Hierarchical Collaborative Fine-Tuning for Personalized Edge-End Hybrid Inference of Stable Diffusion Models

Nan Li [1], Wanting Yang [1], Marie Siew [1],
Zehui Xiong [2], Binbin Chen [1], Shiwen Mao [3], Kwok-Yan Lam [4]
[1]Singapore University of Technology and Design, Singapore
[2]Queen's University Belfast, UK
[3]Auburn University, USA
[4]Nanyang Technological University, Singapore
nan_li@mymail.sutd.edu.sg, {wanting_yang, marie_siew, binbin_chen}@sutd.edu.sg,
z.xiong@qub.ac.uk, smao@auburn.edu, kwokyan.lam@ntu.edu.sg

Abstract

Diffusion models (DMs) have emerged as powerful Artificial Intelligence Generated Content (AIGC) tools for high-quality image synthesis. However, achieving effective edge personalization faces challenges: heterogeneous user preferences, limited local data, and intensive computational demands on resource-constrained devices. To bridge this gap, we first highlight the limitations of existing works in communication efficiency and scalability, and then introduce an edge-assisted collaborative fine-tuning framework, built upon Low-Rank Adaptation (LoRA) for parameter-efficient local tuning. Within a federated learning (FL) framework, we jointly train user-specific models on edge devices and a global model on the server, enabling collaborative personalization while preserving data privacy. The shared global model is enriched with multiple LoRA adapters and can be employed in a hybrid inference process to enhance communication efficiency. To mitigate feature distribution shifts caused by style diversity, the server performs hierarchical client clustering with intra-cluster aggregation for enhanced personalization and inter-cluster interaction for cross-style alignment. Beyond improving inference efficiency, our framework also addresses privacy concerns: transmitting prompts that contain style or label information to a semi-trusted server could inadvertently expose user data. To mitigate this, we derive embeddings from user-specified keywords, reducing the risk of revealing sensitive dataset details. Evaluations show that our framework achieves accelerated convergence and scalable multi-user personalization, making it a practical solution for edge-constrained AIGC services.

1 Introduction

Artificial Intelligence Generated Content (AIGC) models, capable of multi-modal content generation (e.g., text-to-image, image-to-image, and text-to-video), have captured widespread attention due to their capabilities in synthesizing contextually intelligent content. For instance, state-of-the-art image synthesis foundation models like DALL·E, Stable Diffusion Models (SDMs), and GPT-4o leverage advanced architectures to generate high-fidelity images from concise textual prompts.

Driven by real-world application demands, these models are shifting focus from general-purpose tasks (e.g., image synthesis and style transfer) (Ahn et al. 2024; Frenkel et al. 2024) to client-side personalized content generation, emphasizing the need for outputs to adapt to user attributes (e.g., age or preferences) or to be tailored to specialized knowledge domains. For instance, when generating a news poster from an interview video, editors prioritize factual precision, while social media creators emphasize visual appeal. Textual Inversion (Gal et al. 2022) and DreamBooth (Ruiz et al. 2023) enable few-shot (3–5 images) adaptation for such scenarios, but their parameter-isolated learning paradigm—independently optimized per-user embeddings—poses a barrier for dynamic edge deployment.

Unlike Generative Adversarial Networks (GANs), which generate samples in a single forward pass, Diffusion Models (DMs) rely on a computationally intensive iterative denoising process (typically 100 to 1000 steps). This iterative nature, coupled with the considerable storage demands of AIGC models, poses challenges for resource-limited end devices. To address these challenges, in the conventional setting (Fig. 1a), users typically upload raw data and prompts to cloud or capable edge servers for full inference, which, however, raises concerns about raw data leakage. With growing user expectations for low latency and privacy-preserving generation process, server-client hybrid inference frameworks are gaining attention in recent research (Du et al. 2023; Xie et al. 2025; Yan et al. 2024; Yang et al. 2025).

Hybrid inference frameworks aim to distribute computation between the server (for initial processing) and devices (for local denoising), based on task-specific prompts, as depicted in Fig. 1(b-d). However, under a semi-honest server setting, current hybrid implementations incur (i) a certain degree of **storage redundancy** and (ii) **potential information leakage** stemming from user-submitted interaction data (e.g., hybrid inference queries). In particular, EC-Diff (Xie et al. 2025) explores cloud-side strategies (Fig. 1b), yet becomes storage-intensive when scaled to multi-user cases due to maintaining one model per user, and further poses data exposure risks by requiring the upload of plaintext prompts that may expose user-specific preferences (e.g., diagnostic terms like '*dermatofibroma*'). FedBip (Chen et al. 2025)

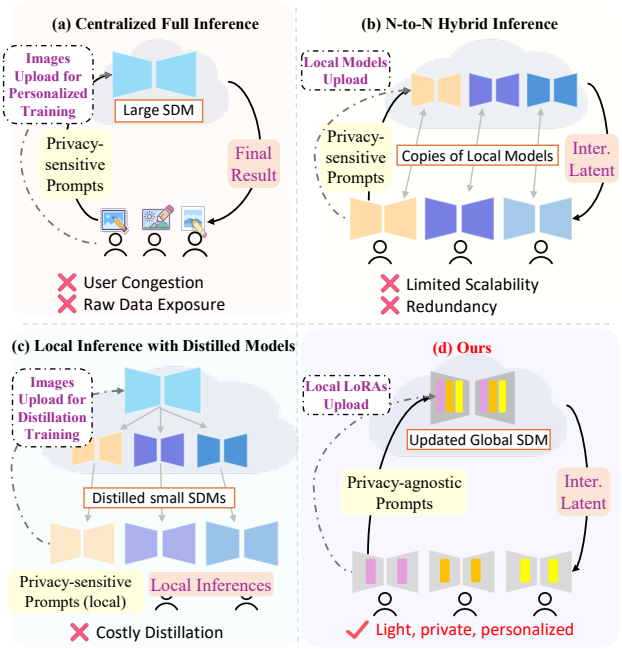

Figure 1: **Four key methods for accelerating the inference process locally of DMs.** Among four mainstream approaches that depend on cloud-side inference, per-user model storage, or user-specific model distillation, our proposed "1-to-N Hybrid Inference" integrates multiple LoRAs into a global DM, enabling parallel multi-user inference via a shared latent—without requiring raw data upload or exposing sensitive prompts.

also requires sharing domain- and instance-level plaintext prompts for server inference. Hybrid-SD (Yan et al. 2024) requires distilling a separate edge model per user (Fig. 1c), causing costly computation on the cloud.

