LOW-RANK CONTINUAL PERSONALIZATION OF DIFFUSION MODELS

Łukasz Staniszewski*
Warsaw University of Technology
luks.staniszewski@gmail.com

Katarzyna Zaleska*
Warsaw University of Technology
katarzyna.zaleska2.stud@pw.edu.pl

Kamil Deja
Warsaw University of Technology
IDEAS NCBR
kamil.deja@pw.edu.pl

ABSTRACT

Recent personalization methods for diffusion models, such as Dreambooth and LoRA, allow fine-tuning pre-trained models to generate new concepts. However, applying these techniques across consecutive tasks in order to include, e.g., new objects or styles, leads to a forgetting of previous knowledge due to mutual interference between their adapters. In this work, we tackle the problem of continual customization under a rigorous regime with no access to past tasks' adapters. In such a scenario, we investigate how different adapters' initialization and merging methods can improve the quality of the final model. To that end, we evaluate the naïve continual fine-tuning of customized models and compare this approach with three methods for consecutive adapters' training: sequentially merging new adapters, merging orthogonally initialized adapters, and updating only relevant task-specific weights. In our experiments, we show that the proposed techniques mitigate forgetting when compared to the naïve approach. In our studies, we show different traits of selected techniques and their effect on the plasticity and stability of the continually adapted model. Repository with the code is available at https://github.com/luk-st/continual-lora.

1 INTRODUCTION

Diffusion models (Sohl-Dickstein et al., 2015) (DMs) have revolutionized image generation with state-of-the-art performance in synthesizing detailed objects and blending them with complex styles. Personalization techniques such as Dreambooth (Ruiz et al., 2023), combined with the Low-Rank Adaptation (LoRA, Hu et al. (2021)), introduce the possibility to easily customize large-scale pre-trained models. In those techniques, a selected subset of model parameters is updated with just a few training examples to enable the generation of concepts not present in the original training set.

Unfortunately, customization of a generative model over several tasks leads to the catastrophic forgetting (French, 1999) of previous knowledge due to mutual interference between tasks' adapters. Recent studies (Shah et al., 2023; Frenkel et al., 2024) try to tackle this issue in terms of object and style customization tasks' adapters by carefully merging the trained adapters, either with optimization (Shah et al., 2023) or by targeting different attention layers (Frenkel et al., 2024). However, for the merging to succeed, the adapters have to be available at the same time to properly scale (Shah et al., 2023) or select (Marczak et al., 2024) the resulting models' weights. Even though LoRA reduces the number of adapted parameters, in practice, with many objects or styles, it is unfeasible to store copies of model weights for each of them. Even in the form of sparse matrices, keeping all the adapters is memory-consuming (e.g., approximately 8 LoRA weight matrices with rank 64 are equal to the size of all adapted parameters in Stable Diffusion XL's (Podell et al., 2024) model).

Therefore, in this work, we propose to study the effectiveness of LoRA weights initialization and merging *under the strict continual learning regime* where only the model or model with a single adapter is passed between tasks. In a series of experiments, we compare (1) Naïve continual fine-

*Equal Contribution.

tuning of the low-rank adapter, and three simple merging baselines mitigating forgetting of concepts and styles: (2) consecutive merging of task-specific adapters, (3) merging LoRAs initialized with orthogonal weights, and (4) merging through a selection of weights with the highest magnitude for the task. Our experiments indicate that adding multiple adapters in a Naïve way can lead to a situation where a model converges, in its performance, towards its base form, while all the evaluated techniques mitigate this issue.

We show the significant distinctions between different initialization methods. For example, orthogonalized initialization results in high plasticity but low stability, whereas selection-based merging exhibits the opposite trend. Our findings indicate that optimal performance is achieved by reinitializing the LoRA weights and merging them using a standard approach. We hypothesize that this effect arises from the specific initialization of LoRA weights, which facilitates sparse adaptation of the base model weights.

2 Related work

Diffusion models personalization and customization The topic of Diffusion Models personalization has been broadly studied in recent years. Several works, including Textual Inversion (Gal et al., 2022) learn new word embeddings to represent custom and unique concepts. This idea is further extended by Li et al. (2023); Kumari et al. (2023), with additional regularization (Wu et al., 2024), or features mapping (Wei et al., 2023). On the other hand, DreamBooth (Ruiz et al., 2023) fine-tunes the entire model to incorporate specific subjects into generated images. This idea is further extended in HyperDreamBooth (Ruiz et al., 2024) where a hypernetwork (Ha et al., 2016) is used to generate personalized weights. Recent personalization techniques employ Low-Rank Adaptation to reduce the computational requirements by fine-tuning small, low-rank matrices and keeping the base model frozen. Nevertheless, the simple merging of weights fine-tuned with LoRA does not prevent task interference. Therefore, in ZipLoRA Shah et al. (2023) optimize separately trained adapters for style and object to simplify their merging, and Frenkel et al. (2024) (B-LoRA) target different attention layers of diffusion models for both style and object adapters. OFT (Qiu et al., 2023) and Orthogonal Adaptation (Po et al., 2024) leverage orthogonalization to merge models seamlessly without compromising fidelity. Finally, Dravid et al. (2024) introduce weights2weights – a personalized adapter space, enabling the creation of entirely new personalized models.

Generating style with diffusion models Apart from introducing new objects to the model, there is a line of research aiming for style generation and editing in Diffusion Models. StyleDrop (Sohn et al., 2023) leverages the Muse transformer to fine-tune text-to-image models for generating style-specific images. StyleAligned (Hertz et al., 2023) employs shared attention to maintain a consistent style across generated images, while Style Similarity (Somepalli et al., 2024b) provides a framework for style retrieval from diffusion models' training data and introduces CSD, an image encoder that accurately captures the style of images. UnlearnCanvas (Zhang et al., 2024) offers a comprehensive dataset for evaluating machine unlearning in stylized image generation. Diff-QuickFix (Basu et al., 2024b) and LOCOEDIT (Basu et al., 2024a) enable precise editing of diffusion models' knowledge of specific styles by identifying style-relevant layers.

Continual Learning of Diffusion Models There is a growing line of research employing DMs in continual learning (CL) mainly as a source of rehearsal examples from past tasks (Gao & Liu, 2023). This idea is extended by Cywiński et al. (2024) through classifier guidance towards easy-to-forget examples. Similarly, Jodelet et al. (2023) uses an externally trained text-to-image diffusion model for the same purpose. On the other hand, Zając et al. (2023) explore if contemporary CL methods prevent forgetting in diffusion models.

