

Appendix for Learning Multi-Source and Robust Representations for Continual Learning

Contents

A	Additional Information for the Experiment Setting	2
A.1	Device Configurations.	2
A.2	Datasets	2
A.3	Performance Metrics.	3
A.4	Implementation	3
A.5	SOTA Methods	3
B	Additional Analysis Results	5
B.1	Analysis with the same backbone	5
B.2	Analysis of forgetting curve	5
B.3	Computational cost analysis	6
B.4	Analysis with MSIDF visualization and interpretability	7
B.5	Ablation Study Analysis	9
B.6	Analysis of MSIDF configuration	10

A Additional Information for the Experiment Setting

A.1 Device Configurations.

All experiments were conducted in the same hardware environment, running Ubuntu 22.04.2 LTS, equipped with 256GB of memory and an Intel Xeon Silver 4320 processor. A single NVIDIA A100 GPU was used for computational acceleration.

A.2 Datasets

We conducted extensive experiments on seven different datasets, including CIFAR-10 (10), Tiny ImageNet (12), MNIST (13), CIFAR-100 (11), CUB-200-2011(CUB-200) (21), ImageNet-R (7), and Stanford Cars 196(Cars196) (9).

- **Standard Datasets:** CIFAR-10, Tiny ImageNet, and MNIST. These datasets, characterized by relatively simple image features and smaller scales, serve as foundational evaluation tools for initially verifying model performance.
- **Complex Datasets:** CIFAR-100, CUB-200, ImageNet-R, and Cars196. These datasets, with their higher difficulty levels and richer visual features, effectively test model performance in more challenging scenarios.

Each dataset was split into tasks based on categories, as detailed below:

- **CIFAR-10:** Divided into 5 tasks, each containing 2 labels, with a total of 10,000 training samples.
- **Tiny ImageNet:** Divided into 10 tasks, each containing 20 labels.
- **Rot-MNIST:** The MNIST dataset was rotated to generate 20 tasks. Each task has 10 label categories, representing digits 0-9.
- **CIFAR-100:** Divided into 10 tasks, each containing 10 labels.
- **CUB-200:** Divided into 10 tasks, each containing 20 labels.
- **ImageNet-R:** A variant of ImageNet containing 200 classes, divided into 10 tasks, each containing 20 classes.

- **Cars196:** Contains 196 classes, divided into 14 tasks, each containing 14 classes.

A.3 Performance Metrics.

To evaluate model performance under Task-IL and Domain-IL setting, we adopt two metrics: "Average" and "Last". The "Average" metric measures the model's average classification accuracy across tasks in a specific scenario, while "Last" indicates the model's classification accuracy on the final task. For Domain-IL, we report the overall accuracy on the R-MNIST, defined as the model's aggregated classification performance across all task-specific test samples.

A.4 Implementation

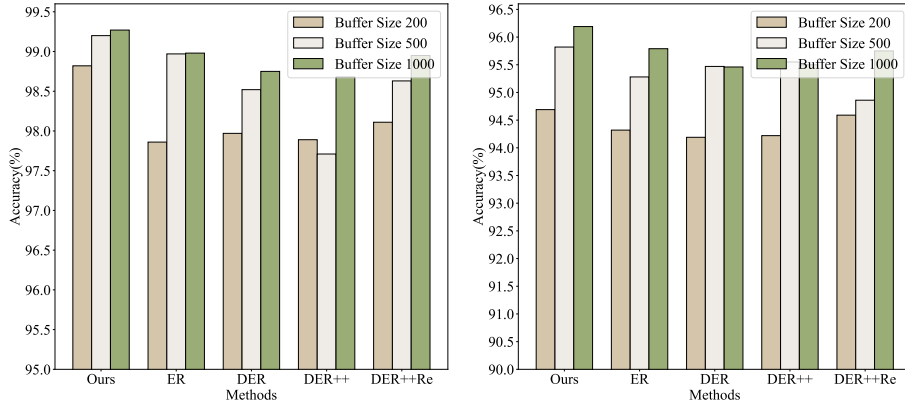
Our proposed model employs three variants of Vision Transformer (ViT) models as the backbone networks, each pre-trained on different datasets. Specifically, we utilized ViT-B/16 models pre-trained on ImageNet-1K and ImageNet-21K. During training and inference, the pre-trained parts of these ViT models were frozen, with only the last three layers remaining trainable. When processing the class_token outputs from multiple ViT models, we used two attention modules with sizes of 3 and 5, respectively. To ensure a fair comparison among different incremental learning methods, we selected the Stochastic Gradient Descent (SGD) optimizer to train all networks, with the hyperparameters set as follows: $\lambda = 0.1$, learning rate at 0.03, and batch size of 32. The input images were resized to a resolution of 224×224 and normalized to have values ranging from 0 to 1.

