RETHINKING SALIENCY IN DATA-FREE CLASS INCREMENTAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Data-Free Class Incremental Learning (DFCIL) aims to sequentially learn tasks with access only to data from the current one. DFCIL is of interest because it mitigates concerns about privacy and long-term storage of data, while at the same time alleviating the problem of catastrophic forgetting in incremental learning. In this work, we rethink saliency in DFCIL and propose a new framework, which we call RObust Saliency Supervision (ROSS), for mitigating the negative effect of saliency drift. Firstly, we use a teacher-student architecture leveraging low-level tasks to supervise the model with global saliency. We also apply boundary-guided saliency to protect it from drifting across object boundaries at intermediate layers. Finally, we introduce a module for injecting and recovering saliency noise to increase robustness of saliency preservation. Our experiments demonstrate that our method can achieve state-of-the-art results on the CIFAR-100, Tiny-ImageNet and ImageNet-Subset DFCIL benchmarks. Code will be made publicly available.

1 INTRODUCTION

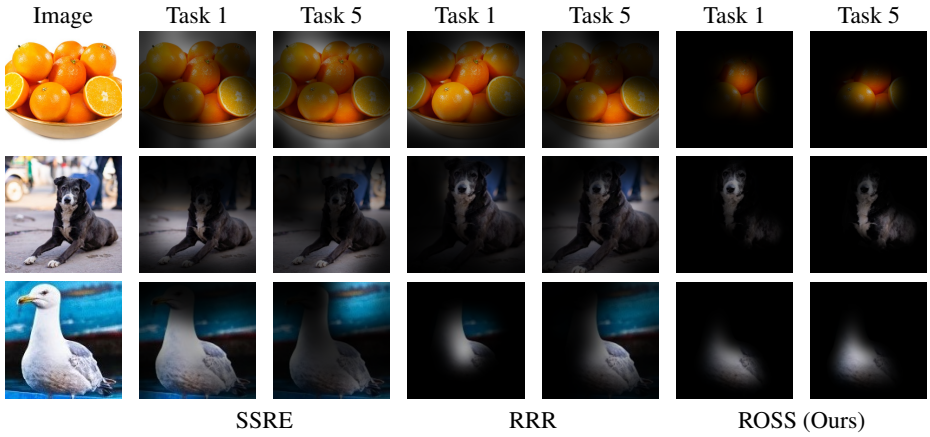
Deep neural networks achieve state-of-the-art performance on many computer vision tasks. However, most of these tasks consider a static world in which tasks are well-defined, stationary, and all training data is available in a single training session. The real world consists of dynamically changing environments and data distributions, which – especially given the computational burden of training large CNNs – has led to renewed interest learning new tasks incrementally while avoiding catastrophic forgetting of old ones (McCloskey & Cohen, 1989; Goodfellow et al., 2013).

Class Incremental Learning (CIL) (Masana et al., 2020; Belouadah et al., 2021) is one such incremental learning scenario that considers the possibility of adding new classes to already-trained models. Most CIL methods rely on a memory buffer that stores data from past tasks (Rebuffi et al., 2017; Castro et al., 2018; Douillard et al., 2020; Wu et al., 2019). In this paper, we consider Data-Free Class Incremental Learning (DFCIL), which is a more challenging scenario in which *no data from previous tasks is retained*. This is a realistic scenario and of great interest due to privacy concerns or restrictions on long-term storage of data. The inability to retain examples from past tasks, however, significantly exacerbates the problem of catastrophic forgetting.

There are several recent works that consider the DFCIL problem. DeepInversion (Yin et al., 2020) inverts trained networks from random noise to generate images as exemplars mixed with current samples for training. SDC (Yu et al., 2020) updates previous prototypes of each learned class by hypothesizing that semantic drift of previous classes can be approximated and estimated using new data. Other previous works propose representation learning methods for overcoming catastrophic forgetting (Zhu et al., 2021a;b). As pointed out in IL2A (Zhu et al., 2021a), learning better representations can reduce representation bias when transferred to new tasks. Incorporating self-supervised learning tasks, such as Barlow Twins (Pham et al., 2021) and rotation prediction (Zhu et al., 2021b), has also been proposed to achieve more stable representations and alleviate forgetting.

CNNs naturally learn to *attend* to features that are discriminative for the tasks they are trained to solve. Catastrophic forgetting also occurs when learning new tasks because the model's attention to *salient* features drifts to features specific to the new task. Standard regularization approaches do little to prevent this saliency drift when learning new tasks. One direct method of regularizing saliency is to apply distillation on saliency maps of old samples (Ebrahimi et al., 2021). However, this is complicated by the inability to save old samples in the DFCIL setting. Another method

Figure 1: Illustration of saliency drift across 5 tasks on ImageNet-Subset. SSRE (Zhu et al., 2022) as a representative method without saliency supervision, which results in random saliency drift toward the background. RRR (Ebrahimi et al., 2021) is a method applying vanilla saliency supervision, which fails to avoid saliency drift due to boundary regions. These phenomena lead to incorrect predictions: Orange to Bowl, Dog to Shoe, Bird to Boat. In comparison, our method maintains robust saliency while preventing saliency drift across boundaries.



is to apply saliency distillation on current task samples with previous task attention (Dhar et al., 2019). This method however suffers from the semantic gap between current and old classes when enforcing saliency consistency. The lack robust saliency regularization may also lead to attention drifting toward the background in future tasks. As demonstrated in Table 1, a conventional baseline (e.g. SSRE (Zhu et al., 2021b)) results in random saliency drift toward the background, and vanilla saliency supervision (e.g. RRR (Ebrahimi et al., 2021)) fails to avoid saliency drift across boundary regions in future tasks. Our Robust Saliency Supervision (ROSS) approach, however, keeps saliency focused on the foreground (for more details, see the Section 3).