Building upon this, our work focuses on reducing server-side storage without sacrificing personalization accuracy in edge-AIGC systems, as depicted in Fig. 1(d). Specifically, our framework supports lightweight local fine-tuning through Low-Rank Adaptation (LoRA) (Hu et al. 2022), ensuring privacy by uploading only model updates and privacy-agnostic prompts for collaborative training without exposing raw data. By updating only a small subset of parameters while keeping the main model frozen, LoRA significantly reduces computational overhead. Federated Learning (FL) (Chen et al. 2023) enables the collaborative training of personalized local models and the construction of a shared global model, which can be further exploited for hybrid inference across users with similar tasks, minimizing redundant server-side storage.

Under the semi-trusted server assumption, we leverage DMs as the generative backbone and propose an FL-based fine-tuning framework for collaborative edge personalization. The key contributions are as follows:

- We first emphasize the emerging trend for personalized content synthesis in edge-AIGC systems, while discussing the limitations of existing methods. We investigate two key challenges in (i) the performance degradation of standard and personalized FL under **increasing data heterogeneity** and **the growing scale of AIGC model architectures**, and (ii) the **protection of user preferences** contained in user-submitted information such as prompts (Section II).
- We then present our cluster-aware hierarchical federated aggregation framework for scalable multi-user edge personalization. It employs intra-cluster aggregation for users sharing the same style to enhance personalized feature representation; inter-cluster aggregation is subsequently performed to generate mixed-style personalized content or produce distribution-neutral results to accelerate hybrid inference (Section III).
- Experiments on real-world data confirm that our multi-style LoRA-enhanced global model shows around 40% better latent space alignment (best-case 0.6× Fréchet Inception Distance (FID) on sketch generation) than the baseline. It successfully serves multiple devices for parallel personalized generation, reducing edge-side computation while preserving style precision (Section IV).

2 System Overview

While integrating FL into edge-AIGC frameworks offers promising benefits—particularly in preserving the privacy of raw data and supporting multi-user coordination—it simultaneously poses several technical challenges. In response to these challenges outlined below, we design a cluster-aware hierarchical architecture that enables scalable and personalized AIGC deployment across heterogeneous edge devices.

2.1 Problem Formulation

We identify three key design drivers that shape our design:

Design Driver 1 – Dual Heterogeneity: Feature Distribution & Device Resource Availability. When targeting multi-user personalization in FL-based edge-AIGC systems, heterogeneity extends beyond the traditional label heterogeneity assumption in FL: (1) *Feature-level heterogeneity*—even identically labeled inputs can exhibit varying feature distributions, creating complex non-IID effects. For example, data may differ fundamentally in both characteristics (e.g., cartoon stylization for primary school education vs. photorealism for undergraduate medical education) and structure (e.g., 2D sequential layouts vs. 3D spatial organizations). (2) *Device-driven heterogeneity.* High-capacity devices prefer to employ higher LoRA ranks (64/128) for richer parameterization, while resource-limited ones use lower ranks (4/8/16) to conserve resources. Such heterogeneity undermines standard aggregation methods (Wang et al. 2024); for example, FedAvg fails to align divergent LoRA parameters without tailored adaptation (Fig. 2).

To mitigate the performance degradation caused by such heterogeneity, clustering-based approaches can facilitate more efficient model aggregation by grouping clients with similar tasks or data style preferences. Our architecture aims to avoid the limitations of conventional client clustering methods (Yuan et al. 2025; Tu, Wang, and Hu 2024), which typically rely on the need for pre-defined k,client-uploaded

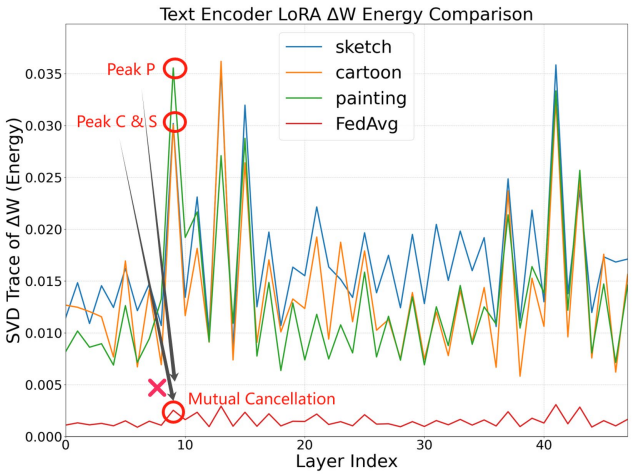

Figure 2: **Feature energy (SVD) of LoRAs vs. FedAvg by layer.** A higher value indicates stronger personalized feature representation. After aggregating LoRA adapters trained on different styles using FedAvg, the resulting model exhibits significantly weakened personalization, reflecting severe feature dilution across styles.

statistics, or server-side public datasets—practices that introduce privacy risks and scalability challenges in real-world deployments.

Design Driver 2 – Classic PFL Methods Fail in AIGC. Personalized FL (PFL) frameworks enhance standard FL by supporting client-specific personalization, aligning with our goal. Common PFL approaches (Tan et al. 2022) include adding a regularization term, performing additional fine-tuning on each client, or decoupling parameters into globally shared and client-specific components. While these approaches have shown success with conventional neural networks (e.g., CNNs), they face challenges when applied to AIGC due to fundamental differences in model complexity and scale. For example, Phoenix (Stanley Jothiraj and Mashhadi 2024) adopts parameter decoupling by assigning the U-Net's final decoder block for client-specific adaptation while keeping the other layers shared in the global model. However, this design overlooks the critical role of attention layers in capturing fine-grained features (Frenkel et al. 2024), limiting its effectiveness for nuanced personalization.

