ConceptSplit: Decoupled Multi-Concept Personalization of Diffusion Models via Token-wise Adaptation and Attention Disentanglement

Supplementary Material

Algorithm 1 Denoising steps with Latent Optimization for Disentangled Attention (LODA)

Input: initial latent z_T, prompt $\mathcal{P}$, set of token indices S, set of timesteps $t = \{T, \ldots, 0\}$, threshold γ, Stage1 end step N, Stable Diffusion model SD.
Output: Denoised latent z_0
step $\leftarrow 0$
for $t = T$ down to 0 **do**
  if step $< N$ **then**
    $A_t \leftarrow \mathrm{SD}(z_t, \mathcal{P}, t)$
    for each token index i in S **do**
      $A_t^i \leftarrow A_t[:, :, i]$
      $A_t^i \leftarrow Gaussian(A_t^i)$
      $P_t^i \leftarrow Normalize(A_t^s)$
    end for
    $\mathrm{KL}_t^H \leftarrow \mathrm{HM}(\{\mathrm{KL}_t^{(i,j)} \mid i, j \in S,\ i \neq j\})$
    $\mathcal{L}_{KL} \leftarrow ReLU(\gamma - \mathrm{KL}_t^H)$
    $z_t' \leftarrow z_t - \eta_t \nabla_{z_t} \mathcal{L}_{KL}$
  end if
  $z_{t-1} \leftarrow \mathrm{SD}(z_t', \mathcal{P}, t)$
  step $\leftarrow$ step $+ 1$
end for
return z_0

Method	Dog	Cat
SD	-0.003600	-0.004531
ToVA	-0.003041	-0.005061
Modifying K	-0.000266	-0.000280
Modifying K,V	-0.000359	-0.000176
Cones2	-0.000385	-0.000421
Textual Inversion	-0.000933	-0.001004

Table 1. **Average entropy change of attention maps ($\Delta\mathcal{H}$) across diffusion steps.** We extract attention maps corresponding to the tokens "*cat*" and "*dog*" from the prompt "A photo of a dog sitting next to a cat".

A. Experimental Detail

We used the RTX3090 graphic card for training and inference. For inference, we used DDIM scheduler [11] with 50 steps and 7.5 classifier-free guidance weight [4]. We use Stable Diffusion v2-1[1] with 768x768 resolution as the pretrained model.

[1] https://huggingface.co/stabilityai/stable-diffusion-2-1

Textual Inversion [2]. We use the third-party implementation of huggingface [12] for Textual Inversion. We train each setting with a learning rate of 5×10^{-4}, step size of 3000, and batch size of 4.

DreamBooth [10]. We use the third-party implementation of huggingface [12] for Dreambooth. We train each setting with a learning rate of 1×10^{-5}, step size of $400\times$ number of subjects, and batch size of 1. Prior preservation loss was used, and a loss weight of 1 was used. 200 class images were used for prior preservation loss.

Custom Diffusion [5]. We use the official implementation of custom diffusion[2]. We train each setting with a learning rate of 1×10^{-4}, step size of $500\times$ number of subjects, and batch size of 1. Generated images shown in our paper were trained with prior preservation loss that prevents the leak of personalized concepts when generating other concepts in diffusion models. Loss weight of 1 was used for prior preservation loss. 200 class images were used for prior preservation loss.

Cones2 [6]. We use the official implementation for Cones2[3]. We train each setting with a learning rate of 1×10^{-4}, step size of 1500, and batch size of 1. With prompt regularization. We implement layout guidance for inference 2 objects and 3 objects.

EDLoRA [3]. We use the official implementation for EDLoRA[4]. We implement this code to SD2.1 for comparison. We train each EDLoRA with setting with learning rate of 1×10^{-3} for text embedding, 1×10^{-5} for text encoder and 1×10^{-4} for unet. We set every rank of LoRA to 4. We set the alpha value for gradient fusion to 1 for the U-net and text encoder.

ConceptSplit (Ours)

ToVA. We used LoRA with rank 64 for our ToVA, and we used prompt regularization, as proposed in Cones2 [6] We utilized 200 prompts using ChatGPT [8] and apply different prompt for each iteration. We trained with 300 iterations for our experiments. With a learning rate of 1e-4 and batch size of 1.