A.5 SOTA Methods

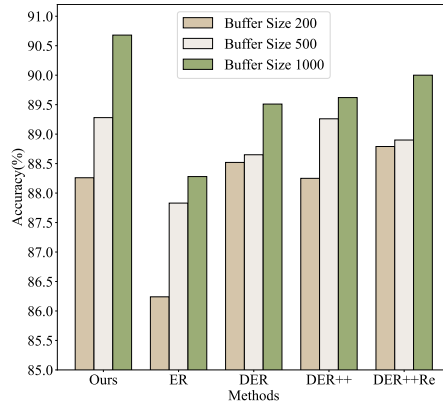
We conducted a comparative analysis of our proposed method with various SOTA methods in continual learning. Specifically, we compared our method with two regularization-based methods (EWC (20), SI (24)), two knowledge distillation-based methods (iCaRL (17), LwF (14)), one architectural method (PNN (19)), two prompt-based methods (DAP (8), L2P (23)), one multi-model fusion method (CoFiMA (16)), and a range of rehearsal-

based strategies (including ER (18), GEM (15), A-GEM (5), GSS (1), FDR (2), HAL (4), DER (3), DER++ (3), and DER++refresh (22)).

To ensure the rationality and consistency of the experimental setup, for methods that do not involve modifications to the backbone network structure (such as DER and DER++), we used a multilayer perceptron (MLP) with two hidden layers as the backbone on MNIST-related datasets; for other datasets, we employed an untrained simplified ResNet18 (6) as the backbone. For methods based on Vision Transformers (e.g., DAP, L2P, and CoFiMA), we followed their original configurations using pretrained ViT backbones and prompt or fusion modules for fair comparison.



(a) The result with the same backbone on CIFAR-10. (b) The result with the same backbone on CIFAR-100.



(c) The result with the same backbone on CUB-200.

Figure 1: Comparison of performance of various models with varying buffer sizes, where each model uses the same backbone.

B Additional Analysis Results

In this appendix, we provide additional ablation studies to analyze the performance of the proposed framework.

B.1 Analysis with the same backbone

For other SOTA methods that do not involve modifications to the backbone structure, we adopted the same multiple pre-trained ViT models as our method’s backbone. Each pre-trained ViT model is only allowed to update the parameters of the last three feature layers. The feature representations extracted by each pre-trained ViT are concatenated and then fed into a linear classifier to obtain the final output. Figure 1 shows the average accuracy of our method and SOTA models on the CIFAR-10, CIFAR-100, and CUB-200 datasets under different memory buffer configurations. The results indicate that our method consistently achieved the highest accuracy across various buffer sizes and significantly outperformed other models.

B.2 Analysis of forgetting curve

Fig. 2 presents the forgetting curves of our method and other SOTA methods on the CIFAR-10, CIFAR-100, and CUB-200 datasets. For SOTA methods that do not involve modifications to the backbone structure, the backbone settings are consistent with ours. The DAP method uses its original backbone from the source code without any modifications. For methods requiring a memory buffer, the buffer size was set to 500. It can be observed that on these three datasets, as the number of tasks increases, the forgetting curve value of the DAP method remains relatively high, while our method maintains stable and superior performance, achieving the lowest forgetting rate. This result highlights the effectiveness of our proposed MLRO technique, which continuously adjusts the representation optimization process over time, thereby effectively mitigating network forgetting.