Motivated by these observations, we propose the Robust Saliency Supervision (ROSS) approach which incorporates three components to address this problem. ROSS first uses a teacher-student architecture in which the teacher model provides low-level supervision of salient regions and salient region boundary maps. This serves as a stationary supervision signal over the incremental model. Additionally, we apply dilated boundary maps to avoid saliency drift across object boundaries at intermediate layers in the CNNs. Since saliency drift usually happens across tasks, encouraging the model to focus on important foreground regions with dilated boundary supervision reduces possibility of saliency shifting toward the background. Finally, inspired by SDC (Yu et al., 2020), we propose a module to inject saliency noise into some feature channels and train the network to denoise them. This helps the network further resist saliency drift across tasks.

The main contributions of this work are: **(i)** We provide new insight into robust saliency supervision under DFCIL settings. We also show the negative effect of methods with no or trivial saliency supervision, which illustrates the superiority of our method. **(ii)** We propose the Robust Saliency Supervision (ROSS) framework with three components that combine to mitigate the saliency drift problem. **(iii)** We show that ROSS can be easily integrated it into other state-of-the-art methods, such as MUC (Liu et al., 2020), IL2A (Zhu et al., 2021a), PASS (Zhu et al., 2021b), SSRE (Zhu et al., 2022), leading to significant performance gains. **(iv)** Our experiments demonstrate that ROSS outperforms all existing DFCIL methods and even several exemplar-based methods on the CIFAR-100, Tiny-ImageNet, and ImageNet-Subset DFCIL benchmarks.

2 Robust Saliency Supervision

We firstly define the Data-free Class Incremental Learning (DFCIL) scenario and our teacher-student framework for low-level saliency supervision. Then we describe our approach to dilated boundary

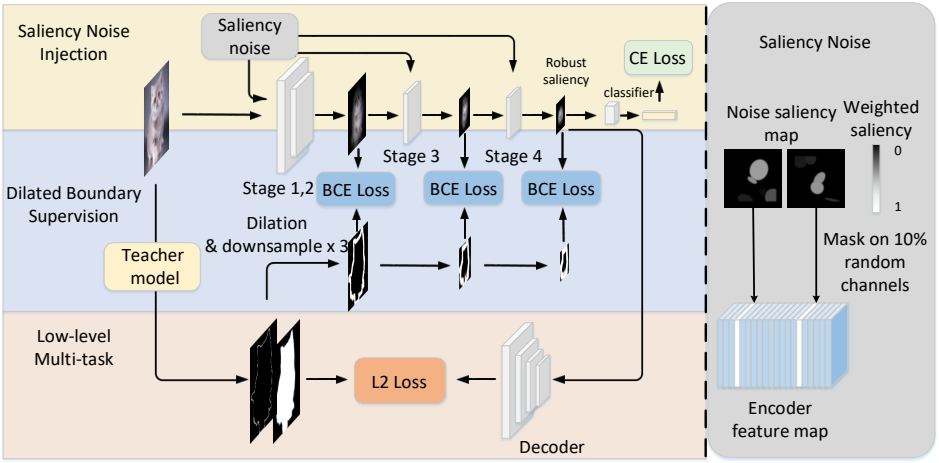


Figure 2: Overall framework of our Robust Saliency Supervision (ROSS). We apply a teacher model to generate saliency and boundary maps. The boundary map is dilated and downsampled to provide supervision in different stages of encoder. A decoder is attached after the encoder for low-level distillation, which serves as the teacher guidance over robust saliency to ensure its validity. To prevent saliency drift in later training phases, we introduce saliency noise into each encoder stage. The model is trained to denoise and reduce the saliency drift on testing current data in future phases.

supervision and saliency noise injection that further mitigate saliency drift in DFCIL. Our overall framework is illustrated in Figure 2.

2.1 Data-Free Class-Incremental Learning (DFCIL)

Class-incremental learning aims to sequentially learn tasks consisting of disjoint classes of samples. Let $t \in \{1, 2, ..T\}$ denote the incremental learning tasks. The training data D_t for each task contains classes C_t with N_t training samples $\{(x_t^i, y_t^i)\}_{i=1}^{N_t}$, where x_t^i are images and $y_t^i \in C_t$ are their labels.

Most deep networks applied to class-incremental learning can be split into two components: a feature extractor F_θ and a common classifier G_ϕ which grows with each new task $t+1$ to include classes C_{t+1}. The feature extractor F_θ first maps the input x to a deep feature vector $z = F_\theta(x) \in \mathbb{R}^d$, and then the unified classifier $G_\phi(z) \in \mathbb{R}^{|C_t|}$ is a probability distribution over classes C_t that is used to make predictions on input x.

Class-incremental learning requires that the model be capable of correctly classifying all samples from previous tasks at *any* training task – that is, when learning task t, the model must not forget how to classify samples from classes from tasks $t' < t$. *Data-free* class-incremental learning additionally restricts models to learn each new task without access to samples from previous ones. This typically leads to learning objectives that minimize a loss function $\mathcal{L}$ defined on current training data D_t:

$$\mathcal{L}_t^{\text{CIL}}(x, y) = \mathcal{L}_{\text{ce}}(G_{\phi_t}(F_{\theta_t}(x)), y) + \mathcal{L}_t^{\text{method}}, \tag{1}$$

where $\mathcal{L}_{\text{ce}}$ is the standard cross-entropy classification loss and L_t^{method} is a method-specific loss that mitigates forgetting of past tasks when learning current task t.