LODA. We set LODA step N to 10. with percent hyperparameter γ 0.9 and ReLU threshold τ to 1.0. We set each strength hyperparameter p, m to +5 and -1e8. Update rate η_t was scheduled linearly with $40 - 20 \cdot \frac{t}{T}$, where T denotes the total steps.

[2] https://github.com/adobe-research/custom-diffusion
[3] https://github.com/ali-vilab/Cones-V2
[4] https://github.com/TencentARC/Mix-of-Show.git

# of Concepts	Method	Capacity	Times
Single concept	Textual Inversion [2]	4.2KB	$\sim$ 2h
	DreamBooth [10]	5.2GB	$\sim$ 7m
	Custom Diffusion [5]	97.5MB	$\sim$ 12m
	Cones 2 [6]	4.2KB	$\sim$ 35m
	EDLoRA [3]	6.6 MB	$\sim$ 28m
	ConceptSplit (Ours)	7.4MB	$\sim$ **3m**
Two concepts	Textual Inversion [2]	8.4KB	$\sim$ 4h
	DreamBooth [10]	5.2GB	$\sim$ 14m
	Custom Diffusion [5]	97.5MB	$\sim$ 24m
	Cones 2 [6]	8.4KB	$\sim$ 70m
	EDLoRA [3]	13.2MB	$\sim$ 56m
	ConceptSplit (Ours)	14.8MB	$\sim$ **6m**
Three concepts	Textual Inversion [2]	12.6KB	$\sim$ 6h
	DreamBooth [10]	5.2GB	$\sim$ 21m
	Custom Diffusion [5]	97.5MB	$\sim$ 36m
	Cones 2 [6]	12.6KB	$\sim$ 105m
	EDLoRA [3]	19.8MB	$\sim$ 84m
	ConceptSplit (Ours)	22.2MB	$\sim$ **9m**
Four concepts	Textual Inversion [2]	16.8KB	$\sim$ 8h
	DreamBooth [10]	5.2GB	$\sim$ 28m
	Custom Diffusion [5]	97.5MB	$\sim$ 48m
	Cones 2 [6]	16.8KB	$\sim$ 150m
	EDLoRA [3]	19.8MB	$\sim$ 112m
	ConceptSplit (Ours)	29.6MB	$\sim$ **12m**

Table 2. **Comparison of model capacity and training time across different personalization methods on Stable Diffusion v2.1.** This table presents a comparison of storage size (capacity) and training time for various personalization techniques, including Textual Inversion [2], DreamBooth [10], Custom Diffusion [5], Cones 2 [6], EDLoRA [3], and ConceptSplit (ours). ConceptSplit consistently demonstrates lower capacity requirements and faster training times.

B. Analysis of Attention Entropy

As shown in Figure 3 in the main paper, We found out that such key-modifying methods show disrupted attention, we first extract the attention map from U-net while forwarding, using the attention store class and storing every attention map for steps. After inference is ended, we aggregate these attention maps which have a resolution of 24 in every layer in U-Net. We extract the attention map from 24, as it is known to have the most semantic information [9]. Then averaged them and applied softmax to make them probability distribution. Then we calculated Entropy $\mathcal{H} = -\sum_{m,n} \hat{\mathbf{A}}(m,n) \log \hat{\mathbf{A}}_{(m,n)}$, where $\hat{A} \in \mathbb{R}^{24\times 24}$ denotes aggregated, and normalized attention map. We calculated the average change of this entropy, which shows a detailed slope on 1. We found out that when we modify these keys, the model gets confused, and through attention map seems to be noisy and disrupted as shown in Figure 4 in the main paper.

C. Qualitative results of ToVA Ablation.

We show our Qualitative results of ToVA ablation in Figure 6. These results show that modifying the key directly, results in degraded images.

D. Algorithm of LODA

We show an algorithm of LODA on Algorithm 1, which we discussed in Section 3.3 in the main paper.

E. Qualitative results of single Object

We show qualitative results of single Object personalization in Figure 1. Our methods show comparable results to existing methods.

F. LODA to Stable Diffusion

In Figure 5, we compare the results of applying our LODA to the pre-trained Stable Diffusion [9] model without personalization against Attend-and-Excite [1]. Attend-and-Excite focuses on increasing the maximum attention values, which often leads to significant concept mixing. In contrast, LODA actively relocates and separates objects, enabling Stable Diffusion to effectively distinguish between them. This demonstrates that LODA is not effective in scenarios of personalization, rather it can boost the performance of the pre-trained Stable Diffusion model.

