Personalized Federated Training of Latent Diffusion Models with Privacy Guarantees

A F M Mahfuzul Kabir Lingxiao Wang
{ak3535, lw324}@njit.edu
New Jersey Institute of Technology

Abstract

We study the problem of federated training of diffusion models (DMs) with privacy guarantees. For high-dimensional data, existing private federated DMs often exhibit a poor privacy-utility tradeoff, since the privacy noise introduced to protect client data becomes increasingly damaging as the dimension of the image representation grows. To address this challenge, we propose **Personalized Federated Training of Latent Diffusion Models (PF-LDM)**, a framework that performs private federated diffusion training in latent space. The key idea is to identify latent representations whose distributional structure is more favorable for private learning. In particular, we show that latent spaces with more discriminative feature representations can better preserve utility under privacy constraints. PF-LDM further combines a shared server-side diffusion model with personalized client-side refinement models: the server captures cross-client generative structure from privatized latent data, while client-specific models refine samples to recover fine-grained local details. Experiments on the CelebAHQ dataset demonstrate that our method enables *high-dimensional image generation*, *improves performance on underrepresented classes across clients*, and *maintains strong privacy protection*.

1. Introduction

Synthetic data generation has emerged as a promising and increasingly common approach for addressing data scarcity in modern machine learning (Lu et al., 2023; Bauer et al., 2024; Kapania et al., 2025). Its use is particularly appealing when real data are scarce, costly to obtain, and difficult to share. Diffusion Models (DMs) have become a dominant paradigm for synthetic image generation because of their stable optimization and strong sample fidelity (Ho et al., 2020). However, achieving strong performance with diffusion-based generative models still typically requires abundant training data. In privacy-sensitive domains like healthcare, finances, and legal organizations, such availability is often limited, mainly because these organizations operate under legal and regulatory constraints that prohibit pooling data across clients for model training (European Parliament & Council of the European Union, 2016; Cal. Legis. Serv., 2020; Grynbaum & Mac, 2023; Wang et al., 2023).

Federated learning (FL) offers a natural direction for such settings by enabling collaborative model training without requiring the exchange of raw data across organizations (McMahan et al., 2017). Therefore, recent works (de Goede et al., 2024; Jothiraj & Mashhadi, 2023; Allmendinger et al., 2024) explored the training of DMs in decentralized data settings using federated learning, implying that privacy is preserved since the raw data do not leave individual clients in collaboration. However, the absence of raw-data sharing does not by itself guarantee privacy. In truth, the privacy-preservation in FL is often considered as a thin facade, as it can be penetrated easily with the successful reconstruction of private training examples from the central server (Boenisch et al., 2023; Pichler et al., 2023; Geiping et al., 2020; Fowl et al., 2021; Melis et al., 2019). This concern is especially important for DMs, because they are prone to memorizing training examples and reproducing them at generation time (Carlini et al., 2023; Gu et al., 2023; Somepalli et al., 2023). These findings highlight the need for diffusion model training methods that provide formal privacy protection in collaborative frameworks.

Accepted to FoGen 2026: Foundations of Deep Generative Models: Understanding Memorization, Generalization, and Reasoning, an ICML 2026 workshop (non-archival).

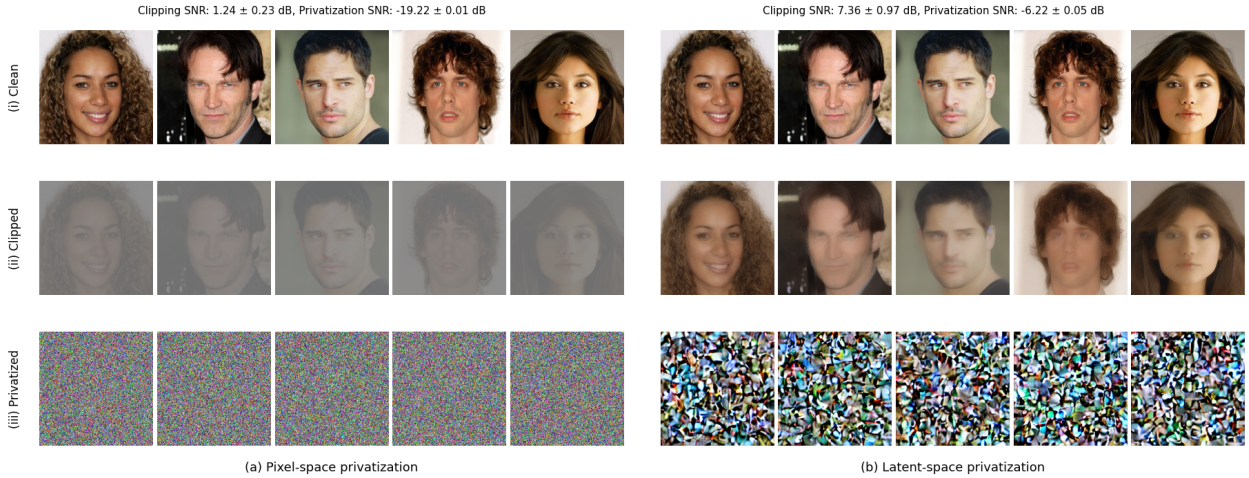

Figure 1. Comparison of pixel-space and latent-space privatization for high-dimensional 256×256 images under the same privacy budget, $\epsilon = 10$. Each panel shows clean images, clipped representations, and privatized representations from top to bottom. The signal-to-noise ratio (SNR), measured in decibels (dB), indicates the strength of the image signal relative to the distortion introduced by clipping and added noise on a logarithmic scale. In pixel-space, the privatization drastically reduces SNR, indicating destruction of perceptual details. In contrast, latent-space privatization preserves coarser semantic structures, highlighting the advantage of moving the privacy mechanism from pixel space to latent space.

A separate line of recent work has studied the training of DMs with formal differential privacy (DP) guarantees in the centralized setting (Dockhorn et al., 2023; Li et al., 2024; Jiang et al., 2025; Liu et al., 2024). These methods typically rely on DP-SGD (Abadi et al., 2016) to protect training data, and many further improve the privacy and utility trade-off by leveraging public data. However, their training pipelines are designed for a centralized setting: they assume that all private data is collected behind a single trust boundary and used to train one centralized generator. This assumption prevents them from being directly applied to federated learning (FL), where data remain distributed across clients, and raw samples cannot be pooled. Extending centralized DP diffusion training to FL is nontrivial because FL introduces additional challenges, including heterogeneous data distributions, privacy-preserving communication, limited local data, and the need to balance cross-client collaboration with client-specific generation control. As a result, centralized DP methods for DMs do not address the decentralized data scarcity that often arises in real-world multi-institution settings, such as rare datasets, small cohorts, and costly experimental data collection (Ching et al., 2018; Rieke et al., 2020; Clark, 2021; Prakash, 2023).

Patel et al. (2026) addressed this issue by proposing a personalized federated framework, PFDM, for diffusion model training in decentralized cross-silo settings (Kairouz et al., 2021b). PFDM splits the reverse diffusion process into client-specific and shared components: client models recover clean images from local noisy images, while the shared model maps Gaussian noise to a mixture of noisy client images. This design keeps the shared model from directly generating client-specific samples, gives clients control over their synthetic data, and allows cross-client collaboration through higher-level shared features. Privacy is enforced by clipping images in pixel space and injecting diffusion forward noise, yielding per-sample privacy guarantees. However, when data are represented in high-dimensional pixel space, this privacy mechanism becomes increasingly restrictive. To maintain privacy in higher dimensions, stronger clipping and forward diffusion noise injection are often required, which can suppress fine-scale attributes that are essential for perceptual quality, as illustrated in Figure 1(a). This limitation makes direct pixel-space privatization increasingly impractical for high-dimensional image generation within this framework.

