_i = G(x)$. These attackers are trained with full in-distribution data, ensuring they represent strong and specialized inversion models. (2) *Defender update.* With D_i frozen, the defender is optimized to both minimize the task loss $\mathcal{L}_{\text{task}}$ and maximize reconstruction errors, using the anti-reconstruction objective $\mathcal{L}_{\text{ar}}$:

$$\mathcal{L}_{\text{ar}} = \frac{1}{N}\sum_{i=1}^{N}\big[\text{SSIM}(x, \tilde{x}_i) - \text{MSE}(x, \tilde{x}_i)\big], \qquad (1)$$

where $\tilde{x}_i = D_i(s_i)$. The total loss for the defender is

$$\mathcal{L}_{\text{defender}} = \mathcal{L}_{\text{task}} + \lambda \mathcal{L}_{\text{ar}}, \qquad (2)$$

where λ controls the privacy-utility balance.

$\mathcal{L}_{\text{ar}}$ combines pixel-wise mean squared error (MSE) and the structural similarity index (SSIM): MSE discourages pixel-level matches by forcing the defender to obscure fine details, while SSIM penalizes the preservation of structural patterns such as shapes and textures. Together, these penalties remove both local appearance and global structure from branch-wise reconstructions. Through iterative adversarial training, PrivDFS-AT uses this objective to progressively harden the defender: as the attacker becomes stronger, the defender adapts to produce feature shares that are fundamentally less invertible.

3.4 Defense via Keyed Policy Diversification

Despite adversarial hardening, a static feature-sharing policy constitutes a fundamental single point of failure: under unbounded computational resources, a Level 3 adversary can eventually tailor an inversion model to that policy. The natural remedy is to break this monoculture by introducing diversity: instead of a single, fixed policy, we employ multiple user-specific policies so that an attacker faces many disjoint reconstruction problems rather than one. Accordingly, we propose **PrivDFS-KD**, which strengthens PrivDFS through *policy diversification*.

Core Idea. PrivDFS-KD extends PrivDFS with a family of k feature-sharing policies $\{\Psi_1, \ldots, \Psi_k\}$, each deterministically derived from an independent key (seed). The seed drives the pseudo-random components of the DFS pipeline—including orthogonal mixing matrices, channel permutations, and patch rearrangements— so that every policy induces a *policy-specific, non-transferable feature distribution*. During training, the policy index is sampled for each mini-batch, forcing the model to learn task representations that are invariant across policies, while making any inversion mapping strictly tied to one policy. As a result, an inversion model trained on one policy Ψ_j cannot generalize

to another Ψ_l ($l \neq j$): even with unlimited in-distribution data, the adversary must now solve k separate reconstruction problems instead of one. The parameter k therefore acts as a tunable privacy budget: larger k improves isolation at the cost of more capacity to support all policies.

4 Evaluations

We evaluate our framework on three aspects: (1) *utility*, measured by classification accuracy on multiple benchmark datasets; (2) *privacy*, assessed using state-of-the-art inversion attacks to quantify the fidelity of recovered inputs; and (3) *efficiency*, evaluated in terms of client-side computation and latency. All models are implemented in PyTorch (Paszke et al. 2019). Unless otherwise specified, the number of server branches is set to $N = 3$. All experiments are repeated 10 times, and we report the mean and standard deviation of the results. Experiments are conducted on a high-performance cluster with Intel Xeon Gold 6330 CPUs, 1TB RAM, and 8 NVIDIA 4090 GPUs.

4.1 Experimental Settings

Datasets and Model. We evaluate our framework on two benchmarks: CIFAR-10 (Krizhevsky, Hinton et al. 2009) for multi-class image classification and CelebA (Liu et al. 2015) for multi-attribute face prediction. To train strong reconstruction adversaries in Level 1 threat model, we provide large similar-distribution auxiliary datasets (Tiny ImageNet (Le and Yang 2015) for CIFAR-10 and FFHQ (Karras, Laine, and Aila 2021) for CelebA). The client-side encoder M_c^{enc} is deliberately lightweight, using a single convolutional layer to extract local features, while each server branch $M_{S,i}$ adopts a ResNet-18 (He et al. 2016) backbone unless stated otherwise.