LoRA is employed in our framework to update the attention-related layers throughout the UNet network, enabling efficient learning of user-specific image feature representations. Unlike methods that require pinpointing specific personalized layers, this approach provides greater scalability and flexibility.

Design Driver 3 – Privacy Risks in Hybrid-Inference Requests. During a hybrid inference-based content generation, local clients inevitably upload their generation requests to coordinate with the server-side model. However, directly transmitting plaintext prompts to semi-trusted servers introduces privacy risks. When such requests include explicit style descriptors or domain-specific category labels (e.g., artistic styles or medical terms like "*dermatofibroma*"), they

Algorithm 1 Cluster-Aware FL for Personalized SDMs

Require: Pre-trained SDM $\mathcal{M}_0$, clients N, rounds R, lr η
Ensure: Global LoRA $\Delta\Theta_g$, cluster LoRAs $\{\Delta\Theta_c\}_{c=1}^{K_R}$

Server executes:
Initialize $\Delta\Theta_g^0$
for round $r = 1, \ldots, R$ **do**
  $S_r \leftarrow$ random client subset
  for each client $i \in S_r$ **in parallel do**
    $\Delta\Theta_i^r, \mathbf{z}_i^r \leftarrow \text{ClientUpdate}(i, \Delta\Theta_g^{r-1})$
  end for
  Step 1: Dynamic Clustering by Style
  $\{\mathcal{C}_1, \ldots, \mathcal{C}_{K_r}\} \leftarrow \text{Cluster}(\{\mathbf{z}_i^r\}_{i \in S_r})$
  / Outliers form singleton clusters or merge into nearest cluster */*
  Step 2: Intra-Cluster Aggregation
  for each cluster $c = 1, \ldots, K_r$ **do**
    $\Delta\Theta_c^r = \sum_{i \in \mathcal{C}_c} \frac{n_i}{n_c} \Delta\Theta_i^r$
    where $n_i = |\mathcal{D}_i|,\ n_c = \sum_{i \in \mathcal{C}_c} n_i$
  end for
  Step 3: Inter-Cluster Aggregation (Dual-Path)
  / Path 1: Domain-Weighted Fusion */*
    $\Delta\Theta_{g,\text{mix}}^r = \sum_{c=1}^{K_r} \beta_c \Delta\Theta_c^r$
  where $\beta_c = \frac{w_c}{\sum_j w_j}, w_c = \alpha \cdot \text{DED}_c^{-1} + (1-\alpha) \cdot \text{SNT}_c$
  / Path 2: Statistically Neutral Model */*
  $\Delta\Theta_{g,\text{neu}}^r = \arg\min_{\Delta\Theta} \sum_{c=1}^{K_r} \|\Delta\Theta - \Delta\Theta_c^r\|_2^2$
  yields $\Delta\Theta_{g,\text{neu}}^r = \text{stack}(\Delta\Theta_1^r, \Delta\Theta_2^r, \ldots, \Delta\Theta_{K_r}^r)$
  // $\Delta\Theta_{g,\text{mix}}^r$: cluster-aware; $\Delta\Theta_{g,\text{neu}}^r$: neutral latent
  $\Delta\Theta_g^r \leftarrow \Delta\Theta_{g,\text{mix}}^r$
  if converged **then**
    break
  else
    send $\Delta\Theta_c^r$ to clients in $\mathcal{C}_c$
  end if
end for

ClientUpdate$(i, \Delta\Theta)$**:**
$\mathcal{D}_i = \{(p, x)\}$ // *Prompt-image pairs*
$\Delta\Theta_i \leftarrow \arg\min_{\Delta\Theta} \mathbb{E}_{(p,x)\sim\mathcal{D}_i}[\mathcal{L}_{\text{SD}}(\mathcal{M}_0(p; \Delta\Theta), x)]$
$\mathbf{z}_i \leftarrow \text{Mean}(\text{CLIP}(p) \text{ for } p \in \mathcal{D}_i)$
return $\Delta\Theta_i, \mathbf{z}_i$

may inadvertently reveal user-specific data traits—allowing adversaries to conduct property inference or semantic leakage attacks.

To protect style attributes and user-sensitive labels from disclosure to semi-trusted servers, our framework incorporates the textual inversion method(Gal et al. 2022) to encode explicit descriptors (e.g., "$sketch$") into private latent tokens (e.g., "$\langle user\text{-}s \rangle$"). A detailed overview of privacy considerations is provided in Section 3.1.

2.2 System Overview

Following the discussion of key challenges, Fig. 3 illustrates our main system model, which consists of two components: (1) a ***cluster-aware hierarchical*** FL training process (Fig. 3a), and (2) an ***optimized hybrid inference*** pipeline (Fig. 3b). During training, as shown in Algorithm 1, each client first fine-tunes its LoRA parameters on its own dataset

$\mathcal{D}_i = \{(p, x)\}$ to minimize the noise prediction loss:

$$\Delta\Theta_i^r = \arg\min_{\Delta\Theta^r} \mathbb{E}_{(p,x)\sim\mathcal{D}_i}\left[\|\epsilon - \epsilon_\theta(x_t, t, p; \Delta\Theta^r)\|_2^2\right],$$

where ϵ is the ground-truth noise, ϵ_θ is the predicted noise, x_t is the noisy latent at the timestep t. $\Delta\Theta^r$ represents the LoRA parameters optimized at FL round r, and the resulting client-specific parameters are denoted by $\Delta\Theta_i^r$.