2.2 A Teacher-student Framework for Low-level Supervision

We propose to learn stable features from low-level stationary tasks shared across all incremental tasks during class-incremental learning. Low-level vision tasks like salient object detection require useful representations of input images. By learning these feature representations across tasks, the model can focus on key area of input images and exploit learned, stable features with less representation drift since the low-level features change very little between tasks.

Figure 3: We dilate the teacher boundary map and apply a binary cross entropy loss at three stages in the CNN backbone prevent mid-level student attention from drifting into boundary regions.

Saliency map prediction is relevant to image classification since the foreground largely determines the results of image classification, while the background is comparatively less important. When learning new tasks with new classes, the background of images of new classes may contain new visual concepts that introduce undesirable noise and lead to forgetting essential previous knowledge. The effectiveness of saliency features for learning classification tasks was demonstrated by Saliency Guided Training (Ismail et al., 2021). Additional supervision of salient region boundaries can aid salient object detection tasks for both segmentation and localization (Zhao et al., 2019). The positive interaction between these two tasks brings richer attention to features relevant to the main classification task. It can provide positive guidance in the form of stationary knowledge across class-incremental tasks. Some examples are illustrated in Figure 5.

We incorporate low-level vision tasks into the network using a teacher-student model. A teacher model T generates low-level representations of each input image x (saliency and boundary maps in our experiments). We use CSNet (Cheng et al., 2021) to generate saliency and boundary maps, as it is lightweight and efficient. The boundary map is computed with a Laplacian filter over the estimated saliency map. We add a decoder D_φ (Li et al., 2020) after the backbone F_θ to predict low-level saliency and boundary maps for input images. The average L2 distance between the student and teacher maps is used as a low-level distillation loss:

$$\mathcal{L}_t^{\text{lm}}(x) = \frac{||D_\varphi(F_\theta(x)) - T(x)||_2}{\sqrt{N}}, \tag{2}$$

where $T(x)$ denotes the output of the teacher network on input x, $D_\varphi(x)$ are combined saliency and boundary maps produced by the decoder, and N is the number of pixels in the student and teacher saliency maps.

2.3 Boundary-guided Mid-level Saliency Drift Regularization

The multi-task supervision of salient regions and boundaries described in the previous section encourages the network to learn representation sufficient to reconstruct the teacher outputs. However, it does little to guide attention at intermediate layers in the CNN. To guard against saliency drift at these intermediate layers in the backbone, we use the generated boundary maps as a type of adaptive supervision as shown in Figure 3. When applying dilated boundary supervision, we add a penalty term on object boundaries and avoid drift to the background. We firstly use 0.5 as the threshold to binarize the teacher boundary map and then obtain dilated teacher boundary maps by:

$$B_d(x) = \text{Dilate}(\text{Laplace}(T_b(x)), d) \tag{3}$$

where $T_b(x)$ is the original teacher boundary map of image x, and d denotes the dilation radius applied on the teacher boundary map for controlling the strictness of boundary-guided saliency.

Rather than use a decoder at each layer as described above, the student saliency map is generated using Grad-CAM (Selvaraju et al., 2017) at three stages of the CNN backbone (see Figure 2). We also experiment with several other methods for generating student saliency maps and report on these experiments in Appendix A.1. The dilated teacher boundary map $B_d(x)$ is downsampled to match the feature map dimensions at these three stages in order to compare the Grad-CAM generated saliency boundary maps with the teacher. We use the binary cross entropy loss for supervision on

dilated boundary regions. The loss is defined as:

$$\mathcal{L}_t^{\text{dbs}}(x) = -\frac{\sum_{j=1}^{N} B_d(x,j)\log(1-S_t(x,j))}{\sum_{j=1}^{N} B_d(x,j)}, \tag{4}$$

where $S_t(x,j)$ denotes the student saliency map of image x at pixel j, $B_d(x,j)$ is the dilated teacher boundary map at pixel j, and N is the number of pixels in x. We compute this loss only within dilated boundary regions, that is where $B_d(x,j)=1$. This loss helps the student saliency map have no intersection with the dilated teacher boundary region.

2.4 Saliency Noise Injection

Although we apply low-level teacher-student distillation and dilated boundary supervision to maintain robust saliency representations across tasks, there is still the possibility that the model forgets saliency on samples from previous tasks. To address this, we force the model to recover the correct saliency estimation from injected saliency noise.

At each task there is no available training data from previous or future tasks, and therefore we can not directly know the accurate saliency drift on these samples. Instead of supervising the model with ground-truth saliency drift signals, we introduce saliency noise on random feature channels. We use a random ellipse to approximate the potential saliency drift in future tasks and the model is trained to denoise within each stage. Therefore the model can effectively reduce real saliency drift.

We generate elliptical noise using a very simple approach. There are six parameter dimensions: the center coordinate (x,y), the major and minor axis lengths (a,b), the rotation angle θ, and the mask weight w. A detailed explanation of this process is given in Appendix A.2. With the help of dilated boundary supervision, each stage learns to denoise the additional saliency noise and generalizes this ability in test.

2.5 Final Learning Objective

The overall learning objective combines the low-level multi-task learning, dilated boundary supervision, and random saliency noise injection modules:

$$\mathcal{L}_t^{\text{all}} = \mathcal{L}_t^{\text{CIL}} + \mathcal{L}_t^{\text{lm}} + \mathcal{L}_t^{\text{dbs}}. \tag{5}$$

3 Experimental Results

In this section we first describe the experimental setup first and then we compare ROSS to other state-of-the-art methods on several DFCIL benchmarks. In Section 3.3 we give further analysis over the different components of our proposed approach.