Baselines. We compare our framework with a series of baselines, including:

- **Split Inference (SI)** (Kang et al. 2017). Standard split inference, which achieves the highest accuracy since features are sent unmodified, but its privacy depends on the split depth, creating a trade-off between leakage and on-device cost.
- **Shredder** (Mireshghallah et al. 2020). A classical perturbation-based defense that learns task-aware noise to obscure private information while preserving accuracy, serving as a representative noise-injection baseline.
- **Naive Channel Split (NCS).** A variant that replaces DFS with a random channel shuffle followed by an equal split into N chunks. This baseline isolates the effect of structured DFS transformations by testing a purely partition-based approach.

Adversary Setting. To rigorously test privacy, we adopt diffusion-based inversion attacks as our primary adversary. Diffusion models have recently become the *state-of-the-art* in image reconstruction and inversion, consistently outperforming GAN- and VAE-based approaches in recovering fine-grained visual details from highly compressed or partial representations (Zhang et al. 2025). Their iterative denoising process and strong learned priors make them particularly effective at exploiting even weak feature cues, thus providing a powerful and realistic attacker.

Attack implementation. We implement each adversary as a conditional U-Net diffusion model (Ronneberger, Fischer, and Brox 2015) with four down- and up-sampling stages, conditioned on the feature shares. The attack follows the defined threat levels: (1) Level 1 adversaries are trained on a small, similar-distribution dataset (e.g., Tiny ImageNet for CIFAR-10 or FFHQ for CelebA); (2) Level 2 adversaries are trained on in-distribution subsets of the target training data, representing attackers with stronger priors; and (3) Level 3 adversaries represent the worst case, with access to the entire target training set.

Evaluation Metrics. We assess each method along three complementary dimensions: (1) *Utility* is reported as Top-1 classification accuracy (%↑); (2) *Client cost* is measured by FLOPs (↓) on the client-side encoder; and (3) *Privacy* is quantified by the fidelity of reconstructions from the diffusion-based attacker, using PSNR (↓) (Horé and Ziou 2010), SSIM (↓) (Wang et al. 2004), and LPIPS (↑) (Zhang et al. 2018). Lower PSNR/SSIM or higher LPIPS indicates stronger privacy (poorer reconstruction).

4.2 Performance under Level 1 Adversary

Privacy Evaluation. Table 2 shows that PrivDFS offers substantially stronger privacy protection than all baselines while maintaining task accuracy close to that of SI. When 33% of feature shares are exposed, reconstruction quality collapses: on CIFAR-10, the SSIM of inverted images drops from 0.952 (SI) and 0.782 (Shredder) to 0.432 with PrivDFS, and on CelebA from 0.840 (SI) to 0.129. Figure 2 confirms this trend qualitatively: images reconstructed from PrivDFS shares are heavily degraded and visually unrecognizable, in stark contrast to the clear structure preserved by SI and Shredder. Even when the adversary compromises 50% of the shares, the gain is tiny (SSIM $0.432 \rightarrow 0.436$ on CIFAR-10 and $0.129 \rightarrow 0.134$ on CelebA), highlighting that partial feature access remains statistically uninformative. Notably, these strong privacy guarantees come with almost no loss in accuracy: 92.5% on CIFAR-10 and 90.8% on CelebA.

Utility-Privacy Trade off. As shown in Table 3, PrivDFS offers a more favorable utility-privacy trade-off than Shredder. For a fair comparison, we increase Shredder's noise level until it achieves a privacy level comparable to PrivDFS. Even under this stronger obfuscation, PrivDFS consistently retains higher task accuracy. This gap reflects a fundamental design difference. PrivDFS achieves privacy not by perturbing features, but by securely distributing it across multiple non-colluding servers. By contrast, Shredder and other perturbation-based defenses rely on irreversible corruption of the feature space. As the noise level increases to ensure privacy, essential representations are inevitably suppressed.

Client-Side Cost. Table 4 compares PrivDFS with SI configured to achieve a similar level of privacy by moving the split point deep into the network (after Block5 of ResNet-18). This deep split substantially increases client computation: on CIFAR-10, SI requires 206.7M FLOPs, whereas PrivDFS achieves the same privacy with only 2.95M FLOPs,

Table 2: Accuracy and privacy metrics under Level 1 attacks on CIFAR-10 and CelebA. For PrivDFS, 33% and 50% correspond to the fraction of compromised servers (i.e., 1 of 3 servers and 3 of 6 servers, respectively).