After local fine-tuning (lines 28–31 in Algorithm 1), the Training Edge Server (TES) performs clustering based on user preferences $\mathbf{z}_i^r$ (e.g., sketch-style vs. cartoon-style) (line 8), followed by intra-cluster aggregation for enhanced personalization (line 11), and inter-cluster aggregation to produce both cluster-based personalized models $\Delta\Theta_g^{\text{mix}}$ (line 15) and a shared global model $\Delta\Theta_g^{\text{neu}}$ (line 18). The overall global loss function combines two objectives:

$$\mathcal{L}_{\text{g}}(\theta_{\text{agg}}, \{\text{LoRA}_k\}) = \lambda_{\text{neu}}\, \mathcal{L}_{\text{neu}}^{\text{LoRA}}(\theta_{\text{agg}}) + \lambda_{\text{mix}} \sum_{k=1}^{K_r} w_k\, \mathcal{L}_{\text{mix}}^k(\theta_{\text{agg}}, \text{LoRA}_k),$$

where $\mathcal{L}_{\text{neu}}^{\text{LoRA}}(\theta_{\text{agg}})$ is the neutralization loss that encourages the shared global model with neutral LoRA $\Delta\Theta_{g,\text{neu}}$ to generate intermediate latent representations $\mathbf{x}_{\text{mid}}$ following a standard Gaussian distribution $\mathcal{N}(0, I)$. Specifically:

$$\mathcal{L}_{\text{neu}}^{\text{LoRA}}(\theta_{\text{agg}}) = \mathbb{E}_{\mathbf{x}_{\text{mid}}\sim\mathcal{M}_0(\cdot;\theta_{\text{agg}},\Delta\Theta_{g,\text{neu}})}\left[D_{\text{KL}}(\mathbf{x}_{\text{mid}}\|\mathcal{N}(0, I))\right],$$

where D_{KL} denotes the Kullback-Leibler divergence. This ensures that $\Delta\Theta_{g,\text{neu}}$ produces statistically neutral outputs that can serve as a common foundation for all style-specific adaptations. The second term $\mathcal{L}_{\text{mix}}^k$ measures the reconstruction quality when combining the global model with each cluster-specific LoRA.

During hybrid inference on the Inference Edge Server (IES), the shared global LoRA $\Delta\Theta_g^{\text{neu}}$ works with the pre-trained base model $\mathcal{M}_{\text{SD}}$ to perform early denoising steps, producing an intermediate latent $\mathbf{x}_{\text{mid}}$ with neutral feature characteristics. Subsequently, cluster-aware personalized LoRAs $\Delta\Theta_g^{\text{mix}}$ can optionally be applied in later denoising steps to enhance user-specific style generation.

3 Proposed approach

3.1 Overview: Privacy Considerations

To limit sensitive data exposure, server roles are decoupled into TES (training) and IES (inference), with no collusion assumed. TES handles LoRA aggregation while being restricted from accessing the full base model. IES retains the pre-trained base model and the generalized global LoRA, which produces only style-neutral intermediate representations, preventing leakage of user-specific features. To further protect privacy, client-submitted data (e.g., domain keywords or prompts) is transmitted as encoded representations. For instance, a style term *sketch* is obfuscated as <*user-s*>, rendering it semantically opaque to the servers.

3.2 Cluster-aware Hierarchical Co-training

The co-training framework focuses on: (a) client-side LoRA fine-tuning (*Step 2 in Fig. 3*), and (b) server-side cluster-aware hierarchical LoRA aggregations (*Step 4-6*). Three sequential phases are executed:

1. Initialization and Local LoRA Training.

To enable end clients with limited computation and data to collaboratively personalize DMs without exposing private data, in this phase, only the added LoRA parameters on each client's local dataset $\mathcal{D}_i = (p, x)$ are updated. The LoRA modules are added to the attention layers of both the text encoder and U-Net components in the pre-trained diffusion model $\mathcal{M}_0$—where self-attention layers capture stylistic nuances while cross-attention layers align outputs with text prompts. Specifically, as shown in **Algorithm 1** (lines 1–4), each client i locally fine-tunes $\mathcal{M}_0$ by updating only the LoRA $\Delta\Theta_i^r$, while keeping all base model weights frozen. Each client possesses a private dataset $\mathcal{D}_i = (p, x, d)$, where p is the text prompt, x is the corresponding image, and d denotes the domain or visual style label (e.g., *sketch*, *cartoon*), later used for clustering in the server stage.

The local fine-tuning objective is defined as:

$$\Delta\Theta_i^r = \arg\min_{\Delta\Theta^r} \mathbb{E}_{(p,x)\sim\mathcal{D}_i}\left[\mathcal{L}_{\text{SD}}(\mathcal{M}_0(p; \Delta\Theta^r), x)\right].$$

Here, $\mathcal{L}_{\text{SD}}$ represents the standard denoising loss on the LoRA-augmented model $\mathcal{M}_0$:

$$\mathcal{L}_{\text{SD}} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0,I),t}\left[\|\epsilon - \epsilon_\theta(x_t, t, p)\|_2^2\right],$$

where $x_t = \sqrt{\bar{\alpha}_t}x + \sqrt{1-\bar{\alpha}_t}\epsilon$. After the r-th local fine-tuning, each client transmits only the LoRA updates $\Delta\Theta_i^r$ and a style embedding $\mathbf{z}_i$. The $\mathbf{z}_i$ is obtained by averaging CLIP-encoded vectors of a subset of its local images: ($\mathbf{z}_i = \frac{1}{|\mathcal{D}_i|}\sum_{x_j\in\mathcal{D}_i}\text{CLIP}(x_j)$) (Radford et al. 2021). This vector summarizes the visual style and enables effective clustering at the server while preserving privacy.

