FORGETTING OF TASK-SPECIFIC KNOWLEDGE IN MODEL MERGING-BASED CONTINUAL LEARNING

Timm Hess, Gido M van de Ven, Tinne Tuytelaars
Department of Electrical Engineering (ESAT), KU Leuven, Belgium
{timmfelix.hess, gido.vandeven, tinne.tuytelaars}@esat.kuleuven.be

ABSTRACT

This paper investigates the linear merging of models in the context of continual learning (CL). Using controlled visual cues in computer vision experiments, we demonstrate that merging largely preserves or enhances shared knowledge, while unshared task-specific knowledge rapidly degrades. We further find that merging models from an incremental training process consistently outperforms merging models trained in parallel.

1 INTRODUCTION

Methods that involve averaging the parameters of different models, often termed weight-space ensembling or model merging, have received significant attention in deep learning (Yang et al., 2024). A common application targets improving performance for a single task (Izmailov et al., 2018; Wortsman et al., 2022b). When multiple (well-performing) model variants are trained starting from a common initialization – for instance, through multiple fine-tuning runs with different random seeds or different hyperparameter settings – averaging their weights tends to yield a final model with improved accuracy and robustness over any individual model, without increasing inference cost. This benefit is generally attributed to the geometry of the loss landscape, as models fine-tuned from the same initialization have been found to often reside in the same wide, and relatively flat basin (Goodfellow et al., 2015; Frankle & Carbin, 2019). A common interpretation is that averaging the weights of such models produces a solution closer to the center of this basin, smoothing out noise or over-specifications learned by individual training runs, leading to better generalization (Izmailov et al., 2018).

Beyond improving performance for single tasks, model merging has also been explored for integrating knowledge from multiple tasks into a single model (Ilharco et al., 2022; Matena & Raffel, 2022). This renders model merging a potential mechanism for continual learning (CL), where the goal is to learn multiple tasks sequentially without catastrophically forgetting earlier ones (McCloskey & Cohen, 1989; Parisi et al., 2019). With CL in mind, two approaches for merging models that are trained or adapted for different tasks can be distinguished. One option is to merge different states of an incrementally trained model, by averaging its weights after learning one task with its subsequent weights after learning a later task. Another option is to merge separate models trained in parallel on different tasks. Some studies have already shown promising results for model merging in specific CL contexts (Marouf et al., 2024; Udandarao et al., 2024; Kozal et al., 2024). Notably, the merging mechanism differs fundamentally from the typical approach to CL, which is to make changes to the loss function to sequentially approximate a joint objective over all observed tasks (Hess et al., 2023). Instead, weight-space merging is typically applied post-hoc, combining independently or sequentially trained models that have been specialized for individual tasks. The general conditions under which such post-hoc merging preserves or degrades knowledge across diverse tasks remain poorly understood. To better understand when merging is suitable for CL, we conceptualize the knowledge of a model trained on multiple tasks as comprising both *shared* components (e.g., general features from pre-training, or knowledge common across multiple tasks) and *task-specific* components (e.g., features or decision boundaries unique to a single task). The central question of our study then is: *how do these shared and task-specific knowledge components react during weight merging?*

In this work, we conduct controlled experiments targeting the computer vision domain. We propose a methodology that allows to instantiate either shared or task-specific knowledge via synthetic visual cues injected directly into the input image space. We empirically show that during linear weight interpolation, shared knowledge tends to be largely preserved or even enhanced, while unshared task-specific knowledge is significantly degraded. These results align with recent research by Zaman et al. (2024), who conduct related experiments with large language models. We further compare merging incrementally trained models with merging parallel trained ones. Our findings provide insight into the suitability and limitations of weight-space ensembling as a mechanism within various CL scenarios, potentially informing the design of more effective strategies for knowledge accumulation and retention.

2 BACKGROUND

Continual learning. An important goal of continual learning (CL) is enabling models to learn sequentially from a stream of tasks, without catastrophically forgetting previously acquired knowledge (De Lange et al., 2022; Wang et al., 2024; van de Ven et al., 2025). The dominant approach in CL involves incrementally training a single, evolving model by attempting to constantly approximate a joint learning objective across all encountered tasks (Hess et al., 2023). In contrast, model merging as a post-hoc method, offers a distinct approach by combining separately adapted model states, rather than modifying the sequential training process itself.

Shortcuts. To empirically study the interaction of different knowledge types during merging, we draw inspiration from research on shortcut learning. 'Shortcuts' refer to spurious or overly simple correlations in data, which the models exploit rather than learning more generalizable features (Geirhos et al., 2020). By intentionally injecting synthetic visual cues that are correlated with class labels but distinct from core image content, scenarios can be created where models learn to rely on these shortcuts.

For a more comprehensive review of related literature, we refer the reader to Appendix B.

3 METHODOLOGY

Our methodology is designed to investigate the fate of shared (common) and unshared (task-specific) knowledge during model merging in computer vision settings. To create controllable and distinct knowledge components in our models, we augment a base image dataset by superimposing synthetic visual cues onto the input images. Examples are shown in Figure 1a. A combination of the base dataset with a specific visual cue constitutes a distinct 'task' for the model to learn. Interpolating between endpoint models that learned tasks with either different or the same visual cues governs our investigation of task-specific and shared knowledge retention. For all our experiments, CIFAR-100 (Krizhevsky et al., 2009) is used as the base dataset. The full implementation details are presented in Appendix A.

Shared pre-trained initialization. All model trainings begin from a common base model, θ_{PT}, which has been pre-trained on the base dataset. The primary purpose of pre-training is to ensure that all endpoint models share a significant initial learning trajectory, a condition motivated by linear mode connectivity research, which suggests it facilitates meaningful weight-space merging without requiring complex re-parameterization techniques (Goodfellow et al., 2015; Frankle & Carbin, 2019; Ainsworth et al., 2023). In addition, the pre-training on the base dataset (without cues) yields a broad set of general features, which allows us to evaluate the preservation of this foundational 'shared knowledge' as an independent measure alongside our visual cue-specific evaluations, which we detail next.

Visual cues. We define a visual cue as an $N \times N$ pixel patch superimposed onto an image at a specific location, providing a simple visual pattern that is consistently correlated with the sample's ground-truth class label, irrespective of the underlying image content. To create two different types of shortcuts, we define two families of cues: (1) Colored patches, consisting of pixel patches of solid color (the *color cue*, denoted as $\mathcal{C}_{\text{color}}$). We utilize the HSV color space, setting Saturation (S) and Value (V) to 1.0, and distributing Hue (H) evenly across classes. (2) Grayscale noise patches, consisting of pixel patches containing grayscale noise patterns (the *noise cue*, $\mathcal{C}_{\text{noise}}$). For each class, a unique pixel noise pattern is generated by sampling each pixel's intensity independently and uniformly from the discrete set $\{0, 255\}$. For both families of cues, we use a patch size of $N = 5$. During training, a cue is superimposed on each image with a probability $p_{\mathcal{C}} = 0.5$ to incentivize models to learn both the cue and the general image features.