Method	CIFAR10				CelebA			
	Acc ↑	PSNR ↓	SSIM ↓	LPIPS ↑	Acc ↑	PSNR ↓	SSIM ↓	LPIPS ↑
SI	**96.79%**	23.008±2.189	0.952±0.027	0.008±0.005	**91.31%**	19.717±1.594	0.840±0.056	0.049±0.012
Shredder	92.35%	17.594±1.580	0.782±0.040	0.035±0.015	90.76%	15.461±1.336	0.615±0.045	0.230±0.034
NCS	94.33%	22.040±1.571	0.888±0.028	0.015±0.005	91.27%	18.986±1.761	0.710±0.055	0.194±0.028
PrivDFS(33%)	92.53%	**13.750±0.472**	**0.432±0.032**	**0.100±0.009**	90.81%	**8.919±0.168**	**0.129±0.012**	**0.566±0.012**
PrivDFS(50%)	92.43%	**13.832±0.481**	**0.436±0.025**	**0.098±0.007**	90.71%	**8.974±0.208**	**0.134±0.015**	**0.554±0.014**

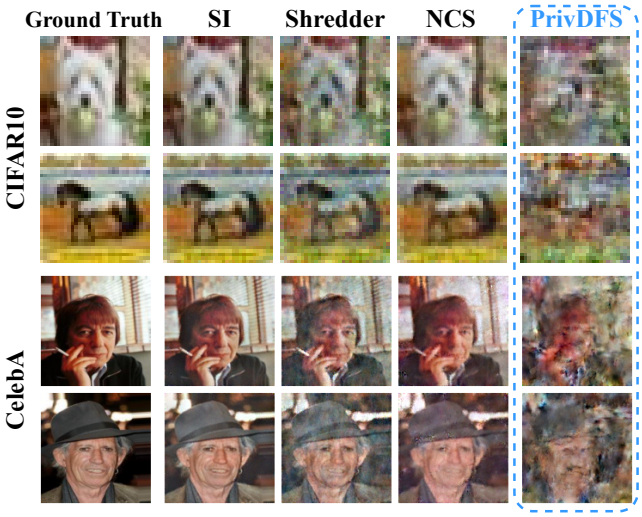

Figure 2: Qualitative reconstruction results under a Level 1 adversary on CIFAR-10 and CelebA. Columns correspond to different defense methods.

Table 3: Accuracy vs. Privacy (matched strong privacy).

Dataset	Method	Acc ↑	PSNR ↓	SSIM ↓	LPIPS ↑
CIFAR10	Shredder	91.39%	13.855	0.491	0.090
	PrivDFS	**92.53%**	**13.750**	**0.432**	**0.100**
CelebA	Shredder	90.52%	9.509	0.159	0.526
	PrivDFS	**90.81%**	**8.919**	**0.129**	**0.566**

over 70× less. On the higher-resolution CelebA, the gap is nearly 100× (4.39G vs. 47.2M). These results demonstrate that PrivDFS decouples privacy from client cost, overcoming the inherent trade-off in traditional split inference and enabling private inference on resource-constrained devices.

4.3 Performance Under Level 2 Adversary

We further explore the security of PrivDFS-AT against a stronger Level 2 adversary, which has access to in-distribution data and can thus train highly specialized inversion models. This setting is substantially more challenging than Level 1, as it removes the data-distribution mismatch that naturally limits reconstruction accuracy.

As shown in Table 5, introducing adversarial hardening

Table 4: Efficiency vs. Privacy (matched strong privacy)

Dataset	Method	FLOPs ↓	PSNR ↓	SSIM ↓	LPIPS ↑
CIFAR10	SI	206.730M	13.974	0.479	0.089
	PrivDFS	**2.949M**	**13.750**	**0.432**	**0.100**
CelebA	SI	4.387G	9.112	0.364	0.360
	PrivDFS	**47.186M**	**8.919**	**0.129**	**0.566**

Table 5: PrivDFS and PrivDFS-AT under Level 2 adversary.