Latent diffusion models (LDMs) provide a natural way to alleviate this bottleneck by using a perceptual compression stage that maps images into a lower-dimensional and computationally efficient latent space (Rombach et al., 2022). This compression is beneficial not only for efficiency, but also for privacy-preserving collaboration. As shown in Figure 1(b), clipping and noising the latent representation of an image is much less destructive than applying the same operations in pixel space, while still preserving perceptually important details of high-dimensional data. Leveraging this compressed latent space can therefore improve the privacy–utility trade-off of PFDM when applied to high-dimensional image generation.

Motivated by these observations, we revisit privacy-preserving federated diffusion through the lens of latent representations. Specifically, we extend the personalized collaborative structure of Patel et al. (2026) from pixel space to

latent space, where high-dimensional images are first encoded by a public pretrained image tokenizer. This allows the framework to preserve client-specific control over generation while enabling collaborative learning of shared cross-client structure. We study how the choice of latent representation affects the privacy–utility trade-off, with the goal of identifying latent spaces that are more suitable for private collaborative diffusion training. Our main contributions are:

1. **Personalized Federated Latent Diffusion Models (PF-LDM).** We introduce PF-LDM, a framework for training diffusion models in the latent space for *high-dimensional*, *decentralized*, and *formally private* image generation. PF-LDM privatizes local data by clipping and noising latent representations rather than raw pixels, reducing the destructive effect of pixel-space privatization and improving the privacy and utility trade-off for per-sample privacy preservation. We provide formal differential privacy guarantees for the proposed latent-space federated training procedure. During generation, a shared global model captures cross-client structure, while personalized client models refine samples to recover client-specific details without exposing private data.
2. **High-Dimensional Generation and Downstream Evaluation.** We empirically validate PF-LDM on CelebA-HQ at 256×256 resolution, demonstrating strong utility for high-dimensional image generation under formal privacy constraints. We further evaluate the practical usefulness of generated data through downstream task performance and compare PF-LDM against relevant baselines. These results show that PF-LDM can generate useful high-dimensional synthetic data while preserving privacy in decentralized settings.
3. **Latent Representation Study and Privacy–Utility Trade-off Analysis.** One central focus of this work is to study which latent representations are more suitable for privacy-preserving federated DMs. We compare different image tokenizers and analyze how their induced latent distributions affect generation quality under the same privacy constraints. Our empirical findings suggest that latent representations with more discriminative feature structure can better preserve utility after clipping and noise injection. We further vary the privacy budget to characterize the privacy and utility trade-off and identify regimes where PF-LDM provides practical benefits for private synthetic data generation.

2. Preliminaries.

Denoising Latent Diffusion Models (LDMs). Image generation with LDMs typically consists of two stages. First, an image compressor maps images into a lower-dimensional latent space. Second, a diffusion model is trained in this latent space. Thus, instead of directly modeling the pixel-space distribution $p(x_0)$, LDMs model the latent distribution $p(z_0)$. New images are generated by sampling a latent through the reverse diffusion process and decoding it back to pixel space.

The compressor is usually implemented as an autoencoder with an encoder Enc and a decoder Dec. Given an image x_0, the encoder produces a latent representation $z_0 = \mathrm{Enc}(x_0)$, and the decoder reconstructs the image as $\tilde{x}_0 = \mathrm{Dec}(z_0)$. The diffusion model is then trained on z_0. In the forward process, z_0 is gradually corrupted by Gaussian noise:

$$z_t = \sqrt{\bar{\alpha}_t}\, z_0 + \sqrt{1-\bar{\alpha}_t}\, \xi, \qquad \xi \sim \mathcal{N}(0, I), \tag{1}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$ is determined by a noise schedule $\{\beta_t\}_{t=1}^{T}$. For sufficiently large T, z_T is close to a Gaussian distribution and serves as the starting point for sampling.

The reverse process learns to denoise z_t back toward z_0. Following Ho et al. (2020); Rombach et al. (2022), this can be formulated as training a neural denoiser ϵ_θ to predict the injected noise:

$$L_{\mathrm{LDM}} = \mathbb{E}_{z_0 \sim p(z),\, \xi \sim \mathcal{N}(0,I),\, t}\left[\left\|\xi - \epsilon_\theta(z_t, t)\right\|_2^2\right], \tag{2}$$

where $p(z)$ denotes the latent distribution and t is uniformly sampled from $\{1, \ldots, T\}$. After training, generation starts from Gaussian noise, iteratively applies the learned denoising model to obtain z_0, and decodes z_0 into an image. Further details appear in Appendix B.

Image Compression with Autoencoders. As mentioned before, LDMs train diffusion models in a compressed latent space rather than directly in pixel space. The compression factor f controls the latent resolution: an image of size

$H \times W$ is encoded into a latent tensor of size $H/f \times W/f$. Larger f gives stronger compression and lower-dimensional latents, while smaller f preserves more details at higher computational and privacy cost.

The autoencoder determines the latent distribution used for diffusion training, which directly affects the privacy and utility trade-off studied in this work. Following the LDM framework (Rombach et al., 2022), we consider two variants: KL-regularized and VQ-regularized autoencoders. KL-regularized autoencoders can be viewed as variational autoencoders (VAEs) (Kingma & Welling, 2022), where the latent distribution is encouraged to stay close to a Gaussian prior. This yields a continuous Gaussian-regularized latent space. In contrast, VQ-regularized autoencoders can be viewed as VQ-GANs (Esser et al., 2021), where encoder features are mapped to a learned codebook of discrete visual tokens. This induces a codebook-structured latent distribution that can be more clustered and potentially more discriminative.

Differential Privacy. We use differential privacy (DP) (Dwork et al., 2006) to formalize privacy in federated DMs.

Definition 2.1 ((ϵ, δ)-DP)**.** A randomized mechanism $\mathcal{A}$ satisfies (ϵ, δ)-DP if, for any adjacent inputs u and u', and any measurable set O, $\mathbb{P}[\mathcal{A}(u) \in O] \leq e^{\epsilon}\mathbb{P}[\mathcal{A}(u') \in O] + \delta$.

The adjacency relation specifies the privacy model. In central DP, u and u' are adjacent datasets, typically differing in one record, and privacy is enforced by a trusted curator. In federated learning, central DP can protect either individual records, known as sample-level DP, or an entire client's dataset, known as client-level DP (McMahan et al., 2018; Kairouz et al., 2021a). Client-level DP protects a larger unit, but usually requires more noise and can reduce utility.

In this work, we focus on local DP (Kasiviswanathan et al., 2011), where u and u' are individual inputs and each client privatizes its data before communication. This removes the need for a trusted server: the server only receives privatized outputs. This is well-suited to cross-silo settings, where each silo may contain records from many individuals. Per-sample local DP also implies sample-level central DP by post-processing, making it a stronger privacy protection.

Problem Setup. We study denoising LDMs in a federated learning setting with M clients. Each client $m \in [M]$ holds a private dataset $D_m = \{x^{i,m}\}_{i=1}^{n_m}$, where samples are drawn from a client-specific data distribution q_0^m. The goal is to learn personalized generative models $\{p^m\}_{m=1}^M$ that approximate the client distributions $\{q_0^m\}_{m=1}^M$, while preserving the privacy of each client's data. We assume access to a public pretrained autoencoder with encoder Enc and decoder Dec. For each image $x^{i,m} \in D_m$, the encoder maps the image to a latent representation $z_0^{i,m} = \mathrm{Enc}(x^{i,m})$. Diffusion training is then performed in the latent space.