3.1 Experimental Setup

We follow standard experimental protocols for DFCIL on three benchmark datasets.

Datasets. We perform experiments on CIFAR-100 (Krizhevsky et al., 2009), Tiny-ImageNet (Le & Yang, 2015), and ImageNet-Subset (Deng et al., 2009). For most experiments, we train the model on half of the classes for the first task, and then equally distribute the remaining classes across each of the subsequent tasks. The convention we use is: $F + C \times T$ means that the first task contains F classes, and the next T tasks each contain C classes. We consider three configurations for CIFAR-100 and ImageNet-Subset: 50 + 5 $\times$ 10, 50 + 10 $\times$ 5, 40 + 20 $\times$ 3. For Tiny-ImageNet we generate three settings: 100 + 5 $\times$ 20, 100 + 10 $\times$ 10, and 100 + 20 $\times$ 5.

State-of-the-art methods. Since we focus on DFCIL, we mainly compare with data-free state-of-the-art approaches: SSRE (Zhu et al., 2022), PASS (Zhu et al., 2021b), IL2A (Zhu et al., 2021a), EWC (Kirkpatrick et al., 2017), LwF-MC (Rebuffi et al., 2017), and MUC (Liu et al., 2020). To demonstrate the effectiveness of our method, we also compare its performance with several exemplar-based methods like iCaRL (both nearest-mean and CNN) (Rebuffi et al., 2017),

Table 1: Average top-1 accuracy and forgetting on CIFAR-100 for different numbers of tasks. Replay-based methods storing 20 exemplars from each previous class are denoted by †. The best overall results are in **bold**. We run all experiments three times and report average accuracy and standard deviations.

	Metric:	**Accuracy ↑**			**Average Forgetting ↓**		
	Method	**5 tasks**	**10 tasks**	**20 tasks**	**5 tasks**	**10 tasks**	**20 tasks**
Exemplar-based	iCaRL-CNN†	40.12±1.0	39.65±0.8	35.47±0.8	42.13±0.8	45.69±0.8	43.54±0.7
	iCaRL-NCM†	49.74±0.8	45.13±0.7	40.68±0.6	24.90±0.9	28.32±0.7	35.53±0.7
	LUCIR†	55.06±1.0	50.14±0.9	48.78±0.9	21.00±1.5	25.12±1.3	28.65±1.3
	EEIL†	52.35±0.6	47.67±0.5	41.59±0.5	23.36±0.8	26.65±0.9	32.40±0.7
	RRR†	57.22±0.8	55.74±0.8	51.35±0.7	18.05±0.8	18.59±0.8	18.40±0.7
Data-free	LwF_MC	36.17±0.9	17.04±0.9	15.88±0.8	44.23±1.2	50.47±1.0	55.46±1.0
	EWC	9.32±0.7	8.47±0.5	8.23±0.5	60.17±0.8	62.53±0.7	63.89±0.5
	MUC	38.45±0.9	19.57±0.8	15.65±0.8	40.28±1.3	47.56±1.1	52.65±1.0
	IL2A	55.13±0.7	45.32±0.7	45.24±0.6	23.78±1.1	30.41±1.0	30.84±0.7
	PASS	55.67±1.2	49.03±0.9	48.48±0.7	25.20±0.8	30.25±0.7	30.61±0.7
	SSRE	56.33±0.9	55.01±0.7	50.47±0.6	18.37±1.1	19.48±1.0	19.00±1.0
	ROSS (Ours)	**59.26±0.5**	**57.93±0.4**	**53.78±0.4**	**16.42±0.7**	**17.66±0.8**	**17.78±0.6**

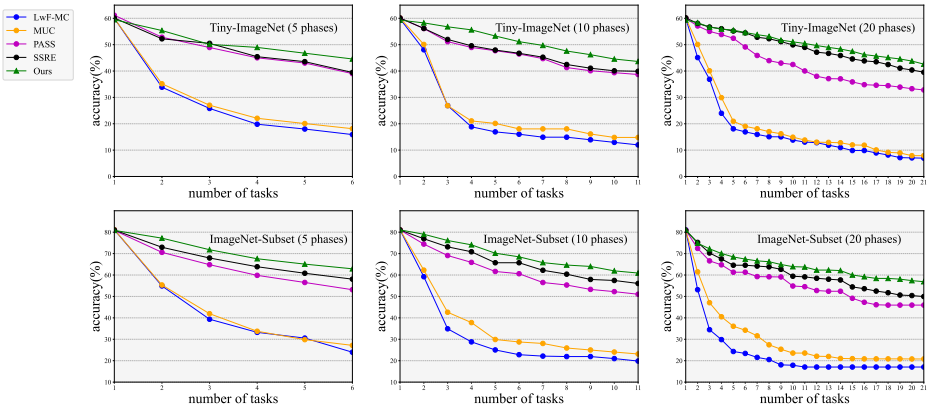


Figure 4: Results on Tiny-ImageNet and ImageNet-Subset for different numbers of tasks. Our method outperforms others, especially on longer task sequences (i.e. more, but smaller, tasks).

EEIL (Castro et al., 2018), LUCIR (Hou et al., 2019). We also compare with RRR (Ebrahimi et al., 2021) integrated with SSRE, which focuses on preserving saliency using exemplar replay.

