A Implementation Details and Additional Results

We organize additional supporting experimental findings as follows:

- Appendix B provides details on the datasets used in this paper.
- Appendix C provides additional implementation and training details for all of the methods.
- Appendix D includes experimental results on IID CL and demonstrates that RSGM's efficacy is not specific to CIL.
- Appendix E provides additional CIL experiments and results with rehearsal, including an analysis of learning curves, studying alternative sampling strategies for rehearsal, using a non-self-supervised backbone CNN, using a vision transformer, and using a balanced dataset. We find that RSGM works well across these experiments and analysis compared to vanilla rehearsal.
- Appendix F includes findings on memory constrained learning from scratch experiments. In this setting, RSGM outperforms vanilla rehearsal in all criteria.
- Appendix G studies RSGM in memory constrained online CL setting using a state-of-the-art online learning method, REMIND. We observe that RSGM combined with REMIND enhances performance in all metrics under various memory constraints.
- Appendix H studies the behavior of our stability gap mitigation method when used with Learning without Forgetting (LwF), a popular regularization method used in CL. We find that our method greatly improves results, illustrating that the mitigation strategy is not specific to rehearsal.
- Appendix I studies both CIL and IID CL on a long-tailed dataset when the memory buffer is constrained to only 100K samples. We find that RSGM's performance is almost entirely unaffected with this memory constraint, whereas vanilla rehearsal's performance decreases across all metrics.

B Dataset Details

This paper uses five benchmark datasets e.g., ImageNet-1K, Places365-LT, Places365-Standard, CUB-200, and CIFAR-10. ImageNet-1K (Russakovsky et al., 2015) has 1.2 million images from 1000 categories, each with $732-1300$ training images and 50 validation images. Places365-LT (Liu et al., 2019) is a long-tailed dataset with an imbalanced class distribution. It is a long-tailed variant of the Places-2 dataset (Zhou et al., 2017). Places365-LT has 365 classes and 62500 training images with 5 to 4980 images per class. For its test set, we use the Places365-LT validation set from (Liu et al., 2019) which consists of a total of 7300 images with a balanced distribution of 20 images per class. Places365-Standard (Zhou et al., 2017) has over 1.8 million training images from 365 classes with $3068-5000$ images per class. We use the validation set consisting of 100 images per class to test the models. CUB-200 (Wah et al., 2011) has RGB images of 200 bird species with 5994 training images and 5794 test images. CIFAR-10 (Krizhevsky et al., 2009) consists of 10 classes with 50000 training images and 10000 test images.

C Additional Implementation Details

In this section we provide additional implementation details for the models presented in the main text.

Main Experiments. For both CIL and IID experiments, we train RSGM, vanilla and head using cross-entropy loss for 600 iterations per rehearsal cycle. During each iteration model is updated on 128 samples. All methods use the same ConvNextV2 backbone [1], use AdamW optimizer with weight decay of 0.05 and initial learning rates of 10^{-3} (RSGM and vanilla) and 10^{-2} (head). LR is reduced in earlier layers by a layer-wise decay factor of 0.9. LR scheduler is not applied for vanilla and head due to poor performance. On the other hand, RSGM uses OneCycle LR scheduler (Smith & Topin, 2017). The offline model is trained for 12500 iterations on all data i.e., ImageNet-1K and Places365-LT combined using initial LR of 10^{-4} without LR scheduler. For all experiments, we set

[1] Pre-trained weights are available here: https://github.com/facebookresearch/ConvNeXt-V2

the rank of the LoRA weight matrices to 48. In all cases, all metrics are based on Top-1 accuracy (%). In general, most CL experiments including those in Sec. 6.1 adhere to the aforementioned settings unless otherwise noted.

