

Mix-of-Show: Decentralized Low-Rank Adaptation for Multi-Concept Customization of Diffusion Models

Yuchao Gu¹, Xintao Wang³, Jay Zhangjie Wu¹, Yujun Shi², Yunpeng Chen²,
 Zihan Fan¹, Wuyou Xiao¹, Rui Zhao¹, Shuning Chang¹, Weijia Wu¹,
 Yixiao Ge³, Ying Shan³, Mike Zheng Shou^{1*}

¹Show Lab, ²National University of Singapore ³ARC Lab, Tencent PCG

<https://showlab.github.io/Mix-of-Show>

Abstract

Public large-scale text-to-image diffusion models, such as Stable Diffusion, have gained significant attention from the community. These models can be easily customized for new concepts using low-rank adaptations (LoRAs). However, the utilization of multiple concept LoRAs to jointly support multiple customized concepts presents a challenge. We refer to this scenario as decentralized multi-concept customization, which involves single-client concept tuning and center-node concept fusion. In this paper, we propose a new framework called Mix-of-Show that addresses the challenges of decentralized multi-concept customization, including concept conflicts resulting from existing single-client LoRA tuning and identity loss during model fusion. Mix-of-Show adopts an embedding-decomposed LoRA (ED-LoRA) for single-client tuning and gradient fusion for the center node to preserve the in-domain essence of single concepts and support theoretically limitless concept fusion. Additionally, we introduce regionally controllable sampling, which extends spatially controllable sampling (*e.g.*, ControlNet and T2I-Adaptor) to address attribute binding and missing object problems in multi-concept sampling. Extensive experiments demonstrate that Mix-of-Show is capable of composing multiple customized concepts with high fidelity, including characters, objects, and scenes.

1 Introduction

Open-source text-to-image diffusion models, such as Stable Diffusion [1], empower community users to create customized models by collecting personalized concept images and fine-tuning them with low-rank adaptation (LoRA) [2, 3]. These tailored LoRA models achieve unparalleled quality for specific concepts through meticulous data selection, preprocessing, and hyperparameter tuning. While existing concept LoRAs serve as plug-and-play plugins for pretrained models, the challenge lies in utilizing multiple concept LoRAs to extend the pretrained model and enable joint composition of those concepts. We refer to this scenario

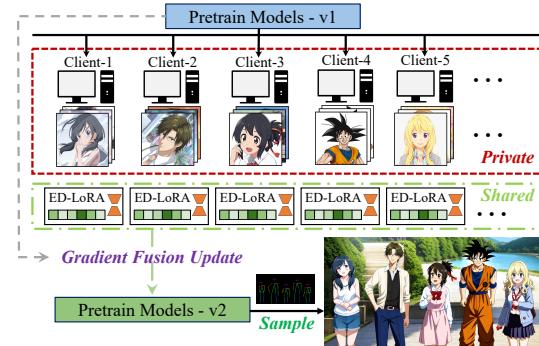


Figure 1: Illustration of decentralized multi-concept customization via Mix-of-Show.

*Corresponding Author.



Figure 2: How to generate Harry Potter and Thanos, these two (or even more) concepts from different shows, in the same image? Our Mix-of-Show enables complex compositions of multiple customized concepts (e.g., characters, objects, scenes) with individually trained concept LoRAs.

as decentralized multi-concept customization. As shown in Fig. 1, it involves two steps: single-client concept tuning and center-node concept fusion. Each client retains their private concept data while sharing the tuned LoRA models. The center node leverages these concept LoRAs to update the pretrained model, enabling joint sampling of these customized concepts. Decentralized multi-concept customization facilitates maximum community engagement in producing high-quality concept LoRAs and offers flexibility in reusing and combining different concept LoRAs.

However, the existing LoRA tuning and weight fusion techniques [3] fail to address the challenges of decentralized multi-concept customization. We have identified two main challenges: concept conflict and identity loss. Concept conflict arises because current LoRA tuning methods do not differentiate between the roles of embeddings and LoRA weights. Our research reveals that embeddings effectively capture concepts within the pretrained models' domain, while LoRA weight assist in capturing out-of-domain information (*e.g.*, styles or fine details cannot be directly modeled by the pretrained model). However, existing LoRA tuning methods place excessive emphasis on LoRA weights while overlooking the importance of embeddings. Consequently, the LoRA weights encode a significant portion of the identity of a given concept, resulting in semantically similar embeddings being projected onto concepts with different appearances. This, in turn, leads to conflicts during model fusion. Furthermore, existing weight fusion strategies compromise each concept's identity and introduce interference from other concepts by performing a weighted average of all concept LoRAs.

To overcome the challenges of decentralized multi-concept customization, we propose Mix-of>Show, which involves embedding-decomposed LoRA (ED-LoRA) for single-client tuning and gradient fusion for center-node fusion. In single-client tuning, ED-LoRA is designed to address concept conflicts by preserving more in-domain essence within the embedding. To achieve this, we enhance the expressive ability of the concept embedding by decomposing it into layer-wise embeddings [4] and multi-word representations. Additionally, we introduce LoRA dropout to prevent co-adaptation between the embedding and LoRA weight. At the central node, gradient fusion leverages multiple concept LoRAs to update the pretrained model. Since the diffusion model includes both forward and reverse diffusion processes, we can obtain the input/output features of each layer through sampling, even in the absence of data. Features from multiple concept LoRAs are combined to generate the fused gradient, which is used for layer-wise updating. Compared to weight fusion [3], gradient fusion aligns the inference behavior of each individual concept, significantly reducing identity loss.

To demonstrate the capabilities of Mix-of>Show, we introduce regionally controllable sampling for multi-concept generation. Direct multi-object generation often encounters issues such as missing objects and attribute binding [5, 6]. Recently, spatially controllable sampling (*e.g.*, Controlnet [7], T2I-Adaptor [8]) have been introduced to guide diffusion models using spatial hints(*e.g.*, keypose or sketch), which resolve the problem of missing objects but still faces challenges with attribute binding in multi-concept generation. Considering that spatial layout is pre-defined when adopting spatial conditions, we propose injecting region prompts through regional-aware cross-attention. Powered by Mix-of>Show and regionally controllable sampling, we can achieve complex compositions of multiple customized concepts, including characters, objects, and scenes, as illustrated in Fig. 2. In summary, our contributions are as follows: 1) We analyze the challenges of decentralized multi-concept customization. 2) We propose the Mix-of>Show framework, consisting of an embedding-decomposed LoRA (ED-LoRA) and gradient fusion, to address the concept conflict and identity loss in decentralized multi-concept customization. 3) We introduce regionally controllable sampling to demonstrate the potential of Mix-of>Show in composing multiple customized concepts.

2 Related Work

2.1 Concept Customization

Concept customization aims to extend pretrained diffusion models to support personalized concepts using only a few images. There are two main types of concept tuning methods: embedding tuning (*e.g.*, Textual Inversion [9] and P+ [4]) and joint embedding-weight tuning (*e.g.*, Dreambooth [10] and Custom Diffusion [11]). Additionally, the community [3] adopts low-rank adaptor (LoRA) [2] for concept tuning, which is lightweight and can achieve comparable fidelity to full weight tuning.

Although significant progress has been made in single-concept customization, multi-concept customization remains a challenge. Custom Diffusion [11] proposes co-training of multiple concepts or constrained optimization of several existing concept models. Following this, SVDiff [12] introduces

data augmentation to prevent concept mixing in co-training multi-concepts, and Cones [13] discovers concept neurons that can be added to support multiple concepts. However, their methods are typically restricted to fuse 2-3 semantically distinct concepts. In contrast, Mix-of-Show can combine theoretically limitless customized concepts, including those within the same semantic category.

Another approach to concept customization, as explored in studies by Instantbooth [14], ELITE [15], and Jia et al. [16], focuses on achieving fast test-time customization. These methods involve pretraining an encoder on a large-scale dataset specific to the desired category. During inference, when provided with a few representative concept images from the trained category, the encoder extracts low-level features that complement the pretrained diffusion models and support customized generation. However, these methods require training a separate encoder for each category, typically limited to common categories (*e.g.*, person or cats). This limitation hinders their ability to customize and compose more diverse and open-world subjects.

2.2 Decentralized Learning

Decentralized or federated learning aims to train models collaboratively across different clients without sharing data. The *de facto* algorithm for federated learning, FedAvg, was proposed by [17]. This method simply averages the weights of each client’s model to obtain the final model. However, we find that directly applying this simple weight averaging is not ideal for fusing LoRAs of different concepts. To improve over FedAvg, previous works have either focused on local client training [18, 19, 20, 21, 22, 23, 24] or global server aggregation [25, 26, 27, 28, 29, 30]. Motivated by this, we explore the optimal design of single-client tuning and center-node fusion for decentralized multi-concept customization.