3. Methodology

3.1. Training Procedure of PF-LDM

Our federated training framework is divided into two stages. In both stages, we use a public pretrained autoencoder.

Stage 1 - Training of Client-specific Denoisers. In the first stage, we train the client-specific local denoisers, denoted by $\{\xi_{\theta_m}\}_{m=1}^M$, on the client-side private dataset $\{D_m\}_{m\in[M]}$ (Line 1–3 in Algorithm 1). Each client first uses the public encoder (shared across clients) to map its private images into latent representations and then trains its local denoiser using the standard latent diffusion training routine T-LDM (Algorithm 3 in Appendix B). These client-specific denoisers capture local data characteristics and are never shared.

Stage 2 - Privatization and Shared Denoiser Training. To enable collaboration, each client constructs a privatized version of its latent dataset. For each latent $\tilde{z}_0^{i,m}$, the client first clips it to have a bounded ℓ_2-norm with clipping radius C_z. The clipped latent is then perturbed using Gaussian forward diffusion noise at time step t_0 (lines 4–7 in Algorithm 1):

$$\tilde{z}_0^{i,m} = \sqrt{\bar{\alpha}_{t_0}}\,\mathrm{CLIP}(z_0^{i,m}, C_z) + \sqrt{1-\bar{\alpha}_{t_0}}\,\xi_{t_0}. \tag{3}$$

Here, t_0 controls the amount of noise injected into the latent representation. The clipping radius C_z bounds the sensitivity of each latent, while the added Gaussian noise provides the privacy guarantee. Therefore, the privacy budget is determined jointly by $C_z, \bar{\alpha}_{t_0}$ (see Theorem 4.1).

Algorithm 1 Personalized Federated Training of Latent Diffusion Models (PF-LDM)

input Training datasets $\{D_m\}_{m\in[M]}$, public encoder Enc, shared model parameter θ_g, local model parameters $\{\theta_m\}_{m=1}^{M}$, noise schedule $\{\beta_t\}_{t=1}^{T}$, latent clipping parameter C_z

1: **on client** $m \in [M]$ **do**
2: Encode private images into latents: $Z_m = \{z_0^{i,m} = \mathrm{Enc}(x^{i,m}) \mid x^{i,m} \in D_m\}$
3: Train a personalized latent denoiser: $\xi_{\theta_m} = \text{T-LDM}(Z_m, T, \{\beta_t\}_{t=1}^{T}, \theta_m)$
4: Sample n_m latent data from Z_m indexed by $\mathcal{B}_m$ and set $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$
5: **for** $i \in \mathcal{B}_m$ **do**
6: $\tilde{z}_0^{i,m} = \sqrt{\bar{\alpha}_{t_0}}\,\mathrm{CLIP}(z_0^{i,m}, C_z) + \sqrt{1-\bar{\alpha}_{t_0}}\,\xi_{t_0}$, where $\xi_{t_0} \sim \mathcal{N}(0, I)$ and $\mathrm{CLIP}(z_0^{i,m}, C_z) = z_0^{i,m} \cdot \min\left\{1, \frac{C_z}{\|z_0^{i,m}\|_2}\right\}$
7: **end for**
8: **Send** $\tilde{Z}_m = \{\tilde{z}_0^{i,m}\}_{i\in\mathcal{B}_m}$ to the server
9: **end on client**
10: **on server do**
11: Obtain the privatized latent dataset $\tilde{Z} = \{\tilde{Z}_m\}_{m\in[M]}$
12: Train the shared global latent denoiser: $\xi_{\theta_g} = \text{T-LDM}(\tilde{Z}, T, \{\beta_t\}_{t=1}^{T}, \theta_g)$
13: **end on server**

output Global denoiser ξ_{θ_g} and private denoisers $\{\xi_{\theta_m}\}_{m=1}^{M}$

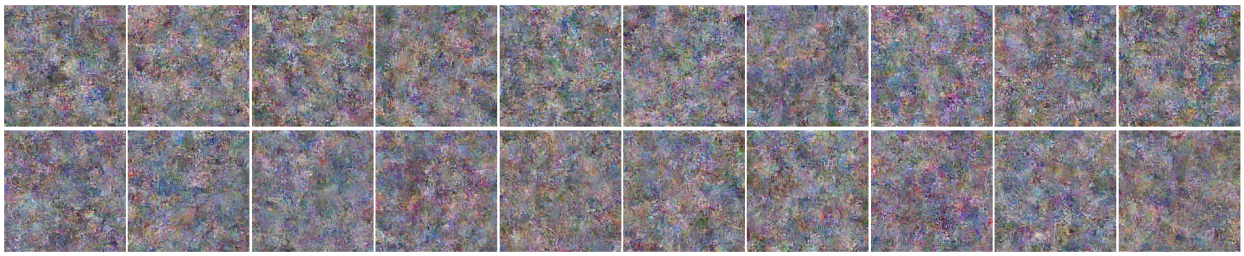

Figure 2. Privatization in PF-LDM for privacy budget $\epsilon = 10$. The top row indicates privatized training samples used to train the global denoiser, while the bottom row indicates samples generated by the global denoiser during PF-LDM sampling. Samples are decoded using the public autoencoder for visualization.

Clients send only these privatized latents to the server. The server then combines the privatized latent datasets $\tilde{Z}_m$ across all $m \in [M]$ clients and trains a shared global denoiser ξ_{θ_g} (lines 8–13 in Algorithm 1). Although the global model is trained as a diffusion model on the privatized latent distribution, its generated samples correspond to noisy intermediate latents rather than clean client-specific latents.

3.2. Sampling with PF-LDM

Image generation in PF-LDM can be performed either *collaboratively* or *non-collaboratively* for each client m. In the collaborative setting, the shared global denoiser first performs reverse denoising over T timesteps to produce an intermediate latent representation, denoted by $\tilde{z}_0$ (lines 3 in Algorithm 2). This intermediate latent is then communicated to the client-specific local denoiser ξ_{θ_m}, which continues the reverse process from timestep t_0 to recover the final clean latent (lines 4–8 in Algorithm 2). The recovered latent is subsequently decoded through the public decoder Dec to generate the final image. In the non-collaborative setting, each client instead performs the full reverse denoising process over all T timesteps using only its own local denoiser ξ_{θ_m}, without relying on the shared global denoiser (lines 9–11 in Algorithm 2). The sampling algorithm S-LDM appears in Appendix B.

This split-sampling strategy is effective because different stages of the diffusion process capture different levels of image structure. The fine-grained image details are perturbed more rapidly in the forward diffusion process, whereas coarser and more global structures tend to persist longer (Patel et al., 2026; Rissanen et al., 2022). Hence, the shared global denoiser can learn large-scale and cross-client structural patterns from the communicated latents without access to client-specific fine-grained details of the latents. The local denoisers then refine these intermediate latents to recover the client-specific details required for personalized image generation. This design allows collaboration to improve generation quality while preserving privacy. The shared model contributes a broadly useful structure, while sensitive details remain recoverable only through the private client model. Without collaboration, local denoisers can still generate samples from their own distributions, but they lose the benefit of the shared structural knowledge learned across clients.