Memory Constrained CL with DERpp and GDumb. We describe settings used in Sec. 6.2 where we combine RSGM with DERpp and GDumb while using identical settings e.g., same ImageNet-1K pre-trained ConvNeXt V2 Femto network and same optimizer settings. Each model pre-trained on ImageNet-1K learns Places365-Standard in 5 batches subsequently (73 categories per batch). Each rehearsal cycle contains total 1200 iterations with 256 samples per iteration. DERpp employs distillation and regularization along with rehearsal to prevent catastrophic forgetting. It regularizes loss on old samples and uses an additional distillation loss on logits of old samples for promoting consistency. We set coefficients $\alpha = 0.1$ and $\beta = 0.9$ for distillation and regularization respectively. GDumb randomly removes a sample from the largest class when buffer reaches its maximum capacity and maintains a class-balanced memory buffer. For all methods, memory buffer is bounded by maximum number of samples (80% ImageNet-1K + 20% Places365-Standard). DERpp, GDumb, and RSGM use initial LR of 1×10^{-3}, 1×10^{-3}, and 1.5×10^{-3} respectively for batch size 256. The offline model uses initial LR of 10^{-2} and 12K iterations with 256 samples per iteration. We assess performance during rehearsal every 100 minibatches to compute the metrics.

Class-balanced Rehearsal. For class balanced rehearsal experiments in Appendix E.3, RSGM and vanilla use initial LR of 10^{-3} and 10^{-4} respectively.

Non-SSL Backbone CNN. For non-SSL backbone experiments with ConvNeXt V1-Tiny (Liu et al., 2022) in Appendix E.4, initial learning rates for RSGM, vanilla, and offline model are 4×10^{-3}, 3×10^{-3}, and 10^{-4} respectively. ConvNeXt V1-Tiny has been pre-trained on ImageNet-1K using supervised learning [2].

ViT Backbone. For ViT backbone experiments in Appendix E.5, we select MobileViT-Small (Mehta & Rastegari) (5.6M) pretrained on ImageNet-1K using supervised learning [3]. We freeze first four MobileNetv2 blokcs and one MobileViT block for extracting universal features and keep the remaining blocks i.e., one MobileNetv2 block and two MobileViT blocks along with head plastic which consists of total 5.4M parameters. We apply LoRA (rank=48) to query, key and value projection matrices in the self-attention module of MobileViT transformer blocks. All methods use AdamW optimizer with weight decay of 0.01. Vanilla and RSGM use initial LR of 3×10^{-3} and 4×10^{-3} respectively. Initial LR for offline model is 10^{-2}. Places365-LT data is learned over 5 rehearsal cycles (73 classes per cycle) where each cycle includes 1200 iterations with 32 samples per iteration. Offline model is trained for 25K iterations with 64 samples per iteration. All other settings follow above mentioned settings for main experiments.

Balanced (Non-LT) Dataset. For experiments with balanced dataset in Appendix E.6, RSGM and vanilla use initial LR of 1.5×10^{-3} and 10^{-3} respectively. Each continual learner pretrained on ImageNet-1K learns Places365-Standard in 5 batches subsequently (73 categories per batch). Each rehearsal cycle contains total 1200 iterations with 256 samples per iteration. At the end of CL, total number of samples seen by a network is 50% of entire dataset (ImageNet and Places-Standard combined). We assess performance during rehearsal every 100 minibatches to measure the stability gap, plasticity gap, and continual knowledge gap. The offline model uses initial LR of 10^{-2} and 12K iterations with 256 samples per iteration.

Memory Constrained Learning from Scratch. In Appendix F, both RSGM and vanilla use AdamW optimizer with initial LR of 0.005 and weight decay of 0.05 for batch size 64. We reduce LR for old class units in output layer by a factor of 0.9. We create Femto version of ConvNeXt V1 following ConvNext V2 Femto configuration and modify stem layer with 3×3 kernels and stride 1 for 32×32 image resolution of CIFAR-10. Following ConvNeXt V1 (Liu et al., 2022), we use cosine scheduler for LR decay and weight decay. For learning from scratch, we do not use any pre-trained weights or backbone. Each model learns CIFAR-10 in 5 incremental batches with 2 classes per batch. We use 50 epochs for each batch and 10 linear warmup epochs for the first batch only. We assess performance during rehearsal every 5 epochs to measure the stability gap, plasticity gap, and