2.3 Controllable Multi-Concept Generation

Direct multi-concept generation using text prompts alone faces challenges such as missing objects and attribute binding [6, 31, 32, 33, 34]. Previous approaches, like Attend-and-Excite [5] and Structure Diffusion [6], have attempted to address these issues, but the problem still persists, limiting the effectiveness of multi-concept generation. Recent works, such as Controlnet [7] and T2I-adaptor [8], introduce spatial control (*e.g.*, keypose and sketch) and enable more accurate compositions, resolving the problem of missing objects in multi-concept generation. However, attribute binding remains a challenge. In our work, we tackle this challenge through regionally controllable sampling.

3 Methods

In this section, we provide a brief background on text-to-image diffusion models and concept customization in Sec. 3.1. We then introduce the task formulation of decentralized multi-concept customization in Sec. 3.2, followed by a detailed description of our method in Sec. 3.3 and Sec. 3.4.

3.1 Preliminary

Text-to-Image Diffusion Models. Diffusion models [35, 36, 37, 38, 39, 40, 41] belong to a class of generative models that gradually introduce noise into an image during the forward diffusion process and learn to reverse this process to synthesize images. When combined with pretrained text embeddings, text-to-image diffusion models [1, 42, 43, 44, 45, 46] are capable of generating high-fidelity images based on text prompts. In this paper, we conduct experiments using Stable Diffusion [1], which is a variant of the text-to-image diffusion model operating in the latent space. Given a condition $c = \psi(P^*)$, where P^* is the text prompt and ψ is the pretrained CLIP text encoder [47], the training objective for stable diffusion is to minimize the denoising objective by

$$\mathcal{L} = \mathbb{E}_{z,c,\epsilon,t}[\|\epsilon - \epsilon_\theta(z_t, t, c)\|_2^2], \quad (1)$$

where z_t is the latent feature at timestep t and ϵ_θ is the denoising unet with learnable parameter θ .

Embedding Tuning for Concept Customization. Textual Inversion [9] represents the input concept using a unique token V . When provided with a few images of the target concept, the embedding of V is tuned using Eq. 1. After tuning, the embedding for V encodes the essence of the target concept and functions like any other text in the pretrained model. To achieve greater disentanglement and control, P+ [4] introduces layer-wise embeddings for concept tokens, denoted as V^+ in this paper.

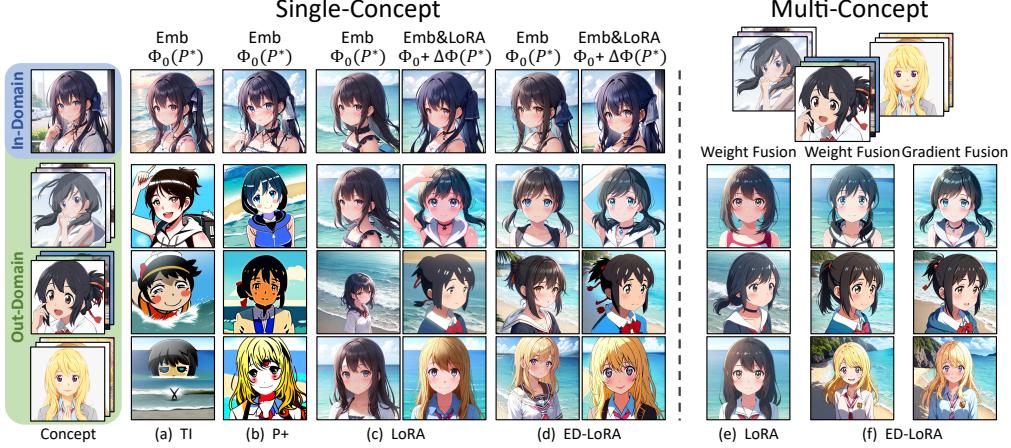


Figure 3: Single- and multi-concept customization between the embedding tuning (*i.e.*, Textual Inversion (TI) [9] and P+ [4]), and joint embedding-weight tuning (*i.e.*, LoRA [3] and our ED-LoRA). P^* = “Photo of a V , near the beach”. Φ_0 and $\Delta\Phi$ denotes the pretrained model and LoRA weight.

Low-Rank Adaptation. Low-rank adaptation (LoRA) [2] was initially proposed to adapt large-language models to downstream tasks. It operates under the assumption that weight changes during adaptation have a low “intrinsic rank” and introduces a low-rank factorization of the weight change to obtain the updated weight W , which is given by $W = W_0 + \Delta W = W_0 + BA$. Here, $W_0 \in \mathbb{R}^{d \times k}$ represents the original weight in the pretrained model, and $B \in \mathbb{R}^{d \times r}$ and $A \in \mathbb{R}^{r \times k}$ represent the low-rank factors, with $r \ll \min(d, k)$. Recently, the community [3] has adopted LoRA for fine-tuning diffusion models, leading to promising results. LoRA is typically used as a plug-and-play plugin in pretrained models, but the community also employs weight fusion techniques to combine multiple LoRAs:

$$W = W_0 + \sum_{i=1}^n w_i \Delta W_i, \quad \text{s.t. } \sum_{i=1}^n w_i = 1, \quad (2)$$

where w_i denotes the normalized importance of different LoRAs.

3.2 Task Formulation: Decentralized Multi-Concept Customization

While custom diffusion [11] has attempted to merge two tuned concepts models into a pretrained model, their findings suggest that co-training with multiple concepts yields better results. However, considering scalability and reusability, we focus on merging single-concept models to support multi-concept customization. We refer to this setting as decentralized multi-concept customization.

Formally, decentralized multi-concept customization involves a two-step process: single-client concept tuning and center-node concept fusion. As shown in Fig. 1, each of the n clients possesses its own private concept data and tunes the concept model ΔW_i . Here, ΔW_i represents the changes in network weights, which specifically refers to LoRA weights in our work. We omit discussing the merging of text embeddings, as the tuned embeddings can be seamlessly integrated into the pretrained model without conflicts.

After tuning, the center node gathers all LoRAs to obtain the updated pretrained weight W by:

$$W = f(W_0, \Delta W_1, \Delta W_2, \dots, \Delta W_n), \quad (3)$$

where f represents the update rule that operates on the original pretrained model weight W_0 and the n concept LoRAs $\{\Delta W_i, i = 1 \dots n\}$. One straightforward updating rule f is weight fusion, as illustrated in Eq. 2. Once updated, the new model W should be capable of generating all the concepts introduced in the n LoRAs.

3.3 Mix-of-Show

In this section, we introduce Mix-of-Show, containing ED-LoRA (in Sec. 3.3.1) for single-client concept tuning, gradient fusion (in Sec. 3.3.2) for center-node concept fusion.

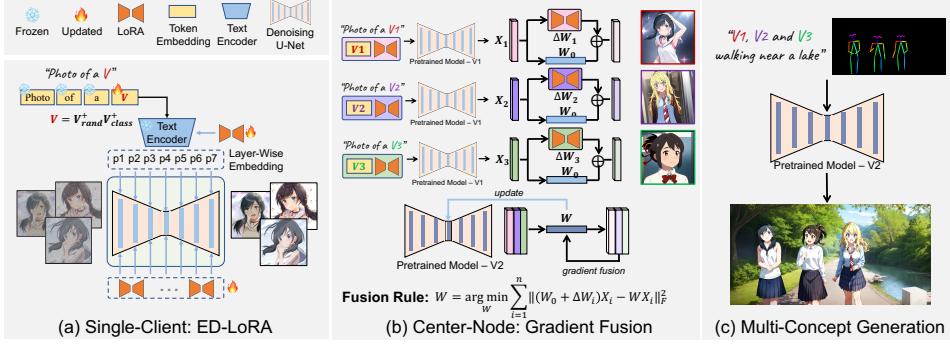


Figure 4: Pipeline of Mix-of-Show. In single-client concept tuning, the ED-LoRA adopts the layer-wise embedding and multi-word representation. In center node, gradient fusion is adopted to fuse multiple concept LoRAs and then support composing those customized concepts.

3.3.1 Single-Client Concept Tuning: ED-LoRA

Vanilla LoRA [3] is not suitable for decentralized multi-concept customization due to the issue of concept conflict. To better understand this limitation, we start by examining the distinct roles of embeddings and LoRA weights in concept tuning.

Single-Concept Tuning Setting. We investigate embedding tuning (*i.e.*, Textual Inversion [9] and P+ [4]) and the joint embedding-weight tuning (*i.e.*, LoRA [3]) on single concept customization. We conduct experiments on both in-domain concept (*i.e.*, directly sampled from the pretrained model), and out-domain concepts. The weights of the pretrained model, including the unet θ and the text encoder ψ , are denoted as $\Phi_0 = \{\theta_0, \psi_0\}$. Given a text prompt P^* containing the concept V , we visualize the tuned embedding of concept V using the pretrained weights $\Phi_0(P^*)$, and visualize the tuned embedding along with the LoRA weight using $(\Phi_0 + \Delta\Phi)(P^*)$.

Analysis. Based on the experiment results in Fig. 3, we draw the following two observations regarding existing embedding tuning and joint embedding-weight tuning approaches.