Smith et al. (2023) introduce the task of continual customization, with the goal of fine-tuning the diffusion model on consecutive personalization tasks. They introduce C-LoRA, where consecutive tasks are learned with a self-regularization technique that penalizes updates corresponding to the already altered weights. STAMINA (Smith et al., 2024) introduces learnable adapter's attention masks parameterized with low-rank MLPs for better scaling to longer task sequences. CIDM (Dong et al., 2024) develop a concept-consolidation loss that explores both the distinctive characteristics of concepts within each task (Task-Specific Knowledge) and shared information across different tasks with similar concepts (Task-Shared Knowledge). Finally, Jha et al. (2024) propose to leverage clas-

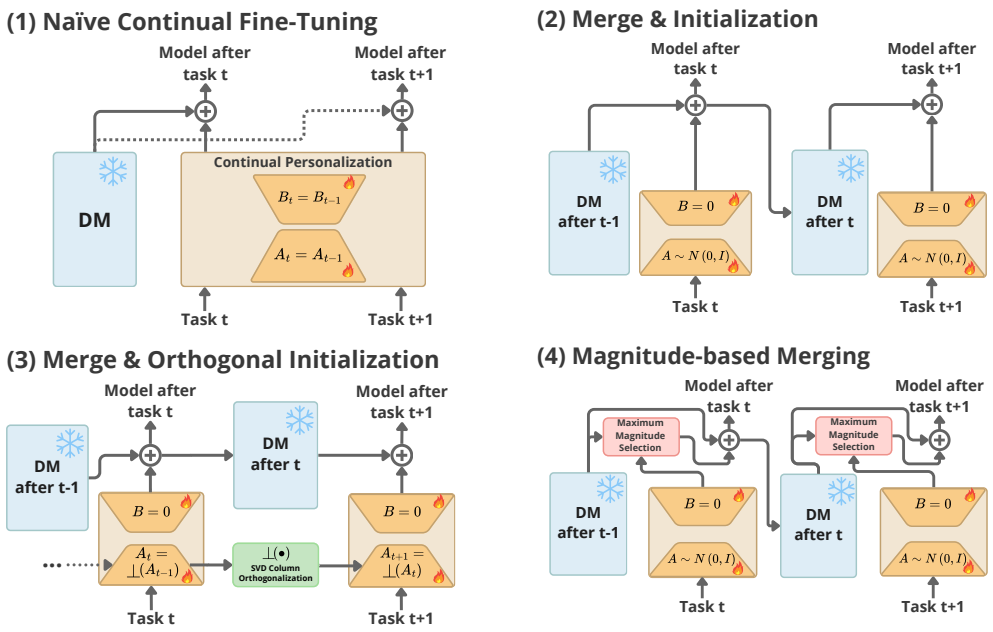


Figure 1: Comparison of initialization and merging in four evaluated methods. The expressions in the adapters refer to how they are initialized with a new task. Numbers next to each schema refer to textual descriptions in Section 3.

sifier scores into both parameter-space (to improve the task-specific Fisher information estimates of EWC) and function-space regularizations (creating a double-distillation framework), achieving high performance in continual personalization. While prior methods improve knowledge consolidation using additional regularizations, in this work, we study the fundamental properties of the most basic components for continual personalization which are merging and initialization techniques.

Model merging The concept of editing models through arithmetic operations applied on models' weights is explored by Ilharco et al. (2023), where authors describe mathematical operations between task vectors, defined as the difference between the weights of the fine-tuned and the base model. To limit the interference between tasks, recent methods replace straightforward averaging with summation of trimmed parameters (Yadav et al., 2024), Fisher-information-based merging (Matena & Raffel, 2022) or with guidance from discriminative model (Jin et al., 2022). In this work, we adapt MAGMAX (Marczak et al., 2024) to our experimental setting, the method achieving state-of-the-art performance in both class-incremental and domain-incremental learning in discriminative modeling. MAGMAX sequentially fine-tunes the model and consolidates the knowledge thanks to maximum magnitude weight selection from several low-rank task adapters.

3 METHODS

We evaluate four different approaches to continual diffusion model personalization. Notation-wise, we denote W_0 as the base model weights, while $\{A_t, B_t\}$ represent the fine-tuned low-rank adapter resulting from personalizing DM on the t-th task. New adapter weights $\{A_t, B_t\}$ are calculated through fine-tuning on data from task t, given the frozen base model weights W_0, and an adapter fine-tuned over $t-1$ tasks denoted as $\{A_{1\ldots t-1}, B_{1\ldots t-1}\}$. The final model can be calculated as $W_{1\ldots T} = W_0 + B_{1\ldots T}A_{1\ldots T}$. All the evaluated strategies differ in either initialization or merging strategy, which we overview in Fig. 1, and describe further in this section.

(1) Naïve continual fine-tuning of LoRA adapters. In the naïve approach, we sequentially fine-tune the adapter weights from the previous task $\{A_{t-1}, B_{t-1}\}$ with the current task data without any reinitialization. Thus, the intermediate adapters, which are continually changed, can be merged into the base model after completing all the T tasks.

(2) Continual adapters merging and standard LoRA reinitialization. We create a new LoRA adapter for each task with a standard initialization technique ($A \sim \mathcal{N}(0, \mathcal{I}), B = 0$). Then, at the end of each task, we merge the adapter with the base model and pass it to the next task. This approach is similar to naïve finetuning. However, thanks to the reinitialization of the LoRA parameters with random values for the A matrix, each task is optimized in its own independent low-dimensional space. Therefore, compared to a naïve approach, the Merge & Initialization technique naturally reduces task interference, as each adapter independently learns only a minimal set of task-specific parameters. Marczak et al. (2025) show that such situation has positive impact on the final model perfromance in discriminative modeling. Intuitively, this can be related to the two-stage training (Kamra et al., 2017) introduced for various CL methods.

(3) Continual adapters merging and orthogonalized reinitialization. We further extend the merging and reinitialization approach in order to limit the interference between tasks by initializing, for the given task t, B_t weights with zeros and setting A_t weights' columns as orthogonal to columns of $A_{1\ldots t-1}$ matrix using Singular Value Decomposition (SVD). We decompose the i-th column of $A_{1\ldots t-1} = \Sigma_{s \in 1\ldots t-1} A_s$ as

$$A^{(i)}_{1\ldots t-1} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \tag{1}$$

and select the last row of the conjugate transpose of the right singular vectors $\mathbf{V}$, corresponding to the smallest singular value. The underlying assumption of this approach is to pre-direct the fine-tuning process toward the non-conflicting adapters, thus limiting the mutual drift of weights that are introduced by consecutive tasks.