Shared vs. task-specific knowledge. We use the visual cues to construct two knowledge protocols that are at the core of our experimental setups. (1) The *unshared, task-specific knowledge protocol* consists of two tasks constructed using distinct visual cues: in one task the color cue is added to the base dataset (denoted as T_{color}), and in the other task the noise cue is added (T_{noise}). To minimize information transfer, the cues are placed at non-overlapping positions (top-left and bottom-right). (2) The *shared knowledge protocol* consists of two tasks with the same visual cue: in both tasks, the color cue is added to the base dataset at the same position (top-left). This controlled use of cues allows for a more explicit separation of shared versus unshared specific knowledge compared to typical CL benchmarks based on semantic splits, where disentangling preserved from re-discovered knowledge can be challenging (Hess et al., 2024).

Incremental vs. parallel training. Starting from the pre-trained weights θ_{PT}, we train models on one of the knowledge protocols to generate *endpoint models* (i.e., models that are trained or adapted for a task with a particular visual cue) for our merging analysis. As illustrated in Figure 1b, we compare two distinct training scenarios. With incremental training, which represents continual learning without specific forgetting mitigating, factors like catastrophic forgetting or knowledge transfer between the sequential training stages can influence the characteristics of the endpoint models. In contrast, parallel training serves as a controlled baseline where both endpoint models are trained independently

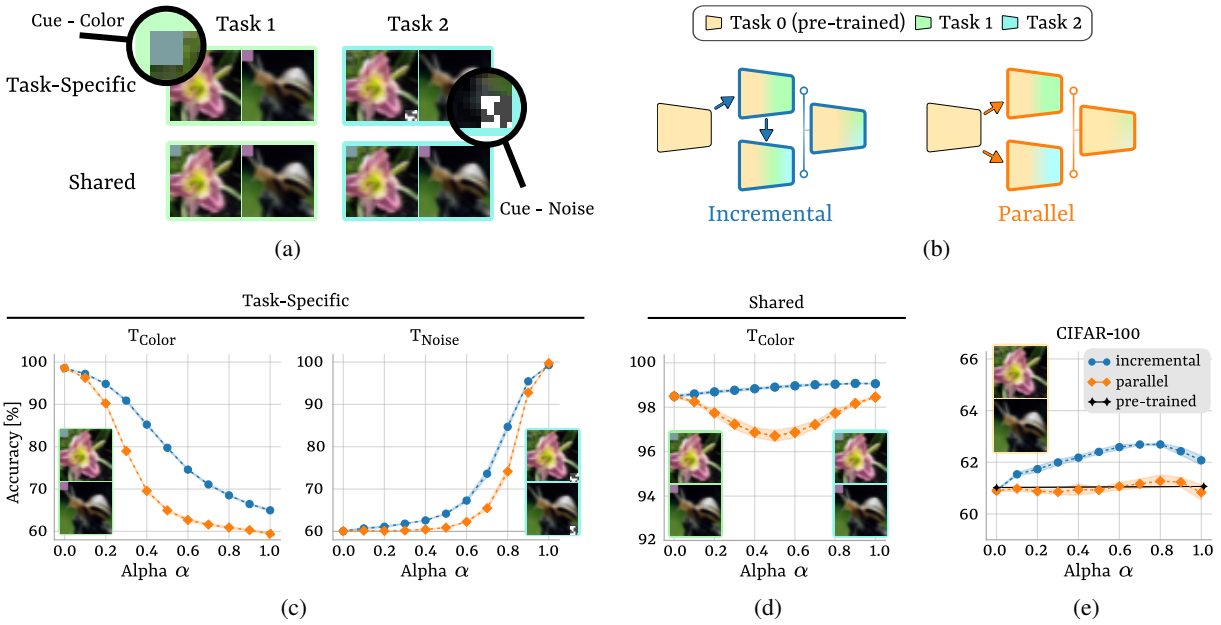

Figure 1: **Experimental Protocol and Main Results. (a)** Example of shared and task-specific knowledge instantiated with visual cues. The shared knowledge protocol uses the same cue for both tasks, while the task-specific knowledge protocol uses distinct cues. **(b)** Schematic illustration of 'incremental' (blue) and 'parallel' (orange) training. Both scenarios start from a common pre-trained model (Task 0, yellow) and adapt models for subsequent tasks (Task 1, green; Task 2, turquoise) that involve specific visual cues. **(c)** Accuracy (y-axis) vs. interpolation coefficient α (x-axis) for the task-specific knowledge protocol. Performance is evaluated in the presence of the color (left) and noise (right) visual cues, comparing incremental (blue circles) and parallel (orange diamonds) training. The endpoint models of the interpolation are specialized for T_{Color} ($\alpha = 0$) and T_{Noise} ($\alpha = 1$). **(d)** Accuracy vs. α for the shared knowledge protocol, where both endpoint models of the interpolation are specialized for T_{Color}. **(e)** Performance on the base dataset (no cues) when interpolating T_{Color} ($\alpha = 0$) and T_{Noise} ($\alpha = 1$) endpoints; the solid black line marks accuracy of the pre-trained model. The plotted lines in the bottom three panels represent the mean accuracy across three independent runs with different random seeds, while the shaded areas indicate the standard error. All evaluations are on the full CIFAR-100 test set. ***These results show that while unshared task-specific knowledge degrades rapidly when models are merged (panel c), shared knowledge components are largely preserved or even enhanced (panel d,e). Moreover, merging incrementally trained models consistently leads to better knowledge preservation compared to merging models trained in parallel.***

from the identical pre-trained state θ_{PT}. This setup allows for a direct analysis of merging effects with and without sequential dependencies.

Evaluation protocol. We evaluate all models (pre-trained, endpoints, and interpolated) using classification accuracy on the held-out test set. To probe shared and task-specific knowledge induced by visual cues, we evaluate performance on the test set with either the color cue ($\mathcal{C}_{\text{color}}$) or the noise cue ($\mathcal{C}_{\text{noise}}$) deterministically ($p_C = 1.0$) applied. To measure the preservation of general shared knowledge, we evaluate performance on the original test set without any visual cues ($p_C = 0.0$) applied. Interpolated models $\theta(\alpha) = \alpha\theta_{T_1} + (1-\alpha)\theta_{T_2}$ are generated via linear interpolation of appropriate task-specific endpoints, with $\alpha \in [0, 1]$ the interpolation coefficient.

4 Results

Here we present the results of the merging experiments, with the endpoint models generated via either parallel or incremental training and according to either the shared or task-specific knowledge protocol. Key trends are summarized in Figure 1, with full numerical details presented in Table 1 in Appendix C.