Observation 1: *The embeddings are capable of capturing concepts within the domain of pretrained models, while the LoRA helps capture out-domain information.*

In Fig. 3(a, b), we observe that embedding tuning approaches such as Textual Inversion and P+ struggle to capture out-domain concepts. This is because they attempt to encode all out-domain details (*e.g.*, anime styles or details not modeled by the pretrained model Φ_0) within the embedding, resulting in semantic collapse. However, for in-domain concepts sampled from the model, embedding tuning accurately encodes the concept identity within the embedding, benefiting from the accurate modeling of concept details by the pretrained model weights Φ_0 . Furthermore, when jointly tuning the embedding with LoRA, the embedding no longer produces oversaturated outputs. This is because the out-domain information is captured by the pretrained model with LoRA weight shift (*i.e.*, $\Phi_0 + \Delta\Phi$).

Observation 2: *Existing LoRA weights encode most of the concept identity and project semantically similar embeddings to visually distinct concepts, leading to conflicts during concept fusion.*

In the joint embedding-LoRA tuning results shown in Fig. 3(c), we observe that directly visualizing the embedding with the pretrained model $\Phi_0(P^*)$ yields semantically similar results. However, when the LoRA weights are loaded $(\Phi_0 + \Delta\Phi)(P^*)$, the target concept can be accurately captured. This suggests that the majority of the concept identity is encoded within the LoRA weights rather than the embedding itself. However, when attempting to support multiple semantically similar concepts within a single model, it becomes problematic to determine which concept to sample based on similar embeddings, resulting in concept conflicts. As shown in Fig. 3(e), when fused into one model, the identity of each individual concept is lost.

Our Solution: ED-LoRA. Based on the aforementioned observations, our ED-LoRA is designed to preserve more in-domain essence within the embedding while capturing the remaining details using LoRA weights. To achieve this, we enhance the expressiveness of the embedding through decomposed embedding. As illustrated in Fig. 4, we adopt a layer-wise embedding similar to [4] and create a multi-world representation for the concept token ($V = V_{rand}^+ V_{class}^+$). Here, V_{rand}^+ is

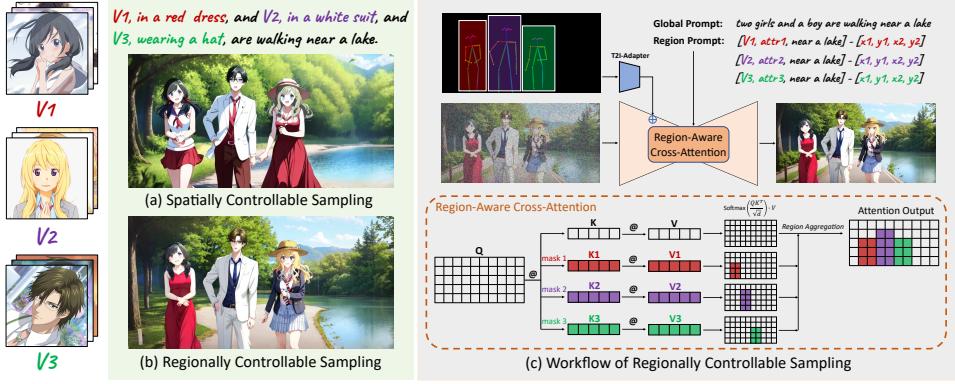


Figure 5: Regionally controllable sampling for multi-concept generation.

randomly initialized to capture the variance of different concepts, while V_{class}^+ is initialized based on its semantic class to maintain semantic meaning. Both tokens are learnable during concept tuning.

Since both the decomposed embedding and LoRA weights are capable of capturing low-level details, they face a co-adaptation issue during joint learning. The dominance of LoRA weights often leads to the embedding losing the concept identity. To address this, we introduce the DropLoRA strategy, which randomly drops the key and value LoRA layers in cross-attention. This strategy encourages the original embedding to retain sufficient details without excessive reliance on LoRA. As shown in Fig. 3(d), the learned embedding of ED-LoRA effectively preserves the essence of the given concept within the domain of the pretrained model, while LoRA helps capture the out-domain details.

3.3.2 Center-Node Concept Fusion: Gradient Fusion

At the center node, we have access to all the concept LoRAs and can use these models to update the pretrained model, enabling multi-concept customization. However, the existing weight fusion strategy described in Eq. 2 is insufficient to achieve this goal, as we will discuss further below.

Multi-Concept Fusion Setting. In this experiment, we apply the weight fusion described in Eq. 2 to weighted average n concept LoRAs or ED-LoRAs $\{\Delta\Phi_i, i = 1 \dots n\}$ into the pretrained model Φ_0 , resulting in a new model Φ . We then use the new model $\Phi(P_i^*)$ to sample each concept and compare its identity with the corresponding single-concept sample $(\Phi_0 + \Delta\Phi)(P_i^*)$.

Analysis. Based on the results in Fig. 3, we make the following observation about fusion strategy.

Observation 3: *Weight fusion leads to identity loss of individual concepts in concept fusion.*

As shown in Fig. 3 (multi-concept), we can observe that weight fusion in the case of LoRA leads to significant loss of concept identity due to conflicts between concepts. Even when combined with our ED-LoRA, weight fusion still compromises the identity of each individual concept. In theory, if a concept achieves its complete identity through LoRA weight shift $\Delta\Phi(P^*)$, fusing it with $n-1$ other concept LoRAs requires reducing its weight to $\frac{1}{n}\Delta\Phi(P^*)$ and introducing other concept LoRA weights, which ultimately diminishes the concept’s identity.

Our Solution: Gradient Fusion. Based on the previous analysis, our objective is to preserve the identity of each concept in the fused model by aligning their single-concept inference behavior. Unlike in federated learning [17, 19], where models are typically single-direction classification models that cannot access gradients without data, text-to-image diffusion models have the inherent capability to decode concepts from text prompts. Leveraging this characteristic, we first decode the individual concepts using their respective LoRA weights, as depicted in Fig. 4(b). We then extract the input and output features associated with each LoRA layer. These input/output features from different concepts are concatenated, and fused gradients are used to update each layer W using the following objective:

$$W = \arg \min_W \sum_{i=1}^n \|(W_0 + \Delta W_i)X_i - WX_i\|_F^2, \quad (4)$$

where X_i represents the input activation of the i -th concept, and $|\cdot|_F$ denotes the Frobenius norm. By adopting this approach, we can fuse different concept LoRAs without accessing the data and



Figure 6: Qualitative comparison on single- and multi-concept customization.

without considering their differences during training. The results of our gradient fusion are shown in Fig. 3(f), demonstrating improved preservation of each concept’s identity and consistent stylization across different concepts.

3.4 Regionally Controllable Sampling

Direct multi-concept sampling often encounters challenges of missing objects and attribute binding [6, 31, 32, 33, 34]. While spatially controllable sampling methods (*e.g.*, ControlNet [7] and T2I-Adaptor [8]) can address the issue of missing objects in multi-concept generation, they cannot accurately bind concepts to specific keyposes or sketches. Merely indicating the desired concept and attribute through a text prompt can lead to attribute binding problems, as in Fig. 5(a), where the identities of three people are mixed, and the “red dress” is incorrectly assigned to other concepts.

To address these challenges, we propose a method called regionally controllable sampling. This approach utilizes both a global prompt and multiple regional prompts to describe an image based on spatial conditions. The global prompt provides the overall context, while the regional prompts specify subjects within specific regions, including their attributes and contextual information from the global prompt (*e.g.*, “near a lake”). To achieve this, we introduce region-aware cross-attention. Given a global prompt P_g^* and n regional prompts $P_{r_i}^*$, we first incorporate the global prompt via cross-attention with the latent z by $h = \text{softmax}\left(\frac{Q(z)K(P_g^*)}{\sqrt{d}}\right) \cdot V(P_g^*)$. Then, we extract the regional latent feature by $z_i = z \odot M_i$, where M_i represents the binary mask associated with the region specified by $P_{r_i}^*$. We obtain regional features using $h_i = \text{softmax}\left(\frac{Q(z_i)K(P_{r_i}^*)}{\sqrt{d}}\right) \cdot V(P_{r_i}^*)$. Finally, we replace the features in the global output with the regional features: $h[M_i] = h_i$. For overlapping regions, we average the features from multiple associated region prompts. As shown in Fig. 5(b), regionally controllable sampling allows for precise assignment of subjects and attributes, while maintaining a harmonious global context.