[2]The pre-trained weights are available here: https://github.com/facebookresearch/ConvNeXt

[3]The pre-trained weights are available here: https://github.com/apple/ml-cvnets

Table 4: **IID Continual Learning**. Results after learning ImageNet-1K followed by Places365-LT over 5 batches with 12500 samples per batch. Here μ and α denote average accuracy and final accuracy respectively. $\#P$ denotes trainable parameters in Millions. Reported value is the average of 5 runs with standard deviation (SD) placed in parentheses as ($\pm$SD).

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	5.08	—	—	—	—	70.69
Vanilla	5.08	0.014 ($\pm$0.0004)	0.173 ($\pm$0.0056)	0.034 ($\pm$0.0005)	68.45 ($\pm$0.0470)	68.77 ($\pm$0.0517)
RSGM	**1.45**	**−0.004** ($\pm$0.0005)	**0.131** ($\pm$0.0017)	**0.003** ($\pm$0.0004)	**70.81** ($\pm$0.0151)	**71.23** ($\pm$0.0664)

continual knowledge gap. The offline model is trained on entire CIFAR-10 dataset for 100 epochs with 20 linear warmup epochs. It uses same settings as used by CL models.

Memory Constrained Online CL. In Appendix G, we use identical settings and hyperparameters for both REMIND and REMIND + RSGM methods. We use ImageNet-1K pre-trained ConvNeXt V2 Femto with similar network configurations and LoRA configurations as used in main experiments. We use AdamW optimizer and REMIND's default per-class learning rate scheduler. We set initial LR to 1×10^{-3}, final LR to 1×10^{-5}, and weight decay to 0.05. Following REMIND, we perform rehearsal with a mini-batch of 51 samples including 50 old samples and 1 new sample. Each method learns CUB-200 dataset in *sample-by-sample* manner. For all methods, memory buffer is bounded by maximum number of samples ($75\% - 93\%$ ImageNet).

Regularization Methods. In Appendix H, LwF has similar configurations as vanilla except initial LR (6×10^{-5}). LwF + SGM has similar configurations as RSGM except initial LR (2×10^{-4}). During each iteration model is updated on 64 new samples without any rehearsal of old samples.

All other settings adhere to above mentioned general settings for main experiments unless otherwise mentioned. Hyperparameters are tuned to maximize performance for each method. We run all experiments on same hardware with a single GPU (NVIDIA RTX A5000).

D IID Continual Learning

To understand if RSGM would be useful for other CL data distributions, we examine its behavior in an IID ordering where each of the 5 CL batch contains randomly sampled classes from Places365-LT. During IID CL, the model sequentially learns 5 incremental batches of data from Places365-LT where each incremental batch contains 12500 examples. Our results are summarized in Table 4. In terms of final accuracy, RSGM achieves a final accuracy of 71.23%, outperforming vanilla rehearsal's 68.77% accuracy, and surprisingly even the offline model's 70.69% accuracy. RSGM achieves a negative stability gap, which indicates knowledge transfer from new classes to old classes. In contrast, we found there was a small stability gap in class-incremental learning, likely due to the dissimilarity among subsequent batches.

E Additional CIL Analysis & Experiments

In this section we conduct additional analysis of the CIL experiments in the main text as well as present additional experiments.

E.1 Learning Curves

In our main text, our figures are averaged across rehearsal cycles. In Fig. 4, we instead present all of the learning curves in sequence, where we denote when the next batch containing new classes is received.

When rehearsal begins, accuracy on ImageNet-1k for vanilla rehearsal drops dramatically and gradually decreases throughout the rehearsal cycles. At the end, vanilla fails to recover the original performance using total 3K iterations. In contrast, RSGM shows better performance throughout

rehearsal cycles with reduced stability gap and full recovery compared to the offline model. Models are evaluated every 10 iterations. After each rehearsal cycle, RSGM outperforms vanilla and matches or exceeds offline accuracy.