Algorithm 2 Sampling with PF-LDM

input Shared global latent denoiser ξ_{θ_g}, client-specific latent denoiser ξ_{θ_m}, public decoder Dec, local timestep t_0, noise schedule $\{\beta_t, \sigma_t\}_{t=1}^T$, sampling mode $\in \{\text{COLLABORATIVE, NON-COLLABORATIVE}\}$

Set $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$
if mode = COLLABORATIVE **then**
  $\tilde{z}_0 = \text{S-LDM}(\xi_{\theta_g}, T, \{\beta_t, \sigma_t\}_{t=1}^T)$
  Set $z_{t_0} = \tilde{z}_0$
  for $t = t_0, t_0 - 1, \ldots, 1$ **do**
    $z_{t-1} = \frac{1}{\sqrt{\alpha_t}}\Big(z_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\,\xi_{\theta_m}(z_t, t)\Big) + \sigma_t \eta,$
    where $\eta \sim \mathcal{N}(0, I)$ if $t > 1$, else $\eta = 0$
  end for
else if mode = NON-COLLABORATIVE **then**
  $z_0 = \text{S-LDM}(\xi_{\theta_m}, T, \{\beta_t, \sigma_t\}_{t=1}^T)$
end if
Recover the clean latent $\hat{z}_0 = z_0$
Decode the generated image $\hat{x} = \text{Dec}(\hat{z}_0)$

output Generated image $\hat{x}$

3.3. Effect of the Local Timestep t_0 in Collaboration

The local timestep t_0 serves as the main knob controlling the privacy and utility trade-off in PF-LDM. A larger t_0 injects more noise into the communicated latents, which strengthens privacy but reduces the useful information available to the shared global denoiser. This can weaken the benefit of collaboration during split sampling. Conversely, a smaller t_0 preserves more information in the communicated latents, allowing the global denoiser to learn richer shared structure and provide stronger sampling guidance, but at the cost of weaker privacy protection.

4. Privacy Guarantee in PF-LDM

We first state the privacy guarantee of PF-LDM. The guarantee applies to the released shared global denoiser, which is trained only on privatized latent representations. The client-specific denoisers remain local and are not communicated.

Theorem 4.1 (Privacy Guarantee of PF-LDM)**.** *Consider Algorithm 1 with local diffusion timestep t_0, noise schedule $\{\beta_t\}_{t=1}^T$, and latent clipping radius C_z. Let $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$, $\sigma^2 = (1-\bar{\alpha}_{t_0})/\bar{\alpha}_{t_0}$. Then, for any $\delta \in (0,1)$, the shared global denoiser ξ_{θ_g} returned by Algorithm 1 satisfies*

$$\left(\frac{2C_z^2}{\sigma^2} + C_z\sqrt{\frac{8\log(1/\delta)}{\sigma^2}}, \delta\right) \text{-local DP.}$$

Theorem 4.1 gives an explicit upper bound on the privacy parameter ϵ. The quantity $\sigma^2 = (1-\bar{\alpha}_{t_0})/\bar{\alpha}_{t_0}$ is the effective variance of the Gaussian noise. Since $\bar{\alpha}_{t_0}$ decreases as t_0 increases, a larger t_0 yields a larger effective noise variance and therefore stronger privacy. Thus, t_0, together with the clipping radius C_z, controls the privacy and utility trade-off in PF-LDM.

For example, with $T = 1000$, a linear noise schedule, and clipping radius $C_z = 35$, choosing $t_0 = 850$ yields an (ϵ, δ)-local DP guarantee with $\epsilon = 10$ and $\delta = 10^{-5}$. Although $\epsilon = 10$ may appear large compared with common central DP settings, it is a local DP guarantee, which is a stronger privacy protection than sample-level central DP with a trusted server. In practice, even moderate or large local-DP privacy budgets can provide meaningful protection against privacy attacks (Jayaraman et al., 2020; Lowy et al., 2024).

A key advantage of performing privatization in latent space is that the latent representation is much lower-dimensional than the original pixel-space image. As a result, the clipping radius C_z can be chosen much smaller than the corresponding pixel-space clipping radius while still preserving the essential semantic and perceptual information needed for generation. This directly improves the privacy bound, since ϵ scales with C_z. Moreover, as illustrated in Figure 1, clipping in latent space is less destructive than clipping in pixel space: the clipped latent representation can still retain meaningful image structure, whereas pixel-space clipping tends to wash out fine visual details. Therefore, latent-space privatization improves privacy not only by reducing the effective sensitivity, but also by preserving more useful information after clipping.

Figure 3. Samples generated with and without collaboration from client 1. All images are 256×256. (a) Samples generated by the non-collaborative LDM showing that the minority class (female) features are weaker. (b) Samples generated by the collaborative framework of PF-LDM show strong generation in both majority and minority classes.

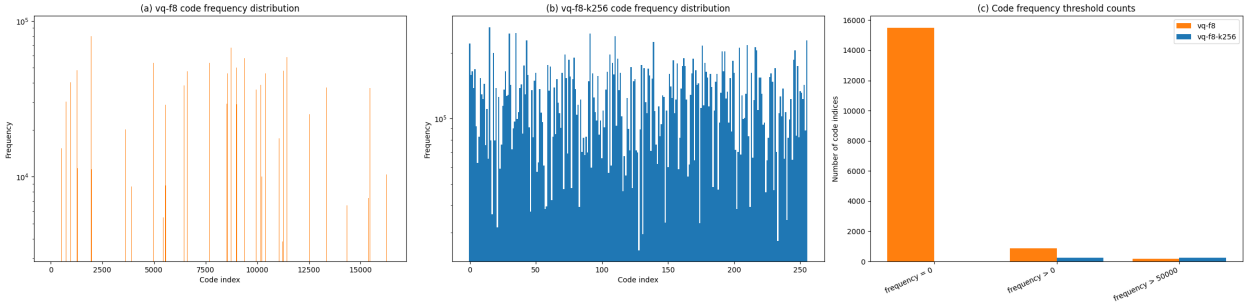

Figure 4. Frequency distribution in VQ-$f8$ regularized latent space for CelebA-HQ dataset. From left, frequency distribution of code indices for (a) VQ-$f8$, (b) VQ-$f8$-k256, and (c) Code indices threshold by frequency.

In Figure 2, we present privatized samples decoded to pixel space in PF-LDM at privacy budget $\epsilon = 10$. The samples remain noisy and lack fine perceptual details, reflecting the strong privacy-preservation of our method.

5. Experiments

5.1. Data Preparation

We evaluate our framework on the CelebA-HQ dataset (Karras et al., 2017) at 256×256 resolution, which contains 30,000 face images. To simulate a decentralized two-client setting, we partition the dataset into two disjoint client datasets with imbalanced class distributions. Each client is assigned one majority group and one minority group in order to model underrepresented local data. Specifically, client 1 contains 5,000 male images as the majority class and 50 female images as the minority class, while client 2 contains 5,000 female images as the majority class and 50 male images as the minority class. This construction induces strong heterogeneity across clients and creates a realistic setting in which each client has only limited access to one subgroup. We show that our framework can enable improved minority class generation without requiring either client to access the other client's private data.

5.2. Choice of Autoencoders

Following Rombach et al. (2022), we evaluate KL-regularized and VQ-regularized autoencoders pretrained on Open-Images (Kuznetsova et al., 2020) with compression factors $f = 4$ and $f = 8$. For $3 \times 256 \times 256$ images, these autoencoders produce latents of size $3 \times 64 \times 64$ for $f = 4$ and $4 \times 32 \times 32$ for $f = 8$. Additional details on the selected autoencoders and their codebook sizes are provided in Appendix C.1.

Table 1. FID scores for PF-LDM with $f8$ compressed autoencoders.

Metric	Major	Minor	Average
KL-$f8$	21.21	23.81	22.51
VQ-$f8$	16.21	21.55	18.88
VQ-$f8$-k256	19.26	22.81	21.03

Table 2. FID scores for three baselines with KL-$f4$ and VQ-$f4$ enabled LDMs.