4 Experiments

4.1 Datasets, Implementation and Evaluation Details

To conduct a systematic investigation of Mix-of>Show, we collect a dataset containing characters, objects, and scenes. For single-concept ED-LoRA tuning, we incorporate LoRA layer into the linear

	Methods	Real-Objects Single→Fused	Real-Characters Single→Fused	Real-Scenes Single→Fused	Mean Change
	Upper Bound	0.811	0.767	0.834	0.804
Text-alignment	P+ [4]	0.771→0.771 (-)	0.686→0.686 (-)	0.759→0.759 (-)	0.739→0.739 (-)
	Custom Diffusion [11]	0.745→0.747 (+0.002)	0.674→0.650 (-0.024)	0.748→0.738 (-0.010)	0.722→0.712 (-0.010)
	LoRA [3]	0.720→0.795 (+0.075)	0.654→0.700 (+0.046)	0.717→0.760 (+0.043)	0.697→0.752 (+0.055)
	Mix-of-Show (Ours)	0.724→0.745 (+0.021)	0.632→0.662 (+0.030)	0.716→0.736 (+0.020)	0.691→0.714 (+0.024)
	Lower Bound	0.721	0.471	0.595	0.596
Image-alignment	P+ [4]	0.790→0.790 (-)	0.670→0.670 (-)	0.796→0.796 (-)	0.752→0.752 (-)
	Custom Diffusion [11]	0.842→0.808 (-0.034)	0.714→0.694 (-0.020)	0.804→0.750 (-0.054)	0.787→0.751 (-0.036)
	LoRA [3]	0.864→0.778 (-0.086)	0.761→0.555 (-0.206)	0.824→0.769 (-0.055)	0.816→0.701 (-0.115)
	Mix-of-Show (Ours)	0.868→0.846 (-0.022)	0.802→0.770 (-0.032)	0.858→0.838 (-0.020)	0.843→0.818 (-0.025)

Table 1: *Text-alignment* and *image-alignment* vary between the single-client tuned model and the center-node fused model. The upper bound of text-alignment and the lower bound of image-alignment are computed by replacing the concept’s token with its class token and sampling using the pretrained model. For instance, to assess the upper-bound text alignment and lower-bound image alignment for the concept " V^{dogA} ," we substitute " V^{dogA} " in the sample prompts with the class token "dog" and sample it using the pretrained model.

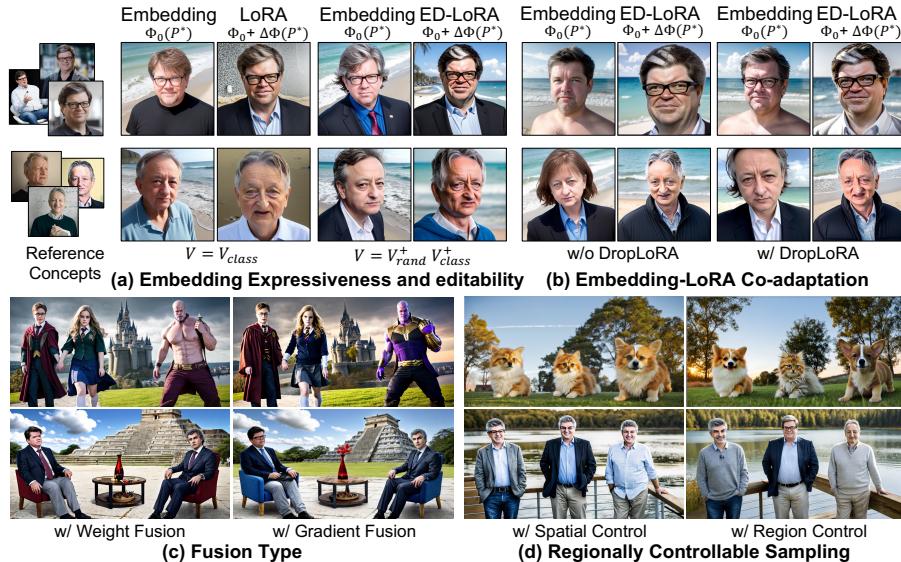


Figure 7: Ablation study of Mix-of-Show. P^* = “Photo of a V , near the beach”. Φ_0 and $\Delta\Phi$ denotes the pretrained model and LoRA weight, respectively.

layer in all attention modules of the text encoder and Unet, with a rank of $r = 4$ in all experiments. We use the Adam [48] optimizer with a learning rate of 1e-5 and 1e-4 for tuning the text encoder and Unet, respectively. For gradient fusion, we use the LBFGS optimizer [49] with 500 and 50 steps to optimize the text encoder and Unet, respectively. For evaluation, we follow the Custom Diffusion [11] to compute the *text-alignment* and *image-alignment* on 1000 samples per concept. More details are provided in the Sec. 6.1.

4.2 Quantitative Results.

Based on the evaluation results presented in Table. 1, our Mix-of>Show method demonstrates superior image alignment compared to previous methods, while maintaining comparable text alignment in the *single-client tuned model*. On the other hand, the P+ method achieves the best text alignment since it focuses solely on tuning the text embedding. The custom diffusion method ranks second as it tunes both the embedding and the K-V projection in cross attention. In contrast to P+ and custom diffusion, our Mix-of>Show and LoRA have the capability to tune all attention layers in the text encoder and Unet of diffusion models. This broader scope of tuning allows them to achieve better image alignment compared to the other methods.

However, there is a notable difference between the LoRA and Mix-of>Show when it comes to fusing all 14 concepts into the pretrained model (*i.e.*, the *center-node fused model*). In the single-client tuned

model, both LoRA and Mix-of>Show achieve similar levels of image alignment. However, when it comes to the center-node fused model, the LoRA experiences a significant loss of image alignment for each concept and degrades towards the lower bound. On the other hand, our Mix-of>Show method experiences much less degradation in image alignment. Furthermore, when comparing to the custom diffusion method, which exhibits degradation in both image and text alignment after fusion, our Mix-of>Show method shows an improvement in text alignment after fusion.

In summary, the Mix-of>Show method achieves the best image alignment in both the single-client tuned model and the center-node fused model, indicating minimal loss of concept identity. For detailed evaluation results to derive the Table. 1, please refer to Table. 2.

4.3 Qualitative Comparison

Single-Concept Results. The qualitative comparison of Mix-of>Show and other methods on the single-client tuned model and center-node fused model is presented in Fig. 6, Fig. 9 and Fig. 10. From the results, it is evident that the LoRA [3] experiences the most significant loss of concept identity after concept fusion. Additionally, due to the limited tunability of positions in the P+ [4] and Custom Diffusion [11], they exhibit oversaturated results or semantic collapse in some examples, as observed in the restylization examples in Fig. 10. Conversely, Mix-of>Show consistently achieves the best concept identity and quality across various examples, while also minimizing the loss of identity after the center-node concept fusion.

Multi-Concept Results. We compare Mix-of>Show with LoRA [3], Custom Diffusion [11], and P+ [4] for decentralized multi-concept customization. For LoRA, we utilize weight fusion to combine the different concepts. In the case of P+, we directly incorporate the tuned concept embedding into the pretrained model. And for Custom Diffusion, we follow their approach of constrained optimization to merge the key and value projections in cross-attention. To ensure consistent evaluation, we employ the same regionally controllable sampling for multi-concept generation across all models and the results are summarized in Fig. 6.

For anime characters, the limited scope of tuning in P+ and Custom Diffusion leads to an excessive encoding of out-domain low-level details within the embedding. This leads to unnatural and less desirable outcomes compared to LoRA and Mix-of>Show. For real-world characters, P+ exhibits oversaturation and produces unnatural results. The fused model of Custom Diffusion compromises text control, leading to the generation of unnatural context and subjects. On the other hand, LoRA fails to effectively preserve the identity of each concept. In comparison, Mix-of>Show successfully maintains identity in multi-concept customization and demonstrates the ability to generalize across different data domains.

4.4 Ablation Study

Single-Client Tuning: Embedding Expressiveness and Editability. In Fig. 7(a), we observe that our decomposed embeddings better preserve the identity of the given concept compared to the vanilla text embeddings in LoRA. Additionally, the use of decomposed embeddings allows for greater editability and higher fidelity, as the LoRA no longer needs to memorize all context details.

Single-Client Tuning: Embedding-LoRA Co-adaptation. Both LoRA and the proposed decomposed embedding are capable of capturing low-level details. However, our objective is to shift more in-domain information to the embedding. As shown in Fig. 7(b), without DropLoRA, the embedding loses its identity, even though it achieves a similar result with LoRA.

Center-Node Fusion: Fusion Type. Using the same ED-LoRA, we conduct experiments with weight fusion and gradient fusion. As shown in Fig. 7(c), gradient fusion effectively preserves the concept identity and details for multi-concept generation.

Regionally Controllable Sampling. As shown in Fig. 7(d), direct sampling lead to attribute binding issues, where the concept identities are mixed. However, our regionally controllable sampling overcomes this problem and achieves correct attribute binding in multi-concept generation.

5 Conclusion

This paper explores decentralized multi-concept customization and highlights the limitations of existing methods such as LoRA tuning and weight fusion, which suffer from concept conflicts and identity loss in this scenario. To overcome these challenges, we propose Mix-of-Show, a framework that combines ED-LoRA for single-client concept tuning and gradient fusion for center-node concept fusion. ED-LoRA preserves the in-domain essence of target concepts within the embedding, thereby avoiding concept conflicts, while gradient fusion minimizes identity loss and achieves consistent stylization of given concepts during concept fusion. Additionally, we introduce regionally controllable sampling to handle attribute binding in multi-concept generation. Experiments demonstrate that Mix-of-Show can successfully generate complex compositions of multiple customized concepts, including characters, objects, and scenes.