(4) Magnitude-based selection of LoRA weights. Finally, we adapt and *modify* the magnitude-based parameter selection method introduced in Marczak et al. (2024). As the original MAGMAX approach involves merging adapters into the model after all the T tasks, continuing the method for $(T+1)$-th task is impossible without storing base model weights θ_0, weights obtained after T tasks θ_T (before selection operation), and the running max-magnitude statistics τ_{MAGMAX}. To that end, after *each task* t, we run the MAGMAX selection of parameters with the highest magnitude by comparing the sum of already merged adapters $\{A_{1\ldots t-1}, B_{1\ldots t-1}\}$ with the current one $\{A_t, B_t\}$:

$$\{A_{1\ldots t}, B_{1\ldots t}\} = \{\text{MAGMAX}(A_{1\ldots t-1}; A_t), \text{MAGMAX}(B_{1\ldots t-1}; B_t)\}. \tag{2}$$

The resulting adapter is passed together with a base model into the next task. At the beginning of a new task, the so far selected adapters are temporarily merged into the frozen Base model, while the new adapters' parameters are initialized in the standard way, optimized, and passed to the selection procedure (2).

4 EXPERIMENTS

Experimental setup. In our experiments, we use the DreamBooth (Ruiz et al., 2023) dataset for fine-tuning on objects, and UnlearnCanvas (Zhang et al., 2024) for styles. We perform fine-tuning on the 10 consecutive tasks and average results over 4 random tasks ordering, with 2 random seeds each. To measure the performance of the model, we calculate cosine similarity between model outputs and reference images with DINO (Caron et al., 2021) and CLIP (Radford et al., 2021) feature extractors for object alignment measure and DINO and CSD (Somepalli et al., 2024a) for style similarity. We report the performance of the final model fine-tuned over all 10 tasks with the Average Score and Average Forgetting metrics. More details about the experimental setup are provided in the Appendix A.

Results. In Table 1 and Table 2, we present the results for the continual object and style personalization, respectively. For both tasks, we observe that the merge & initialization method, either with or without orthogonalization, outperforms naïve continual fine-tuning. On the other hand, magnitude-based merging has the lowest average forgetting at the expense of lower general performance as a result of the lower adaptation capabilities of the method. While magnitude-based weights selection yields state-of-the-art performance in discriminative tasks (Marczak et al., 2024), selecting a limited number of weights hinders the precise merging of new generative traits. Detailed analysis in Appendix B on performance after each task shows that the merge & orthogonalization approach exhibits the best plasticity. However, a simple summation of orthogonal task vectors makes Merge & Orthogonalization fail in retaining original performance on past tasks, leading to the highest forgetting. The Merge & Initialization method balances well a relatively high plasticity with stability, as observed by well-preserved past knowledge.

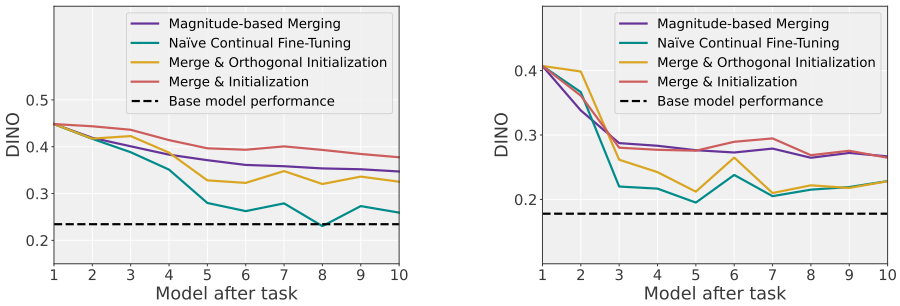
Adapter fine-tuning method	Average Score $\overline{S}_T$ (↑)		Average Forgetting $\overline{F}_T$ (↓)	
	CLIP-I	DINO	CLIP-I	DINO
Base model (reference)	0.586	0.304	-	-
Naïve Continual Fine-Tuning	$0.670_{\pm.013}$	$0.402_{\pm.029}$	$0.063_{\pm.013}$	$0.144_{\pm.039}$
Merge & Initialization	$\mathbf{0.675}_{\pm.005}$	$\mathbf{0.457}_{\pm.011}$	$\underline{0.026}_{\pm.003}$	$\underline{0.056}_{\pm.009}$
Merge & Orthogonal Initialization	$\underline{0.673}_{\pm.014}$	$0.403_{\pm.025}$	$0.072_{\pm.015}$	$0.162_{\pm.033}$
Magnitude-based Merging	$0.643_{\pm.002}$	$\underline{0.408}_{\pm.006}$	$\mathbf{0.018}_{\pm.002}$	$\mathbf{0.036}_{\pm.006}$

Table 1: Average Score and Average Forgetting in CLIP-I and DINO alignment metrics for continual object personalization. The best results are in **bold**, while the second best are underlined.

Adapter fine-tuning method	Average Score $\overline{S}_T$ (↑)		Average Forgetting $\overline{F}_T$ (↓)	
	CSD	DINO	CSD	DINO
Base model (reference)	0.088	0.146	-	-
Naïve Continual Fine-Tuning	$0.345_{\pm.032}$	$0.249_{\pm.010}$	$0.184_{\pm.039}$	$0.136_{\pm.013}$
Merge & Initialization	$\mathbf{0.385}_{\pm.019}$	$\mathbf{0.285}_{\pm.019}$	$\underline{0.131}_{\pm.019}$	$\underline{0.085}_{\pm.021}$
Merge & Orthogonal Initialization	$\underline{0.349}_{\pm.014}$	$\underline{0.252}_{\pm.012}$	$0.204_{\pm.013}$	$0.141_{\pm.013}$
Magnitude-based Merging	$0.289_{\pm.015}$	$0.240_{\pm.009}$	$\mathbf{0.093}_{\pm.013}$	$\mathbf{0.052}_{\pm.015}$

Table 2: Average Score and Average Forgetting in CSD and DINO alignment metrics for continual style personalization. The best results are in **bold**, while the second best are underlined.

In Fig. 2, we compare the performance of different methods on the first task after continual adaptation across multiple tasks in object (left) and style (right) personalization. The results show that the naïve approach degrades overall quality to the baseline model's performance, while other techniques mitigate forgetting more effectively. Notably, knowledge retention varies significantly between object and style personalization. While the Merge & Initialization method reduces forgetting in both cases, resetting weights with orthogonal initialization performs significantly worse in style personalization. We hypothesize that this is due to task vectors being more aligned in style personalization. In such cases, more orthogonal vectors lead to higher forgetting.