	KL-$f4$	VQ-$f4$
Baseline	**FID (Maj/Min/Avg)↓**	**FID (Maj/Min/Avg)↓**
NP	16.68 / 14.67 / 15.67	19.91 / 15.47 / 17.69
NC	17.27 / 20.88 / 19.08	20.32 / 23.96 / 22.14
PF-LDM	**18.45 / 19.50 / 18.97**	**16.18 / 20.11 / 18.14**

5.3. Modeling & Training Details

We use a U-Net with 2 layers of ResNet blocks as our main denoiser for all experiments. Attention-based spatial transformer blocks are used at downsampling factors $2, 4$, and 8. Each transformer block has depth 1 and is conditioned on a 512-dimensional context embedding, with 32 channels per attention head. We use class-conditional generation and treat the labels as public information since we care about protecting the image contents and not the designated image labels. For collaborative training, a separate conditioning signal is used with the client-specific local models only to recover from clipped latents generated by the global denoiser.

We train the denoisers following the original latent-diffusion codebase (Rombach et al., 2022). All methods are trained on a linear noise schedule with $T = 1000$. For privatization in the CelebA-HQ dataset, we use $t_0 = 850$ and $C_z = 35$, which yields the privacy budget $\epsilon = 10, \delta = 10^{-5}$. All models are trained for a maximum of 500 epochs with a learning rate 2×10^{-6} on a single NVIDIA A100 (80GB) with AdamW optimization, and the results are collected as the best metric across all checkpoints.

5.4. Baselines

We present three different baselines for comparison following Patel et al. (2026), from private extreme to non-private extreme. Our method resides between these two baselines with a privacy-utility trade-off described (Section 3).

1. **Non-private Centralized Baseline (NP):** Non-private extreme; a single model trained on the union of all clients' datasets, designating no privacy-preservation.
2. **Non-Collaborative Baseline (NC):** Private extreme; independent client-specific models trained only on the client's private dataset, designating extreme privacy and no collaboration.
3. **Collaborative (PF-LDM):** Collaborative baseline without sharing raw data. The data is shared through our privatization method, and sampling is done through the described process in Section 3. An additional conditioning signal is included on the client-side models only to recover clean latents from clipped ones.

5.5. Evaluation Metrics

We evaluate each baseline with the Fréchet Inception Distance (FID) (Heusel et al., 2017; Ho et al., 2020). We report the FID score for the majority and minority groups evaluated on a total of 10,000 generated samples. For reference images, we use a set 10,000 images from the CelebA-HQ dataset, disjoint from the training dataset. We also evaluate accuracy for the downstream classification task on generated images to further evaluate utility.

5.6. Results

Comparative analysis on Autoencoders. Table 1 compares PF-LDM with different autoencoders under the same compression factor, $f = 8$. Under identical training settings, the KL-regularized autoencoder underperforms the VQ-regularized alternatives, even though their latent tensors have the same spatial dimensions. This gap suggests that the structure of the latent distribution, not only its dimensionality, plays an important role in the privacy–utility trade-off.

The KL-regularized autoencoder produces a continuous latent space that is weakly regularized toward a Gaussian prior. In such a space, visual information is represented through continuous variations across latent coordinates. Clipping and isotropic Gaussian noise can therefore perturb these coordinate-level variations directly, making it harder for the denoiser to recover fine visual structure. In contrast, VQ-regularized autoencoders introduce a learned codebook, which encourages latent features to cluster around discrete visual tokens. This codebook structure can make the latent

Table 3. FID results for different privacy budgets using a VQ-$f8$-enabled LDM.

Method		FID (Maj./Min./Avg.)↓
NP	Centralized	18.88 / 15.60 / 17.24
NC	Local-only	18.59 / 22.86 / 20.73
PF-LDM	$\epsilon = 5$	16.73 / 21.84 / 19.28
PF-LDM	$\epsilon = 10$	16.21 / 21.55 / 18.88
PF-LDM	$\epsilon = 50$	14.11 / 20.42 / 17.26
PF-LDM	$\epsilon = 100$	14.50 / 20.21 / 17.36

Table 4. Classification accuracy for CelebA-HQ generated samples with different classifiers.

Classifier	CelebA Baseline	NC Baseline	PF-LDM
CNN	73.24	68.30	**69.62**
ResNET-9	93.78	89.34	**90.54**
ResNET-18	98.54	96.60	**97.12**

representation more stable under perturbation, since the representation is organized around reusable visual patterns rather than arbitrary continuous variations. It can also yield more discriminative features by separating different visual patterns into different codebook regions.

The code-frequency analysis in Figure 4(a) further shows that the larger VQ-$f8$ codebook is sparsely activated, suggesting that different codewords capture more specialized visual patterns. Even the smaller VQ-$f8$-k256 autoencoder, which uses a more constrained and less sparse codebook, outperforms KL-$f8$ under the same compression factor. These results indicate that VQ-regularized latents are better suited for PF-LDM, likely because their clustered and more discriminative feature structure is more robust to latent clipping and privacy noise.

Baseline Comparison. In Table 2, we present the main baseline comparison for both KL-regularized and VQ-regularized LDMs. The non-private baseline represents the upper-bound setting without privacy preservation, while the non-collaborative baseline represents the private extreme where each client trains without benefiting from cross-client collaboration. The results show that our collaborative method consistently improves over the non-collaborative setting, particularly for the minority group, and achieves performance close to the non-private baseline. We provide additional qualitative comparisons with the non-collaborative methods in Figure 3.

Privacy–Utility Trade-off. We study the privacy–utility trade-off of PF-LDM by varying the privacy budget $\epsilon \in \{5, 10, 50, 100\}$ for VQ-$f8$-based LDMs. Table 3 shows that PF-LDM consistently outperforms the non-collaborative baseline across all privacy budgets, demonstrating that collaboration remains beneficial even under strong privacy constraints. Moreover, as the privacy budget increases, the FID of PF-LDM improves and approaches the non-private centralized baseline. Notably, even at the strict privacy setting of $\epsilon = 5$, PF-LDM achieves substantially better average FID than the non-collaborative baseline, indicating that the proposed collaborative mechanism retains meaningful utility while enforcing strong privacy.

Downstream task evaluation. We further evaluate the utility of our framework with downstream task performance on generated samples and the original CelebAHQ dataset. For the original CelebAHQ dataset, we sample 5000 male and 5000 female images to train classifiers, while for the Non-collaborative and Collaborative baselines, we generate 5000 samples for each of the male and female classes. We train three different classifiers: a 3-layer CNN with average pooling and a linear head, a ResNet-9, and a ResNet-18 pretrained on ImageNet. From Table 4, we can see that the CelebA baseline establishes an upper bound for downstream accuracy, while our collaborative framework not only outperforms the non-collaborative baseline, but also achieves downstream performance close to the upper bound.

6. Conclusion

In this work, we present a framework for decentralized training of latent diffusion models with formal privacy guarantees. Our framework takes advantage of latent space image compression to improve the privacy-utility trade-off in high-dimensional image generation through personalized local models and a shared global model. With empirical evaluations, we demonstrate that our framework can generate high-dimensional images and improve performance for minority groups while maintaining the privacy of each client in collaboration. Future developments in our work involve providing utility guarantees, designing latent space, and applying the framework to super-resolution images.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*,

pp. 308–318, 2016.

Allmendinger, S., Zipperling, D., Struppek, L., and Kühl, N. Collafuse: Collaborative diffusion models. *arXiv preprint arXiv:2406.14429*, 2024.

Bauer, A., Trapp, S., Stenger, M., Leppich, R., Kounev, S., Leznik, M., Chard, K., and Foster, I. Comprehensive exploration of synthetic data generation: A survey, 2024. URL https://arxiv.org/abs/2401.02524.