Implementation details. We use ResNet-18 (He et al., 2016) as a feature extraction backbone. This is the same base network used in SSRE Zhu et al. (2022) and PASS (Zhu et al., 2021b), two state-of-the-art DFCIL approaches. We use the decoder in (Li et al., 2020) to estimation student low-level maps. All experiments are trained from scratch using Adam for 100 epochs with an initial learning rate 0.001. The learning rate is reduced by a factor 10 at epochs 45 and 90. For exemplar-based approaches, we use herding (Rebuffi et al., 2017) to select and store 20 samples per class following common settings (Rebuffi et al., 2017; Hou et al., 2019). We implement RRR Ebrahimi et al. (2021) with SSRE to fairly compare it with our ROSS. We report two common metrics for class incremental learning: top-1 accuracy and average forgetting for all classes learned up to task t. We perform three runs of all experiments and report mean performance and variance. For dilated boundary supervision, we set d of three stages to be 5%, 10% and 15% of the image size.

Table 2: We report the performance gain of top-1 accuracy by applying ROSS to other DFCIL methods in a plug-and-play way. Absolute gains are marked in (red).

Dataset	**CIFAR-100**			**Tiny-ImageNet**		
Method	**5 tasks**	**10 tasks**	**20 tasks**	**5 tasks**	**10 tasks**	**20 tasks**
MUC	38.45	19.57	15.65	18.95	15.47	9.14
+ROSS	49.17 (+10.72)	40.34 (+20.77)	37.86 (+22.21)	32.47 (+13.46)	30.13 (+14.66)	27.70 +18.56
IL2A	55.13	45.32	45.24	36.77	34.53	28.68
+ROSS	58.74 (+3.61)	53.24 (+7.92)	53.07 (+7.83)	42.49 (+5.72)	41.34 (+6.81)	40.59 (+11.91)
PASS	55.67	49.03	48.48	41.58	39.28	32.78
+ROSS	59.10 (+3.43)	54.45 (+5.42)	52.37 (+3.89)	44.05 (+2.47)	43.06 (+3.78)	42.57 (+9.79)
SSRE	56.33	55.01	50.47	41.45	40.07	39.25
+ROSS	59.26 (+2.93)	57.93 (+2.92)	53.78 (+3.31)	44.13 (+2.68)	43.86 (+3.79)	43.55 (+4.30)

Table 3: Ablative experiments on each component of our proposed method. Experiments are on CIFAR-100 in the 10 task setting and we report the top-1 accuracy in %. We use DBS, LM, SNI to denote the three components of ROSS: dilated boundary supervision, low-level multi-task supervision, and saliency noise injection.

Method & Tasks	DBS	LM	SNI	1	2	3	4	5	6	7	8	9	10	11
Baseline				78.7	73.6	71.8	68.4	64.1	62.7	59.7	57.6	56.8	55.6	55.0
Variants	✓			78.9	75.3	72.9	68.7	65.4	63.9	61.9	59.9	58.8	56.7	55.8
		✓		78.7	75.3	72.9	69.1	65.8	64.3	62.3	60.5	58.6	57.1	56.2
			✓	78.7	75.1	72.8	69.3	65.9	64.4	62.5	60.5	58.8	57.6	56.7
	✓	✓		79.0	76.0	72.8	67.3	65.1	63.7	61.6	60.9	59.8	58.9	57.3
	✓		✓	79.1	75.9	73.0	69.8	66.3	64.7	62.9	61.3	59.7	57.8	57.0
		✓	✓	79.1	76.3	72.9	69.7	65.1	64.3	61.4	60.1	59.2	58.9	57.6
	✓	✓	✓	79.1	76.1	72.9	69.7	65.0	63.6	62.5	60.7	59.5	59.0	57.9

3.2 Comparison with the State-of-the-art

We report the comparative performance on CIFAR-100 in Table 1 and on Tiny-ImageNet and ImageNet-Subset in Figure 4. ROSS outperforms all data-free approaches. For exemplar-based methods like iCaRL (Rebuffi et al., 2017), EEIL (Castro et al., 2018), and LUCIR (Hou et al., 2019), our method still has significantly better performance. On longer sequences (i.e. 10 and 20 tasks), our methods significantly reduce forgetting when learning new classes compared with other DFCIL methods. Although our method has the similar top-1 accuracy on the first task, it has better performance in most intermediate tasks and the final task. For longer sequences in Figure 4, the gap between our method and the best baseline is kept large consistently, showing the effectiveness of our method on relieving forgetting. The performance gain is larger on Tiny-ImageNet and ImageNet-Subset compared to CIFAR100, and this demonstrates the generality of our method to dataset with larger spatial size and scale. It is worth mentioning that our ROSS also produces results with smaller variance. We conclude this as the contribution of reducing saliency drift to background regions, which may incorporate some random noise.

3.3 Additional Analysis

In this section we take a deeper look at three approaches we propose. If not specified, the results are produced under settings of ROSS integrated SSRE.

Plug-and-play with other DFCIL methods. Some existing DFCIL methods, for example PASS, IL2A and SSRE, focus on reducing forgetting via embedding regularization. Considering the importance of saliency to image classification, it is natural consider whether ROSS can be integrated into these methods. The results in Table 2 show the performance gain brought by applying our method. Adding ROSS doubles the performance for MUC in many cases and significantly improves IL2A