2. Style-based Cluster & Intra-cluster Aggregation.

Let $\mathcal{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N\}$ denote all client embeddings, and the server computes pairwise cosine similarities $s_{ij} = \frac{\mathbf{z}_i^\top \mathbf{z}_j}{\|\mathbf{z}_i\|_2\|\mathbf{z}_j\|_2}$ to construct K clusters $\{\mathcal{C}_1, \ldots, \mathcal{C}_K\}$ by minimizing $\sum_{k=1}^K \sum_{i\in\mathcal{C}_k} \|\mathbf{z}_i - \boldsymbol{\mu}_k\|_2^2$, where $\boldsymbol{\mu}_k$ is the mean embedding of cluster $\mathcal{C}_k$ (Kim, Kwon, and Ye 2022). For clarity, the domain representations are visualized in plain-text format in Fig. 3. Within each cluster $\mathcal{C}_k$, LoRA updates $\{\Delta\Theta_i \mid i \in \mathcal{C}_k\}$ are aggregated through weighted averaging to enhance domain-specific consistency:

$$\Delta\Theta_c^{(r)} = \sum_{i\in\mathcal{C}_c} \frac{|\mathcal{D}_i|}{|\mathcal{D}_c|}\,\Delta\Theta_i^{(r)}, \qquad |\mathcal{D}_c| = \sum_{i\in\mathcal{C}_c} |\mathcal{D}_i|.$$

This operation enhances style-specific feature expressiveness through intra-cluster consistency, while suppressing redundancy from overlapping parameters. It should be noted that directly extending weighted averaging to inter-cluster fusion ($\sum_c w_c\,\Delta\Theta_c^r$) may lead to *feature cancellation* caused by multi-style interference (see Fig. 2). To mitigate inconsistencies caused by rank heterogeneity in LoRA modules, we recommend a *dynamic median-aligned padding* strategy—the median rank across clients within cluster $\mathcal{C}_k$ is computed as $r_{\text{median}} = \text{median}\{r_i \mid i \in \mathcal{C}_k\}$, and components exceeding r_{median} are truncated.

3. Inter-cluster Aggregation.

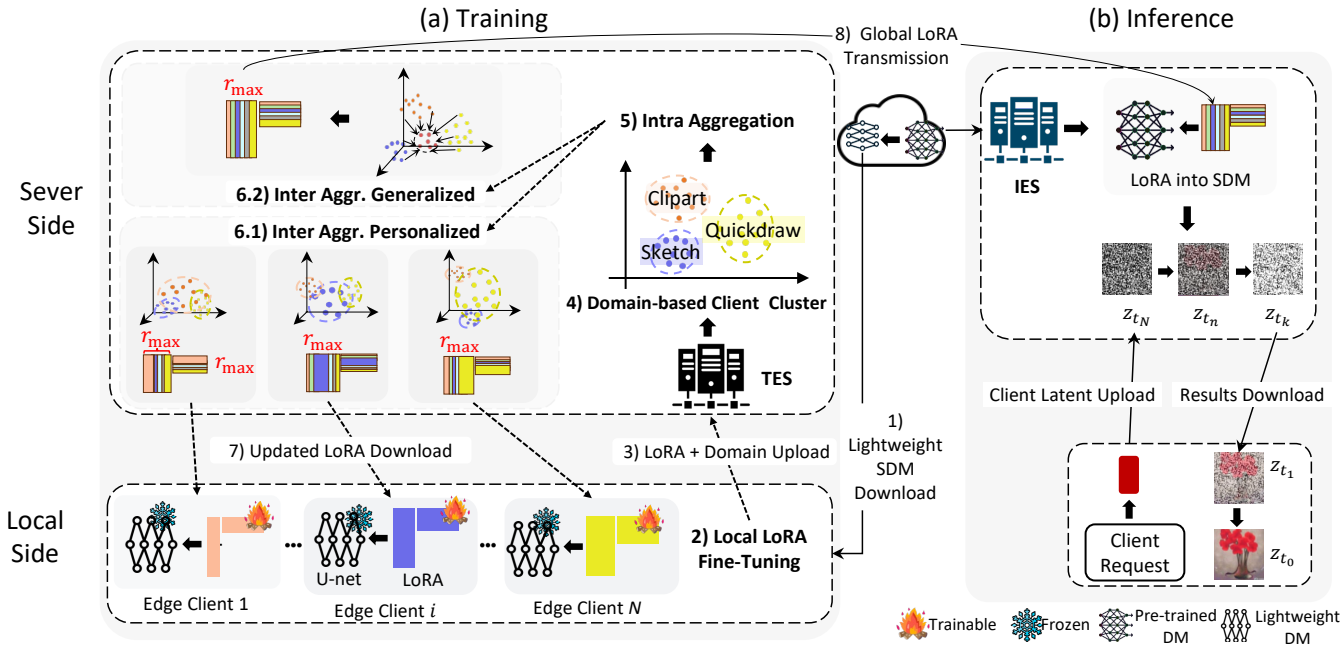

Figure 3: **A federated framework for fine-tuning personalized diffusion models across edge devices.** Clients are grouped by domain (e.g., image style) based on their dataset characteristics. After cluster-level aggregation, cluster-aware models are sent back to each group, while a shared global LoRA is delivered to the IES for hybrid inference.

Following the intra-cluster aggregation phase, which yields cluster-specific LoRAs $\{\Delta\Theta_c\}_{c=1}^{K_r}$, we proceed to inter-cluster aggregation to facilitate cross-domain knowledge transfer. Through adjustable coefficients, this phase produces two distinct outcomes: (1) cluster-aware personalized models $\Delta\Theta_{g,\text{mix}}$ capturing local preferences and (2) a shared, generalized global LoRA module $\Delta\Theta_{g,\text{neu}}$ producing style-neutral representations. Clients with limited training data can thereby benefit from diverse, privacy-preserving style patterns without performing local training.

1) **Cluster-specific Model (Style-Weighted Fusion):** For clients lacking training data for certain styles, we introduce an optional inter-cluster aggregation module to enable privacy-preserving and efficient style transfer. Inspired by Flora (Wang et al. 2024), we preserve style-specific features via stacking and use two complementary metrics to determine each style's contribution: semantic distance and feature importance. Let $d_k = \|\mathbf{z}_k - \mathbf{z}_g\|_2$ denote the semantic distance from cluster k's style embedding to the global style centroid $\mathbf{z}_g$. Let $\tau_k = \|\Sigma_k\|_1/\text{rank}(\Delta\Theta_k)$ denote the normalized trace of cluster k's LoRA singular values, which quantifies the average feature importance (or expressiveness) captured by cluster k's LoRA parameters. The style-weighted global LoRA is then formulated as:

$$\Delta\Theta_{g,\text{mix}} = \sum_{k=1}^{K_r} \beta_k \Delta\Theta_k = \sum_{k=1}^{K_r} \frac{w_k}{\sum_{j=1}^{K_r} w_j}, \Delta\Theta_k,$$

where the weight is computed as: $w_k = \alpha \cdot \frac{1}{d_k} + (1 - \alpha) \cdot \tau_k$. Here, $\alpha \in [0, 1]$ balances the contribution from semantic distinctiveness and feature expressiveness.