Boenisch, F., Dziedzic, A., Schuster, R., Shamsabadi, A. S., Shumailov, I., and Papernot, N. 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.

Cal. Legis. Serv. The California Privacy Rights Act of 2020, 2020. URL https://oag.ca.gov/system/files/initiatives/pdfs/19-0021A1%20%28Consumer%20Privacy%20-%20Version%203%29_1.pdf.

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramèr, F., Balle, B., Ippolito, D., and Wallace, E. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

Chen, H., Han, Y., Chen, F., Li, X., Wang, Y., Wang, J., Wang, Z., Liu, Z., Zou, D., and Raj, B. Masked autoencoders are effective tokenizers for diffusion models, 2025. URL https://arxiv.org/abs/2502.03444.

Ching, T., Himmelstein, D. S., Beaulieu-Jones, B. K., Kalinin, A. A., Do, B. T., Way, G. P., Ferrero, E., Agapow, P.-M., Zietz, M., Hoffman, M. M., et al. Opportunities and obstacles for deep learning in biology and medicine. *Journal of the Royal Society Interface*, 15(141):20170387, 2018.

Clark, M. Machine learning needs big data to revolutionise drug discovery. *Drug Discovery World*, 2021. URL https://www.ddw-online.com/machine-learning-needs-big-data-to-revolutionise-drug-discovery-14537-202111/. Accessed: 2024-11-10.

de Goede, M., Cox, B., and Decouchant, J. Training diffusion models with federated learning. *arXiv preprint arXiv:2406.12575*, 2024.

Dockhorn, T., Cao, T., Vahdat, A., and Kreis, K. Differentially private diffusion models. *Transactions on Machine Learning Research*, 2023.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pp. 265–284. Springer, 2006.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12873–12883, 2021.

European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance), May 2016. URL http://data.europa.eu/eli/reg/2016/679/2016-05-04/eng. Legislative Body: OP_DATPRO.

Fowl, L., Geiping, J., Czaja, W., Goldblum, M., and Goldstein, T. Robbing the fed: Directly obtaining private data in federated learning with modified models. *arXiv preprint arXiv:2110.13057*, 2021.

Geiping, J., Bauermeister, H., Dröge, H., and Moeller, M. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in neural information processing systems*, 33:16937–16947, 2020.

Grynbaum, M. M. and Mac, R. The times sues openai and microsoft over ai use of copyrighted work. *The New York Times*, 27, 2023.

Gu, X., Du, C., Pang, T., Li, C., Lin, M., and Wang, Y. On memorization in diffusion models. *arXiv preprint arXiv:2310.02664*, 2023.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jayaraman, B., Wang, L., Knipmeyer, K., Gu, Q., and Evans, D. Revisiting membership inference under realistic assumptions. *arXiv preprint arXiv:2005.10881*, 2020.

Jiang, T., Li, C., Ma, F., and Wang, T. Rapid: Retrieval augmented training of differentially private diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

Jothiraj, F. V. S. and Mashhadi, A. Phoenix: A federated generative diffusion model. *arXiv preprint arXiv:2306.04098*, 2023.

Kairouz, P., McMahan, B., Song, S., Thakkar, O., Thakurta, A., and Xu, Z. Practical and private (deep) learning without sampling or shuffling. In *International Conference on Machine Learning*, pp. 5213–5225. PMLR, 2021a.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021b.

Kapania, S., Ballard, S., Kessler, A., and Vaughan, J. W. Examining the expanding role of synthetic data throughout the ai development pipeline. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, pp. 45–60, 2025.

Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

Kasiviswanathan, S. P., Lee, H. K., Nissim, K., Raskhodnikova, S., and Smith, A. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes, 2022. URL https://arxiv.org/abs/1312.6114.

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., Duerig, T., and Ferrari, V. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128, 03 2020. doi: 10.1007/s11263-020-01316-z.

Li, K., Gong, C., Li, Z., Zhao, Y., Hou, X., and Wang, T. Privimage: Differentially private synthetic image generation using diffusion models with semantic-aware pretraining. In *33rd USENIX Security Symposium (USENIX Security 24)*, pp. 4837–4854, 2024.

Liu, M. F., Lyu, S., Vinaroz, M., and Park, M. Differentially private latent diffusion models. *Transactions on Machine Learning Research*, 2024.

Lowy, A., Li, Z., Liu, J., Koike-Akino, T., Parsons, K., and Wang, Y. Why does differential privacy with large epsilon defend against practical membership inference attacks? *arXiv preprint arXiv:2402.09540*, 2024.

Lu, Y., Chen, L., Zhang, Y., Shen, M., Wang, H., Wang, X., van Rechem, C., Fu, T., and Wei, W. Machine learning for synthetic data generation: a review. *arXiv preprint arXiv:2302.04062*, 2023.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. Pmlr, 2017.

McMahan, H. B., Ramage, D., Talwar, K., and Zhang, L. Learning differentially private recurrent language models. In *International Conference on Learning Representations*, 2018.

Melis, L., Song, C., De Cristofaro, E., and Shmatikov, V. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE symposium on security and privacy (SP)*, pp. 691–706. IEEE, 2019.

Mironov, I. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pp. 263–275. IEEE, 2017.

Patel, K. K., Jiang, B., Kabir, A. F. M. M., Zhang, W., Zou, D., and Wang, L. Personalized federated training of diffusion models with privacy guarantees. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Pichler, G., Romanelli, M., Vega, L. R., and Piantanida, P. Perfectly accurate membership inference by a dishonest central server in federated learning. *IEEE Transactions on Dependable and Secure Computing*, 21(4):4290–4296, 2023.

Prakash, A. Exploring new chemical space for the treatments of tomorrow. *American Pharmaceutical Review*, 2023. URL https://www.americanpharmaceuticalreview.com/Featured-Articles/597596-Exploring-New-Chemical-Space-for-the-Treatments-of-Tomorrow/. Accessed: 2024-11-10.

Razavi, A., van den Oord, A., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2, 2019. URL https://arxiv.org/abs/1906.00446.

Rieke, N., Hancox, J., Li, W., Milletari, F., Roth, H. R., Albarqouni, S., Bakas, S., Galtier, M. N., Landman, B. A., Maier-Hein, K., et al. The future of digital health with federated learning. *NPJ digital medicine*, 3(1):1–7, 2020.

Rissanen, S., Heinonen, M., and Solin, A. Generative modelling with inverse heat dissipation. *arXiv preprint arXiv:2206.13397*, 2022.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Understanding and mitigating copying in diffusion models. *Advances in Neural Information Processing Systems*, 36:47783–47803, 2023.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Wang, W., Bell, J. J., Dotson, J. P., and Schweidel, D. A. Generative ai and artists: Consumer preferences for style and fair compensation. *Available at SSRN 4428509*, 2023.