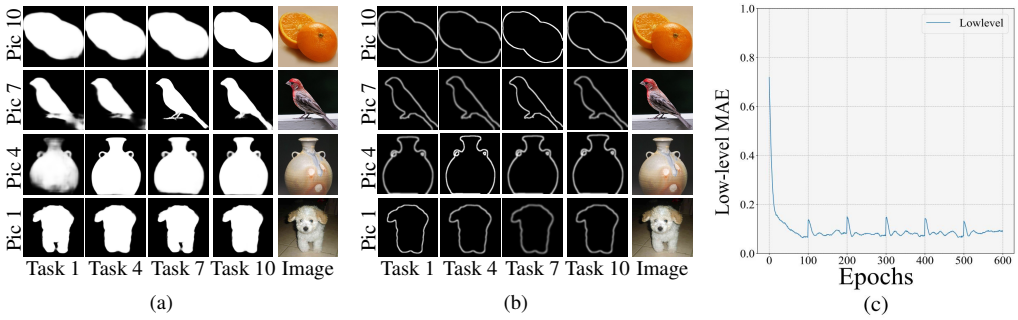


Figure 5: Visualization of the saliency (a) and boundary (b) maps from our student encoder-decoder network with original images from different tasks at different stages of incremental learning. Our method also produces stable low-level results while reducing forgetting in classification. (c) The MAE loss of low-level tasks between the student and teacher network across tasks.

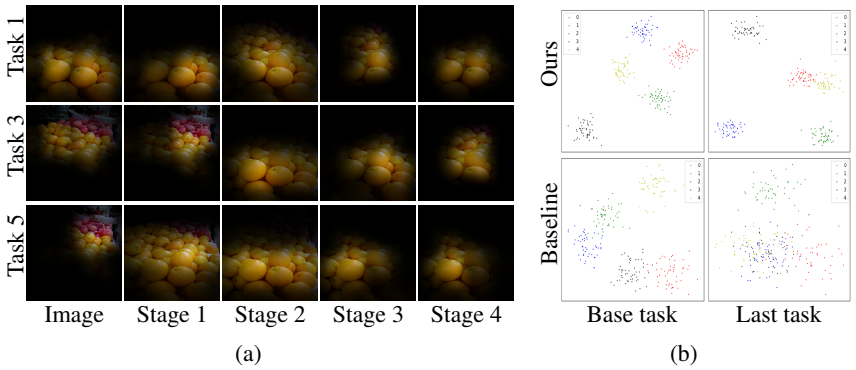


Figure 6: Visualization of the saliency across tasks within different stages, as shown in (a). Some saliency drifts at later task are recovered through stages. Dilated boundary supervision also shrink the saliency map in deeper stages, increasing their robustness. (b): Visualization of the refined embedding with our method. Compared with the baseline, our method can preserve more discriminative representations between classes from one task and between different tasks.

and PASS. When we incorporate it into the best baseline SSRE, it yields a consistent gain of by about 3%.

Ablation Study. To assess each component ROSS, we performed a set of ablations using the 10-task setting on CIFAR-100 (see Table 3). We consider has eight configurations with different subsets of the three modules. For settings with a single component, we see that saliency denoising works well to relieve saliency drift on later phases and performs better. For settings with two modules, we see that the low-level multi-task saliency supervision provides the most saliency signals for supervision and further enhances the performance. The dilated boundary supervision also helps saliency regularization by further guarding against saliency drift across boundaries.

Low-level multi-task. To analyze the effect of our proposed low-level multi-task, we make detailed experiments on ImageNet-Subset of 5 and 10 tasks setting. We first plot the loss across tasks in Figure 5 (c). After learning to predict boundary and saliency maps along with the first task, the network maintains good performance for the rest of the 5 tasks sequence. This shows that the low-level tasks we are stable during continual learning. Furthermore, we visualize the results of saliency and boundary map prediction during the incremental learning in Figure 5 (a) and (b). We show some samples of predicted boundary and saliency maps after learning different tasks. Although CIL involves samples of different classes, we can see that the low-level outputs are relatively stable and class-agnostic. If the model is able to extract these low-level features continuously, it can preserve useful prior knowledge for learning samples of new classes.

Saliency denoising across stages. To show the effect of saliency denoise, we select the 5-task setting on ImageNet-Subset. We visualize a sample and its saliency at different training phases and encoder stages in Figure 6a. With the help the denoising process, the encoder is able to recover intermediate saliency drift and maintain accurate attention in the output. When testing samples of old classes, the network reduces the saliency drift and focuses on important regions.

Discriminative embeddings over different classes. Since our method helps model focus on the foreground, more class-specific pixels are used to computing embeddings. This makes them more discriminative with less distracting background information. We use T-SNE to visualize embeddings of 5 initial classes right after learning the base task and the last task with the 10-task setting on ImageNet-Subset in Figure 6b. At the base task, both baseline and Ours perform well. When we evaluate it after the last task, it is obvious that Ours can still keep discriminative between tasks but the baseline has overlapping embeddings.

4 Related Work

We discuss previous work on incremental learning from the recent literature, and then describe the state-of-the-art in DFCIL.

Incremental Learning. Various methods have been proposed for incremental learning in the past few years (Delange et al., 2021; Belouadah et al., 2021). Recent works can be coarsely grouped into three categories: replay-based, regularization-based, and parameter-isolation methods. Replay-based methods mitigate the task-recency bias by retaining training samples from previous tasks. In addition to replaying samples BiC (Wu et al., 2019), PODNet (Douillard et al., 2020), and iCaRL (Rebuffi et al., 2017) apply a distillation loss to prevent forgetting and enhance model stability. GEM (Lopez-Paz & Ranzato, 2017), AGEM (Chaudhry et al., 2019), and MER (Riemer et al., 2019) exploit past-task exemplars by modifying gradients on current training samples to match old samples. Rehearsal-based methods may cause models to overfit to stored samples. Regularization-based approaches such as LwF (Li & Hoiem, 2016), EWC (Kirkpatrick et al., 2017), and DMC (Zhang et al., 2020) offer ways to learn better representations while leaving enough plasticity for adaptation to new tasks. Parameter-isolation methods (Mallya & Lazebnik, 2018; Xu & Zhu, 2018) use models with different computational graphs for each task. With the help of growing models, new model branches mitigates catastrophic forgetting at the cost of more parameters and computational costs.