G. Ours on SDXL

We also implement our method to SDXL[7], showing feasibility for both vanilla and personalized settings as shown in figure 4.

H. Hyperparameter Ablations

H.1. The Effect of the Percentile γ

In Figure 2, we illustrate the visual effects of varying the percentile hyperparameter γ. A low γ value means we consider broader attention regions for each token. When applied to Attention Fixing Guidance, this broad consideration leads to excessive removal of overlapping areas, hindering successful personalization. As γ increases to around 80 or higher, proper fixation of attention occurs. However, setting γ too high, such as at 99, results in only very localized regions being fixed, failing to adequately suppress the influence of the "cat" token. Consequently, this leads to images where, for example, a dog has cat whiskers, indicating that the "cat" token's influence was not sufficiently reduced.

H.2. The Effect of p, m

In Figure 3, we present how the images change based on the parameters p and m, which are used to strengthen or weaken

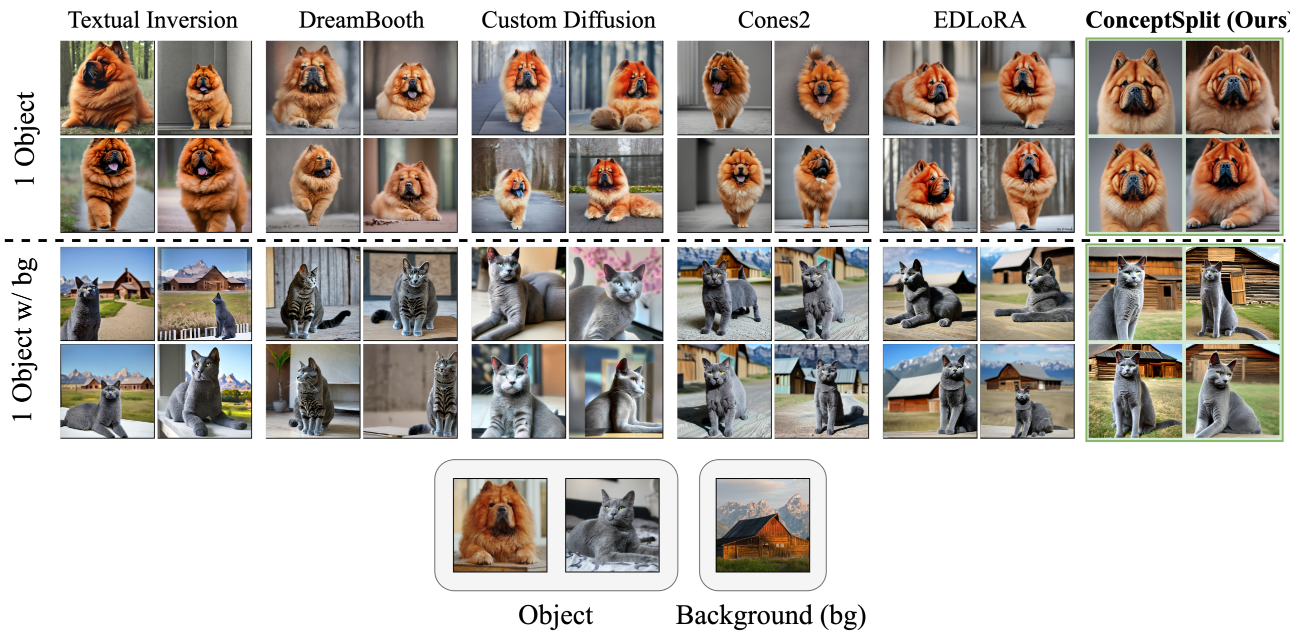

Figure 1. **Qualitative comparison in single-object scenarios on Stable Diffusion 2.1.** In single-object scenarios, our approach ensures that the background is appropriately generated alongside the target concept, maintaining contextual integrity.

attention, respectively. Adjusting the value of p slightly enhances individual features but does not produce significant overall differences in the generated images. However, applying the parameter m has a substantial impact; when m is applied, the influence of each token on other tokens diminishes, causing the learned concepts to appear more distinctly. In contrast, without applying m, the resulting images exhibit a mixed form due to overlapping token influences.