Endpoint models successfully learn specific cues while retaining general knowledge. Before investigating model interpolation, we confirm that the endpoint models behave as expected. Models adapted to a specific visual cue demonstrate high performance when tested on that cue, with cue-specific accuracies consistently above 98.5% (Ta-

ble 1), while performance on the shared general knowledge (CIFAR-100 without cues) remains close to the pre-trained baseline of 61.3%. An exception is the endpoint model incrementally trained on both cues, whose performance on CIFAR-100 without cues surpasses that of the pre-trained baseline, indicating positive transfer from the sequential training process.

Unshared specific knowledge degrades while shared knowledge is preserved or enhanced. A central finding of our work is the starkly different effect of linear interpolation on different knowledge types. As shown in Figure 1c, unshared task-specific knowledge rapidly degrades, i.e. sensitivity to a specific cue decays sharply as the interpolation moves towards the other endpoint, a transition that is particularly abrupt in the parallel setting. Conversely, *shared knowledge* is robust to interpolation. As shown in Figure 1d, when merging models that were trained on tasks with the same cue (T_{Color}), performance is maintained well across the interpolation path. Furthermore, as shown in Figure 1e, performance on the common CIFAR-100 task remains stable (in the parallel scenario) or is even enhanced (in the incremental scenario), peaking at an accuracy of 62.69%, which is above both endpoints.

Merging incrementally trained models preserves knowledge better than merging parallel-trained ones. We also find a key distinction between training scenarios, as merging models from an incremental training process consistently outperforms merging models trained in parallel (Figure 1c to 1e). In particular, when merging two models that are adapted for the same specific cue (T_{Color}), the incremental scenario maintains high performance across the entire interpolation path, whereas the parallel scenario exhibits a slight concave dip in accuracy around the midpoint (Figure 1d). Furthermore, for general shared knowledge (CIFAR-100 without cues, Figure 1e), the incremental scenario produces a notable synergistic effect: accuracy rises to a peak of 62.69%, surpassing both endpoints and the original pre-trained model. In contrast, the parallel scenario's performance remains largely flat, showing no significant benefit from interpolation. This finding, supported by related work of (Marouf et al., 2024), is particularly relevant for CL, as it indicates that merging states from a single, evolving model can be more advantageous than merging independently trained specialists.

To assess the generality of these observations, in Appendix D we conduct additional experiments under varied conditions. We analyze the reverse order of cue adaptations (i.e., first T_{Noise}, then T_{Color}), and we explore a 'chunking'-setup (Lee & Storkey, 2025), where the pre-trained model's feature base is weaker and parts of the previously shared common knowledge of CIFAR-100 are divided over the different tasks. The results for both configurations corroborate the distinct behaviors of shared and task-specific knowledge during interpolation that we observe in our main experiments.

5 Discussion

Our experiments demonstrate that linear model merging distinctly impacts different knowledge types. On one hand, shared knowledge is largely preserved and can be enhanced, aligning with earlier findings of consolidating commonalities within shared loss basins (Izmailov et al., 2018). In stark contrast, unshared task-specific knowledge rapidly degrades upon interpolation due to interference between divergent parameter adaptations, a finding consistent with prior work on large language models (Zaman et al., 2024). Another important finding of our work is that merging incrementally trained models yields better knowledge consolidation than merging parallel-trained ones, suggesting that the process of sequential adaptation, even without explicit CL mechanisms, guides models along trajectories that are more amenable to beneficial merging. However, also when incrementally trained models are merged, unshared task-specific knowledge is still rapidly degraded. For CL, this implies that while naive merging might strengthen the generality of shared features, it risks catastrophic forgetting of unique past task knowledge. This raises critical questions about the 'utility' of what is preserved versus forgotten, an issue orthogonal to the stability-plasticity dilemma. Therefore, the suitability of simple merging for CL appears limited to scenarios prioritizing general shared capabilities or involving highly similar tasks, unless model scale, endpoint training, or merging techniques are specifically tailored to mitigate these trade-offs. We provide an extended discussion in Appendix E.

6 Limitations and Future Work

Our study offers initial insights into how linear merging of models differentially affects shared and task-specific knowledge in the vision domain. While our controlled setting based on visual cues enabled clear distinctions, several avenues warrant further exploration to better understand the applicability of merging in continual learning. One is to investigate whether the observed dynamics of shared versus specific knowledge scale to larger, more diverse architectures (e.g., vision transformers) and more complex, real-world task sequences. While merging benefits are known for large models, the precise nature of knowledge interaction as identified here needs validation at scale. Another compelling direction for future work is to explore methods for achieving more explicit internal disentanglement of general versus task-specific knowledge within models during their (continuous) training.

ACKNOWLEDGMENTS

This work has been supported by funding from the European Union under the Horizon 2020 research and innovation program (ERC project KeepOnLearning, grant agreement No. 101021347) and by a senior postdoctoral fellowship from the Research Foundation – Flanders (FWO) under grant number 1266823N.

REFERENCES

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations*, 2023.

Florian Peter Busch, Roshni Kamath, Rupert Mitchell, Wolfgang Stammer, Kristian Kersting, and Martin Mundt. Where is the truth? the risk of getting confounded in a continual world. In *International Conference on Machine Learning*, 2025.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2022.

Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In *International Conference on Machine Learning*, pp. 1309–1318. PMLR, 2018.

Sebastian Dziadzio, Vishaal Udandarao, Karsten Roth, Ameya Prabhu, Zeynep Akata, Samuel Albanie, and Matthias Bethge. How to merge your multimodal models over time? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20479–20491, 2025.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of DNNs. *Advances in Neural Information Processing Systems*, 31, 2018.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Ian J. Goodfellow, Oriol Vinyals, and Andrew M. Saxe. Qualitatively characterizing neural network optimization problems. In *International Conference on Learning Representations*, 2015.

Timm Hess, Tinne Tuytelaars, and Gido M van de Ven. Two complementary perspectives to continual learning: Ask not only what to optimize, but also how. In *Proceedings of the 1st ContinualAI Unconference*, volume 249 of *Proceedings of Machine Learning Research*, pp. 37–61, 2023.

Timm Hess, Eli Verwimp, Gido M van de Ven, and Tinne Tuytelaars. Knowledge accumulation in continually learned representations and the issue of feature forgetting. *Transactions on Machine Learning Research*, 2024.

Gabriel Ilharco, Mitchell Wortsman, Samir Yitzhak Gadre, Shuran Song, Hannaneh Hajishirzi, Simon Kornblith, Ali Farhadi, and Ludwig Schmidt. Patching open-vocabulary models by interpolating weights. *Advances in Neural Information Processing Systems*, 35:29262–29277, 2022.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence*, 2018.

Jedrzej Kozal, Jan Wasilewski, Bartosz Krawczyk, and Michał Woźniak. Continual learning with weight interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 4187–4195, 2024.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Thomas L Lee and Amos Storkey. Chunking: Continual learning is not just about distribution shift. In *Proceedings of The 3rd Conference on Lifelong Learning Agents*, volume 274 of *Proceedings of Machine Learning Research*, pp. 915–937, 2025.