A. Related Works

Image Compression with Autoencoders. The perceptual compression in LDMs is achieved by the means of various autoencoders such as Variational Autoencoder (VAE), Vector-quantized Autoencoders (VQ-VAE), VQGAN, etc. (Kingma & Welling, 2022; Razavi et al., 2019; Van Den Oord et al., 2017; Chen et al., 2025). These autoencoders enable LDMs to apply the denoising process to high-resolution image synthesis in a compressed latent space. Hence, they act as an image compression for perceptual details in a lower dimensional latent space. In order to avoid arbitrarily high-variance latent spaces, LDMs use mainly two types of regularizations for autoencoders: KL-reg and VQ-reg (Rombach et al., 2022). KL-reg imposes a slight KL-penalty toward a standard normal on the learned latent, while VQ-reg uses a vector quantization layer within the decoder which can be interpreted as a VQGAN. While the original LDM work focuses on these two kinds of compression techniques, several other techniques had emerged such as Masked Autoencoders that use tokenization of images in the latent space similar to tokenization in LLMs (Chen et al., 2025). For our work, we focus on the KL-reg and VQ-reg adapted from the original LDM work and treat this as a public autoencoder trained on OpenImages dataset (Kuznetsova et al., 2020) because of their versatility. The compression of autoencoders is defined by a compression factor f, which denotes the factor of compression of dimensionality from the pixel space to the latent space. Larger f such as $f = 8$ or $f = 16$ provide stronger compression at the cost of reconstruction fidelity, while smaller f such as $f = 4$ provide lower compression but better fidelity. For our work, we evaluate $f = 4$ and $f = 8$.

Image Generation with Latent Diffusion Models. In recent years, we have seen rapid progress in image synthesis, with diffusion models (DMs) emerging as a dominant approach due to their stable training dynamics (Ho et al., 2020). However, DMs face a limitation in high-resolution images due to the computational requirement in sampling from the raw pixel space. To overcome the limitation, Rombach et al. (Rombach et al., 2022) introduced latent diffusion models (LDMs), addressing the challenge by mapping images into the latent space of a pretrained autoencoder. The quality and practicality of latent diffusion depend strongly on the underlying autoencoding scheme. Prior work has explored both continuous latent representations, typically based on variational autoencoders (Kingma & Welling, 2022), and discrete latent representations, such as vector-quantized autoencoders (Van Den Oord et al., 2017). Our work builds on this line of research, but differs in the objective: we study the privacy-preservation in LDMs for decentralized and collaborative framework.

Federated Learning in Diffusion Models. A growing body of work studies DMs in federated environments, showcasing collaboration in image generation without pooling data across different clients. Existing approaches explored FedAvg-style training of DMs, personalized diffusion architectures, and split generation pipelines. de Goede et al. (de Goede et al., 2024) adapt federated averaging to train DDPM collaboratively while also proposing communication-efficient variants for the UNet structure called USplit, ULatDec, and UDec to reduce number of parameters exchanged. They report 74% less parameter exchange than naive FedAvg while keeping the image quality close to centralized baselines. Jothiraj et al. (Jothiraj & Mashhadi, 2023) in their work *Phoenix* show that for collaboration, the individual client data do not need to be IID, emphasizing data diversity in collaborative frameworks. Lastly, Allmendinger et al. (Allmendinger et al., 2024) in CollaFuse show that collaboration can reduce client-side computational burden by moving expensive computation to a shared server. While these works focus on the utility of collaboration, none of them focus on the privacy factor in collaborative training. Our work takes motivation from the utility of these collaborative frameworks, but with the goal of privacy protection for each client in a collaborative framework.

Differential Privacy in Diffusion Generative Models. As discussed in 1, DMs can memorize training examples and reproduce them during generation, compromising the privacy of the training data. To tackle this, recent works have focused on training DMs with differential privacy guarantees. Broadly, these methods can be grouped into two categories: the first trains DMs privately through gradient clipping and noise injection, while the second leverages public-data pretraining before privately fine-tuning selected model components. Earlier works, such as DPDM uses differentially private SGD (DP-SGD) to apply clip and noise to gradients during training, like standard private deep learning (Dockhorn et al., 2023). DP-LDM extends this idea to a latent space to take advantage of latent diffusion by combining public pretraining with private fine-tuning of only the attention modules, improving scalability and efficiency relative to DPDM (Liu et al., 2024). PrivImage views the privacy-utility problem from a public-data pretraining side, proposing the selection of a public dataset that is semantically aligned with the sensitive private data for pretraining, followed by private fine-tuning with DP-SGD (Li et al., 2024). RAPID builds a knowledge base of trajectories from

Algorithm 3 Training of Latent Diffusion Model (T-LDM)

input Latent dataset $Z = \{z_0^i\}_{i=1}^N$, noise schedule $\{\beta_t\}_{t=1}^T$, denoiser parameters θ, learning rate η
output Trained latent denoiser ϵ_θ
1: Set $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$
2: **for** each training iteration **do**
3: Sample mini-batch $\{z_0^i\}_{i\in\mathcal{B}} \sim Z$, timesteps $t_i \sim \text{Uniform}(\{1, \dots, T\})$, and noise $\epsilon_i \sim \mathcal{N}(0, I)$
4: Form noisy latents $z_{t_i}^i = \sqrt{\bar{\alpha}_{t_i}} z_0^i + \sqrt{1 - \bar{\alpha}_{t_i}} \epsilon_i$
5: Compute noise-prediction loss

$$\mathcal{L}_{\text{LDM}}(\theta) = \frac{1}{|\mathcal{B}|} \sum_{i\in\mathcal{B}} \left\| \epsilon_i - \epsilon_\theta(z_{t_i}^i, t_i) \right\|_2^2$$

6: Update $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{LDM}}(\theta)$
7: **end for**
output Learned denoiser ϵ_θ

Table 5. Choice of autoencoders used in our framework.

AE	**KL-$f4$**	**KL-$f8$**	**VQ-$f4$**	**VQ-$f8$**	**VQ-$f8$-k256**
Latent shape	3×64×64	4×32×32	3×64×64	4×32×32	4×32×32
Codebook size	–	–	8,192	16,384	256

public data, and then, during private DM training, using the early denoising steps as queries to retrieve similar public trajectories (Jiang et al., 2025). This helps their method to focus the expensive private training on the later steps. While DPDM, PrivImage, and RAPID provide methods for privatized training of DMs, they lack experiments that extends to higher resolution image generation, particularly limited upto 64×64 images. DP-LDM is the only one among these works to report experiments at higher resolution, specifically 256×256 image generation; however, all of these works are developed in centralized settings and therefore do not address the decentralized challenges of data imbalance, heterogeneity, and rare local modes discussed in Section 1. Patel et al. (2026) address this gap through PFDM, a decentralized diffusion framework governed by formal privacy and utility guarantees, which we discuss next.

B. Missing Details from Section 2

Denoising Latent Diffusion Models (LDMs). The training of LDMs follows a two-stage procedure. First, an image compressor maps images from pixel space to a lower-dimensional latent space. Given an image dataset $X = \{x_0^i\}_{i=1}^N$, an autoencoder with encoder Enc and decoder Dec encodes each image as $z_0^i = \text{Enc}(x_0^i)$, forming the latent dataset $Z = \{z_0^i\}_{i=1}^N$. Second, a denoiser is trained directly on this latent dataset to generate the clean latents that can be converted to image using the Dec. The process involves sampling a mini-batch from Z, timesteps t_i from a uniform distribution, and noise ϵ_i from a standard distribution (line 3 in Algorithm 3). The sampled noise and timesteps are then used to construct noisy latents in the diffusion forward process, and the denoiser is trained using a noise-prediction objective (lines 4–6 in Algorithm 3).

To generate clean latents from the trained LDM, sampling follows the reverse diffusion process in latent space. Starting from Gaussian noise $\hat{z}_T \sim \mathcal{N}(0, I)$ (line 2 in Algorithm 4), the denoiser ϵ_θ iteratively predicts the noise component in $\hat{z}_t$. Following the DDPM noise-parameterization trick (Ho et al., 2020), this predicted noise is used to parameterize the reverse mean $\mu_\theta(\hat{z}_t, t)$ instead of directly predicting $\hat{z}_{t-1}$ (lines 4–5). The previous latent $\hat{z}_{t-1}$ is then sampled by adding Gaussian noise at intermediate timesteps, while setting the noise to zero at the final step (lines 6–7). After all reverse steps, the algorithm returns the generated latent $\hat{z}_0$, which is decoded to image space as $\hat{x} = \text{Dec}(\hat{z}_0)$.