(a) Object personalization

(b) Style personalization

Figure 2: DINO score on the first task over continual fine-tuning on next tasks for the object and styles customization.

5 Conclusions

In this work, we analyze the continual adaptation capabilities of a DreamBooth method coupled with the Low-Rank Adaptation (LoRA) technique. We evaluate three different approaches to merging and initialization of the individual adapters. We show that naïve continual training of a LoRA leads to catastrophic forgetting, while other techniques can mitigate this issue.

ACKNOWLEDGMENTS

The project was funded by the National Science Centre, Poland, grants no: 2023/51/B/ST6/03004 and 2022/45/B/ST6/02817. The computational resources were provided by PLGrid grant no. PLG/2024/017114 and no. PLG/2024/017266.

REFERENCES

Samyadeep Basu, Keivan Rezaei, Priyatham Kattakinda, Vlad I. Morariu, Nanxuan Zhao, Ryan A. Rossi, Varun Manjunatha, and Soheil Feizi. On mechanistic knowledge localization in text-to-image generative models. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024a. URL https://openreview.net/forum?id=fsVBsxjRER.

Samyadeep Basu, Nanxuan Zhao, Vlad I. Morariu, Soheil Feizi, and Varun Manjunatha. Localizing and editing knowledge in text-to-image generative models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024b. URL https://openreview.net/forum?id=Qmw9ne6SOQ.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021.

Bartosz Cywiński, Kamil Deja, Tomasz Trzciński, Bartłomiej Twardowski, and Łukasz Kuciński. Guide: Guidance-based incremental learning with diffusion models. *arXiv preprint arXiv: 2403.03938*, 2024. URL https://arxiv.org/abs/2403.03938v1.

Jiahua Dong, Wenqi Liang, Hongliu Li, Duzhen Zhang, Meng Cao, Henghui Ding, Salman Khan, and Fahad Khan. How to continually adapt text-to-image diffusion models for flexible customization? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=O4RCFjVUBJ.

Amil Dravid, Yossi Gandelsman, Kuan-Chieh Wang, Rameen Abdal, Gordon Wetzstein, Alexei A. Efros, and Kfir Aberman. Interpreting the weight space of customized diffusion models. *arXiv preprint arXiv: 2406.09413*, 2024.

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 1999.

Yarden Frenkel, Yael Vinker, Ariel Shamir, and Daniel Cohen-Or. Implicit style-content separation using b-lora, 2024. URL https://arxiv.org/abs/2403.14572.

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.

R. Gao and Weiwei Liu. Ddgr: Continual learning with deep diffusion-based generative replay. *International Conference on Machine Learning*, 2023. URL https://openreview.net/forum?id=RlqgQXZx6r.

David Ha, Andrew Dai, and Quoc V. Le. Hypernetworks. *arXiv preprint arXiv: 1609.09106*, 2016.

Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. Style aligned image generation via shared attention. *arXiv preprint arxiv:2312.02133*, 2023.

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, 2021.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic, 2023. URL https://arxiv.org/abs/2212.04089.

Saurav Jha, Shiqi Yang, Masato Ishii, Mengjie Zhao, Christian Simon, Muhammad Jehanzeb Mirza, Dong Gong, Lina Yao, Shusuke Takahashi, and Yuki Mitsufuji. Mining your own secrets: Diffusion classifier scores for continual personalization of text-to-image diffusion models. *arXiv preprint arXiv: 2410.00700*, 2024.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. *arXiv preprint arXiv:2212.09849*, 2022.

Quentin Jodelet, Xin Liu, Yin Jun Phua, and T. Murata. Class-incremental learning using diffusion model for distillation and replay. *IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, 2023. doi: 10.1109/ICCVW60793.2023.00367.

Nitin Kamra, Umang Gupta, and Yan Liu. Deep generative dual memory network for continual learning. *arXiv preprint arXiv:1710.10368*, 2017.

Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. *CVPR*, 2023.

Dongxu Li, Junnan Li, and Steven C. H. Hoi. Blip-diffusion: Pre-trained subject representation for controllable text-to-image generation and editing, 2023. URL https://arxiv.org/abs/2305.14720.

Daniel Marczak, Bartłomiej Twardowski, Tomasz Trzciński, and Sebastian Cygert. Magmax: Leveraging model merging for seamless continual learning. *arXiv preprint arXiv: 2407.06322*, 2024.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D. Bagdanov, and Joost van de Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. *arXiv preprint arXiv: 2502.04959*, 2025.

Michael S Matena and Colin A Raffel. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022.

Ryan Po, Guandao Yang, Kfir Aberman, and Gordon Wetzstein. Orthogonal adaptation for modular customization of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7964–7973, June 2024.

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. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=di52zR8xgf.

Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. In *NeurIPS*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

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, 2023. URL https://arxiv.org/abs/2208.12242.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6527–6536, 2024.

Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. Ziplora: Any subject in any style by effectively merging loras. *arXiv preprint arxiv:2311.13600*, 2023.

James Seale Smith, Yen-Chang Hsu, Lingyu Zhang, Ting Hua, Zsolt Kira, Yilin Shen, and Hongxia Jin. Continual diffusion: Continual customization of text-to-image diffusion with c-lora. *arXiv preprint arXiv: 2304.06027*, 2023.

James Seale Smith, Yen-Chang Hsu, Zsolt Kira, Yilin Shen, and Hongxia Jin. Continual diffusion with stamina: Stack-and-mask incremental adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1744–1754, 2024.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR, 2015.

Kihyuk Sohn, Nataniel Ruiz, Kimin Lee, Daniel Castro Chin, Irina Blok, Huiwen Chang, Jarred Barber, Lu Jiang, Glenn Entis, Yuanzhen Li, Yuan Hao, Irfan Essa, Michael Rubinstein, and Dilip Krishnan. Styledrop: Text-to-image generation in any style. *arXiv preprint arXiv: 2306.00983*, 2023.

Gowthami Somepalli, Anubhav Gupta, Kamal Gupta, Shramay Palta, Micah Goldblum, Jonas Geiping, Abhinav Shrivastava, and Tom Goldstein. Measuring style similarity in diffusion models, 2024a. URL `https://arxiv.org/abs/2404.01292`.

Gowthami Somepalli, Anubhav Gupta, Kamal Gupta, Shramay Palta, Micah Goldblum, Jonas Geiping, Abhinav Shrivastava, and Tom Goldstein. Measuring style similarity in diffusion models. *arXiv preprint arXiv: 2404.01292*, 2024b.

Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and W. Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. *IEEE International Conference on Computer Vision*, 2023. doi: 10.1109/ICCV51070.2023.01461.