References

- [1] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. [1](#), [4](#)
- [2] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. [1](#), [3](#), [5](#)
- [3] Simo Ryu. Low-rank adaptation for fast text-to-image diffusion fine-tuning. <https://github.com/cloneofsimo/lora>. [1](#), [3](#), [5](#), [6](#), [9](#), [10](#), [16](#), [17](#), [18](#)
- [4] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. $p+$: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. [3](#), [4](#), [5](#), [6](#), [9](#), [10](#), [16](#), [17](#), [18](#)
- [5] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *arXiv preprint arXiv:2301.13826*, 2023. [3](#), [4](#)
- [6] Weixi Feng, Xuehai He, Tsu-Jui Fu, Varun Jampani, Arjun Akula, Pradyumna Narayana, Sugato Basu, Xin Eric Wang, and William Yang Wang. Training-free structured diffusion guidance for compositional text-to-image synthesis. *arXiv preprint arXiv:2212.05032*, 2022. [3](#), [4](#), [8](#)
- [7] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. [3](#), [4](#), [8](#)
- [8] Chong Mou, Xiantao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. [3](#), [4](#), [8](#)
- [9] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. [3](#), [4](#), [5](#), [6](#)
- [10] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. [3](#), [14](#), [15](#)
- [11] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. *arXiv preprint arXiv:2212.04488*, 2022. [3](#), [5](#), [9](#), [10](#), [14](#), [15](#), [16](#), [17](#), [18](#)
- [12] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. *arXiv preprint arXiv:2303.11305*, 2023. [3](#)
- [13] Zhiheng Liu, Ruili Feng, Kai Zhu, Yifei Zhang, Kecheng Zheng, Yu Liu, Deli Zhao, Jingren Zhou, and Yang Cao. Cones: Concept neurons in diffusion models for customized generation. *arXiv preprint arXiv:2303.05125*, 2023. [4](#)
- [14] Jing Shi, Wei Xiong, Zhe Lin, and Hyun Joon Jung. Instantbooth: Personalized text-to-image generation without test-time finetuning. *arXiv preprint arXiv:2304.03411*, 2023. [4](#)

- [15] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *arXiv preprint arXiv:2302.13848*, 2023. 4
- [16] Xuhui Jia, Yang Zhao, Kelvin CK Chan, Yandong Li, Han Zhang, Boqing Gong, Tingbo Hou, Huisheng Wang, and Yu-Chuan Su. Taming encoder for zero fine-tuning image customization with text-to-image diffusion models. *arXiv preprint arXiv:2304.02642*, 2023. 4
- [17] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 4, 7
- [18] Qinbin Li, Bingsheng He, and Dawn Song. Model-contrastive federated learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10713–10722, 2021. 4
- [19] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020. 4, 7
- [20] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020. 4
- [21] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas Navarro, Matthew Mattina, Paul N Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. *arXiv preprint arXiv:2111.04263*, 2021. 4
- [22] Maruan Al-Shedivat, Jennifer Gillenwater, Eric Xing, and Afshin Rostamizadeh. Federated learning via posterior averaging: A new perspective and practical algorithms. *arXiv preprint arXiv:2010.05273*, 2020. 4
- [23] Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. Addressing class imbalance in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10165–10173, 2021. 4
- [24] Yujun Shi, Jian Liang, Wenqing Zhang, Vincent YF Tan, and Song Bai. Towards understanding and mitigating dimensional collapse in heterogeneous federated learning. *arXiv preprint arXiv:2210.00226*, 2022. 4
- [25] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems*, 33:7611–7623, 2020. 4
- [26] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv preprint arXiv:1909.06335*, 2019. 4
- [27] Mi Luo, Fei Chen, Dapeng Hu, Yifan Zhang, Jian Liang, and Jiashi Feng. No fear of heterogeneity: Classifier calibration for federated learning with non-iid data. *Advances in Neural Information Processing Systems*, 34, 2021. 4
- [28] Hongyi Wang, Mikhail Yurochkin, Yuekai Sun, Dimitris Papailiopoulos, and Yasaman Khazaeni. Federated learning with matched averaging. *arXiv preprint arXiv:2002.06440*, 2020. 4
- [29] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020. 4
- [30] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H Brendan McMahan. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295*, 2020. 4
- [31] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 4, 8
- [32] Kimin Lee, Hao Liu, Moonkyung Ryu, Olivia Watkins, Yuqing Du, Craig Boutilier, Pieter Abbeel, Mohammad Ghavamzadeh, and Shixiang Shane Gu. Aligning text-to-image models using human feedback. *arXiv preprint arXiv:2302.12192*, 2023. 4, 8

- [33] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Better aligning text-to-image models with human preference. *arXiv preprint arXiv:2303.14420*, 2023. 4, 8
- [34] Wan-Duo Kurt Ma, JP Lewis, W Bastiaan Kleijn, and Thomas Leung. Directed diffusion: Direct control of object placement through attention guidance. *arXiv preprint arXiv:2302.13153*, 2023. 4, 8
- [35] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 4
- [36] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 4
- [37] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 4
- [38] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4
- [39] Kushagra Pandey, Avideep Mukherjee, Piyush Rai, and Abhishek Kumar. Diffusevae: Efficient, controllable and high-fidelity generation from low-dimensional latents. *arXiv preprint arXiv:2201.00308*, 2022. 4
- [40] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *arXiv preprint arXiv:2206.00927*, 2022. 4
- [41] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022. 4
- [42] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 4, 20
- [43] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 4
- [44] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 4
- [45] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022. 4
- [46] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 4
- [47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PMLR, 2021. 4, 14
- [48] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 9, 14
- [49] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989. 9, 14

6 Appendix

6.1 Dataset, Implementation and Evaluation Details

6.1.1 Dataset Details

Previous works, such as Dreambooth [10] and Custom Diffusion [11], have primarily focused on object customization. In contrast, our research encompasses a more comprehensive investigation that involves characters, objects, and scenes. To facilitate our experiments, we curate a dataset comprising 19 different concepts, including 6 real-world characters, 5 anime characters, 6 real-world objects, and 2 real-world scenes. The object parts is borrowed from Dreambooth [10] and Custom Diffusion [11].

6.1.2 Implementation Details

Pretrained Models. In light of the well-known quality issues associated with Stable-Diffusion v1-5 on human faces, we make the decision to switch and adopt the Chilloutmix² as the pretrained model for real-world concepts. Additionally, we employ the Anything-v4³ as the pretrained model for anime concepts. To ensure fair comparisons with other methods, all of our experiments and evaluations are conducted using the same pretrained model.

Single-Client Concept Tuning. In the implementation of ED-LoRA tuning, we incorporate LoRA layers into the linear layers of all attention modules within the text encoder and Unet. Throughout our experiments, we maintain a consistent rank of $r = 4$ for the LoRA layers for simplicity. To optimize the different components, we utilize the Adam optimizer [48] with specific learning rates: 1e-3 for text embedding, 1e-5 for the text encoder, and 1e-4 for the Unet. We add a 0.01 noise offset during single-concept tuning to mitigate the presence of overly bright or dark images.

Center-Node Concept Fusion. In the center-node concept fusion, we apply layer-wise optimization to the layer connected with the LoRA layer. Each layer for optimization is initialized with pretrained weights, and we use the LBFGS optimizer [49]. In detail, we optimize the text encoder layer through 500 steps, while the Unet layer requires 50 steps for optimization.

Sample Details. In all the experiments and evaluations conducted in this paper, we utilize the DDIM sampler with 50 steps. To filter out undesired variations in diffusion models, we employ the same negative prompt for both our and the comparison methods during sampling: "longbody, lowres, bad anatomy, bad hands, missing fingers, extra digit, fewer digits, cropped, worst quality, low quality."

Running Times. In single-client concept tuning, the process of tuning each concept takes approximately 5-10 minutes on two Nvidia-A100 GPUs, taking into account variations in data volume. As for the center-node concept fusion, our experiment involves merging 14 concepts within the pretrained model, and it takes around 50 minutes to complete on a single Nvidia-A100 GPU. It's worth noting that all experiments and running times are conducted in full-precision mode.

6.1.3 Evaluation Setting.

Our evaluation focuses on investigating the each concept in the *single-concept tuned model* and the *center-node fused model*. To assess the performance, we employ the evaluation metric, which includes image-alignment and text-alignment, as outlined in Custom Diffusion [11]. Specifically, for text-alignment, we evaluates the text-image similarity of the sampled image with the corresponding sample prompt in the CLIP feature space [47] by CLIP-Score toolkit⁴. For image-alignment, we evaluate the pairwise image similarity between the sampled image and the target concept data in the CLIP Image feature space [47].