C. Missing Details from Section 5

C.1. Choice of Autoencoders

Following Rombach et al. (2022), we evaluate two types of autoencoders for our framework: KL-regularized and VQ-regularized. Particularly, our work uses autoencoders pretrained on the OpenImages dataset (Kuznetsova et al., 2020) for two different compression factors, $f = 4$ and $f = 8$. For image dimension of $3 \times 256 \times 256$, we get latents in the shape $3 \times 64 \times 64$ for $f = 4$ and $4 \times 32 \times 32$ for $f = 8$ compression. Hence, our KL-$f4$ and KL-$f8$

Algorithm 4 Sampling from Latent Diffusion Model (S-LDM)

input Trained latent denoiser ξ_θ, noise schedule $\{\beta_t\}_{t=1}^T$
output Generated latent sample $\hat{z}_0$
1: Set $\alpha_t = 1 - \beta_t$, $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$, and $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$
2: Sample initial latent $\hat{z}_T \sim \mathcal{N}(0, I)$
3: **for** $t = T, T-1, \ldots, 1$ **do**
4: Predict noise $\hat{\epsilon}_t = \xi_\theta(\hat{z}_t, t)$
5: Compute mean

$$\mu_\theta(\hat{z}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\hat{z}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\hat{\epsilon}_t\right)$$

6: Sample $\eta_t \sim \mathcal{N}(0, I)$ if $t > 1$, else set $\eta_t = 0$
7: Compute previous latent

$$\hat{z}_{t-1} = \mu_\theta(\hat{z}_t, t) + \sqrt{\tilde{\beta}_t}\,\eta_t$$

8: **end for**
output Generated latent $\hat{z}_0$

autoencoders transform the images to a standard Gaussian latent of the mentioned dimension, while the VQ-$f4$ and VQ-$f8$ autoencoders transform images into a vector of discrete code indices. We use two different set of VQ-$f8$ autoencoders based on the size of the codebook Z (Rombach et al., 2022), the first one has a 16,384 indices in the codebook, while the second one has 256 indices in the codebook. As for the VQ-$f4$ autoencoder, we use 8,192 indices for the discrete codebook. To summarize, our choice of autoencoders is listed in detail in Table 5.

D. Proof of Theorem 4.1

We first recall the notion of Rényi differential privacy (RDP) (Mironov, 2017), which is convenient for analyzing the Gaussian mechanism. We will derive an RDP guarantee for the privatization step and then convert it to an (ϵ, δ)-DP guarantee.

Definition D.1 (Rényi Differential Privacy)**.** A randomized mechanism $\mathcal{A}$ satisfies (γ, ρ)-Rényi differential privacy, for $\gamma > 1$ and $\rho > 0$, if for any adjacent datasets $D, D' \in \mathcal{D}$ that differ in one element,

$$D_\gamma\big(\mathcal{A}(D) \,\|\, \mathcal{A}(D')\big) = \frac{1}{\gamma - 1}\log \mathbb{E}_{o\sim\mathcal{A}(D')}\left[\left(\frac{\mathbb{P}[\mathcal{A}(D) = o]}{\mathbb{P}[\mathcal{A}(D') = o]}\right)^\gamma\right] \leq \rho.$$

The following standard result converts RDP to approximate DP.

Lemma D.2 (RDP to DP (Mironov, 2017))**.** *If a randomized mechanism $\mathcal{A}$ satisfies (γ, ρ)-RDP, then for any $\delta \in (0, 1)$, $\mathcal{A}$ satisfies*

$$\left(\rho + \frac{\log(1/\delta)}{\gamma - 1}, \delta\right)\text{-DP.}$$

We also use the RDP guarantee of the Gaussian mechanism.

Lemma D.3 (Gaussian Mechanism (Mironov, 2017))**.** *Let q be a function with ℓ_2-sensitivity*

$$S_2 = \sup_{D,D'} \|q(D) - q(D')\|_2,$$

where the supremum is over adjacent datasets D, D'. Then the Gaussian mechanism

$$\mathcal{A}(D) = q(D) + z, \qquad z \sim \mathcal{N}(0, \sigma^2 I),$$

satisfies

$$\left(\gamma, \frac{\gamma S_2^2}{2\sigma^2}\right)\text{-RDP.}$$

Proof of Theorem 4.1. The shared global denoiser is trained only on the privatized latent dataset

$$\widetilde{Z} = \{\widetilde{Z}_m\}_{m\in[M]}.$$

For each client m and sample i, the privatized latent is generated as

$$\widetilde{z}_0^{i,m} = \sqrt{\bar{\alpha}_{t_0}}\, z_0^{i,m} + \sqrt{1-\bar{\alpha}_{t_0}}\, \xi, \qquad \xi \sim \mathcal{N}(0, I).$$

Thus, the privatization step is a Gaussian mechanism applied to the clipped latent representation. Assuming $\|z_0^{i,m}\|_2 \leq C_z$, the sensitivity of the deterministic part

$$q(z_0) = \sqrt{\bar{\alpha}_{t_0}}\, z_0$$

is bounded by

$$S_2 \leq \sup_{\|z\|_2, \|z'\|_2 \leq C_z} \sqrt{\bar{\alpha}_{t_0}} \|z - z'\|_2 \leq 2\sqrt{\bar{\alpha}_{t_0}}\, C_z.$$

The Gaussian noise variance is

$$\sigma^2 = 1 - \bar{\alpha}_{t_0}.$$

By Lemma D.3, each privatized latent sample satisfies

$$\left(\gamma, \frac{\gamma S_2^2}{2\sigma^2}\right)\text{-RDP} = \left(\gamma, \gamma \cdot \frac{2\bar{\alpha}_{t_0} C^2}{1-\bar{\alpha}_{t_0}}\right)\text{-RDP}.$$

Let

$$\tau = \frac{2\bar{\alpha}_{t_0} C^2}{1-\bar{\alpha}_{t_0}}.$$

Then each privatized latent is $(\gamma, \gamma\tau)$-RDP. Applying Lemma D.2, we obtain an (ϵ, δ)-DP guarantee with

$$\epsilon = \gamma\tau + \frac{\log(1/\delta)}{\gamma - 1}.$$

Optimizing over $\gamma > 1$ gives

$$\gamma = 1 + \sqrt{\frac{\log(1/\delta)}{\tau}},$$

and therefore

$$\epsilon = \tau + 2\sqrt{\tau \log(1/\delta)}.$$

Substituting the value of τ, each privatized latent sample satisfies

$$\left(\frac{2\bar{\alpha}_{t_0} C^2}{1-\bar{\alpha}_{t_0}} + C\sqrt{\frac{8\bar{\alpha}_{t_0}\log(1/\delta)}{1-\bar{\alpha}_{t_0}}}, \delta\right)\text{-DP}.$$

Since this randomization is applied locally to each sample before communication, the construction of $\widetilde{Z}$ satisfies the same per-sample local DP guarantee. Finally, the shared global denoiser is trained only as a function of $\widetilde{Z}$. Therefore, by the post-processing property of differential privacy, the learned shared denoiser satisfies

$$\left(\frac{2\bar{\alpha}_{t_0} C_z^2}{1-\bar{\alpha}_{t_0}} + C_z\sqrt{\frac{8\bar{\alpha}_{t_0}\log(1/\delta)}{1-\bar{\alpha}_{t_0}}}, \delta\right)\text{-local DP}.$$

This proves the theorem. □