Feize Wu, Yun Pang, Junyi Zhang, Lianyu Pang, Jian Yin, Baoquan Zhao, Qing Li, and Xudong Mao. Core: Context-regularized text embedding learning for text-to-image personalization, 2024. URL `https://arxiv.org/abs/2408.15914`.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36, 2024.

Michał Zając, Kamil Deja, Anna Kuzina, Jakub M. Tomczak, Tomasz Trzciński, Florian Shkurti, and Piotr Miłoś. Exploring continual learning of diffusion models. *arXiv preprint arXiv: 2303.15342*, 2023.

Yihua Zhang, Chongyu Fan, Yimeng Zhang, Yuguang Yao, Jinghan Jia, Jiancheng Liu, Gaoyuan Zhang, Gaowen Liu, Ramana Rao Kompella, Xiaoming Liu, and Sijia Liu. Unlearncanvas: Stylized image dataset for enhanced machine unlearning evaluation in diffusion models, 2024. URL `https://arxiv.org/abs/2402.11846`.

APPENDIX

A EXPERIMENTAL SETUP

A.1 DATASETS

DreamBooth. To fine-tune the model on personalized objects, the DreamBooth (Ruiz et al., 2023) dataset is used, which contains images of 30 objects (9 of them being dogs and cats, and 21 being other objects) belonging to 15 different classes. Each object is represented by several images (from 4 up to 6), which represent it in either a different environment or from a different camera angle. Due to the quantity limits, we follow Ruiz et al. (2023) and use the same dataset of images for both evaluation and fine-tuning.

UnlearnCanvas. For style-related tasks, we use the UnlearnCanvas collection (Zhang et al., 2024), which contains 20 objects (each with 20 images) in 60 different styles. When evaluating methods for the style tasks, we select 10 images representing 10 different objects in the same style but the ones that were not used during training.

We generate datasets for our experiments by randomly selecting classes and objects. For object-related tasks, we select 10 classes, each represented by a unique object and paired with a unique token. For style-based tasks, we select 10 distinct styles, each paired with one object and five corresponding images. Object names, classes, and styles were uniquely selected to avoid repetition. Selected classes are presented in Table 3.

Style name	byzantine, palette knife, ink art, blossom season, artist sketch, winter, dadaism, dapple, meteor shower, pastel
Object name	sneaker, boot, backpack, stuffed animal, candle, can, glasses, bowl, cartoon, teapot

Table 3: Objects and styles names used during experiments.

A.2 FINETUNING

We perform the experiments in a class-incremental setup, as illustrated in Figure 1. Starting with a base text-to-image Stable Diffusion XL model, we sequentially train it on the following tasks. After each task, depending on the approach, we either merge the obtained LoRA weights with the base model and pass it to the next task or store only the LoRA weights to initialize adapters for the subsequent steps.

We train the model on $T = 10$ tasks. For each object task, we use distinct DreamBooth tokens, while for styles, we retain the real names. The learning rate is set to 1e-5, with 850 train steps and a batch size of 1. During fine-tuning, we don't leverage the prior preservation regularization Ruiz et al. (2023).

In the original work introducing LoRA (Hu et al., 2021), the authors investigate how both the rank and the target matrices of the transformer's self-attention module, to which the LoRA is applied, influence method efficiency. They show that, while the impact of rank r is small, it is recommended to apply the LoRA to W^Q, W^K, W^V, and W^O. In our experiments, we similarly apply LoRA to these targets within both the self-attention and cross-attention layers of the U-Net and set $r = 64$.

To ensure the robustness of our experiments, all four methods for low-rank continual personalization methods were evaluated with two seeds (0, 5) and four task ordering seeds (0, 5, 10, 42).

A.3 Evaluation metrics

To evaluate fine-tuned model generations, we use image alignment metrics to calculate the cosine similarity between embeddings of generated and reference images. For subject alignment, we use features extracted from the DINO Caron et al. (2021) and CLIP Radford et al. (2021) models. When assessing style, we continue using DINO but replace CLIP with CSD Somepalli et al. (2024a), as CLIP has demonstrated a bias towards image content, making it unsuitable for style-related tasks. When sampling from the text-to-image diffusion model, we generate $N = 8$ images from each of $P = 5$ prompts and calculate the mean cosine similarity of models' embeddings between each of $N \cdot P = 40$ samples and the object/style reference images. When generating an image, we do it with 50 denoising diffusion steps.

When investigating model performance in a continual learning regime, we use two metrics: average score $\overline{S}_T = \frac{1}{T}\sum_{j=1}^{T} S_j^T$ (with S_j^i being performance on j-th task of a model that was trained on i consecutive tasks) and average forgetting $\overline{F}_T = \frac{1}{T-1}\sum_{j=1}^{T-1} \max_{1\leq k\leq T}(S_j^k - S_j^T)$, with S_j^T being score (CLIP-I, DINO, CSD) on j-th task of a model that was trained on T consecutive tasks.

B Style personalization results

The model generally achieves its highest performance on each task immediately after fine-tuning, but this performance tends to degrade as new tasks are added. Despite this decline, our evaluated techniques can retain knowledge from previous tasks. As presented in Figures 9 and 14 for object personalization and Figures 19 and 24 for style personalization, we can observe that the merge & orthogonal initialization method exhibits the highest plasticity in all metrics for both objects and styles. For some cases (tasks 1 and 2 in merge & initialization method in Figure 9), we can observe backward transfer, a situation where the performance on the previous task improves thanks to the adaptation to the new task.

C Prompts for styles

Due to the complex nature of style personalization, this task is characterized by the easy overwriting of previous styles when new ones come adapted by the Dreambooth method. This fact is extremely harmful especially in the context of continued learning, where it implies catastrophic forgetting of the model. In our experiments, we note that a significant problem in the context of personalization of styles is the way they are expressed in the form of a prompt. To this end, we investigate how different forms of prompts affect the average score, average forgetting, and task scores of a diffusion model that learns the next styles sequentially.

We evaluate models using six different prompts while training them across five consecutive style-related tasks, aiming to identify the prompt that achieved the highest accuracy and lowest forgetting. The results are summarized in Table 4. We also measure the average CSD alignment score and its changes across successive tasks - results in Table 5. We observe that using classic Dreambooth prompts like 'image in the S^* style' introduces worse results than specifying the full names of the styles, especially with the token 'style' inside.