For each concept, we utilize 20 evaluation prompts, which can be roughly categorized into four types: Recontextualization, Restylization, Interaction, and Property Modification. In Recontextualization, we assess the concept's performance by changing its context to different settings, such as the Eiffel Tower or Mount Fuji. In Restylization, we explore the concept's ability to adapt to various artistic styles. In Interaction, we investigate the concept's capability to interact with other objects, such as associations or actions like sitting on a chair. In Property Modification, we modify the internal state of

²<https://civitai.com/models/6424/chilloutmix>

³<https://huggingface.co/andite/anything-v4.0/tree/main>

⁴<https://github.com/jmhessel/clipscore>

	prompts for characters	prompts for pets	prompts for table
Recontextualization	A photo of <TOK> on the beach, small waves, detailed symmetric face, beautiful composition A <TOK>, in front of Eiffel tower A <TOK>, near the mount fuji A <TOK>, in the forest A <TOK>, walking on the street	A <TOK>, in the swimming pool A <TOK>, in front of Eiffel tower A <TOK>, near the mount fuji A <TOK>, in the forest A <TOK>, walking on the street	A <TOK>, in the swimming pool A <TOK>, in front of Eiffel tower A <TOK>, near the mount fuji A <TOK>, in the forest A <TOK>, walking on the street
	A <TOK>, cyberpunk 2077, 4K, 3d render in unreal engine A watercolor painting of a <TOK>	A <TOK>, cyberpunk 2077, 4K, 3d render in unreal engine A watercolor painting of a <TOK>	A <TOK>, cyberpunk 2077, 4K, 3d render in unreal engine A watercolor painting of a <TOK>
	A painting of a <TOK> in the style of Vincent Van Gogh A painting of a <TOK> in the style of Claude Monet A <TOK> in the style of Pixel Art	A painting of a <TOK> in the style of Vincent Van Gogh A painting of a <TOK> in the style of Claude Monet A <TOK> in the style of Pixel Art	A painting of a <TOK> in the style of Vincent Van Gogh A painting of a <TOK> in the style of Claude Monet A <TOK> in the style of Pixel Art
	A <TOK> sit on the chair A <TOK> ride a horse A <TOK>, wearing a headphone A <TOK>, wearing a sunglass A <TOK>, wearing a Santa hat	A <TOK> sit on the chair A <TOK> on the boat A <TOK>, wearing a headphone A <TOK>, wearing a sunglass A <TOK> playing with a ball	A <TOK> sit on the chair A <TOK> on the boat A <TOK>, wearing a headphone A <TOK>, wearing a sunglass A <TOK> playing with a ball
Restylization	A smiling <TOK> An angry <TOK>	A sad <TOK> An angry <TOK>	A sad <TOK> An angry <TOK>
	A running <TOK> A jumping <TOK> A <TOK> is lying down	A running <TOK> A jumping <TOK> A <TOK> is lying down	A running <TOK> A jumping <TOK> A <TOK> is lying down
Interaction			
Property Change			
	prompts for chair	prompts for vase	prompts for scene
Recontextualization	A <TOK>, in the snow A <TOK>, at night A <TOK>, in autumn A <TOK>, in a sunny day A <TOK>, in thunder and lightning	A <TOK>, in the snow A <TOK>, at night A <TOK>, in autumn A <TOK>, in a sunny day A <TOK>, in thunder and lightning	A <TOK>, in the snow A <TOK>, at night A <TOK>, in autumn A <TOK>, in a sunny day A <TOK>, in thunder and lightning
	A <TOK>, cyberpunk 2077, 4K, 3d render in unreal engine A watercolor painting of a <TOK>	A <TOK>, cyberpunk 2077, 4K, 3d render in unreal engine A watercolor painting of a <TOK>	A <TOK>, cyberpunk 2077, 4K, 3d render in unreal engine A watercolor painting of a <TOK>
	A painting of a <TOK> in the style of Vincent Van Gogh A painting of a <TOK> in the style of Claude Monet A <TOK> in the style of Pixel Art	A painting of a <TOK> in the style of Vincent Van Gogh A painting of a <TOK> in the style of Claude Monet A <TOK> in the style of Pixel Art	A painting of a <TOK> in the style of Vincent Van Gogh A painting of a <TOK> in the style of Claude Monet A <TOK> in the style of Pixel Art
	A girl near the <TOK> A boy near the <TOK> A dog near the <TOK> A cat near the <TOK> Many people near the <TOK>	A girl near the <TOK> A boy near the <TOK> A dog near the <TOK> A cat near the <TOK> Many people near the <TOK>	A girl near the <TOK> A boy near the <TOK> A dog near the <TOK> A cat near the <TOK> Many people near the <TOK>
Restylization	A <TOK> in rainbow colors A <TOK> made of metal A close view of <TOK>	A <TOK> in rainbow colors A <TOK> made of metal A close view of <TOK>	A <TOK> in rainbow colors A <TOK> made of metal A close view of <TOK>
	A top view of <TOK> A bottom view of <TOK>	A top view of <TOK> A bottom view of <TOK>	A top view of <TOK> A bottom view of <TOK>
Interaction			
Property Change			

Figure 8: Summarization of our evaluation prompts for each concept.

the concept, including expressions or states like running or jumping. Each type consists of 5 prompts, some of which are borrowed from previous work, resulting in a total of 20 prompts per concept. We sample 50 images for each prompt, ensuring reproducibility by fixing the random seed within the range of [1, 50]. This yields a total of 1000 images for each concept. The evaluation prompts for each concept are presented in Fig. 8.

6.2 Additional Experiments

6.3 Less Language Drift in Mix-of-Show

Previous methods, such as Dreambooth [10] and Custom Diffusion [11], have observed the issue of language drift during concept tuning. This phenomenon occurs when the original class associated with a concept gradually loses its original meaning throughout the tuning process. To tackle this challenge, these methods employ prior-preservation loss to regularize the original class using images sampled from the pretrained model or retrieved from the dataset. However, this problem arises because the concept tokens used in these methods cannot accurately represent the given concept, and the primary source of identity encoding is the weight shift. As a result, the weight shift can impact semantically similar concepts. In contrast, Mix-of-Show mitigates language drift by utilizing decomposed embeddings that effectively capture the in-domain essence of the given concept. As depicted in Fig. 11, our Mix-of-Show exhibits less language drift when compared to other methods.

6.4 Sample without Condition

As shown in Fig. 12, our Mix-of-Show method can also generate multi-concept samples without spatial conditions, particularly when there is a clear distinction in concept semantics.

Single-Concept Model	Methods	Cat (5)	DogA (5)	Chair (5)	Table (4)	DogB (5)	Vase (6)	Mean
Text-alignment	Upper Bound	0.832	0.821	0.795	0.792	0.821	0.807	0.811
	P+ [4]	0.828	0.801	0.726	0.696	0.799	0.776	0.771
	Custom Diffusion [11]	0.784	0.744	0.698	0.689	0.786	0.768	0.745
	LoRA [3]	0.761	0.673	0.655	0.670	0.779	0.779	0.720
Mix-of-Show (Ours)		0.771	0.703	0.666	0.671	0.772	0.759	0.724
Image-alignment	Lower Bound	0.753	0.755	0.674	0.682	0.769	0.692	0.721
	P+ [4]	0.783	0.753	0.809	0.810	0.825	0.761	0.790
	Custom Diffusion [11]	0.869	0.830	0.825	0.888	0.848	0.794	0.842
	LoRA [3]	0.859	0.871	0.872	0.917	0.887	0.778	0.864
Mix-of-Show (Ours)		0.874	0.864	0.890	0.879	0.889	0.811	0.868
Fused Model	Methods	Cat (5)	DogA (5)	Chair (5)	Table (4)	DogB (5)	Vase (6)	Mean
Text-alignment	Upper Bound	0.832	0.821	0.795	0.792	0.821	0.807	0.811
	P+ [4]	0.828	0.801	0.726	0.696	0.799	0.776	0.771
	Custom Diffusion [11]	0.756	0.735	0.726	0.724	0.782	0.758	0.747
	LoRA [3]	0.827	0.803	0.757	0.766	0.814	0.801	0.795
Mix-of-Show (Ours)		0.801	0.738	0.673	0.709	0.786	0.761	0.745
Image-alignment	Lower Bound	0.753	0.755	0.674	0.682	0.769	0.692	0.721
	P+ [4]	0.783	0.753	0.809	0.810	0.825	0.761	0.790
	Custom Diffusion [11]	0.871	0.813	0.774	0.794	0.823	0.773	0.808
	LoRA [3]	0.805	0.800	0.761	0.778	0.808	0.715	0.778
Mix-of-Show (Ours)		0.852	0.863	0.864	0.816	0.880	0.800	0.846

(a) Quantitative results from single-concept model and center-node fused model on *real-world objects*.