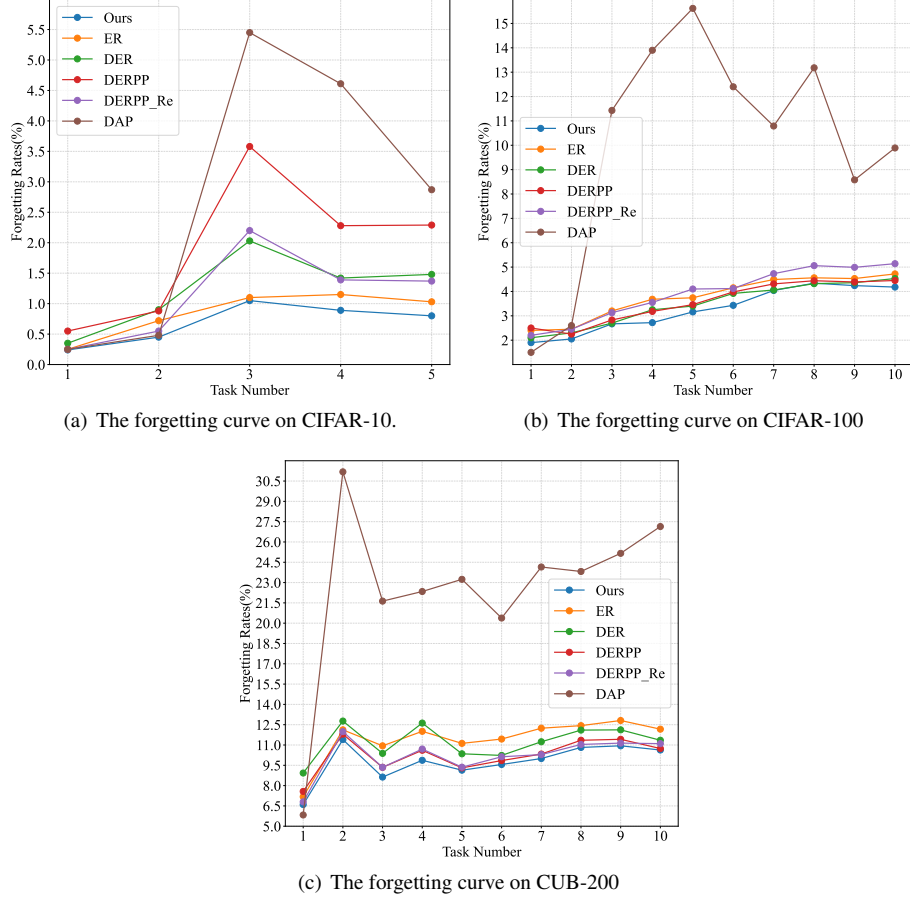


Figure 2: The comparison of the forgetting rates between ours and other baseline methods.

B.3 Computational cost analysis

To further evaluate the computational efficiency of the proposed LMSRR framework, we compare its resource consumption against representative baselines from four methodological categories, including replay-based, regularization-based, prompt-based, and multi-model ensemble approaches. The evaluation considers the following metrics: trainable parameters (Params, in Millions), average and peak GPU memory usage (GPU Avg/Max, in MiB), average and peak CPU memory usage (CPU Avg/Max, in MiB), and average iteration time (Iteration, in seconds per iteration). All experiments were conducted under identical hardware and training configurations.

Table 1: Computational cost comparison among different continual learning methods. All results are conducted on the CIFAR-100 dataset.

Methods	Params (M)	GPU Avg ↓	GPU Max ↓	CPU Avg ↓	CPU Max ↓	Iteration ↓
ER	21.42M	5876.93 MiB	5876.93 MiB	2279.25 MiB	2342.39 MiB	3.07 s/it
DERPP	21.42M	5607.20 MiB	5607.20 MiB	2202.59 MiB	2266.29 MiB	3.21 s/it
DERPP(Re)	21.42M	5607.12 MiB	5607.12 MiB	2196.54 MiB	2260.47 MiB	3.15 s/it
DAP	0.51M	3666.89 MiB	3666.89 MiB	2694.51 MiB	2724.04 MiB	1.33 s/it
CoFiMA	85.80M	9460.43 MiB	9896.50 MiB	2311.98 MiB	2684.08 MiB	2.41 s/it
LMSRR	42.53M	4672.31 MiB	4672.31 MiB	2499.88 MiB	2563.55 MiB	2.92 s/it