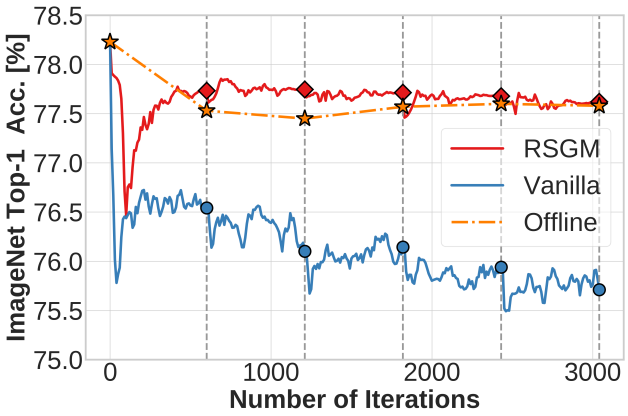
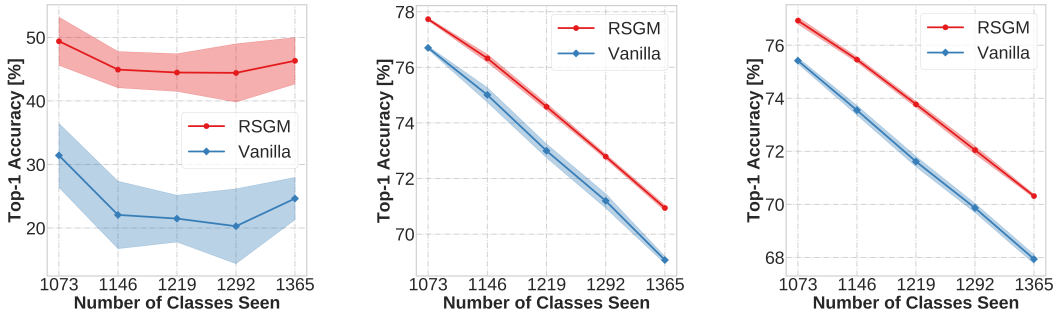


Figure 4: **Stability gap in all rehearsal cycles.** After pretraining on ImageNet-1K, the model learns 365 new classes from Places365-LT over five rehearsal cycles (73 new classes and 600 iterations per rehearsal cycle). RSGM quickly recovers old performance in the beginning of CL whereas vanilla fails to obtain full recovery. After each rehearsal cycle (vertical dotted gray line), final accuracy is highlighted by diamond (RSGM), star (offline), and circle (vanilla).

In Fig. 5, we also illustrate model's accuracy on new, old, and all classes in all rehearsal cycles where RSGM achieves higher accuracy than vanilla. This indicates that RSGM consistently improves model's plasticity (Fig. 5a), stability (Fig. 5b), and knowledge accumulation (Fig. 5c).



(a) Accuracy on new classes (b) Accuracy on old classes (c) Accuracy on all classes

Figure 5: **Stability-plasticity.** After pre-training on ImageNet-1K, the model learns 365 new classes from Places365-LT over five batches (73 new classes per batch) in class incremental setting. The accuracy is averaged over 6 runs and shaded region indicates standard deviation.

E.2 Repeated CIL Experiments

The results in Table 1 are for a single ordering of the CL batches. While it was not computationally feasible to use all CL batch orderings for every method, we repeated this experiment for 6 orderings for RSGM and vanilla rehearsal. We also included head for comparison where we froze all layers except the final layer and trained final layer during rehearsal. The averaged results across runs are given in Table 5, and we find that RSGM consistently mitigates the stability gap achieving an $\mathcal{S}_\Delta$ of 0.001 compared to 0.019 for vanilla rehearsal. Besides that RSGM achieves outperforming scores in every other criteria compared to vanilla. RSGM also outperforms head in all criteria. This indicates that updating representations in earlier layers besides head using RSGM is critical for learning new knowledge and retaining old knowledge.