Dataset	Method	Acc ↑	PSNR ↓	SSIM ↓	LPIPS ↑
CIFAR10	PrivDFS	**92.37%**	14.528	0.472	0.090
	PrivDFS-AT	92.34%	**11.784**	**0.268**	**0.171**
CelebA	PrivDFS	**90.81%**	9.642	0.159	0.564
	PrivDFS-AT	90.43%	**8.384**	**0.111**	**0.632**

produces a marked improvement in privacy with almost no impact on task utility. On CIFAR-10, PrivDFS-AT reduces SSIM from 0.472 to 0.268 and nearly doubles LPIPS (0.090 → 0.171), indicating that reconstructed images lose both structural and perceptual fidelity. CelebA shows a similar pattern, with reconstructions becoming even more degraded. These gains are achieved with only a marginal drop in accuracy (-0.03% on CIFAR-10 and -0.38% on CelebA), demonstrating that the anti-reconstruction objective successfully pushes the learned representations toward a space where predictive signals are preserved but invertible structures are eliminated.

4.4 Performance Under Level 3 Adversary

Level 3 represents the most challenging threat model, where an adversary has unrestricted access to in-distribution data and can devote unbounded resources to train highly specialized inversion models. In this setting, a single static feature-sharing policy becomes a single point of failure. PrivDFS-KD addresses this by introducing *keyed policy diversification*, so that an inversion model overfits to its own key-specific feature distribution and fails to generalize to others.

Table 6 quantifies the trade-off between the number of keys (k) and task accuracy. As k increases, accuracy drops gradually due to the need to learn policy-invariant representations; however, this degradation can be mitigated by using higher-capacity server models (e.g., ResNet-34). This

Table 6: Accuracy of PrivDFS-KD under different numbers of keys (k) and server model capacities.

M_S	2 Keys	4 Keys	8 Keys	16 Keys
ResNet18	91.11%	89.17%	85.79%	85.02%
ResNet34	**91.64%**	**89.40%**	**86.90%**	**86.62%**

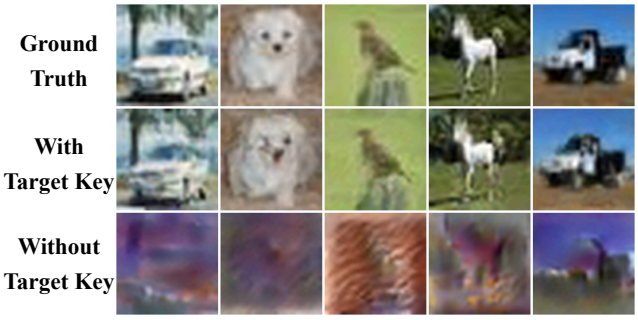

Figure 3: Qualitative reconstruction results under a Level 3 adversary. Without the target key, inversion fails completely.

demonstrates that PrivDFS-KD scales through increased model capacity. Figure 3 illustrates PrivDFS-KD's security. When the attacker's inversion model is trained *with the target Key*, reconstruction remains accurate (e.g., PSNR$=20.3$, SSIM$=0.77$). However, applying the same model *without the target Key* results in a complete collapse of reconstruction (PSNR$=10.6$, SSIM$=0.19$), with outputs degenerating into noise. This contrast confirms that PrivDFS-KD enforces strong *policy isolation*: even a Level 3 adversary cannot transfer an inversion model across keys. These results show that PrivDFS-KD turns a single global reconstruction problem into k disjoint ones, providing a tunable, key-driven defense against the strongest class of adaptive adversaries.

4.5 Extended Experiments

Scalability with Server Count. We evaluate the scalability of PrivDFS by varying the number of server branches (N) while keeping the client-side feature dimensionality fixed. Figure 4 shows that accuracy decreases only slightly as N grows, demonstrating that PrivDFS scales gracefully to different deployment configurations.

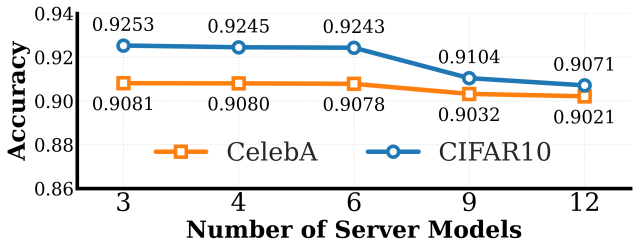

Figure 4: Impact of the number of server models (N) on task accuracy: PrivDFS scales well with more branches.