For other SOTA methods that do not involve modifications to the backbone structure, we employed a single pre-trained ViT model as the backbone, freezing all layers except the last three. Despite integrating multiple pre-trained backbones, LMSRR maintains a moderate number of trainable parameters (42.53M), which is substantially smaller than other multi-model fusion methods while providing superior adaptability and efficiency. LMSRR also achieves the lowest average GPU memory usage among methods utilizing large-scale backbones, confirming its lightweight memory footprint during training. Compared with prompt-based methods such as DAP, which use fewer parameters (0.51M) but are sensitive to prompt initialization and task shifts, LMSRR attains a more stable performance-resource balance.

B.4 Analysis with MSIDF visualization and interpretability

To further verify the interpretability and effectiveness of the proposed MSIDF module, we conducted additional experiments to examine whether MSIDF can effectively capture informative feature interactions and perform dynamic fusion rather than static averaging. Two analyses were performed for this purpose.

First, we evaluated the discriminability and structure of the fused features. Using the CIFAR-10 dataset, features extracted from two pre-trained ViT backbones (ViT_1 and ViT_2) were fused either by simple concatenation or by our proposed MSIDF module. We visualized the fused representations using t-SNE across all ten categories. As shown in Fig. 3, the features obtained through MSIDF exhibited clearer class separation and more compact clusters than those from simple concatenation. These results indicate that MSIDF produces more structured and discriminative representations, sup-

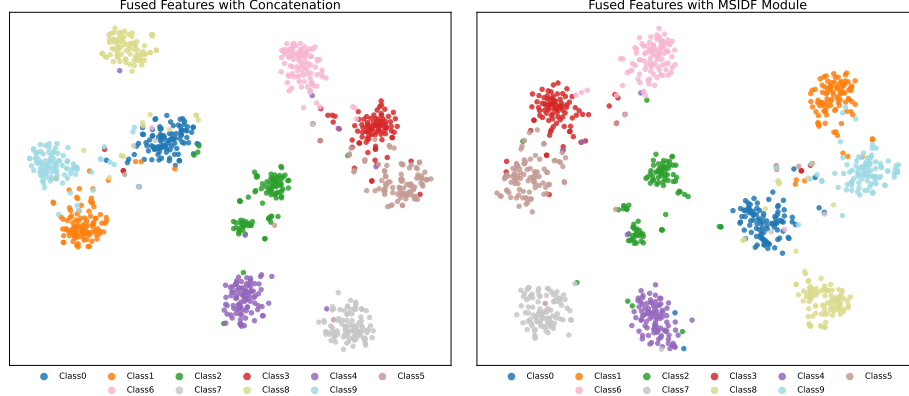


Figure 3: t-SNE visualization of fused features on CIFAR-10. Left: MSIDF fusion. Right: simple concatenation.

porting its ability to capture salient information and reveal intrinsic data structure.

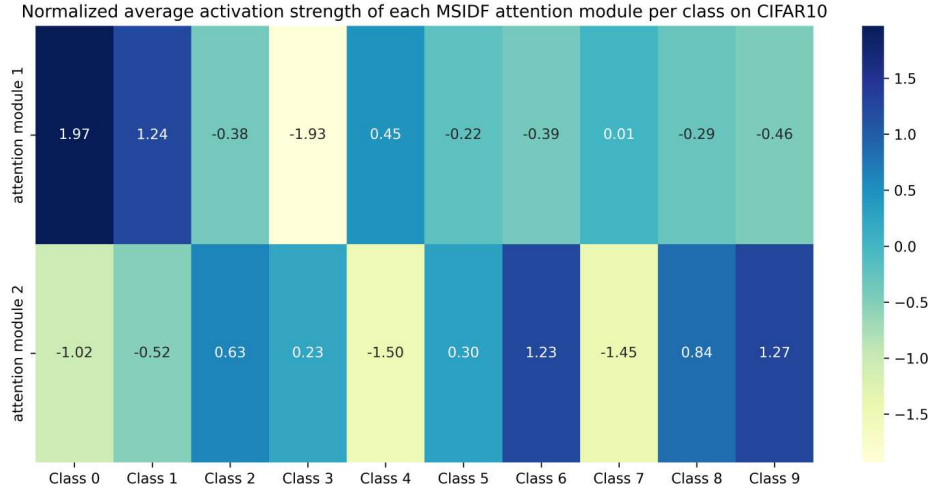


Figure 4: Average activation intensities of two attention modules across CIFAR-10 classes.