Table 5: **Results averaged over 6 runs (CIL)**. Experimental results are based on ImageNet-1K and Places365-LT datasets. A continual learner pre-trained on ImageNet learns Places in 5 batches subsequently (73 categories per batch). Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total number of trainable parameters in Millions. Reported value is the average of 6 runs with standard deviation (SD) placed in parentheses as ($\pm$SD). The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively.

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	5.08	—	—	—	—	70.69
Vanilla	5.08	0.020 ($\pm$0.0017)	0.385 ($\pm$0.0091)	0.031 ($\pm$0.0015)	71.68 ($\pm$0.1236)	67.94 ($\pm$0.1721)
Head	**0.53**	0.021 ($\pm$0.0011)	0.473 ($\pm$0.0250)	0.032 ($\pm$0.0009)	71.28 ($\pm$0.1410)	67.68 ($\pm$0.2710)
RSGM	1.45	**0.001** ($\pm$0.0012)	**0.087** ($\pm$0.0082)	**0.002** ($\pm$0.0007)	**73.71** ($\pm$0.0763)	**70.31** ($\pm$0.0682)

E.3 Class Balanced Uniform Sampling for Rehearsal

In our main results, we sampled randomly during rehearsal without balancing for each class. However, prior work has shown that class balanced random sampling works significantly better than unbalanced uniform sampling for long-tailed datasets (Harun et al., 2023b). We conducted a CIL experiment to examine this in our memory unconstrained rehearsal setup where we learn ImageNet-1K followed by Places365-LT.

Table 6 shows that using class balanced rehearsal, RSGM improves performance in most criteria compared to previous results without class balance (Table 1). When both vanilla and RSGM use class balanced rehearsal, RSGM outperforms vanilla by 7.3$\times$ in stability gap, 3.6$\times$ in plasticity gap and provides continual knowledge transfer ($\mathcal{CK}_\Delta < 0$).

Table 6: **Class Balanced Rehearsal**. Experimental results are based on ImageNet-1K and Places365-LT datasets. A continual learner pre-trained on ImageNet learns Places in 5 batches subsequently (73 categories per batch). Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total number of trainable parameters in Millions. The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively.

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	5.08	—	—	—	—	70.69
Vanilla	5.08	0.022	0.316	0.021	72.24	69.03
RSGM	**1.45**	**0.003**	**0.089**	**−0.003**	**74.00**	**70.61**

E.4 Analysis with a Non-Self-Supervised Backbone CNN

Much of deep learning has moved toward self-supervised pretraining prior to supervised fine-tuning, especially in foundation models (Devlin et al., 2018; Brown et al., 2020; Ramesh et al., 2021), since this has been shown to reduce overfitting on the pretext dataset used for self-supervised learning and to generalize better to downstream tasks. In the main text, we used the self-supervised ConvNextV2 architecture. This may have enabled our system to achieve higher results on Places365-LT than if the CNN was initialized from ImageNet-1K with supervised learning. To determine if our general trends for the methods hold, we conducted another experiment with ConvNeXt V1 Tiny (29M), which is pre-trained on ImageNet-1K without self-supervision.

Experimental results in Table 7 demonstrate that RSGM with supervised backbone mitigates stability gap and enhances performance in all criteria. Therefore efficacy of RSGM does not depend upon self-supervised pre-training.

Table 7: **CIL without Self-Supervised Pre-Training**. This table shows results from ConvNeXt V1-Tiny pre-trained on ImageNet-1K using supervised learning, which then learns Places365-LT in 5 batches subsequently (73 categories per batch) in class-incremental setting. Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total trainable parameters in Millions. The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively.

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	27.00	—	—	—	—	74.16
Vanilla	27.00	0.030	0.396	0.035	74.73	70.67
RSGM	**3.53**	**0.005**	**0.102**	**0.001**	**77.48**	**73.92**

E.5 Analysis with using a Vision Transformer Backbone

In this section we study the behavior of the system for a ViT model pretrained with supervised learning. For this, we select a light-weight transformer, MobileViT small (Mehta & Rastegari). MobileViT learns local and global representations using convolutions and transformers, respectively. It has total 5.6 million parameters and top-1 accuracy of 78.4% on ImageNet-1K.