2) **Shared Global Model (Statistical Neutralization):** This path generates a style-neutral LoRA by solving $\Delta\Theta_{g,\text{neu}} = \arg\min_{\Delta\Theta} \sum_{c=1}^{K_r} \|\Delta\Theta - \Delta\Theta_c\|_2^2$, where each cluster LoRA $\Delta\Theta_c$ is stacked to form the neutral LoRA $\Delta\Theta_{g,\text{neu}}$. $\Delta\Theta_{g,\text{neu}}$ is then inserted into the pre-trained SDM and optimized using the neutralization loss $\mathcal{L}_{\text{neu}}^{\text{LoRA}}(\theta_{\text{agg}})$ to ensure statistically neutral latent generation. Following Flora (Wang et al. 2024), this stacking approach effectively preserves multiple personalized features across diverse domains.

3.3 Hybrid Inference Architecture

In this phase, the IES handles multi-user requests via a hybrid inference strategy that leverages the global LoRAs produced by Algorithm 1. This process can be formalized as:

1. **Neutral Latent Generation (Server-side):** Each user uploads a generic prompt p_{gen}, derived from non-sensitive or anonymized keywords (e.g., *Art_painting* $\rightarrow$ $\langle$*user-a*$\rangle$), to the server. Interpretable by $\mathcal{M}_{\text{SD+SharedLora}}$ and local LoRA modules but opaque to third parties, this prompt activates the corresponding LoRA subspace, allowing the server to generate a style-neutral intermediate latent:

$$\mathbf{x}_{\text{mid}} = \text{SD}_{\text{denoise}}(p_{\text{gen}}, \mathcal{M}_{\text{SD}}, \Delta\Theta_g^{\text{neu}}, t = 1 : T_{\text{early}}),$$

where $\text{SD}_{\text{denoise}}$ denotes the denoising process, and T_{early} specifies the number of initial denoising steps performed on the server side.

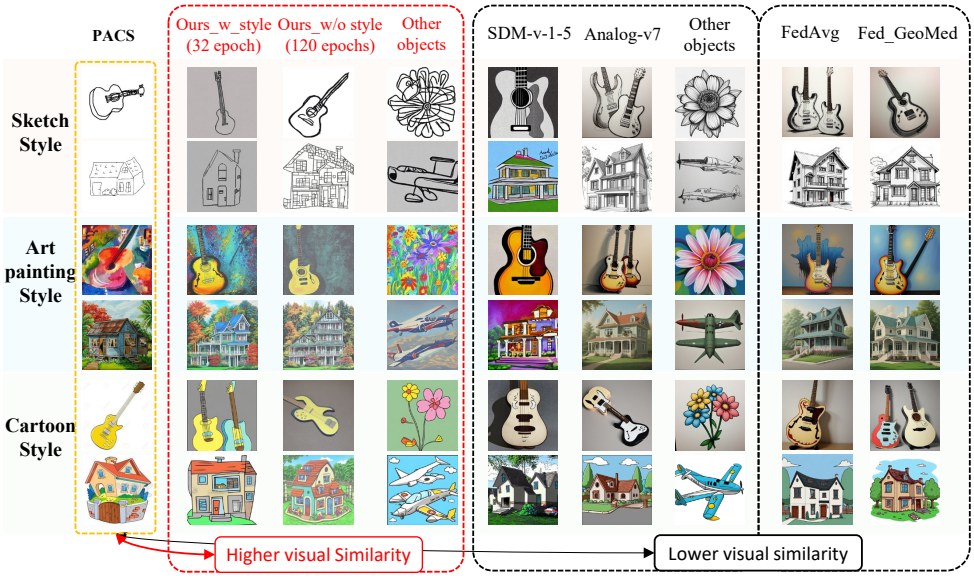

Figure 4: **Comparison on PACS dataset**: (i) Without fine-tuning, two pre-trained SDMs (*SDM-v1-5* and *Analog-v7*) fail to match PACS's styles; (ii) After fine-tuning and applying typical FL aggregation methods (i.e., *FedAvg* and *Fed_GeoMed*), the global outputs show limited feature alignment. (iii) Our outputs (*"Ours_w_style"* and *"Ours_w/o_style"*) better align with PACS's styles, which with "*A* {*label*} *with <user-s>style*" and "*A* {*label*} *with S/A/C style*" as input prompts separately. (iv) Our method achieves better alignment on unseen objects (*"flower", "plane"*) that are absent in training data.

2. **Local Generation (Client-side):** The client resumes denoising from $\mathbf{x}_{\text{mid}}$ using its cluster-aware personalized model $\Delta\Theta_g^{\text{mix}}$ and the prompt p_{gen}. The final latent representation is obtained as:

$$\mathbf{x}_{\text{final}} = \text{SD}_{\text{denoise}}(p, \mathbf{x}_{\text{mid}}, \mathcal{M}_{\text{SD}}, \Delta\Theta_g^{\text{mix}}, t = T_{\text{early}}{+}1 : T)$$

where T denotes the total number of denoising steps.

4 Experimental Evaluation

4.1 Models, Datasets and Experiment Settings.

We evaluate two SDMs (*SDM-v1.5* and *Analog-v7*) as our AIGC base models to assess their fine-tuning potential. The PACS dataset (7 classes across 4 domains: Photo/Art/Cartoon/Sketch) is used for personalized image synthesis. To assess federated fine-tuning performance, we simplify the whole process with rank-16 LoRA for all clients, bypassing intra-cluster and directly performing the inter-cluster aggregation across three LoRAs; the FL simulation follows a zero-shot setting, where each unique style is assigned to a single user (with 100 images each). This setup ensures lightweight adaptation, updating only 1.18M parameters in the text encoder and 3.19M in the U-Net per client. All experiments are conducted on NVIDIA GeForce RTX 4090 GPUs (24GB each) running CUDA 12.6.