Ablation of DFS Stages. To understand the contribution of each transformation in the DFS pipeline, we systematically remove one stage at a time and re-train the model. Table 7 shows that almost all stages are necessary for privacy and accuracy: removing any component leads to higher PSNR/SSIM and lower LPIPS, indicating weaker privacy. In particular, removing *Patch Reorganization* causes a drastic privacy collapse (SSIM rises from 0.432 to 0.909) and an anomalous boost in accuracy, showing that this stage is critical for breaking global structural cues. While some removals slightly improve one metric, they always harm others, confirming that the full DFS pipeline yields the balanced and robust protection.

Table 7: Ablation study of the DFS stages.

Stage (w/o)	Acc ↑	PSNR ↓	SSIM ↓	LPIPS ↑
Full DFS	**92.53%**	**13.750**	**0.432**	**0.100**
LocConf	92.85%	15.134	0.496	0.083
OrthoRcb	92.67%	14.219	0.443	0.108
AdaNoise	92.48%	14.767	0.476	0.091
ChanPerm	92.62%	14.201	0.457	0.104
PatchReorg	96.37%	21.173	0.909	0.016
CrossMix	91.93%	13.889	0.440	0.116

Inference Incapability of a Single Share. Finally, we verify the core security claim that no single branch is informative enough to support prediction on its own. When we directly classify using only one branch of PrivDFS (on CIFAR-10), the accuracies of the three branches are 10.8%, 8.5%, and 10.4%, essentially indistinguishable from random guessing (10%). In contrast, fusing all three branches recovers a full accuracy of 92.5%. This confirms that each share is semantically incomplete, preventing both direct label inference and downstream reconstruction attacks that depend on partial predictive information.

5 Conclusion and Future Work

We have presented PrivDFS, a distributed feature-sharing framework that protects inference privacy by fragmenting intermediate features across non-colluding servers. Through adversarial hardening (PrivDFS-AT) and key diversification (PrivDFS-KD), PrivDFS achieves strong resistance to inversion attacks with negligible accuracy loss and greatly reduced client cost. Experiments on CIFAR-10 and CelebA validate its effectiveness. Despite these advances, several challenges still remain: PrivDFS currently requires end-to-end training for each architecture, the multi-server design introduces additional inference cost, its current form is limited to classification tasks, and policy diversification with many keys, while effective, can slightly reduce accuracy. Extending PrivDFS to generation tasks is also non-trivial due to their strong contextual dependencies. Addressing these challenges would pave the way for private inference that is scalable, versatile, and ready for real-world deployment.

References

Avella-Medina, M.; Bradshaw, C.; and Loh, P.-L. 2023. Differentially private inference via noisy optimization. *The Annals of Statistics*, 51(5): 2067–2092.

Bakhtiarnia, A.; Milosevic, N.; Zhang, Q.; Bajovic, D.; and Iosifidis, A. 2023. Dynamic Split Computing for Efficient Deep EDGE Intelligence. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, 1–5. IEEE.

Cheng, K.; Xi, N.; Liu, X.; Zhu, X.; Gao, H.; Zhang, Z.; and Shen, Y. 2023. Private Inference for Deep Neural Networks: A Secure, Adaptive, and Efficient Realization. *IEEE Transactions on Computers*, 72(12): 3519–3531.

Diaa, A.; Fenaux, L.; Humphries, T.; Dietz, M.; Ebrahimianghazani, F.; Kacsmar, B.; Li, X.; Lukas, N.; Mahdavi, R. A.; Oya, S.; Amjadian, E.; and Kerschbaum, F. 2024. Fast and Private Inference of Deep Neural Networks by Co-designing Activation Functions. In *Proceedings of the 33rd USENIX Security Symposium, USENIX Security 2024, Philadelphia, PA, USA, August 14-16, 2024*. USENIX Association.

Feng, J.; Wu, Y.; Sun, H.; Zhang, S.; and Liu, D. 2025. Panther: Practical Secure Two-Party Neural Network Inference. *IEEE Transactions on Information Forensics and Security*, 20: 1149–1162.

Fredrikson, M.; Jha, S.; and Ristenpart, T. 2015. Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-16, 2015*, 1322–1333. ACM.