Data-free Class Incremental Learning. Compared to conventional class incremental learning, data-free class incremental learning is more practical for applications where training data is sensitive and may not be stored in perpetuity. DAFL (Chen et al., 2019) uses synthesized samples place of stored exemplars introduces an extra GAN architecture for generating synthesized images. DeepInversion, which inverts trained networks using random noise to generate images, is another popular DFCIL method (Yin et al., 2020). Always Be Dreaming further improves on DeepInversion for DFCIL (Smith et al., 2021). SDC attempts to overcome the problems caused by semantic drift when training new tasks on old class samples (Yu et al., 2020). It directly estimates prototypes of each learned class for the nearest class mean classifier. PASS (Zhu et al., 2021b) and IL2A (Zhu et al., 2021a) are prototype-based replay methods for efficient and effective DFCIL. Both introduce an efficient way of generating prototypes of old classes. Since these prototypes are features computed from past training samples, original images are not retained. SSRE (Zhu et al., 2022) introduces another re-parameterization methods for trade-off between old and new knowledge. Our method proposes three new components to reduce saliency drift in DFCIL setting, which is complementary to some approaches mentioned above.

5 Conclusions

This paper rethinks saliency drift and its effect on DFCIL. The insight behind our ROSS is to consider guiding the model to focus on salient regions and to simply apply distillation during training. We show that some precaution and robust supervision is necessary to mitigate forgetting of saliency across tasks. Experiments on several datasets and settings demonstrate that ROSS is effective and surpasses the state-of-the-art. ROSS can be easily combined with other methods, leading to large performance gains over baselines.

6 ETHICS STATEMENT

In this work, we study class-incremental learning under data-free setting, which can improve the efficiency of learning, reduce the computation cost and protect the data privacy. It is an essential ingredient toward general artificial intelligence, which can be useful in many applications.

7 REPRODUCIBILITY STATEMENT

The code for this paper is written in PyTorch 1.9.0. All datasets used for the training and fine-tuning are publicly available. The Grad-CAM maps is generated using pytorch_grad_cam 0.1.0. We will make our code publicly available for reproducibility.

REFERENCES

Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38–54, 2021.

Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 233–248, 2018.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. In *ICLR*, 2019.

Hanting Chen, Yunhe Wang, Chang Xu, Zhaohui Yang, Chuanjian Liu, Boxin Shi, Chunjing Xu, Chao Xu, and Qi Tian. Data-free learning of student networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3514–3522, 2019.

Ming-Ming Cheng, Shanghua Gao, Ali Borji, Yong-Qiang Tan, Zheng Lin, and Meng Wang. A highly efficient model to study the semantics of salient object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. doi: 10.1109/TPAMI.2021.3107956.

Matthias Delange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Greg Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5138–5146, 2019.

Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *ECCV*, 2020.

Sayna Ebrahimi, Suzanne Petryk, Akash Gokul, William Gan, Joseph E. Gonzalez, Marcus Rohrbach, and trevor darrell. Remembering for the right reasons: Explanations reduce catastrophic forgetting. In *International Conference on Learning Representations*, 2021.

Ian J Goodfellow, Mehdi Mirza, Da Xiao, Aaron Courville, and Yoshua Bengio. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 831–839, 2019.

Aya Abdelsalam Ismail, Hector Corrada Bravo, and Soheil Feizi. Improving deep learning interpretability by saliency guided training. *Advances in Neural Information Processing Systems*, 34, 2021.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

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

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Xiangtai Li, Ansheng You, Zhen Zhu, Houlong Zhao, Maoke Yang, Kuiyuan Yang, and Yunhai Tong. Semantic flow for fast and accurate scene parsing. In *ECCV*, 2020.

Yin Li, Xiaodi Hou, Christof Koch, James M Rehg, and Alan L Yuille. The secrets of salient object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 280–287, 2014.

Zhizhong Li and Derek Hoiem. Learning without forgetting. In *14th European Conference on Computer Vision, ECCV 2016*, pp. 614–629. Springer, 2016.

Yu Liu, Sarah Parisot, Gregory Slabaugh, Xu Jia, Ales Leonardis, and Tinne Tuytelaars. More classifiers, less forgetting: A generic multi-classifier paradigm for incremental learning. In *European Conference on Computer Vision*, pp. 699–716. Springer, 2020.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Arun Mallya and Svetlana Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 7765–7773, 2018.

Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost van de Weijer. Class-incremental learning: survey and performance evaluation on image classification. *arXiv preprint arXiv:2010.15277*, 2020.

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.

Quang Pham, Chenghao Liu, and Steven Hoi. Dualnet: Continual learning, fast and slow. *Advances in Neural Information Processing Systems*, 34, 2021.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

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. In *In International Conference on Learning Representations (ICLR)*, 2019.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

James Smith, Yen-Chang Hsu, Jonathan Balloch, Yilin Shen, Hongxia Jin, and Zsolt Kira. Always be dreaming: A new approach for data-free class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9374–9384, 2021.

Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 374–382, 2019.

Ju Xu and Zhanxing Zhu. Reinforced continual learning. *Advances in Neural Information Processing Systems*, 31, 2018.

Qiong Yan, Li Xu, Jianping Shi, and Jiaya Jia. Hierarchical saliency detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1155–1162, 2013.