Weishi Li, Yong Peng, Miao Zhang, Liang Ding, Han Hu, and Li Shen. Deep model fusion: A survey. *arXiv preprint arXiv:2309.15698*, 2023.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in Neural Information Processing Systems*, 30, 2017.

Daniel Marczak, Bartłomiej Twardowski, Tomasz Trzciński, and Sebastian Cygert. MagMax: Leveraging model merging for seamless continual learning. In *European Conference on Computer Vision*, pp. 379–395, 2024.

Imad Eddine Marouf, Subhankar Roy, Enzo Tartaglione, and Stéphane Lathuilière. Weighted ensemble models are strong continual learners. In *European Conference on Computer Vision*, pp. 306–324. Springer, 2024.

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

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Dilan Gorur, Razvan Pascanu, and Hassan Ghasemzadeh. Linear mode connectivity in multitask and continual learning. In *International Conference on Learning Representations*, 2021.

Behnam Neyshabur, Hanie Sedghi, and Chiyuan Zhang. What is being transferred in transfer learning? *Advances in Neural Information Processing Systems*, 33:512–523, 2020.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. *Advances in Neural Information Processing Systems*, 36:66727–66754, 2023.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*, 2022.

Vishaal Udandarao, Karsten Roth, Sebastian Dziadzio, Ameya Prabhu, Mehdi Cherti, Oriol Vinyals, Olivier Henaff, Samuel Albanie, Zeynep Akata, and Matthias Bethge. A practitioner's guide to real-world continual multimodal pretraining. *Advances in Neural Information Processing Systems*, 37:133801–133845, 2024.

Gido M van de Ven, Nicholas Soures, and Dhireesha Kudithipudi. Continual learning and catastrophic forgetting. In John Wixted (ed.), *Learning and Memory: A Comprehensive Reference (Third Edition)*, volume 1, pp. 153–168. Academic Press, Oxford, 2025.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5362–5383, 2024.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pp. 23965–23998. PMLR, 2022a.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7959–7971, 2022b.

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*, 36:7093–7115, 2023.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in LLMs, MLLMs, and beyond: Methods, theories, applications and opportunities. *arXiv preprint arXiv:2408.07666*, 2024.

Kerem Zaman, Leshem Choshen, and Shashank Srivastava. Fuse to forget: Bias reduction and selective memorization through model fusion. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 18763–18783, Miami, Florida, USA, 2024. Association for Computational Linguistics.

APPENDIX

The following Appendix provides supplementary material to accompany the main paper. It offers:

A Further details on our experimental setup

B An extended discussion of related literature

C Comprehensive numerical results for the primary experiments

D Additional experimental validations under varied conditions

E An extended discussion of our results

A EXPERIMENTAL SETUP DETAILS

This section provides further details on the experimental setup and hyperparameters used throughout all training phases (pre-training and subsequent cue adaptations) described in Section 3 and Appendix D, unless otherwise specified.

Model Architecture and Optimization. The core architecture is a slim ResNet-18 model (Lopez-Paz & Ranzato, 2017). In this model, batch normalization layers were replaced with group normalization layers using a single group to approximate layer normalization, thereby avoiding potential confounding effects from batch normalization's running statistics parameters (Kozal et al., 2024). All training stages (pre-training and subsequent cue adaptations) utilize a cross-entropy loss function and the same optimization settings. Optimization is performed using Stochastic Gradient Descent (SGD) with a momentum of 0.9 and weight decay of 5×10^{-4}. Each training stage proceeds for 50 epochs. A linear learning rate warm-up is applied during the first 5% of these epochs, increasing the learning rate from 0.004 to 0.1. Subsequently, a cosine annealing learning rate schedule decays the learning rate from 0.1 down to a minimum of 1×10^{-5} over the remaining epochs. The mini-batch size is 128. All cue adaptation experiments are repeated three times with distinct random seeds, and reported as mean and standard error. Unless explicitly specified, pre-training is not subjected to different random seeds, but the same pre-trained weights are used as initialization for all runs.

Data Augmentation and Visual Cue Application. Standard data augmentations for CIFAR-100 are employed during all training stages: random horizontal flips (with a probability of 0.5) and random crops (image size 32×32 pixels, with padding of four pixels). Visual cues, when applied during the cue adaptation stages, are superimposed onto the images *after* these standard augmentations. For the visual cues (both $\mathcal{C}_{\text{color}}$ and $\mathcal{C}_{\text{noise}}$), we use a patch size of 5×5 pixels. When distinct cues are used to instantiate unshared task-specific knowledge (e.g., T_{Color} vs. T_{Noise} experiments), they are placed at fixed, non-overlapping, and distinct positions: the color patch in the top-left corner and the noise patch in the bottom-right corner of the image.

The code to reproduce our experiments is available at: https://github.com/TimmHess/MergeForget.

B RELATED WORK

Model Merging and Continual Learning. Combining the parameters of multiple neural networks, often referred to as model merging or weight-space ensembling, has gained significant interest (Li et al., 2023; Yang et al., 2024). In single-task settings, averaging model weights, particularly those fine-tuned from a common pre-trained initialization, often improves performance and robustness ('model soups' (Wortsman et al., 2022a)), potentially finding solutions superior to any individual model (Izmailov et al., 2018; Wortsman et al., 2022b). Techniques vary from simple averaging (Wortsman et al., 2022a) and interpolation (Ilharco et al., 2022) to more sophisticated methods like task arithmetic (Ilharco et al., 2023; Ortiz-Jimenez et al., 2023), Fisher-weighted averaging (Matena & Raffel, 2022), and interference resolution strategies (Yadav et al., 2023; Marczak et al., 2024). The potential of merging extends to continual learning (CL), where models must learn sequentially without catastrophic forgetting (McCloskey & Cohen, 1989; Parisi et al., 2019). Commonly, CL methods utilize regularization, replay, architectural, or a combination of these mechanisms, and focus on preserving performance on past tasks during sequential training (De Lange et al., 2022; Wang et al., 2024). Model merging offers a different perspective, as it involves the post-hoc combination of models trained independently or sequentially. Indeed, simple averaging has shown promise in the CL 'chunking' setting (Lee & Storkey, 2025), was combined with replay (Marouf et al., 2024), and has been explored in combination with other techniques for continuous adaptation, sometimes incorporating Fisher information or exponential moving average-like updates sequentially (Udandarao et al., 2024; Dziadzio et al., 2025). However, merging models adapted to different tasks introduces interference (Yadav et al., 2023; Marczak et al., 2024), a key challenge shared with the stability-plasticity dilemma inherent to CL. Building on work distinguishing shared and task-specific knowledge (Zaman et al.,

2024), which found that merging can be used to selectively forget task-specific information in large language models (LLMs), our work investigates these dynamics in the vision domain to improve our understanding of the suitability of model merging for CL.