Second, to confirm the dynamic fusion behavior of MSIDF, we analyzed activation intensities from its two attention branches (corresponding to 3×3 and 5×5 convolutional windows). After completing each task on CIFAR-10, we randomly sampled class-specific images and computed the ℓ_2 -norms of the outputs from both branches to measure activation strength. Fig. 4 illustrates the average normalized activation inten-

sity for each class. The results show distinct activation preferences among categories; for example, Class 0 primarily activates the first attention branch, whereas Classes 6 and 9 tend to activate the second. This observation verifies that MSIDF performs input-adaptive, task-aware fusion, dynamically routing information through different attention paths rather than relying on fixed fusion weights. Together, these analyses provide intuitive and empirical support for the interpretability and effectiveness of MSIDF.

Table 2: Comprehensive ablation study on LMSRR.

Method	CIFAR-100	CUB-200	Cars196
w/o MSIDF	94.21 \pm 0.07	87.33 \pm 0.31	86.41 \pm 0.25
w/o MLRO	94.67 \pm 0.05	87.86 \pm 0.18	88.51 \pm 0.14
w/o ARO	94.92 \pm 0.09	87.79 \pm 0.22	88.66 \pm 0.20
LMSRR (ViT1 + ViT2)	95.76\pm0.08	88.91\pm0.64	90.14 \pm 0.06
ViT1 (In1k)	93.71 \pm 0.04	87.52 \pm 0.27	88.53 \pm 0.11
ViT2 (In21k)	91.58 \pm 0.06	84.61 \pm 0.35	85.64 \pm 0.21
ViT1 + ViT2 + ViT3	95.63 \pm 0.10	88.63 \pm 0.29	90.21\pm0.13

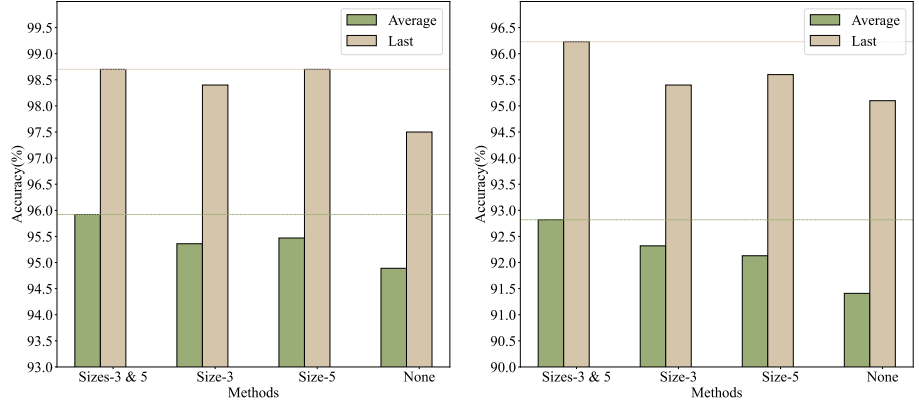
B.5 Ablation Study Analysis

We conducted a comprehensive ablation study on three benchmark datasets (CIFAR-100, CUB-200, and Cars196) to evaluate the individual contributions of each component within the LMSRR framework. The results in Table 2 report the Average accuracy.

Removing key components from LMSRR reveals their distinct functional roles. Without the MSIDF module, performance drops notably across all datasets, confirming that multi-scale attention fusion is crucial for extracting complementary and discriminative features from multiple ViT sources. Excluding the MLRO strategy results in moderate degradation, showing its effectiveness in maintaining inter-task stability by constraining representation drift. Removing the ARO mechanism also slightly reduces performance, particularly on fine-grained datasets, indicating that adaptive regularization via layer-wise switch vectors successfully mitigates over-regularization and preserves flexibility during continual updates.