Garimella, K.; Ghodsi, Z.; Jha, N. K.; Garg, S.; and Reagen, B. 2023. Characterizing and Optimizing End-to-End Systems for Private Inference. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3, ASPLOS 2023, Vancouver, BC, Canada, March 25-29, 2023*, 89–104. ACM.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 770–778. IEEE Computer Society.

He, Z.; Zhang, T.; and Lee, R. B. 2019. Model inversion attacks against collaborative inference. In *Proceedings of the 35th Annual Computer Security Applications Conference, ACSAC 2019, San Juan, PR, USA, December 09-13, 2019*, 148–162. ACM.

Horé, A.; and Ziou, D. 2010. Image Quality Metrics: PSNR vs. SSIM. In *Proceedings of the 20th International Conference on Pattern Recognition, ICPR 2010, Istanbul, Turkey, 23-26 August 2010*, 2366–2369. IEEE Computer Society.

Juuti, M.; Szyller, S.; Marchal, S.; and Asokan, N. 2019. PRADA: Protecting Against DNN Model Stealing Attacks. In *Proceedings of the IEEE European Symposium on Security and Privacy, EuroS&P 2019, Stockholm, Sweden, June 17-19, 2019*, 512–527. IEEE.

Kang, Y.; Hauswald, J.; Gao, C.; Rovinski, A.; Mudge, T. N.; Mars, J.; and Tang, L. 2017. Neurosurgeon: Collaborative Intelligence Between the Cloud and Mobile Edge. In *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS 2017, Xi'an, China, April 8-12, 2017*, 615–629. ACM.

Karras, T.; Laine, S.; and Aila, T. 2021. A Style-Based Generator Architecture for Generative Adversarial Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12): 4217–4228.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images.

Le, Y.; and Yang, X. 2015. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7): 3.

Lee, T.; Edwards, B.; Molloy, I. M.; and Su, D. 2019. Defending Against Neural Network Model Stealing Attacks Using Deceptive Perturbations. In *Proceedings of the 2019 IEEE Security and Privacy Workshops, SP Workshops 2019, San Francisco, CA, USA, May 19-23, 2019*, 43–49. IEEE.

Li, Z.; Yang, M.; Liu, Y.; Wang, J.; Hu, H.; Yi, W.; and Xu, X. 2023. GAN You See Me? Enhanced Data Reconstruction Attacks against Split Inference. In *Proceedings of the Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*. NeurIPS Foundation.

Liu, S.; Wang, Z.; Chen, Y.; and Lei, Q. 2025. Data Reconstruction Attacks and Defenses: A Systematic Evaluation. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, 613–621. PMLR.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep Learning Face Attributes in the Wild. In *Proceedings of the 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 3730–3738. IEEE Computer Society.

Mann, Z. Á.; Weinert, C.; Chabal, D.; and Bos, J. W. 2024. Towards Practical Secure Neural Network Inference: The Journey So Far and the Road Ahead. *ACM Computing Surveys*, 56(5): 117:1–117:37.

Mireshghallah, F.; Taram, M.; Ramrakhyani, P.; Jalali, A.; Tullsen, D. M.; and Esmaeilzadeh, H. 2020. Shredder: Learning Noise Distributions to Protect Inference Privacy. In *Proceedings of the ASPLOS '20: Architectural Support for Programming Languages and Operating Systems, Lausanne, Switzerland, March 16-20, 2020*, 3–18. ACM.

Otroshi-Shahreza, H.; Hahn, V. K.; and Marcel, S. 2024. Vulnerability of State-of-the-Art Face Recognition Models to Template Inversion Attack. *IEEE Transactions on Information Forensics and Security*, 19: 4585–4600.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.;

Desmaison, A.; Köpf, A.; Yang, E. Z.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Proceedings of the Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 8024–8035. NeurIPS Foundation.

Peng, H.; Huang, S.; Zhou, T.; Luo, Y.; Wang, C.; Wang, Z.; Zhao, J.; Xie, X.; Li, A.; Geng, T.; Mahmood, K.; Wen, W.; Xu, X.; and Ding, C. 2023. AutoReP: Automatic ReLU Replacement for Fast Private Network Inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, 5155–5165. IEEE.

Qiu, Y.; Fang, H.; Yu, H.; Chen, B.; Qiu, M. K.; and Xia, S. 2024. A Closer Look at GAN Priors: Exploiting Intermediate Features for Enhanced Model Inversion Attacks. In *Proceedings of the Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XXXII*, volume 15090 of *Lecture Notes in Computer Science*, 109–126. Springer.