Loss Landscapes and Mode Connectivity. The effectiveness of model merging, particularly linear interpolation of weights, is closely tied to the concept of mode connectivity (Garipov et al., 2018; Draxler et al., 2018). Research suggests that minima found via fine-tuning from a common pre-trained initialization often lie within the same loss basin and can be connected by linear paths of low loss (Frankle & Carbin, 2019; Neyshabur et al., 2020). This provides a geometric explanation for why averaging fine-tuned models can be successful (Wortsman et al., 2022a). The situation is less clear when merging models that were adapted to different tasks or data distributions. In that case interference might occur (Yadav et al., 2023; Mirzadeh et al., 2021). Our work uses interpolation across models trained with different task-specific cues to implicitly probe these geometric properties. It reveals the nuance that depending on whether knowledge is shared between models or not, merging either preserves that knowledge or leads to interference. This finding aligns with the intuition that incompatible representations (likely residing in different basins or requiring non-linear paths) are harder to merge effectively via simple averaging.

Merging Shared and Task-Specific Knowledge – Fuse-to-Forget. While model merging is often explored for its potential to aggregate capabilities in a model, the work of Zaman et al. (2024) investigated the inverse potential: using merging as a mechanism to selectively *forget* specific, potentially undesirable knowledge components. They investigated this using LLMs. Motivated by applications such as reducing societal biases learned during pre-training or mitigating privacy risks by erasing memorized training examples, the authors postulate that merging might preferentially discard non-shared information. To analyze this phenomenon systematically, they proposed a framework that distinguishes between *shared knowledge* (common among the models being merged) and *task-specific knowledge* (unique to individual models). Using this framework, Zaman et al. (2024) evaluated the effect of merging LLMs fine-tuned with specific, targeted interventions, namely associating disconnected token pairs. In their work, this property of forgetting knowledge specific to a particular model is framed as *desirable*, e.g. for removing unwanted biases. However, generally speaking, we argue that this property is ambivalent, as forgetting specific task capabilities might be undesirable in other contexts. Our work adapts the concept of shared and task-specific knowledge to the vision domain using a different method to instantiate task-specific knowledge.

Cues, Shortcuts, Confounders. To create distinct task-specific knowledge components in a controlled manner in the vision domain, we draw inspiration from the literature on shortcut learning (Geirhos et al., 2020). This research area studies how deep neural network models are prone to exploiting spurious correlations instead of learning robust features. By injecting synthetic visual cues that are easy to learn and correlated with the class label, but distinct from the core image content, we encourage models to specialize on these particular 'shortcuts'. This setup allows for a cleaner separation between shared (same shortcut in each task) and task-specific (different shortcut in each task) knowledge compared to using standard datasets with semantic splits, where the nature of shared vs. specific features can be ambiguous (Hess et al., 2024). It provides a visual analog to the token-pair association used by Zaman et al. (2024) and allows us to study how merging interacts with models reliant on different, easily controlled, task-specific strategies. This controlled approach is also related to Busch et al. (2025), who study the impact of confounders to CL, but we focus specifically on the interaction of shared and task-specific knowledge when using merging-like approaches.

C Full Numerical Results

This section presents the comprehensive numerical data of the main experiments discussed in Section 4 and visualized in Figure 1. Table 1 details the mean classification accuracies and standard errors (across three runs) for all evaluated conditions: endpoint models ($\alpha = 0$ and $\alpha = 1$) and interpolated models ($\alpha \in (0, 1)$), trained in either the task-specific knowledge protocol ($T_{\text{Color}} \rightarrow T_{\text{Noise}}$) or the shared knowledge protocol ($T_{\text{Color}} \rightarrow T_{\text{Color}}$), for both parallel and incremental training scenarios, and evaluated with or without the visual cues used during training.

D Additional Experiments

This section describes experiments designed to further probe and validate the observations presented in the main paper. These include experiments utilizing a 'chunked' training approach, and an analysis with the reversed cue adaptation order.

Table 1: **Numerical Results of the Experiments in the Main Text.** Displayed for each experiment is the classification accuracy (in %) on the test set, reported as the mean ± standard error across three runs with different random seeds.

			Interpolation Coefficient α										
Training	Evaluation	Scenario	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$T_{\text{Color}} \rightarrow T_{\text{Noise}}$	T_{Color}	*Incr.*	98.58 ± 0.05	97.22 ± 0.07	94.85 ± 0.13	90.89 ± 0.29	85.22 ± 0.29	79.74 ± 0.22	74.60 ± 0.20	71.08 ± 0.20	68.50 ± 0.19	66.46 ± 0.15	64.97 ± 0.24
		Para.	98.57 ± 0.05	96.28 ± 0.10	90.25 ± 0.37	78.96 ± 0.15	69.56 ± 0.23	64.96 ± 0.27	62.69 ± 0.25	61.61 ± 0.22	60.92 ± 0.20	60.29 ± 0.19	59.43 ± 0.26
	T_{Noise}	*Incr.*	60.05 ± 0.17	60.70 ± 0.15	61.11 ± 0.20	61.83 ± 0.10	62.59 ± 0.06	64.19 ± 0.11	67.27 ± 0.36	73.63 ± 0.75	84.70 ± 0.62	95.45 ± 0.26	99.26 ± 0.05
		Para.	60.05 ± 0.16	60.16 ± 0.18	60.12 ± 0.08	60.20 ± 0.03	60.45 ± 0.09	60.88 ± 0.03	62.27 ± 0.15	65.46 ± 0.40	74.15 ± 0.74	92.78 ± 0.21	99.72 ± 0.03
	No Cue	*Incr.*	60.91 ± 0.02	61.53 ± 0.06	61.73 ± 0.12	61.99 ± 0.11	62.18 ± 0.09	62.40 ± 0.09	62.59 ± 0.10	62.69 ± 0.02	62.69 ± 0.05	62.43 ± 0.15	62.08 ± 0.18
		Para.	60.90 ± 0.03	60.98 ± 0.10	60.87 ± 0.05	60.85 ± 0.10	60.94 ± 0.22	60.93 ± 0.09	61.07 ± 0.14	61.17 ± 0.17	61.27 ± 0.24	61.23 ± 0.18	60.83 ± 0.29
$T_{\text{Color}} \rightarrow T_{\text{Color}}$	T_{Color}	*Incr.*	98.50 ± 0.06	98.59 ± 0.04	98.70 ± 0.05	98.76 ± 0.05	98.84 ± 0.05	98.91 ± 0.05	98.96 ± 0.05	99.01 ± 0.03	99.03 ± 0.02	99.07 ± 0.02	99.06 ± 0.01
		Para.	98.49 ± 0.06	98.25 ± 0.05	97.74 ± 0.12	97.24 ± 0.23	96.87 ± 0.22	96.71 ± 0.25	96.87 ± 0.24	97.22 ± 0.16	97.73 ± 0.14	98.17 ± 0.06	98.45 ± 0.04

D.1 Model Merging in the Chunking-Setup

To further investigate the fate of shared and task-specific knowledge under model merging, we investigate a condition where the pre-trained feature base is weaker and where parts of the previously shared and pre-trained knowledge are divided over the different tasks.