Prompt Template	**DINO** $\overline{S}_T$ (↑)	**DINO** $\overline{F}_T$ (↓)	**CSD** $\overline{S}_T$ (↑)	**CSD** $\overline{F}_T$ (↓)
image of {} in {token} style	0.188	<u>0.215</u>	0.237	**0.272**
image of {} in {name} style	0.289	0.249	0.418	0.359
image of {} in {token}	0.260	0.279	0.377	0.433
image of {} in {name}	<u>0.292</u>	0.237	<u>0.423</u>	0.348
{token} image of {}	0.268	0.249	0.387	0.389
{name} image of {}	**0.299**	**0.201**	**0.446**	<u>0.275</u>

Table 4: Performance metrics average score and average forgetting across different style prompts templates for DINO and CSD.

Prompt Template	$\mathbf{Task_1}$	$\mathbf{Task_2}$	$\mathbf{Task_3}$	$\mathbf{Task_4}$	$\mathbf{Task_5}$
image of {} in {token} style	0.365	0.267	0.213	0.117	0.084
image of {} in {name} style	**0.503**	0.362	0.321	0.186	0.261
image of {} in {token}	0.491	0.303	0.236	0.115	0.168
image of {} in {name}	0.493	0.371	0.337	0.191	0.261
{token} image of {}	0.487	0.350	0.250	0.130	0.231
{name} image of {}	0.487	**0.410**	**0.397**	**0.255**	**0.289**

Table 5: The average value of the CSD metric demonstrates how models perform on the selected tasks, where 1 represents the most recently trained task, and 5 represents the earliest task.

D EXAMPLE SAMPLES

Figure 3 presents object images generated by the models fine-tuned on all 10 consecutive tasks using our continual learning methods, while Figure 4 presents it for styles.

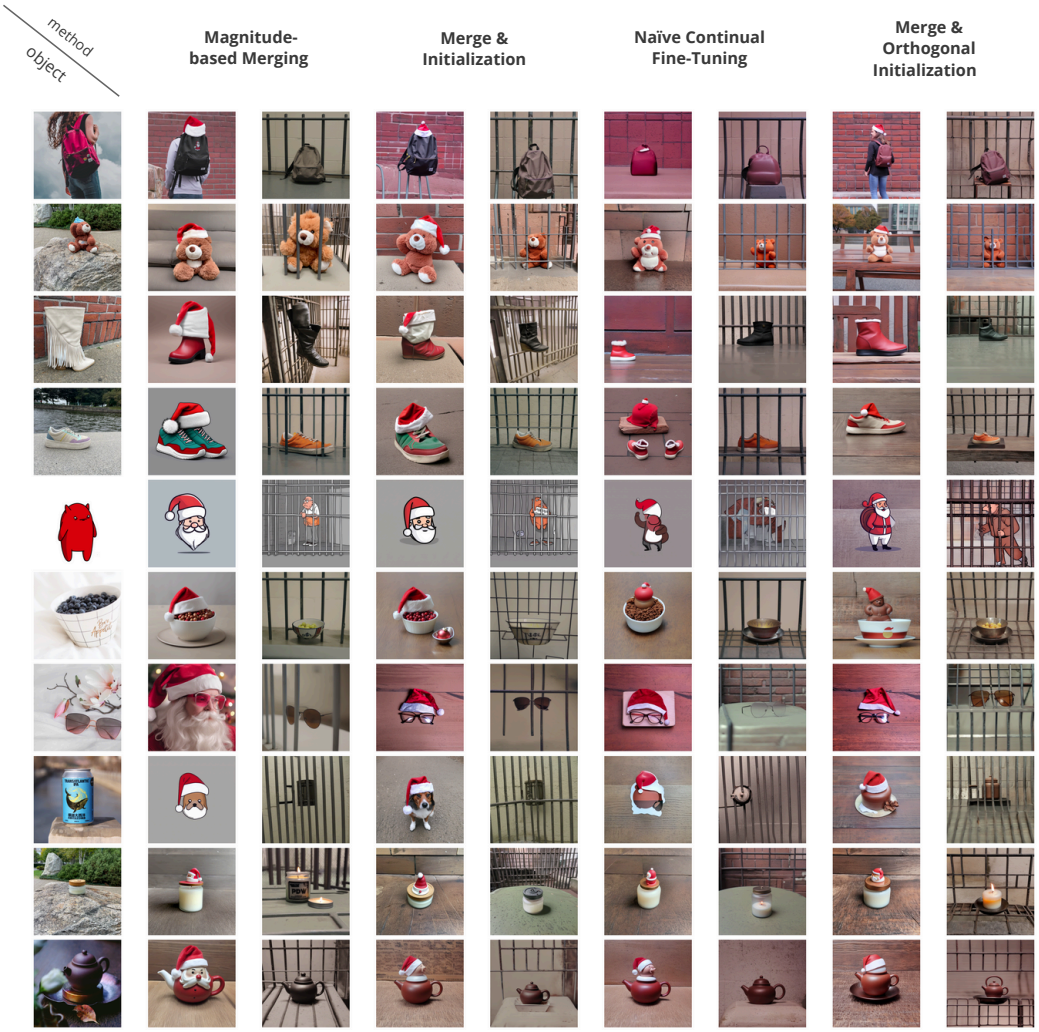


Figure 3: Images generated by the model fine-tuned on 10 tasks. The first left column represents the subsequent tasks, each represented by an image illustrating the object to learn. We show how each method sequentially performs using the prompt templates: 'a V^* wearing a santa hat' and 'a V^* in a jail'.