Regarding backbone combinations, we analyze how the number and type of inte-

grated ViT backbones affect LMSRR performance. We employ three pretrained models: ViT1 (ImageNet-1k), ViT2 (ImageNet-21k), and ViT3 (ViT-L/14 pretrained on ImageNet-21k). Single-backbone variants (ViT1 or ViT2) exhibit limited generalization due to restricted semantic coverage. By combining two heterogeneous ViTs (ViT1 + ViT2), the proposed MSIDF module effectively aligns and fuses their complementary representations through dynamic multi-scale attention, substantially enhancing semantic diversity and discriminability. Introducing a third backbone (ViT3) yields marginal benefit, as the representational diversity from three pretrained sources approaches saturation and increases fusion redundancy. Therefore, the dual-source configuration achieves the optimal balance between representational richness, computational efficiency, and stability.



(a) Results of different configurations of MSIDF on CIFAR-100. (b) Results of different configurations of MSIDF on Tiny ImageNet.

Figure 5: Performance variations of the proposed MSIDF method under different configurations.

B.6 Analysis of MSIDF configuration

In this appendix, to investigate the impact of attention modules on the overall performance of our proposed MSIDF method, we designed and tested the following four configurations: MSIDF with two attention modules of different sizes-3 & 5; MSIDF with only a size-3 attention module; MSIDF with only a size-5 attention module; and

a baseline model without the MSIDF mechanism. These configurations were tested on the CIFAR-100 and Tiny ImageNet datasets. As shown in Fig. 5, the configuration with attention modules of different sizes achieved the best performance, and models using the MSIDF method consistently outperformed those without it. These results highlight the significance of MSIDF in enhancing overall model performance by enabling the capture of more critical feature information through these attention modules. Additionally, using multiple window combinations can further enhance this effect.

References

- [1] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. *Advances in neural information processing systems*, 32, 2019. 4
- [2] Ari S Benjamin, David Rolnick, and Konrad Kording. Measuring and regularizing networks in function space. *arXiv preprint arXiv:1805.08289*, 2018. 4
- [3] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020. 4
- [4] Arslan Chaudhry, Albert Gordo, Puneet Dokania, Philip Torr, and David Lopez-Paz. Using hindsight to anchor past knowledge in continual learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6993–7001, 2021. 4
- [5] Arslan Chaudhry, Marc’Aurelio Ranzato, Marcus Rohrbach, and Mohamed El-hoseiny. Efficient lifelong learning with A-GEM. In *Int. Conf. on Learning Representations (ICLR)*, *arXiv preprint arXiv:1812.00420*, 2019. 4
- [6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

- [7] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. *ICCV*, 2021. 2
- [8] Dahuin Jung, Dongyoon Han, Jihwan Bang, and Hwanjun Song. Generating instance-level prompts for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11847–11857, 2023. 3
- [9] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *2013 IEEE international conference on computer vision workshops*, pages 554–561. IEEE, 2013. 2
- [10] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Univ. of Toronto, 2009. 2
- [11] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 2
- [12] Ya Le and Xuan Yang. Tiny imageNet visual recognition challenge. Technical report, Univ. of Stanford, 2015. 2
- [13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998. 2
- [14] Z. Li and D. Hoiem. Learning without forgetting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017. 3
- [15] David Lopez-Paz and Marc’Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pages 6467–6476, 2017. 4
- [16] Imad Eddine Marouf, Subhankar Roy, Enzo Tartaglione, and Stéphane Lathuilière. Weighted ensemble models are strong continual learners. In *European Conference on Computer Vision*, pages 306–324. Springer, 2024. 3

- [17] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental classifier and representation learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2001–2010, 2017. 3
- [18] Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. *arXiv preprint arXiv:1810.11910*, 2018. 4
- [19] Andrei A Rusu, Neil C Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 3
- [20] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *International conference on machine learning*, pages 4528–4537. PMLR, 2018. 3
- [21] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 2
- [22] Zhenyi Wang, Yan Li, Li Shen, and Heng Huang. A unified and general framework for continual learning. *arXiv preprint arXiv:2403.13249*, 2024. 4
- [23] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 3
- [24] F. Zenke, B. Poole, and S. Ganguli. Continual learning through synaptic intelligence. In *Proc. of Int. Conf. on Machine Learning*, vol. *PLMR 70*, pages 3987–3995, 2017. 3