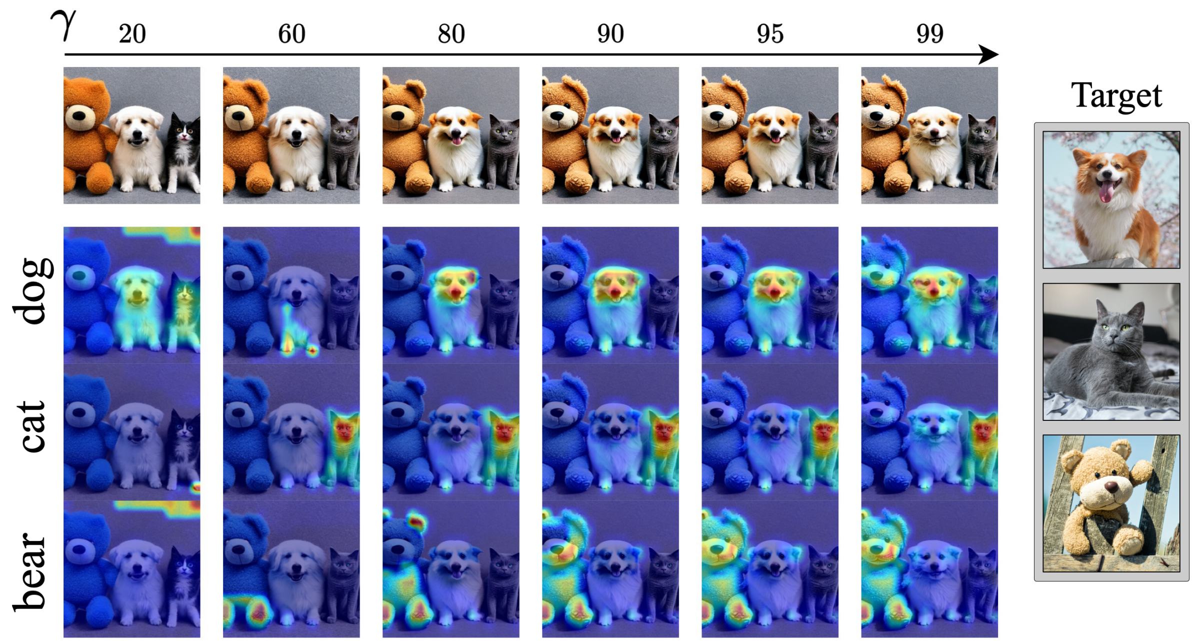

Figure 2. Effect of hyperparameters p and m, which respectively strengthen and weaken the attention scores of each token.

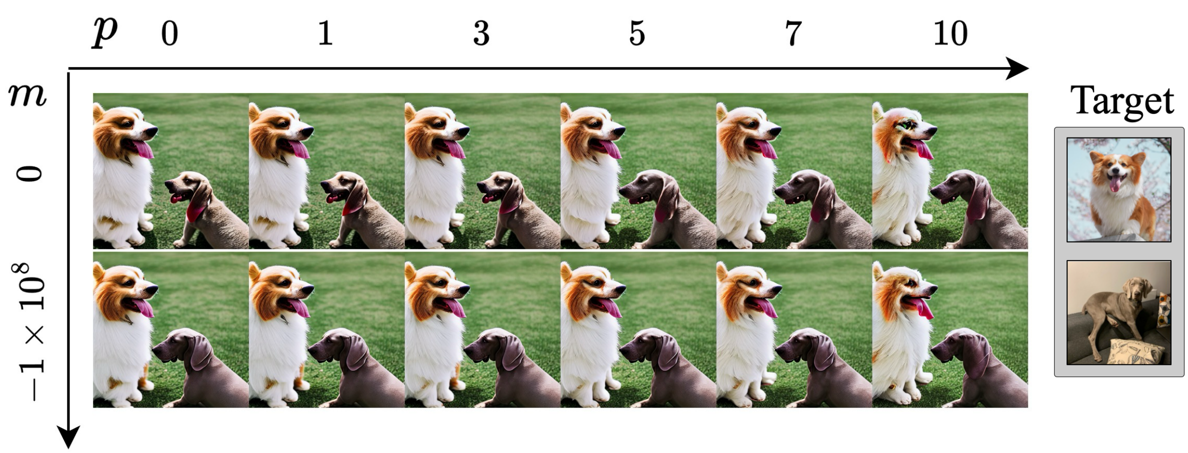

Figure 3. Effect of hyperparameters p and m, which respectively strengthen and weaken the attention scores of each token.