Table 8 shows the comparison between vanilla and RSGM when they have same MobileViT backbone. RSGM shows better performance in all criteria using $3.8\times$ fewer parameters than vanilla.

Table 8: **Vision Transformer Backbone**. Experimental results are based on ImageNet-1K and Places365-LT datasets. A continual learner pre-trained on ImageNet learns Places in 5 batches subsequently (73 categories per batch). Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total number of trainable parameters in Millions. The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively.

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	4.97	—	—	—	—	69.10
Vanilla	4.97	0.039	0.434	0.046	70.18	66.35
RSGM	**1.30**	**0.016**	**0.140**	**0.016**	**72.09**	**67.96**

E.6 Balanced (Non-LT) Dataset

In real-world setting, data distribution is commonly long-tailed and imbalanced, hence we used Places365-LT dataset in the main results. However, our analysis holds for balanced and non-LT dataset as well. We study this using Places365-Standard. Results in Table 9 show that RSGM outperforms vanilla rehearsal in all criteria.

Table 9: **Non-LT Dataset**. Experimental results are based on ImageNet-1K and Places365-Standard datasets. A continual learner pre-trained on ImageNet learns Places in 5 batches subsequently (73 categories per batch). Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total number of trainable parameters in Millions. The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively.

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	5.08	—	—	—	—	65.37
Vanilla	5.08	0.078	0.201	0.082	66.37	56.63
RSGM	**1.45**	**0.054**	**0.091**	**0.047**	**68.47**	**59.21**

F Memory Constrained Learning From Scratch

In the main text, we define our problem setting with a base initialization phase where a model acquires base knowledge using a pre-train dataset. Here we also test another problem setting without the base initialization phase where a model learns from scratch. We study stability gap and efficacy of RSGM when a model is trained from scratch on CIFAR-10 dataset in 5 rehearsal cycles (2 classes per rehearsal cycle). Since the model is trained on small number of training data, it learns less

transferable representations. Therefore, instead of using LoRA and freezing old class units in output layer, we train all layers and update old class units with lower learning rate. Whereas we use dynamic soft targets and data-driven weight initialization as used in our main experiments. We summarize our findings in Table 10 where RSGM achieves higher scores than vanilla rehearsal in all metrics. RSGM outperforms vanilla rehearsal by 3.43% (50K samples in buffer) and 3.45% (5K samples in buffer) in final accuracy.

Table 10: **Learning from scratch**. A model learns CIFAR-10 from scratch in 5 incremental batches (2 classes per batch). Memory buffer is bounded by max number of samples. Here μ, α, and $\#P$ denote average accuracy over batches, final accuracy, and parameters (Millions) respectively.

Method	Buffer	$\#P(\downarrow)$	$\mathcal{S}_{\Delta}(\downarrow)$	$\mathcal{P}_{\Delta}(\downarrow)$	$\mathcal{CK}_{\Delta}(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Vanilla	50K	4.83	0.217	0.047	0.166	77.60	66.91
RSGM	50K	4.83	**0.192**	**0.046**	**0.147**	**78.64**	**70.34**
Vanilla	5K	4.83	0.445	0.061	0.331	70.33	53.63
RSGM	5K	4.83	**0.422**	**0.058**	**0.318**	**71.06**	**57.08**

G Memory Constrained Online Continual Learning

In the main text, we study stability gap in incremental batch learning setting. Here we study stability gap in online continual learning setting using a state-of-the-art online learning method, REMIND (Hayes et al., 2020). We conduct memory constrained CL experiments with CIL data ordering, where we combine RSGM with REMIND while using identical configurations. We summarize the results in Table 11. We observe that RSGM combined with REMIND (REMIND + RSGM) outperforms REMIND (without RSGM) by large margins in all metrics and shows effectiveness in online learning setting. We also observe that RSGM maintains similar effectiveness across various memory constraints.