Experimental Setup. Following Lee & Storkey (2025), we divide the CIFAR-100 training dataset randomly into three chunks of approximately equal size. We checked that each chunk contains examples from all classes. The pre-training phase now consists of training only on the data of chunk 1. As before, this model, denoted $\theta_{\text{PT-chunk}}$, serves as the common starting point for subsequent adaptations.

The methodology is analogous to our experiments in the main text. We adapt models using visual cues, employing both parallel and incremental scenarios originating from $\theta_{\text{PT-chunk}}$. For parallel adaptation, one model branch is trained on chunk 2 augmented with the $\mathcal{C}_{\text{color}}$ cue, and another independent branch is trained on chunk 3 augmented with the $\mathcal{C}_{\text{noise}}$ cue. For incremental adaptation, the model is first trained on chunk 2 with $\mathcal{C}_{\text{color}}$, and then training continues using chunk 3 augmented with $\mathcal{C}_{\text{noise}}$. The visual cues, training objectives, probabilistic cue application ($p_{\mathcal{C}} = 0.5$), and optimization hyperparameters remain the same (see Appendix A). Note that in contrast to the experiments in the main text, here, the pre-training is subject to the random seeds of the individual runs.

Finally, linear interpolation is performed between the endpoint models obtained from the cue-specific adaptation phase for both parallel and incremental scenarios. All models, and interpolated models are evaluated on the full CIFAR-100 test set, using the three conditions defined in Section 3: color cue specific knowledge, noise cue specific knowledge, and shared general knowledge (CIFAR-100 with no cues).

Table 2: **Numerical Results for the "Chunking" Setup.** Displayed for each experiment is the test accuracy (in %), reported as the mean ± standard error across three runs with different random seeds.

			Interpolation Coefficient α										
Training	Evaluation	Scenario	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$T_{\text{Color}} \rightarrow T_{\text{Noise}}$	T_{Color} Cue	*Incr.*	90.62 ± 0.38	89.43 ± 0.40	86.98 ± 0.36	83.55 ± 0.40	79.41 ± 0.56	75.00 ± 0.58	70.64 ± 0.70	66.29 ± 0.78	63.63 ± 0.73	55.76 ± 0.71	53.09 ± 1.04
		Para.	90.95 ± 0.40	87.93 ± 0.30	81.71 ± 0.22	72.95 ± 0.55	62.89 ± 0.67	54.67 ± 0.55	48.58 ± 0.49	46.12 ± 0.44	43.71 ± 0.41	41.35 ± 0.41	39.42 ± 0.61
	T_{Noise} Cue	*Incr.*	41.62 ± 0.63	43.08 ± 0.61	44.49 ± 0.64	45.69 ± 0.60	46.92 ± 0.61	48.60 ± 0.61	52.76 ± 0.62	60.16 ± 0.64	75.12 ± 0.94	92.76 ± 0.23	99.12 ± 0.05
		Para.	41.62 ± 0.63	42.48 ± 0.61	42.82 ± 0.64	43.20 ± 0.60	43.60 ± 0.61	45.25 ± 0.61	49.08 ± 0.62	56.75 ± 0.64	74.12 ± 0.94	92.76 ± 0.23	99.12 ± 0.05
	CIFAR-100	*Incr.*	42.29 ± 0.61	43.60 ± 0.52	44.49 ± 0.48	45.34 ± 0.47	46.12 ± 0.46	46.53 ± 0.68	46.98 ± 0.61	47.02 ± 0.48	46.87 ± 0.49	46.15 ± 0.55	45.64 ± 0.59
		Para.	42.97 ± 0.56	43.18 ± 0.50	43.23 ± 0.52	43.29 ± 0.47	43.33 ± 0.48	43.15 ± 0.32	43.59 ± 0.31	43.66 ± 0.31	43.59 ± 0.30	43.02 ± 0.30	41.62 ± 0.40
$T_{\text{Color}} \rightarrow T_{\text{Color}}$	T_{Color} Cue	*Incr.*	90.62 ± 0.38	91.39 ± 0.39	91.93 ± 0.35	92.53 ± 0.40	93.07 ± 0.40	93.60 ± 0.40	93.94 ± 0.40	94.26 ± 0.40	94.56 ± 0.40	94.72 ± 0.40	94.73 ± 0.40
		Para.	90.81 ± 0.27	90.87 ± 0.28	90.83 ± 0.22	90.66 ± 0.17	90.39 ± 0.16	90.24 ± 0.15	90.36 ± 0.17	90.48 ± 0.17	90.68 ± 0.16	90.82 ± 0.13	90.79 ± 0.04

Results and Observations. The interpolation results for this chunking experiment are presented in Figure 2. The overall trends we observe are largely consistent with our main findings, though with some nuances reflecting the modified pre-training.

Endpoint Performance: The $\theta_{\text{PT-chunk}}$ model (black diamond in the CIFAR-100 panel of Figure 2) establishes the baseline shared knowledge accuracy after seeing only chunk 1. Models subsequently adapted to specific cues on new data chunks (chunks 2 and 3) achieve high accuracy when tested with their respective cues on the full test set, indicating successful learning of these specific visual features. Performance of these endpoints on the general CIFAR-100 task (no cues) reflects both the retained knowledge from chunk 1 and any generalizable features learned from chunks 2 and 3.

Unshared Task-Specific Knowledge: Sensitivity to the distinct T_{Color} and T_{Noise} cues shows rapid degradation during interpolation between the two cue-specialized models, for both parallel and incremental scenarios. This aligns with the main experiments, suggesting that unshared specific knowledge is poorly preserved by merging.