Single-Concept Model	Methods	Potter (14)	Hermione (15)	Thanos (15)	Hinton (14)	Lecun (17)	Bengio (15)	Mean
Text-alignment	Upper Bound	0.765	0.776	0.765	0.765	0.765	0.765	0.767
	P+ [4]	0.640	0.744	0.622	0.708	0.693	0.711	0.686
	Custom Diffusion [11]	0.654	0.720	0.594	0.717	0.683	0.674	0.674
	LoRA [3]	0.580	0.696	0.568	0.716	0.681	0.684	0.654
Mix-of-Show (Ours)		0.575	0.650	0.562	0.665	0.680	0.662	0.632
Image-alignment	Lower Bound	0.485	0.458	0.510	0.422	0.510	0.441	0.471
	P+ [4]	0.778	0.608	0.809	0.582	0.614	0.629	0.670
	Custom Diffusion [11]	0.737	0.663	0.852	0.627	0.694	0.710	0.714
	LoRA [3]	0.866	0.679	0.917	0.683	0.716	0.705	0.761
Mix-of-Show (Ours)		0.869	0.785	0.921	0.731	0.723	0.782	0.802
Fused Model	Methods	Potter (14)	Hermione (15)	Thanos (15)	Hinton (14)	Lecun (17)	Bengio (15)	Mean
Text-alignment	Upper Bound	0.765	0.776	0.765	0.765	0.765	0.765	0.767
	P+ [4]	0.640	0.744	0.622	0.708	0.693	0.711	0.686
	Custom Diffusion [11]	0.604	0.678	0.624	0.699	0.651	0.641	0.650
	LoRA [3]	0.693	0.717	0.656	0.694	0.714	0.725	0.700
Mix-of-Show (Ours)		0.632	0.677	0.611	0.673	0.678	0.701	0.662
Image-alignment	Lower Bound	0.485	0.458	0.510	0.422	0.510	0.441	0.471
	P+ [4]	0.778	0.608	0.809	0.582	0.614	0.629	0.670
	Custom Diffusion [11]	0.765	0.680	0.749	0.603	0.693	0.671	0.694
	LoRA [3]	0.558	0.600	0.792	0.412	0.523	0.447	0.555
Mix-of-Show (Ours)		0.827	0.756	0.867	0.710	0.729	0.733	0.770

(b) Quantitative results from single-concept model and center-node fused model on *real-world characters*.

Single-Concept Model	Methods	Rock (20)	Pyramid (20)	Mean	Fused Model	Methods	Rock (20)	Pyramid (20)	Mean
Text-alignment	Upper Bound	0.869	0.798	0.834	Text-alignment	Upper Bound	0.869	0.798	0.834
	P+ [4]	0.801	0.716	0.759		P+ [4]	0.801	0.716	0.759
	Custom Diffusion [11]	0.809	0.686	0.748		Custom Diffusion [11]	0.793	0.682	0.738
	LoRA [3]	0.737	0.697	0.717		LoRA [3]	0.786	0.734	0.760
Mix-of-Show (Ours)		0.754	0.677	0.716	Mix-of-Show (Ours)		0.770	0.702	0.736
Image-alignment	Lower Bound	0.672	0.517	0.595	Image-alignment	Lower Bound	0.672	0.517	0.595
	P+ [4]	0.821	0.770	0.796		P+ [4]	0.821	0.770	0.796
	Custom Diffusion [11]	0.808	0.800	0.804		Custom Diffusion [11]	0.742	0.757	0.750
	LoRA [3]	0.863	0.784	0.824		LoRA [3]	0.810	0.728	0.769
Mix-of-Show (Ours)		0.859	0.857	0.888	Mix-of-Show (Ours)		0.832	0.844	0.838

(c) Quantitative results from single-concept model and center-node fused model on *real-world scenes*.

Table 2: *Text-alignment* and *image-alignment* of the single-client tuned model and the center-node fused model. The upper bound of text-alignment and the lower bound of image-alignment are computed by replacing the concept's token with its class token and sampling using the pretrained model. For instance, to assess the upper-bound text alignment and lower-bound image alignment for the concept " V^{dogA} ," we substitute " V^{dogA} " in the sample prompts with the class token "dog" and sample it using the pretrained model. (N) means each concept has N images for tuning.

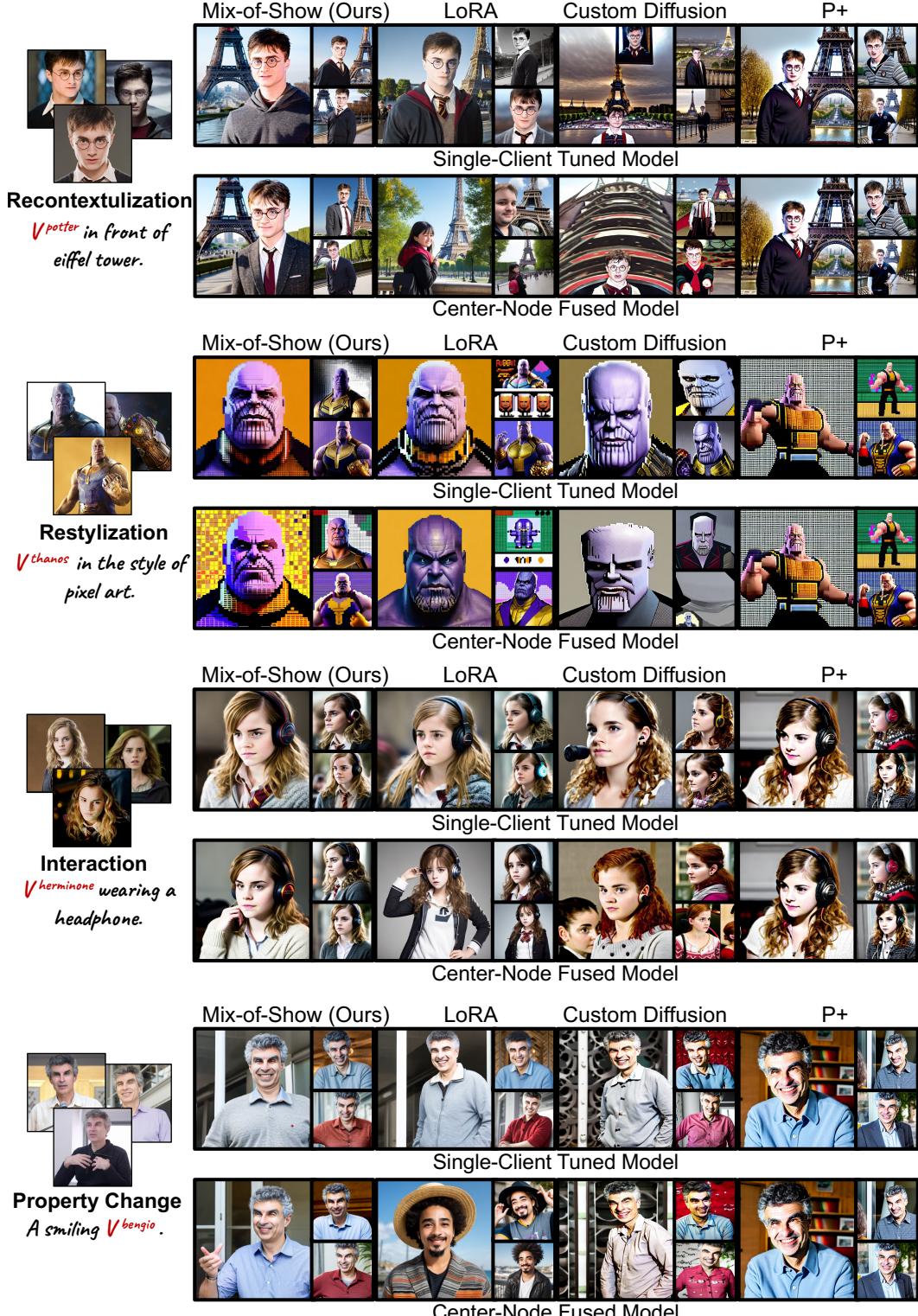


Figure 9: Qualitative comparison of Mix-of-Show vs. P+ [4], Custom Diffusion [11], and LoRA [3]. Our Mix-of-Show demonstrates superior concept identity and quality in single-concept tuned model and achieves the least identity loss after concept fusion.