Table 11: **Online Continual Learning**. A model pre-trained on ImageNet-1K learns CUB-200 sample-by-sample with a replay mini-batch of 51 samples (50 old + 1 new). Here μ, α, and $\#P$ denote average accuracy over batches, final accuracy, and parameters (Millions) respectively.

Method	Buffer	$\#P(\downarrow)$	$\mathcal{S}_{\Delta}(\downarrow)$	$\mathcal{P}_{\Delta}(\downarrow)$	$\mathcal{CK}_{\Delta}(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline model	—	5.08	—	—	—	—	75.99
REMIND	80994	5.08	0.146	0.837	0.162	64.15	62.35
REMIND + RSGM	80994	**1.45**	**0.034**	**0.675**	**0.049**	**72.81**	**69.67**
REMIND	44394	5.08	0.156	0.834	0.172	63.37	60.94
REMIND + RSGM	44394	**1.45**	**0.034**	**0.661**	**0.049**	**72.80**	**69.63**
REMIND	24594	5.08	0.173	0.834	0.189	62.10	59.06
REMIND + RSGM	24594	**1.45**	**0.036**	**0.672**	**0.051**	**72.69**	**69.55**

H Using our Stability Gap Mitigation Method with Regularization Methods

In the main text, we restrict our analysis to rehearsal methods. We hypothesized that our combined mitigation strategy would be helpful for non-rehearsal methods as well. We therefore study stability gap mitigation (SGM), which combines soft targets, weight initialization, OOCF, and LoRA, without rehearsal using Learning without Forgetting (LwF) (Li & Hoiem, 2017), which pioneered using knowledge distillation in CL (Zhou et al., 2023). Instead of rehearsal, LwF stores a copy of the model before learning the new CL batch to update the model with distillation. LwF has been shown to reduce catastrophic forgetting in a range of CL scenarios, although it and other regularization-based methods have not been shown to be effective in the CIL setting (Zhou et al., 2023).

We conducted an experiment to compare vanilla LwF with a version of LwF that uses SGM without rehearsal during CIL of ImageNet and Places365-LT. Overall results are given in Table 12 and a learning curve is given in Fig. 6. As expected based on prior results, rehearsal methods vastly

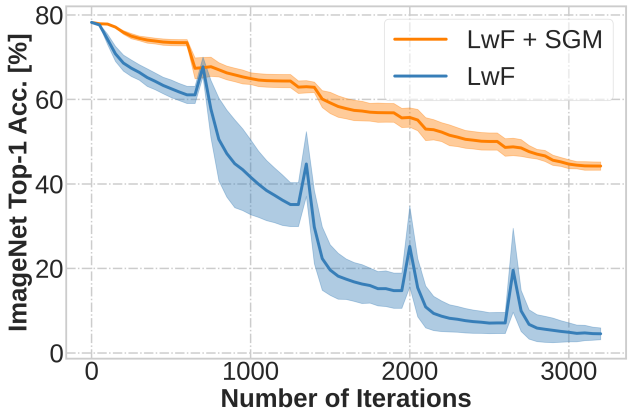


Figure 6: **Comparison with regularization method.** Y axis shows average accuracy of 6 runs with standard deviation (shaded region). The network is trained on ImageNet-1K and then learns 365 new classes from Places-LT over five batches (73 new classes and 600 iterations per batch). When new batch arrives, accuracy on ImageNet-1k for LwF plummets. LwF fails to recover performance and ends up with large stability gap. In contrast, LwF with SGM does not plummet like LwF and shows better performance throughout CL phase with significantly reduced stability gap.

outperform LwF; however, we find that SGM provides an enormous benefit to LwF in terms of reducing the stability gap, resulting in increased accuracy.