4.2 Baselines and Experiment Results.

Style Alignment in Personalized Image Generation: As shown in Fig. 4, when given the same input prompts—"*A* {*label*} *with* {*sketch/art/cartoon* } *style*"—the two pre-trained DMs (e.g., *SDM-V-1-5* and *Analog-v7*) show noticeable style deviations from the PACS. This discrepancy is primarily attributed to the models' varying interpretations of certain terms, particularly the semantic ambiguity of "*sketch*", but our models can effectively correct these deviations, generating images that not only better match the PACS style but also preserve diversity. We also compare two classic FL aggregation strategies—*FedAvg* reconstructs the global model via layer-wise weighted accumulation of client updates; *FedGeoMed* applies geometric medians at each layer to enhance robustness against outlier gradients. As reflected in Fig. 2, due to the inconsistency in the direction of feature representations among local LoRAs, important personalized features tend to cancel out during aggregation, leading to suboptimal results.

In comparison, our fine-tuned variants (*Ours_w_style* and *Ours_w/o_style*, both using stack-based aggregation) produce more stylistically aligned results than the four baselines. Specifically, *Ours_w/o_style* replaces explicit style terms (e.g., "*sketch*") with implicit descriptors (e.g., *<user-specific>*) in the prompts. This setting allows for more privacy-aware control over style, but requires longer fine-tuning epochs. As shown in Fig. 4, with a batch size of 2 and 100 local images, it takes 120 epochs to converge—nearly four times longer than *Ours_w_style* (32 epochs).

Cluster-based Aggregation for Personalized Outputs: Fig. 5 compares hybrid inference results between the pre-trained SDM ($Baseline$) and our LoRA-enhanced SDM ($Ours$) based on generating intermediate latent representations on the server. With a total of 50 inference steps, the server conducts early-stage inference on 20% (10 steps) or 30% (15 steps) of the steps; under these split ratios, our outputs better align with the PACS test data distribution, partic-

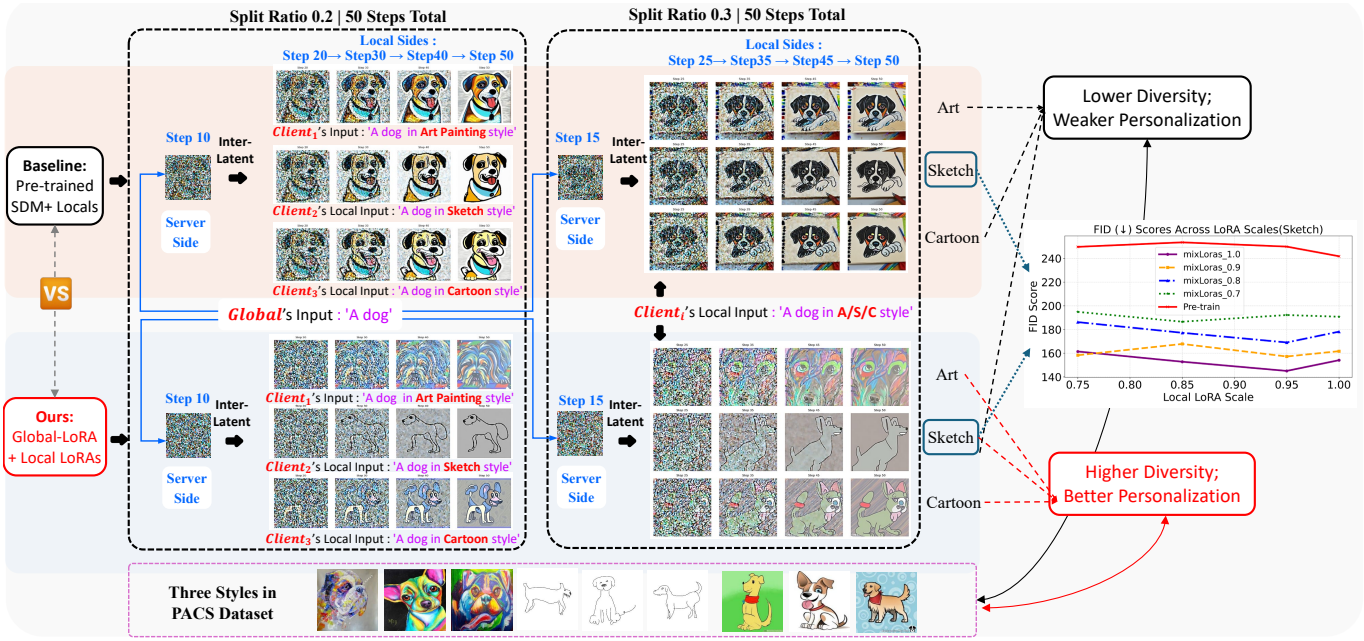

Figure 5: **Hybrid inference comparison: LoRA-enhanced SDM vs. pre-trained SDM.** Given a generic prompt (i.e., "$A\ dog$"), the server performs early-stage inference for 10 (0.2) or 15 (0.3) of the 50 total steps. Based on the same shared intermediate latent, each client applies its own prompt for downstream generation. Ours preserves both diversity and user-specific style features, whereas the vanilla pre-trained model exhibits weaker diversity and lacks personalization. As a representative case, the same result is validated by FID (Fréchet Inception Distance) scores of the $sketch$ style.

ularly in the $sketch$ style.

Quantitative evaluation using Fréchet Inception Distance (FID) (averaged over 100 images) confirms these observations, as seen in Fig. 5—lower scores indicate closer statistical alignment with real images. The $mixLoras \in [0.7, 1.0]$ controls the contribution of the global LoRA to the outputs' stylization, while the local LoRA scale (0.75-0.95) adjusts the strength of client-side personalization; higher scale values indicate stronger personalization effects. Our method achieves optimal performance (FID=145) at $mixLoras$=1.0 with local scale=0.95, surpassing the pre-trained baseline (FID=240) by 40% across all tested $mixLoras$ ranges (0.7-1.0). This demonstrates that the fine-tuned global model can enhance output quality compared to relying solely on localized adjustments.