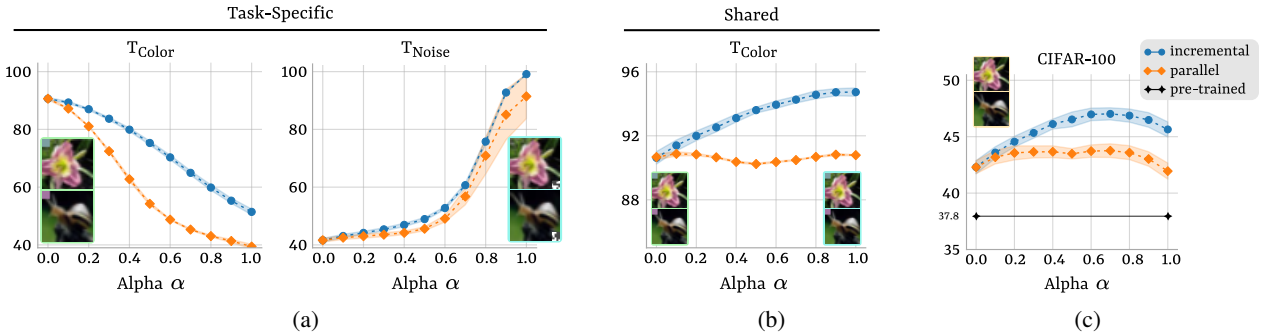

Figure 2: **Weight-Interpolation Results in the CIFAR-100 'Chunking' Setup. (a)** Accuracy (y-axis) vs. interpolation coefficient α (x-axis) for the task-specific knowledge protocol in the chunking setup. Performance is evaluated on the color (left) and noise (right) visual cues, comparing incremental (blue circles) and parallel (orange diamonds) training. The endpoint models are specialized on chunk 2 with T_{Color} ($\alpha = 0$) and chunk 3 with T_{Noise} ($\alpha = 1$), both originating from a base model pre-trained on chunk 1. **(b)** Accuracy vs. α for the shared knowledge protocol, where the endpoint models of the interpolation are specialized for chunk 2 with T_{Color} ($\alpha = 0$) and chunk 3 with T_{Color} ($\alpha = 1$). **(c)** Performance on the base CIFAR-100 dataset (no cues) when interpolating the same endpoints as in (a). The solid black diamond marks the accuracy of the model pre-trained only on chunk 1. In all panels, the plotted lines represent the mean accuracy across three independent runs with different random seeds, while the shaded areas indicate the standard error. All evaluations are on the full CIFAR-100 test set. ***These results corroborate our main findings: also when general knowledge is learned distributively across data chunks, merging leads to the rapid degradation of unshared task-specific knowledge (panel a) while preserving and enhancing the consolidated shared knowledge (panel b,c).***

Shared Task Knowledge: In the incremental scenario, sequentially training with the same color cue yields forward transfer, strengthening the model's ability to use the cue in the context of new data. Consequently, a monotonic improvement in performance along the interpolation path (Figure 2b) is observed. Here, merging does not benefit, but visibly reverts the accumulation of knowledge with respect to the visual cue. Results for the parallel training scenario, without any forward transfer, show the same trends as our experiment in the main text. A slight concave dip in accuracy at the merging midpoint suggests that specializing on the same cue in the context of different data leads to parametrically distinct solutions that are not perfectly compatible under linear averaging.

Shared General Knowledge (CIFAR-100): Performance on the full CIFAR-100 test set (no cues) during interpolation shows improvements for both endpoints, and further substantial gains from interpolation. It contrasts the above results for shared specific task knowledge and re-emphasizes a key finding of the main: while merging may revert the learning of a *specific* shared skill in a sequential setting, it can simultaneously consolidate and improve the more distributed, *general* shared knowledge acquired across the same sequence.

This experiment and its findings reinforce our main observations. Also when the general features relevant to the CIFAR-100 task are learned in a distributed manner across different data chunks (each potentially paired with a specific cue during adaptation), linear model merging demonstrates a capacity to consolidate and enhance this broadly applicable knowledge. Concurrently, it continues to struggle with reconciling unshared, task-specific cue knowledge introduced within individual chunks. The consistency of these shared-versus-specific dynamics, whether general knowledge is established via comprehensive pre-training or learned incrementally across data chunks, underscores their robustness.

D.2 Interpolation Results – Reverse Order of Cue Adaptation

To further assess the robustness of our main findings regarding the interaction of shared and task-specific knowledge, we conducted an additional set of experiments where the order of cue adaptation in the incremental scenario was reversed compared to that primarily presented in Section 3.

Experimental Setup Modification. The core methodology for pre-training, visual cue design (C_{color} and C_{noise}), optimization, and evaluation remains identical to our main experiments (details in Appendix A). The key difference lies in the reversed order of the cues in the task-specific adaptation phase.

Table 3: **Numerical Results for the Reverse Cue Order Experiments.** Shown for each experiment is the classification accuracy (in %) on the test set, reported as mean ± standard error across three runs with different random seeds.

Training	Evaluation	Scenario	Alpha										
			0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
$T_{\text{Noise}} \rightarrow T_{\text{Color}}$	Noise	*Incr.*	99.76 ± 0.04	97.11 ± 0.33	87.44 ± 1.00	76.17 ± 0.99	69.57 ± 0.66	66.34 ± 0.37	64.68 ± 0.15	63.83 ± 0.04	63.36 ± 0.05	62.87 ± 0.18	62.39 ± 0.20
		Para.	99.76 ± 0.04	93.70 ± 0.23	75.86 ± 0.14	66.39 ± 0.17	62.76 ± 0.06	61.31 ± 0.04	60.71 ± 0.06	60.51 ± 0.06	60.51 ± 0.12	60.48 ± 0.09	60.29 ± 0.12
	Color	*Incr.*	59.71 ± 0.12	60.70 ± 0.13	61.84 ± 0.02	63.03 ± 0.17	65.05 ± 0.23	68.59 ± 0.24	74.77 ± 0.28	83.52 ± 0.22	90.32 ± 0.05	94.25 ± 0.14	96.21 ± 0.13
		Para.	59.71 ± 0.12	60.23 ± 0.10	60.60 ± 0.06	60.94 ± 0.04	61.73 ± 0.03	63.36 ± 0.08	67.26 ± 0.28	75.35 ± 0.49	87.22 ± 0.30	94.63 ± 0.26	97.59 ± 0.10
	CIFAR-100	*Incr.*	61.07 ± 0.16	61.66 ± 0.13	61.98 ± 0.13	62.38 ± 0.15	62.68 ± 0.18	62.75 ± 0.16	62.82 ± 0.16	62.99 ± 0.20	63.12 ± 0.17	62.98 ± 0.11	62.81 ± 0.13
		Para.	61.07 ± 0.16	61.35 ± 0.17	61.45 ± 0.20	61.43 ± 0.20	61.15 ± 0.13	61.02 ± 0.12	60.88 ± 0.08	60.94 ± 0.04	60.99 ± 0.06	61.14 ± 0.11	60.85 ± 0.13

Results and Observations. The interpolation results for this reverse order experiment are presented in Figure 3 and Table 3. The observed trends are qualitatively consistent with those reported in our main results (Section 4. Unshared task-specific knowledge (sensitivity to the initial T_{Noise} cue at $\alpha = 0$ or the final T_{Color} cue at $\alpha = 1$) rapidly degrades when interpolating towards the opposing endpoint. Performance on the shared CIFAR-100 task (no cues) again shows that the incremental adaptation scenario can lead to an enhancement of this knowledge around the midpoint of interpolation. The parallel scenario for shared knowledge remains relatively flat.