Chuan Yang, Lihe Zhang, Huchuan Lu, Xiang Ruan, and Ming-Hsuan Yang. Saliency detection via graph-based manifold ranking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3166–3173, 2013.

Hongxu Yin, Pavlo Molchanov, Jose M Alvarez, Zhizhong Li, Arun Mallya, Derek Hoiem, Niraj K Jha, and Jan Kautz. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8715–8724, 2020.

Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6982–6991, 2020.

Junting Zhang, Jie Zhang, Shalini Ghosh, Dawei Li, Serafettin Tasci, Larry Heck, Heming Zhang, and C-C Jay Kuo. Class-incremental learning via deep model consolidation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1131–1140, 2020.

Jia-Xing Zhao, Jiang-Jiang Liu, Deng-Ping Fan, Yang Cao, Jufeng Yang, and Ming-Ming Cheng. Egnet: Edge guidance network for salient object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8779–8788, 2019.

Fei Zhu, Zhen Cheng, Xu-yao Zhang, and Cheng-lin Liu. Class-incremental learning via dual augmentation. *Advances in Neural Information Processing Systems*, 34, 2021a.

Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5871–5880, 2021b.

Kai Zhu, Wei Zhai, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Self-sustaining representation expansion for non-exemplar class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9296–9305, 2022.

Table 4: Ablation on method for generating student saliency maps on Tiny-Imagenet.

	5 Tasks	10 Tasks	20 Tasks
baseline (SSRE)	40.2	40.0	39.3
CAM	41.2	40.7	40.4
SmoothGrad	42.1	41.0	40.4
Grad-CAM	44.1	43.9	43.5

Table 5: Parameters and FLOPs of the teacher model. FLOPs are computed with $3 \times 32 \times 32$ images.

Model	Parameter(M)	FLOPS(G)
Ours	17.9	0.78
Teacher model	0.0941	0.012

A Appendices

A.1 Ablation on Student Saliency Methods

To show the generalization of ROSS, we use several methods to compute student saliency maps and report the results in Table 4. Grad-CAM performs best, although other methods yield performance gains, demonstrating the effectiveness of ROSS.

A.2 Generation of Saliency Noise

For each ellipse there are 6 dimensions: the center coordinate (x, y), the rotation angle θ, the mask weight w, and the major and minor axes (a, b). x, y, θ and w are sampled from a uniform distribution over ranges: $x \in [0, H)$, $y \in [0, W)$, $\theta \in [0, 2\pi)$, $w \in [0, 1]$. H W denote the height and width of input images. To generate ellipses of appropriate size, we draw the major and minor axes from a Gaussian distribution with $\mu_a = \max(H, W)/2$, $\sigma_a = \max(H, W)/6$, $\mu_b = \min(H, W)/2$, $\sigma_b = \min(H, W)/6$. The sampled a, b is clipped to$[0, max(H, W)/2]$ and $[0, min(H, W)/2]$, respectively. For each ellipse, we create a saliency map S_i. We repeat this random generation process 5-10 times and an element-wise max operation on these S_i to get a single saliency map S. For each encoder feature map, 10% of randomly selected channels are directly masked with S, where each selected channel will have an independent S.

A.3 Teacher Model

We use CSNet (Cheng et al., 2021) to compute all the teacher saliency and boundary maps because it is very lightweight. Compared to our main model, the teacher model has fewer than 1% parameters and requires 1.5% of the FLOPs (as shown in Table 5). Note that we compute all low-level maps offline before new tasks, and so the extra FLOPs should be amortized over the number of epochs. Therefore, the additional FLOPs required by the teacher model is only about 0.015% of the main model, which is negligible in practice.

A.4 Ablation on Low-level Teacher Maps

To show the effectiveness of our ROSS, we perform an ablation on the low-level teacher maps. We replace them with the Grad-CAM generated from a ResNet-152 network. To avoid information leakage, ResNet-152 is trained from scratch. Before each new task, we first train it only on task data data and use the Grad-CAM output to supervision saliency in our incremental model. From Table 6 we see that ROSS still outperforms previous methods.

A.5 Ablation on Student Architecture Teacher Pretraining

We select PASS (Zhu et al., 2021b) as our baseline method to apply ROSS to (as shown in Table 7). Experiments are on ImageNet-Subset with 5 tasks. We ablate the teacher pretraining and student

Table 6: Ablation on low-level teacher saliency maps on Tiny-ImageNet with 10 tasks.

Low-level source(Method)	Accuracy(%)
PASS	39.3
SSRE	40.0
ResNet152(Ours)	42.1
CSNet(Ours)	43.9

Table 7: Ablations on student architecture and teacher pretraining.

(a) We compare different student network architectures Method-Res18 denotes applying Method with ResNet18 as its backbone.

Method	Parameter(M)	Accuracy(%)
PASS-Res18	14.5	50.4
PASS-Res32	21.7	51.2
SSRE-Res18	19.4	58.7
Ours-Res18	17.9	61.5

(b) Ablation on teacher network pretaining.

Method	Accuracy(%)
No pretraining	61.5
Pretrained teacher model	62.0

decoder. Since some methods use ImageNet pretrained weights for better saliency map estimation, we train CSNet (Cheng et al., 2021) from scratch on the dataset (with and without pretaining) for salient object detection (Yan et al., 2013; Li et al., 2014; Yang et al., 2013). This allows us to verify that no information leakage happens due to pretraining the saliency network on ImageNet. From the results in Table 7 we see that even with a very small backbone (ResNet-18) ROSS yields significant gains other other approaches. Similarly, the teacher network without pretraining works almost as well as pretraining the saliency network on ImageNet.