5 Future research directions

To fully realize the potential of FL-powered Edge-AIGC systems, several critical open problems are considered:

- **Maintaining Personalization under Dynamic Client Arrivals.** New arrivals call for more adaptive solutions that do not rely on coarse-grained clustering when initializing new users. A promising future direction is to develop redundancy-aware adaptation strategies that determine whether new clients should reuse or align with existing aggregated LoRAs, or set up new clusters.
- **Network-Aware Aggregation for Federated AIGC.** When clients face heterogeneous network conditions and device constraints, imbalance can introduce aggregation bias and weaken personalization in the global LoRA model. Future directions include developing network-aware aggregation methods that incorporate both the communication capabilities of clients and the semantic relevance of their updates.
- **Adversarial Attacks in Federated Edge-AIGC.** The distributed nature of FL-based training makes it vulnerable to adversarial behaviors, such as data inversion and model poisoning by malicious clients through injecting adversarially crafted data or uploading manipulated models (e.g., LoRA parameters). Therefore, robust detection and defense mechanisms are crucial for safeguarding the integrity of federated AIGC systems.

6 Conclusion

In this work, we presented a federated multi-user fine-tuning framework for personalized AIGC at the edge. Built on LoRA-enhanced diffusion models, our approach enables scalable and privacy-preserving content generation across heterogeneous devices and user preferences. By integrating FL with adaptive multi-LoRA in a cluster-aware hierarchical framework, we mitigate redundancy, heterogeneity, and inefficiency in complex AIGC networks. Experimental results confirmed that our method achieves superior personalized inference accuracy and system scalability, highlighting its effectiveness for real-world edge AIGC applications.

Acknowledgements

This research is supported in part by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Communications and Connectivity Bridging Funding Initiative, in part by the Ministry of Education, Singapore, under its Joint SMU-SUTD Grant (22-SIS-SMU-052), and in part by the SUTD-MOE Early Career Award under the Singapore Ministry of Education START Scheme. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore. S. Mao's work is supported in part by the NSF under Grants 200887-131101-2002 and 200823-131101-2002.

References

Ahn, N.; Lee, J.; Lee, C.; Kim, K.; Kim, D.; Nam, S.-H.; and Hong, K. 2024. Dreamstyler: Paint by style inversion with text-to-image diffusion models. In *Proc. AAAI Conf. Artif. Intell.*, volume 38, 674–681.

Chen, H.; Li, H.; Zhang, Y.; Bi, J.; Zhang, G.; Zhang, Y.; Torr, P.; Gu, J.; Krompass, D.; and Tresp, V. 2025. Fedbip: Heterogeneous one-shot federated learning with personalized latent diffusion models. In *Proc. IEEE CVPR '25*, 30440–30450.

Chen, Z.; Yang, H. H.; Tay, Y. C.; Chong, K. F. E.; and Quek, T. Q. S. 2023. The Role of Federated Learning in a Wireless World with Foundation Models. *IEEE Wirel. Commun.*

Du, H.; Zhang, R.; Niyato, D.; Kang, J.; Xiong, Z.; Kim, D. I.; Shen, X.; and Poor, H. V. 2023. Exploring collaborative distributed diffusion-based AI-generated content (AIGC) in wireless networks. *IEEE Netw.*, 38(3): 178–186.

Frenkel, Y.; Vinker, Y.; Shamir, A.; and Cohen-Or, D. 2024. Implicit style-content separation using b-lora. In *ECCV*, 181–198. Springer.

Gal, R.; Alaluf, Y.; Atzmon, Y.; Patashnik, O.; Bermano, A. H.; Chechik, G.; and Cohen-Or, D. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W.; et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2): 3.

Kim, G.; Kwon, T.; and Ye, J. C. 2022. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2426–2435.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; and Aberman, K. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proc. IEEE CVPR '23*, 22500–22510.

Stanley Jothiraj, F. V.; and Mashhadi, A. 2024. Phoenix: A federated generative diffusion model. In *Companion Proceedings of the ACM Web Conference 2024*, 1568–1577.

Tan, A. Z.; Yu, H.; Cui, L.; and Yang, Q. 2022. Towards personalized federated learning. *IEEE Trans. Neural Netw. Learn. Syst.*, 34(12): 9587–9603.

Tu, K.; Wang, X.; and Hu, X. 2024. EntroCFL: Entropy-based clustered federated learning with incentive mechanism. *IEEE Internet of Things Journal*.

Wang, Z.; Shen, Z.; He, Y.; Sun, G.; Wang, H.; Lyu, L.; and Li, A. 2024. FLoRA: Federated Fine-Tuning Large Language Models with Heterogeneous Low-Rank Adaptations. In *NeurIPS*, volume 37.

Xie, J.; Zhang, S.; Zhao, Z.; Wu, F.; and Wu, F. 2025. EC-Diff: Fast and High-Quality Edge-Cloud Collaborative Inference for Diffusion Models. *arXiv preprint arXiv:2507.11980*.

Yan, C.; Liu, S.; Liu, H.; Peng, X.; Wang, X.; Chen, F.; Fu, L.; and Mei, X. 2024. Hybrid sd: Edge-cloud collaborative inference for stable diffusion models. *arXiv preprint arXiv:2408.06646*.

Yang, W.; Xiong, Z.; Guo, S.; Mao, S.; Kim, D. I.; and Debbah, M. 2025. Efficient Multi-user Offloading of Personalized Diffusion Models: A DRL-Convex Hybrid Solution. *IEEE Trans. Mob. Comput.*

Yuan, C.; Liu, Z.; Lv, J.; Shao, J.; Jiang, Y.; Zhang, J.; and Li, X. 2025. Task-Oriented Feature Compression for Multimodal Understanding via Device-Edge Co-Inference. *arXiv preprint arXiv:2503.12926*.