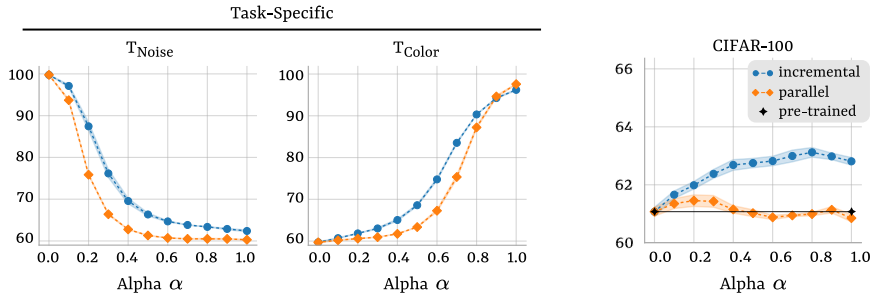

Figure 3: **Weight-Interpolation Results for Reverse Order Cue Adaptation.** Accuracy (y-axis) vs. interpolation coefficient α (x-axis). The $\alpha = 0$ endpoint is specialized on T_{Noise} and the $\alpha = 1$ endpoint on T_{Color}. Results compare 'incremental' (blue circles) and 'parallel' (orange diamonds) training. **(a)** The task-specific panels (T_{Noise}, T_{Color}) show the performance on the respective cue-specific test sets. **(b)** The shared CIFAR-100 panel shows the performance without cues. In both panels, the plotted lines represent the mean accuracy across three independent runs with different random seeds, while the shaded areas indicate the standard error. All evaluations are on the full CIFAR-100 test set.

E Extended Discussion

As discussed in Section 5 of the main text, our experiments investigating the linear merging of models highlight a distinct impact on shared versus task-specific knowledge, particularly when merging different checkpoints of an incrementally trained model. This extended discussion elaborates on these core findings, delving deeper into the interactions observed, the nuances between incremental and parallel adaptation scenarios, the broader implications for applying merging in CL, and considerations regarding when this approach might be most suitable. We begin by re-examining the different behavior of shared and specific knowledge components.

Fate of Shared and Task-Specific Knowledge. A consistent observation across our experiments is the divergent behavior of shared versus task-specific knowledge under linear interpolation. Shared knowledge, whether established during pre-training or by specific features like a common visual cue learned by both endpoint models, is largely preserved and can even be enhanced by interpolation (Figure 1d and 1e). This aligns with findings where merging is thought to consolidate commonalities by finding a more central solution within a shared loss basin, often facilitated by a common training history (Izmailov et al., 2018; Wortsman et al., 2022b). At the same time, the minor performance variations observed, particularly the slight dip in the parallel scenario, hints at a nuanced interaction. While functionally near-identical (both proficient at T_{Color}), their underlying parametric solutions for this relatively simple, cue-specific skill might still diverge. This observation links to early findings on linear mode connectivity (Frankle & Carbin, 2019), where the extent of shared training iterations influences the ease of finding low-loss paths between solutions. In particular, subsequent independent fine-tuning that results in distinct features (T_{Color} cue) can lead to slightly different local minima, ultimately resulting in (minor) barriers upon linear interpolation. Depending on the characteristics of the newly learned features, initialization in a common basin is no guarantee for future merging compatibility. In sharp contrast, *unshared task-specific knowledge* (e.g., when interpolating between a T_{Color}-specialized model and a T_{Noise}-specialized model) rapidly degraded upon interpolation. This highlights that distinct parameter adaptations optimized for unrelated specific cues are easily disrupted by averaging. This differential effect on shared versus unshared components aligns with observations made by Zaman et al. (2024) regarding knowledge dynamics in language models.

Benefits of Merging Sequentially Adapted Models. Another observation is the superior performance of merging models adapted in the *incremental* (naive CL) scenario compared to the *parallel* scenario, particularly for shared knowledge (Figure 1e, CIFAR-100 panel). When interpolating between sequentially trained states, the performance on the general CIFAR-100 task consistently surpassed that of merging parallel-trained endpoints, often exceeding even the original pre-trained model's accuracy. This suggests that the process of sequential adaptation, even without explicit CL mechanisms, moves the models along trajectories that are more amenable to beneficial merging. The shared trajectory of adaptation across tasks in the incremental setup could lead to more compatible or aligned representations of features, which are then effectively consolidated by interpolation. Similar results are presented by Marouf et al. (2024). This finding is particularly relevant for CL, as it indicates that merging states from a single, continually evolving model might be more advantageous than merging independently trained specialists.

Implications for Continual Learning. The observed degradation of unshared task-specific knowledge when merging models presents a complex consideration for CL. The fact that specific learned information can be lost during merging introduces a critical question of 'utility'. If such specific adaptations represent undesirable artifacts like reliance on various biases or spurious correlations, as framed by Zaman et al. (2024), their erosion could be beneficial. However, in many CL contexts, the unique knowledge acquired for distinct past tasks is precisely what needs to be preserved, especially as retraining on all historical data is often infeasible. In these cases, naive merging would exacerbate catastrophic forgetting. This underscores the challenge of determining whether specific knowledge components are valuable or detrimental, a decision crucial for applying merging, but also other CL mechanism, effectively.

When Might Merging Be Suitable for CL? Our results suggest that naive linear interpolation might be most promising for CL scenarios aiming to consolidate general, shared capabilities, or where tasks are sufficiently similar that their specific solutions are not too disconnected. It appears less suited for maintaining strong performance on diverse, specialized tasks if unique knowledge components are crucial. However, several factors could modify this outlook: *Task Similarity and Relatedness:* Merging models from closely related tasks may lead to less degradation. While one might question the gains if tasks are very similar, merging could still offer robustness by averaging out optimization noise or consolidating compatible specializations (Izmailov et al., 2018; Ilharco et al., 2022). *Model Architecture and Scale:* Our study used a variant of ResNet-18. Much recent merging success has been on very large models. As works like Ramasesh et al. (2022); Ilharco et al. (2022) suggest, the effectiveness of patching or merging can improve with scale, possibly due to greater parameter capacity allowing for more disentangled (Ortiz-Jimenez et al., 2023) or robust encoding of specific knowledge. *Merging Techniques:* Simple linear interpolation is a baseline. More advanced methods, including Fisher-weighted averaging (Matena & Raffel, 2022) or interference-resolving techniques (Yadav et al.,

2023; Marczak et al., 2024), might better preserve specific knowledge. However, such methods primarily re-balance existing knowledge components; their utility still depends on the desirability of preserving those specific components. *Nature of Endpoint Models:* The training regimen of the endpoint models is critical. Recent work in continual pre-training suggests that specific optimization strategies for endpoints, such as using exponential moving averages, can significantly improve their suitability for subsequent merging and overall CL performance (Udandarao et al., 2024; Marouf et al., 2024). This points towards co-designing training and merging strategies.

Operationally, linear merging is often applied post-hoc and is computationally inexpensive compared to many online CL methods that modify the training process itself. While this offers flexibility, the success of merging implicitly still relies on the initial training trajectories leading to compatible solutions in parameter space, a condition often facilitated by shared pre-training or task similarity (Frankle & Carbin, 2019).