Rao, B.; Zhang, J.; Wu, D.; Zhu, C.; Sun, X.; and Chen, B. 2025. Privacy Inference Attack and Defense in Centralized and Federated Learning: A Comprehensive Survey. *IEEE Transactions on Artificial Intelligence*, 6(2): 333–353.

Ribeiro, M.; Grolinger, K.; and Capretz, M. A. M. 2015. MLaaS: Machine Learning as a Service. In *Proceedings of the 14th IEEE International Conference on Machine Learning and Applications, ICMLA 2015, Miami, FL, USA, December 9-11, 2015*, 896–902. IEEE.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention - MICCAI 2015 - 18th International Conference Munich, Germany, October 5 - 9, 2015, Proceedings, Part III*, volume 9351 of *Lecture Notes in Computer Science*, 234–241. Springer.

Rudin, L. I.; Osher, S.; and Fatemi, E. 1992. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1): 259–268.

Shamir, A. 1979. How to Share a Secret. *Communications of the ACM*, 22(11): 612–613.

Singh, A.; Sharma, V.; Sukumaran, R.; Mose, J.; Chiu, J.; Yu, J.; and Raskar, R. 2024. SIMBA: Split Inference - Mechanisms, Benchmarks and Attacks. In *Proceedings of the-Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part LXXVI*, volume 15134 of *Lecture Notes in Computer Science*, 214–232. Springer.

Ulyanov, D.; Vedaldi, A.; and Lempitsky, V. S. 2018. Deep Image Prior. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 9446–9454. Computer Vision Foundation / IEEE Computer Society.

Vepakomma, P.; Gupta, O.; Swedish, T.; and Raskar, R. 2018. Split learning for health: Distributed deep learning without sharing raw patient data. *CoRR*, abs/1812.00564.

Vepakomma, P.; Singh, A.; Gupta, O.; and Raskar, R. 2020. NoPeek: Information leakage reduction to share activations in distributed deep learning. In *Proceedings of the 20th International Conference on Data Mining Workshops, ICDM Workshops 2020, Sorrento, Italy, November 17-20, 2020*, 933–942. IEEE.

Wang, Y.; Guo, S.; Deng, Y.; Zhang, H.; and Fang, Y. 2024. Privacy-Preserving Task-Oriented Semantic Communications Against Model Inversion Attacks. *IEEE Transactions on Wireless Communications*, 23(8): 10150–10165.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612.

Xu, T.; Wu, L.; Wang, R.; and Li, M. 2024. PrivCirNet: Efficient Private Inference via Block Circulant Transformation. In *Proceedings of the Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*. NeurIPS Foundation.

Yang, X.; Chen, J.; He, K.; Bai, H.; Wu, C.; and Du, R. 2023. Efficient Privacy-Preserving Inference Outsourcing for Convolutional Neural Networks. *IEEE Transactions on Information Forensics and Security*, 18: 4815–4829.

Yeom, S.; Giacomelli, I.; Fredrikson, M.; and Jha, S. 2018. Privacy Risk in Machine Learning: Analyzing the Connection to Overfitting. In *Proceedings of the 31st IEEE Computer Security Foundations Symposium, CSF 2018, Oxford, United Kingdom, July 9-12, 2018*, 268–282. IEEE Computer Society.

Zhang, L.; Rao, A.; and Agrawala, M. 2023. Adding Conditional Control to Text-to-Image Diffusion Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, 3813–3824. IEEE.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 586–595. Computer Vision Foundation / IEEE Computer Society.

Zhang, S. Q.; Li, Z.; Guo, C.; Mahloujifar, S.; Dangwal, D.; Suh, G. E.; Salvo, B. D.; and Liu, C. 2025. Unlocking Visual Secrets: Inverting Features with Diffusion Priors for Image Reconstruction. *IEEE Transactions on Machine Learning Research*.

Zhu, X.; Luo, X.; Wu, Y.; Jiang, Y.; Xiao, X.; and Ooi, B. C. 2025. Passive Inference Attacks on Split Learning via Adversarial Regularization. In *Proceedings of the 32nd Annual Network and Distributed System Security Symposium, NDSS 2025, San Diego, California, USA, February 24-28, 2025*. The Internet Society.