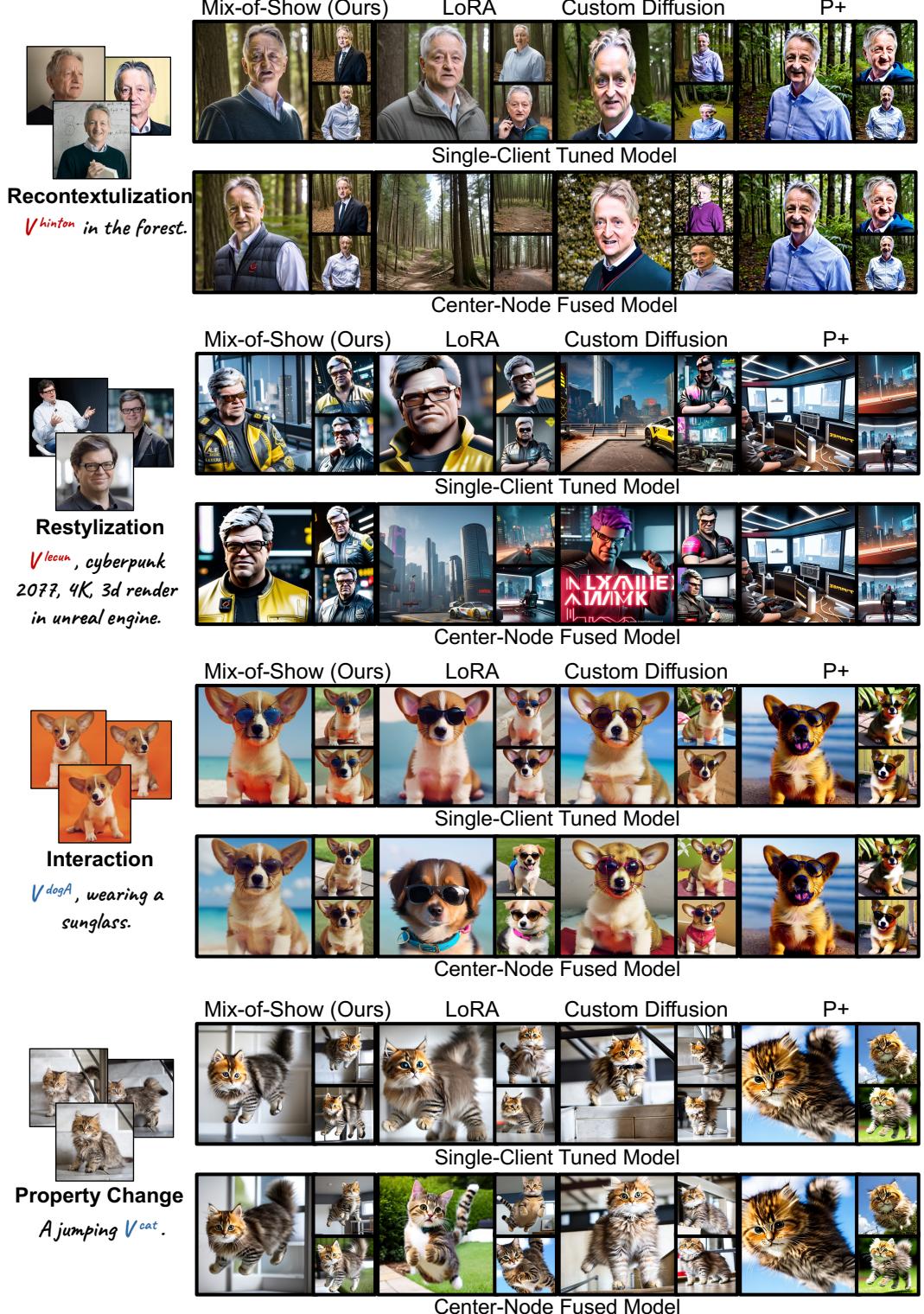


Figure 10: Qualitative comparison of Mix-of-Show vs. P+ [4], Custom Diffusion [11], and LoRA [3]. Our Mix-of-Show demonstrates superior concept identity and quality in single-concept tuned model and achieves the least identity loss after concept fusion.

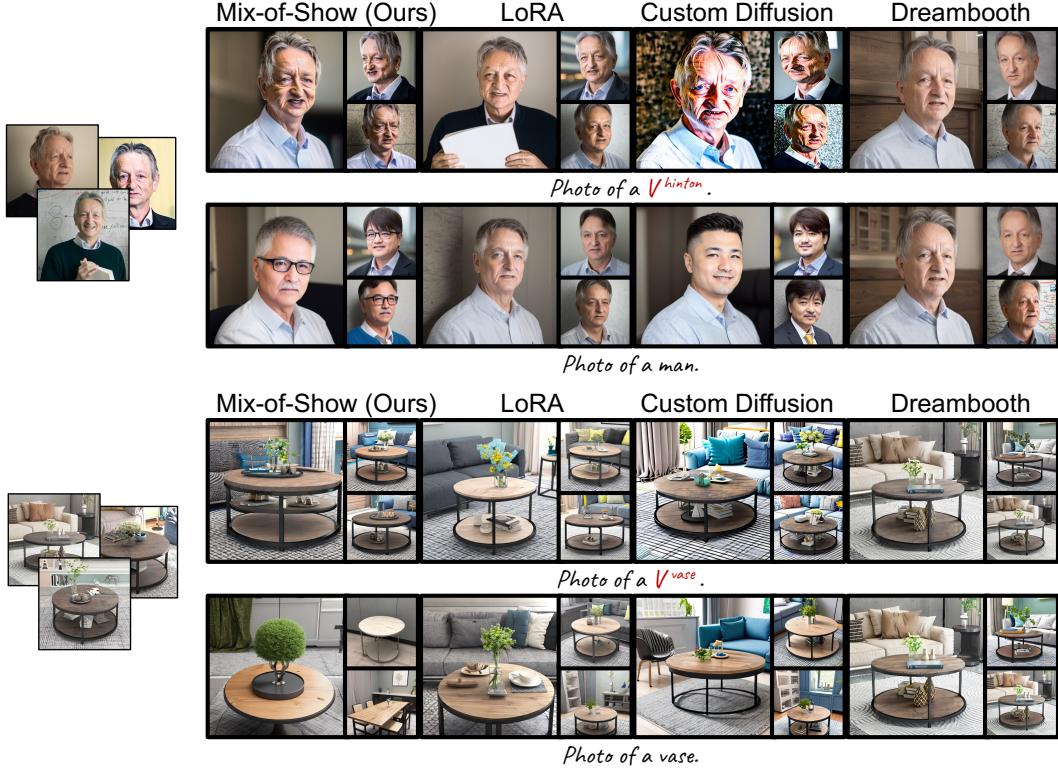


Figure 11: Comparison of language drift among different methods trained without using a regularization dataset on the original class. Mix-of>Show demonstrates the least language drift without generating unnatural results.

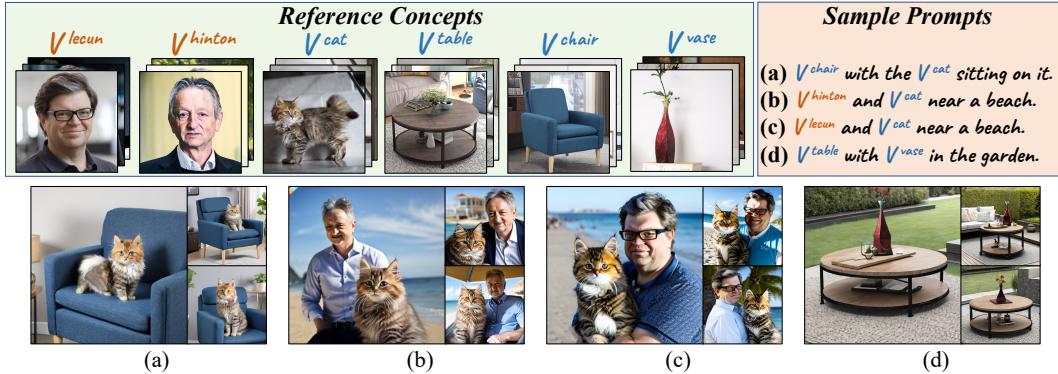


Figure 12: Multi-concept generation of Mix-of>Show without the use of spatial conditions on the semantic-distinct concepts.



Figure 13: Limitation of regionally controllable sampling when the background contains structured elements.

7 Limitation & Future Work

7.1 Limitation

The first limitation is associated with regionally controllable sampling. As illustrated in Fig. 13, when the background contains structured elements, the regionally controllable sampling may disrupt the consistency of these elements. The second limitation pertains to the center-node concept fusion, which requires a relatively long time to merge concepts. The main bottleneck in this process arises from the presence of large spatial features in the Unet layer during layer-wise optimization. To mitigate the fusion time cost, some engineering techniques such as half-precision may help. The final limitation of our method pertains to face identity preservation. In Stable Diffusion, the information lost in the VAE affects the generation of high-quality full-body characters, particularly in the small face region, leading to a loss of facial details. To address this limitation, further exploration of our method’s application to pixel-based text-to-image diffusion models, such as Imagen [42], may provide potential solutions and improvements.

7.2 Future Work

Our Mix-of>Show framework enables the reusability and scalability of tuned concepts, facilitating complex compositions. In future work, it would be interesting to investigate how Mix-of>Show can enhance storybook generation by embedding character and object identities consistently across different plots. Additionally, it is important to explore methods for reducing the computational cost of concept fusion in Mix-of>Show without compromising the identity of the concepts.

7.3 Potential Negative Society Impact

This project aims to provide the community with an effective tool for decentralized creation of high-quality customized concept models and the ability to reuse and combine different concepts to compose complex figures. Our method achieves consistent and superior identity preservation for characters, objects, and scenes, enabling complex compositions. However, there is a risk of malicious parties potentially using this framework to create misleading interactions with real-world characters. This issue is not unique to our approach but is a common concern in other multi-concept customization methods as well. Therefore, future research in multi-concept customization and generation should continue to investigate and address these concerns.