Table 12: **Comparison with regularizaton method**. A continual learner pre-trained on ImageNet-1K learns Places365-LT in 5 batches subsequently (73 categories per batch) in CIL setting. Results are averaged over 6 runs. Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total trainable parameters in Millions. The $(\uparrow)$ and $(\downarrow)$ indicate high and low values to reflect optimum performance respectively. For regularization baseline, we select LwF that regularizes model based on knowledge distillation.

Method	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	5.08	—	—	—	—	70.69
Vanilla Rehearsal	5.08	0.019	0.384	0.030	71.71	68.01
RSGM	**1.45**	**0.001**	**0.087**	**0.002**	**73.71**	**70.31**
LwF	5.08	0.605	0.450	0.607	24.04	4.76
LwF + SGM	**1.45**	**0.236**	**0.072**	**0.235**	**54.87**	**40.00**

I Memory Constrained Experiments with LT Dataset

Since, memory constraint was relaxed in the main results for LT dataset (Places365-LT), here we study the stability gap under memory constraints when learning ImageNet-1K followed by CL of Places365-LT. In memory restricted CL for both class-incremental and IID settings, the learner can store and access only 7.5% of entire dataset (ImageNet and Places combined). Now learner has access to 100K samples (old and current data combined) compared to unconstrained setup where learner had access to 1.34M samples.

Following the common practice of storing 120K instances for ImageNet-1K with rehearsal (Rebuffi et al., 2017), we set memory upper bound to 100K instances where 38K instances are randomly sampled from the ImageNet-1K dataset and stored in the memory buffer and remaining 62K are incrementally added to the buffer as Places365-LT is learned continually.

Our results for memory constrained rehearsal for CIL are summarized in Table 13. And the results for memory constrained rehearsal in the IID setting are summarized in Table 14. Our observations and conclusions about RSGM and vanilla rehearsal made in unconstrained CL still hold for con-

Table 13: **Memory constrained CL (CIL)**. A continual learner pre-trained on ImageNet-1K learns Places365-LT in 5 batches subsequently (73 categories per batch). Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total trainable parameters in Millions. The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively. First two rows are memory unconstrained methods for comparison. Memory is constrained in terms of maximum number of instances (2nd column) a model can store in the buffer.

Method	Max instances	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	1343667	5.08	—	—	—	—	70.69
Vanilla	1343667	5.08	0.028	0.393	0.033	71.52	67.67
RSGM	1343667	**1.45**	**0.006**	0.082	**0.002**	**73.70**	**70.30**
Vanilla	100900	5.08	0.040	0.388	0.044	70.62	65.99
RSGM	100900	**1.45**	**0.006**	**0.081**	**0.002**	73.67	70.23

Table 14: **Memory constrained CL (IID)**. A continual learner pre-trained on ImageNet-1K learns Places365-LT in 5 batches subsequently (12500 samples per batch). Here μ denotes average accuracy over batches and α is final accuracy. $\#P$ denotes total trainable parameters in Millions. The ($\uparrow$) and ($\downarrow$) indicate high and low values to reflect optimum performance respectively. First two rows are memory unconstrained methods for comparison. Memory is constrained in terms of maximum number of instances (2nd column) a model can store in the buffer.

Method	Max instances	$\#P(\downarrow)$	$\mathcal{S}_\Delta(\downarrow)$	$\mathcal{P}_\Delta(\downarrow)$	$\mathcal{CK}_\Delta(\downarrow)$	$\mu(\uparrow)$	$\alpha(\uparrow)$
Offline	1343667	5.08	—	—	—	—	70.69
Vanilla	1343667	5.08	0.014	0.177	0.033	68.45	68.68
RSGM	1343667	**1.45**	**−0.004**	0.129	**0.003**	70.80	**71.14**
Vanilla	100900	5.08	0.027	0.173	0.045	67.50	66.90
RSGM	100900	**1.45**	**−0.004**	**0.128**	**0.003**	**70.81**	71.07

strained CL. In constrained setup, overall accuracy drops and the stability gap worsens for vanilla rehearsal, whereas RSGM is largely unaffected.