Figure 4. **Application of our method to SDXL.** Our approach is implemented on SDXL, demonstrating feasibility in both vanilla and personalized settings.

Figure 5. **Comparison of Attention-and-Excite (AaE) [1] and LODA on Stable Diffusion 1.5.** This figure illustrates the differences in concept preservation and controllability between AaE and our proposed method, LODA. While AaE focuses on attention refinement to better represent multiple objects, LODA further enhances concept disentanglement, reducing interference between personalized concepts.

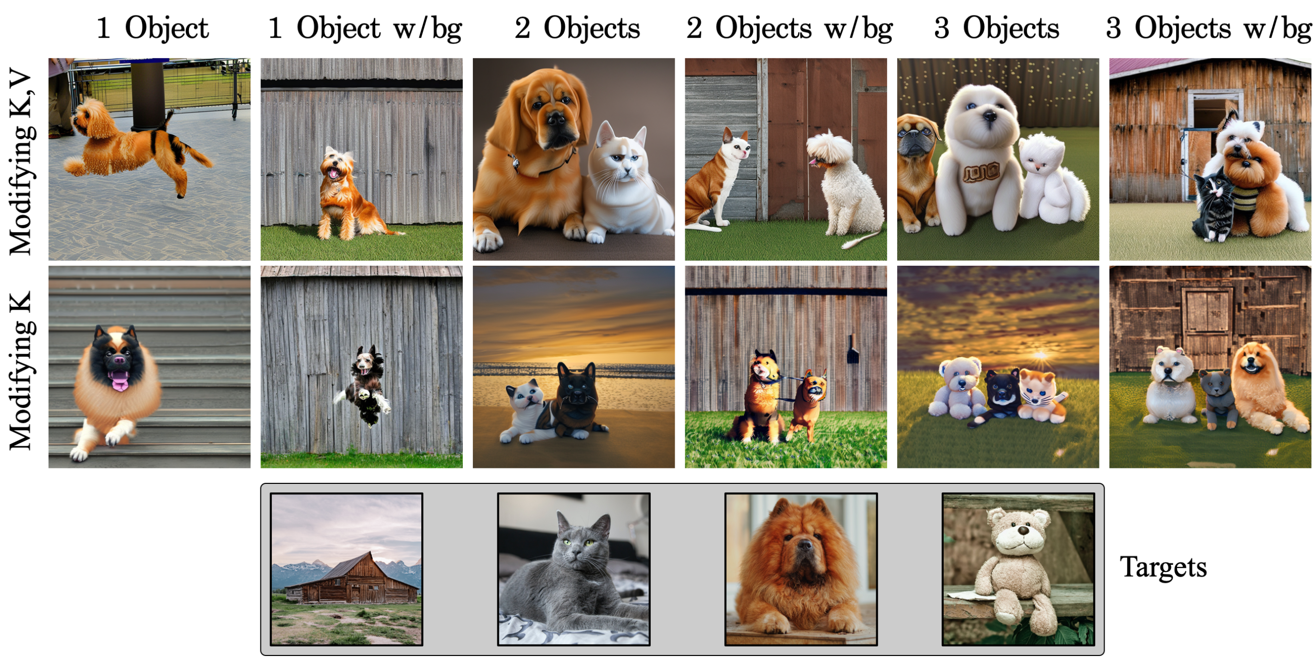

Figure 6. **Qualitative results of ToVA ablation.**

References

[1] 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. *ACM Transactions on Graphics (TOG)*, 42(4):1–10, 2023. 2, 5

[2] 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. 1, 2

[3] Yuchao Gu, Xintao Wang, Jay Zhangjie Wu, Yujun Shi, Yunpeng Chen, Zihan Fan, Wuyou Xiao, Rui Zhao, Shuning Chang, Weijia Wu, et al. Mix-of-show: Decentralized low-rank adaptation for multi-concept customization of diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 2

[4] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 1

[5] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 1, 2

[6] Zhiheng Liu, Yifei Zhang, Yujun Shen, Kecheng Zheng, Kai Zhu, Ruili Feng, Yu Liu, Deli Zhao, Jingren Zhou, and Yang Cao. Cones 2: Customizable image synthesis with multiple subjects. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 57500–57519, 2023. 1, 2

[7] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2

[8] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018. 1

[9] 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. 2

[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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 1, 2

[11] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1

[12] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022. 1