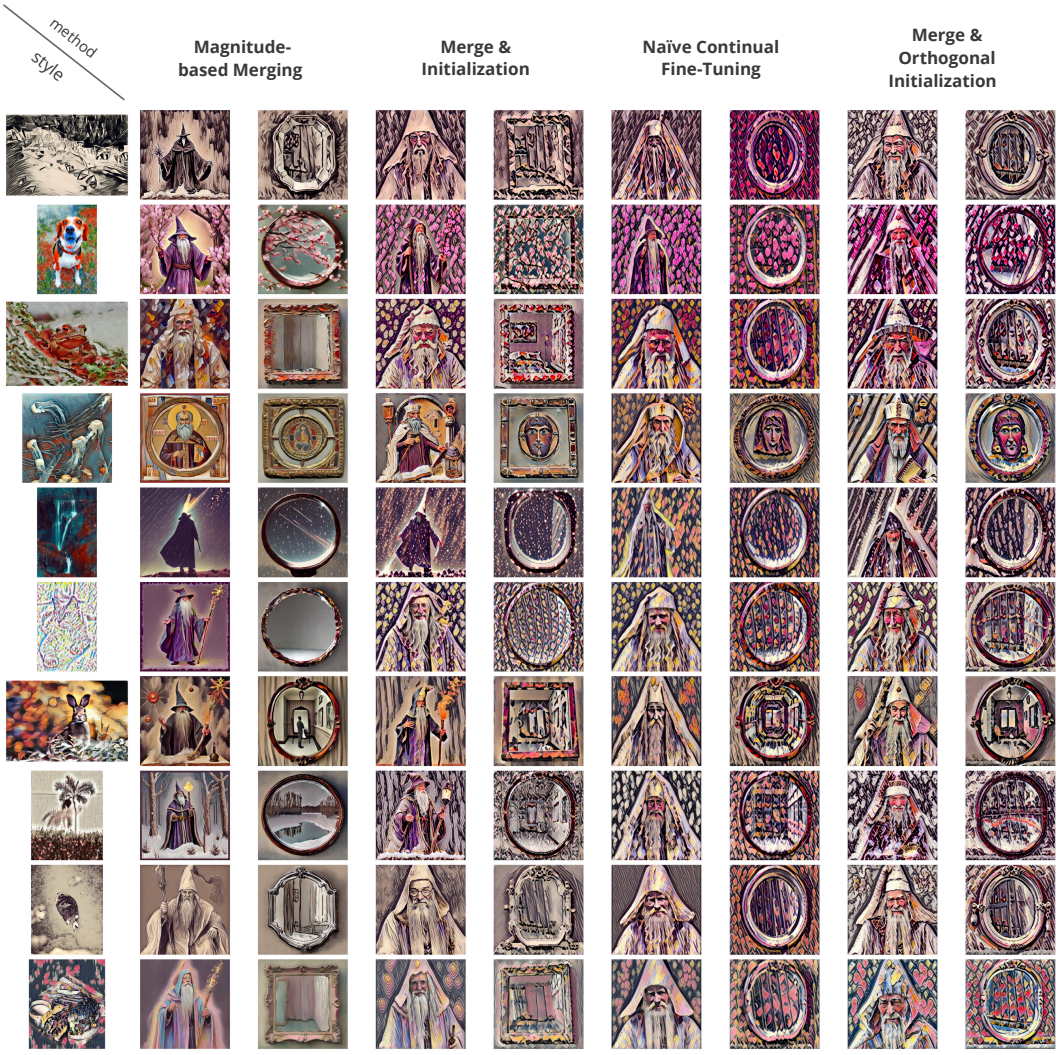


Figure 4: Images generated by the model fine-tuned on 10 tasks. The first left column represents the subsequent tasks, each represented by an image illustrating the style to learn. We show how each method sequentially performs using the prompt templates: '{style_name} image of a wizard' and '{style_name} image of a mirror'.

Figure 5: Naïve continual fine-tuning of LoRA adapters

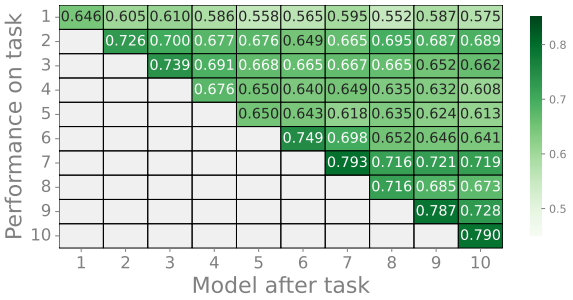


Figure 6: Continual adapters merging and standard LoRA reinitialization

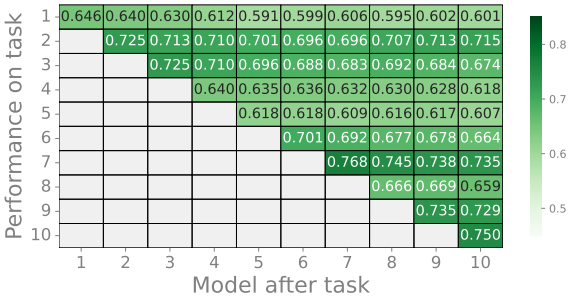


Figure 7: Continual adapters merging and orthogonalized reinitialization

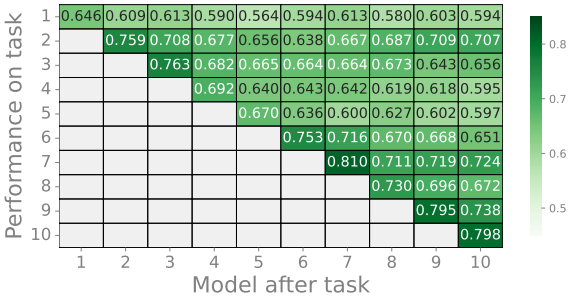


Figure 8: Magnitude-based selection of LoRA weights

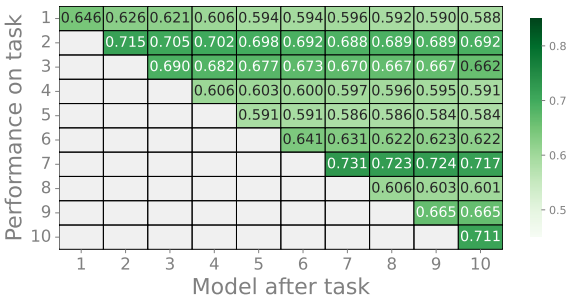


Figure 9: Heatmap of CLIP-I alignment scores for each task in continual object personalization.

Figure 10: Naïve continual fine-tuning of LoRA adapters

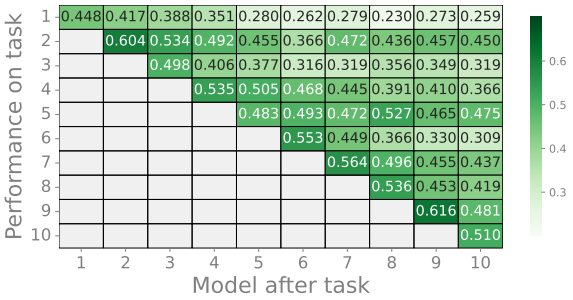


Figure 11: Continual adapters merging and standard LoRA reinitialization

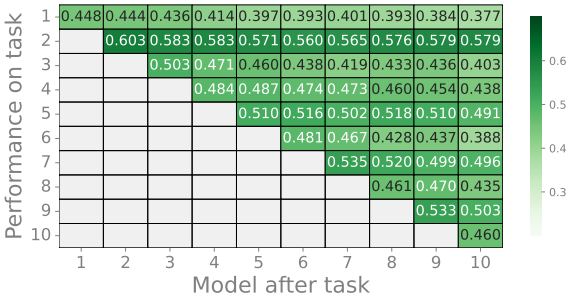


Figure 12: Continual adapters merging and orthogonalized reinitialization

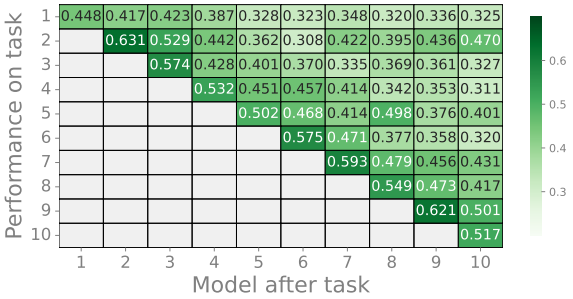


Figure 13: Magnitude-based selection of LoRA weights

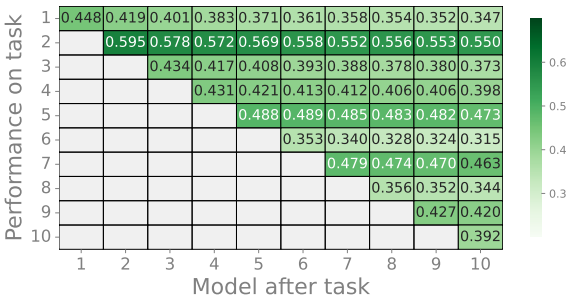


Figure 14: Heatmap of DINO alignment scores for each task in continual object personalization.

Figure 15: Naïve continual fine-tuning of LoRA adapters

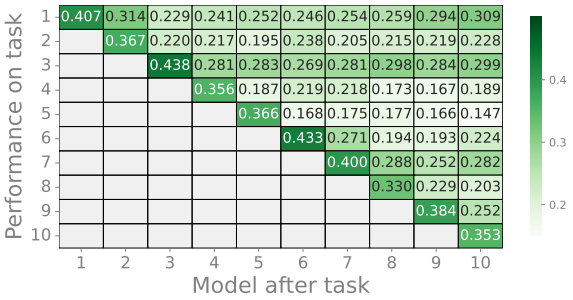


Figure 16: Continual adapters merging and standard LoRA reinitialization

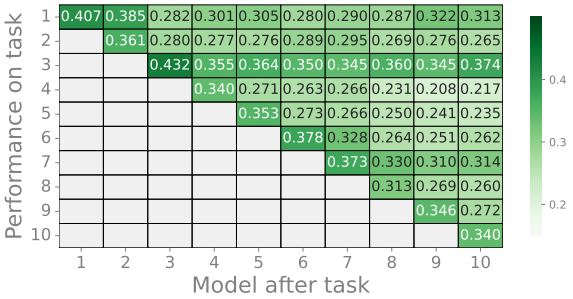


Figure 17: Continual adapters merging and orthogonalized reinitialization

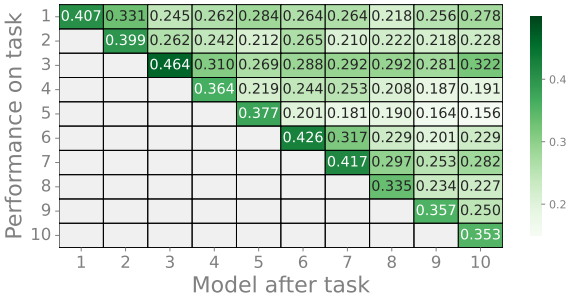


Figure 18: Magnitude-based selection of LoRA weights

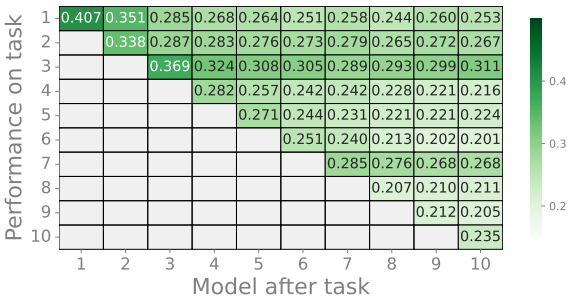


Figure 19: Heatmap of DINO alignment scores for each task in continual style personalization.

Figure 20: Naïve continual fine-tuning of LoRA adapters

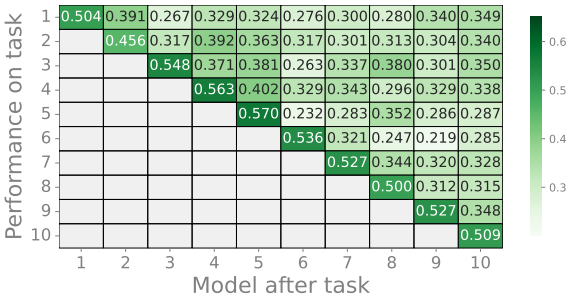


Figure 21: Continual adapters merging and standard LoRA reinitialization

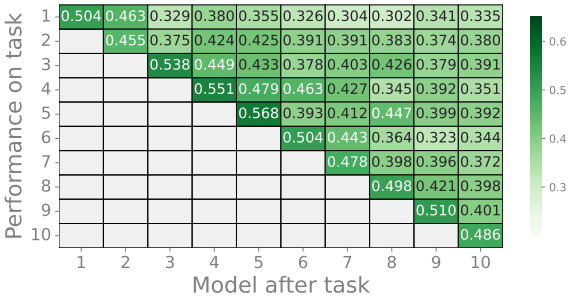


Figure 22: Continual adapters merging and orthogonalized reinitialization

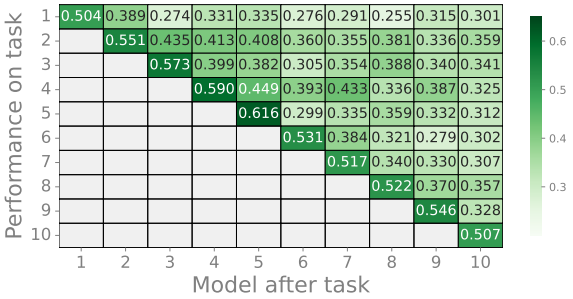


Figure 23: Magnitude-based selection of LoRA weights

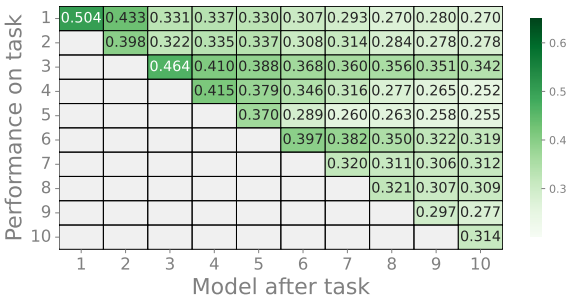


Figure 24: Heatmap of CSD alignment scores for each task in continual style personalization.